Hortonworks Data Platform

IOP to HDP Migration

(October 30, 2017)

docs.cloudera.com

Hortonworks Data Platform: IOP to HDP Migration

Copyright © 2012-2017 Hortonworks, Inc. Some rights reserved.

The Hortonworks Data Platform, powered by Apache Hadoop, is a massively scalable and 100% open source platform for storing, processing and analyzing large volumes of data. It is designed to deal with data from many sources and formats in a very quick, easy and cost-effective manner. The Hortonworks Data Platform consists of the essential set of Apache Hadoop projects including MapReduce, Hadoop Distributed File System (HDFS), HCatalog, Pig, Hive, HBase, ZooKeeper and Ambari. Hortonworks is the major contributor of code and patches to many of these projects. These projects have been integrated and tested as part of the Hortonworks Data Platform release process and installation and configuration tools have also been included.

Unlike other providers of platforms built using Apache Hadoop, Hortonworks contributes 100% of our code back to the Apache Software Foundation. The Hortonworks Data Platform 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 Data Platform 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. Migrating from IOP to HDP

This topic provides an overview of the full process required to migrate from IBM Open Platform (IOP) versions 4.2.0 or 4.2.5 to Hortonworks Data Platform (HDP) version 2.6.3.

Prerequisites for your migration process:

Process Overview Steps:

1. **Back up Metadata**.

   For details on how to back up:

   • metadata for IBM value-add services, see Backing up Value Add Metadata.

   • configurations for open sources components, see Back up Configurations.

   • open source Solr, see Back up Solr.

2. **Upgrade your version of Ambari**.

   For Ambari upgrade details, see Upgrade Ambari.

3. **Clean up IBM value-added services from your environment**.

   For value-add service cleanup details, see Cleaning up IBM value-add services.

4. **Remove deprecated components and services from your environment**.

   For component and service deletion details, see Removing deprecated components and services.

5. **Upgrade the stack**.

   For stack upgrade details, see Upgrade the Stack.

6. **Upgrade and finalize your Big SQL installation**.

   For upgrading and finalizing details, see Upgrading and finalizing Big SQL.

7. **Upgrade IBM DSM**.

   For upgrading details, see Upgrading IBM Data Server Manager (DSM).

2. Prerequisites

Meeting all prerequisites is essential for Ambari to know the cluster is in a healthy operating mode and can successfully manage the upgrade process. To perform an IOP migration using Ambari, your cluster must meet the following prerequisites.

IOP Version	All hosts must have IOP version 4.2.0 or 4.2.5 installed.
Disk Space	Be sure to have adequate space on `/usr/hdp` for the target HDP version. Each complete install of an HDP version will occupy about 2.5 GB of disk space.
Ambari Agent Heartbeats	All Ambari Agents must be communicating and heartbeating to the 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. Run **Service Actions > Run Service Check** on all services and remediate any issues, if necessary, before performing the IOP to HDP upgrade.
Ranger Policy Export/Import DB Requirements	HDP-2.6 and higher versions include Ranger policy import/export. The minimum database requirements for this feature are as follows: • MariaDB: 10.1.16+ • MySQL: 5.6.x+ • Oracle: 11gR2+ • PostgreSQL: 8.4+ • MS SQL: 2008 R2+
Big SQL	• To upgrade Big SQL, you must have a user with the following attributes:

- passwordless sudo access on all nodes of the cluster, including the Ambari server itself.

- The ability to connect passwordlessly through ssh from the Ambari server to all Big SQL nodes. This user can be root. If the user is not root, the user name must be passed to the upgrade script with the -a option. The upgrade script must be run with root user privilege. This can be achieved by using the sudo command. If you have configured Ambari for non-root access (see Configuring Ambari for non-root access), use the -a option with the user name created for that purpose.

- Big SQL installation requires about 2 GB of free disk space on the /usr partition. The upgrade process requires about 4 GB of free disk space on the root partition. This space is used temporarily by the upgrade process. It is released when the upgrade is complete.

- You must disable Big SQL high availability before performing an upgrade.

- Ambari configuration groups are not supported for the Big SQL service. The upgrade script produces a warning message if you use configuration groups. If you override the warning, you must validate that the configuration of all nodes in all configuration groups is updated as part of the upgrade. It is recommended that you remove all configuration groups before performing an upgrade.

- You must disable Yarn and Slider support for Big SQL before performing an upgrade.

3. Back up Value Add Metadata

Please see the IBM Knowledge Center topic:

Back up Value Add Metadata

4. Back Up Configurations

Next, make backups of the following configurations:

4.1. Backup WebHcat Configuration

WebHcat configuration might be erased when HDP rpm package is uninstalled. It is recommended to backup WebHcat configuration on WebHcat node for rollback.

```
mkdir -p /tmp/backup
tar cfvz /tmp/backup/webhcat.tar.gz /etc/hive-webhcat
```

4.2. Back up Hcatalog Configuration

Hcatalog configuration might be erased when HDP rpm package is uninstalled. It is recommended to backup Hcatalog configuration on Hcatlog node for rollback.

```
tar cfvz /tmp/backup/hcatalog.tar.gz /etc/hive-hcatalog
```

4.3. Back up Spark Configuration

Spark configuration will be erased when HDP rpm package is uninstalled. It is recommended to backup Spark configuration on all nodes for rollback.

```
tar cfvz /tmp/backup/spark.tar.gz /etc/spark
```

4.4. Back up Solr

This topic describes the steps necessary to back up your Solr index, configurations, data, and core instance.

Preparing Solr for Backups

1. For each replica, optimize the index.

2. Verify that for each replica, the value of num_docs is equal to the value of total_docs.

3. Verify that all Collections are healthy.

4. Verify that all Shards have live leaders.

5. Verify that all Replicas of a collection are healthy.

6. Record the number of documents in each replica.

7. Stop Solr and run CheckHDFSIndex for each replica to verify index health.

8. Start Solr.

Back up Jars

1. Backup all custom jars added to Solr's classpath.

2. The jars in the classpath can be seen on the Solr UI unde Java properties.

3. For each replica, check status of backup.

Back up Index

1. Create a directory owned by Solr to store the backups.

2. For each replica, create a backup.

3. Do not issue multiple backup commands on the same replica!

Verify Backups

1. For each backup, run **CheckIndex**.

2. Verify index integrity.

Example Code:

```
#CheckIndex
 su - solr
 cd /usr/iop/4.2.0.0/solr
 java -cp "$(hadoop classpath):
 ./server/solr-webapp/webapp/WEB-INF/lib/*:./server/lib/ext/*:/usr/iop/
current/hadoop-client/lib/servlet-api-2.5.jar"
 -ea:org.apache.lucene... org.apache.lucene.index.CheckIndex
 /tmp/backups/snapshot.c1_shard1_replica1/

#CheckHDFSIndex
 su - solr
 cd /usr/iop/4.2.0.0/solr
 java -cp "$(hadoop classpath):
 ./server/solr-webapp/webapp/WEB-INF/lib/*:./server/lib/ext/*:/usr/iop/
current/hadoop-client/lib/servlet-api-2.5.jar"
 -ea:org.apache.lucene...
 org.apache.solr.index.hdfs.CheckHdfsIndex
 hdfs://node1.ibm.com:8020/apps/solr/data/c1/core_node1/data/index

#Backup
 su - solr

 curl
'http://localhost:8983/solr/c1_shard1_replica1/replication?command=backup&
name=c1_shard1_replica1&location=/tmp/backups'

#Backup Status
 curl
 'http://localhost:8983/solr/c1_shard1_replica1/replication?command=details' |
 xmllint --format
```

Backup Configuration:

1. Download `migration_helper.py` from

   https://github.com/hortonworks/iop-solr-stack/releases/download/iop-solr-1.0.0.0-migration/migration_helper.py.

or download the Solr mpack from

http://s3.amazonaws.com/dev.hortonworks.com/IOPSOLR/centos7/1.x/
BUILDS/1.0.0.0-24/tars/iopsolr/iop-solr-mpack-6.3.0.1.0.0.0-24.tar.gz

(replace centos 7 with centos6 in the path if the machine is running centos6 OS). Untar
that tarball, and find the `migration_helper.py` in the `\utilities` folder.

2. Backup configurations

   a. Create a backup folder.

      For example:

      ```
      mkdir -p /tmp/solr-backup
      ```

   b. Run migration_helper.py with `--action backup` flag:

      ```
      /usr/bin/python migration_helper.py --action backup --backup-location /
      tmp/solr-backup
      --cluster <cluster-name> -H <ambari-server-host> --username
      <ambari-admin-username> --password <ambari-admin-password> --port
       <ambari-server-port>
      ```

      > **Note**
      >
      > This operation will generate SOLR-backup-*.json files.
      >
      > Ambari credentials/host/port are used here.
      >
      > Use `-s` if ambari-server uses https.

3. Backup solr core (optional, but recommended):

   ```
   "curl http://<solr-server>:8983/solr/<core_name>/replication?command=backup&
   location=<backup-location>
   ```

   Replace core name and backup location with actual ones.

   > **Note**
   >
   > Do not forget `kinit` and `--negotaiate -u:` flags in the curl command if
   > Solr uses kerberos.

   If you are migrating from IOP 4.2.0, see https://cwiki.apache.org/confluence/display/
   solr/Making+and+Restoring+Backups

5. Upgrade Ambari

This document describes the procedure for upgrading Ambari, before upgrading the Ambari-managed IOP cluster to an Ambari-managed HDP cluster.

⚠️ **Important**

Migrating to HDP-2.6.3 requires Ambari-2.6.0.

To migrate an IOP-4.2.0 cluster, you must upgrade Ambari to version 2.5.2 and then to Ambari-2.6.0 before migrating to HDP-2.6.3.

To migrate an IOP-4.2.5 cluster to HDP-2.6.3, upgrade Ambari to version 2.6.0, then migrate to HDP-2.6.3.

5.1. Preparing to Upgrade Ambari

- Be sure to review the Ambari 2.6.0.0 Release Notes for Known Issues and Behavioral Changes.

- You **must** have root, administrative, or root-equivalent authorization on the Ambari server host and all servers 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`.

- **Plan to upgrade the Ambari Metrics service:**

  - Record the location of the **Metrics Collector** component before you begin the upgrade process.

  - You **must** stop the Ambari Metrics service from **Ambari Web**.

  - After upgrading Ambari, you must also upgrade Ambari Metrics System and add the Grafana component.

- **Optional:** You may add the following property to the `ambari.properties` file.

  stack.upgrade.bypass.prechecks=true

  Adding this property allows the Express Upgrade process to start,if pre-check failures occur.

- If your cluster has no access to the Internet, you must set up a local repository before you continue.

  📝 **Note**

  During Ambari upgrade, the existing `/var/lib/ambari-server/ambari-env.sh` file is overwritten. 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 size), you will need to re-apply your changes to the new file manually.

5.2. Upgrade Ambari

1. If you are running Ambari Metrics service in your cluster, stop the service. From **Ambari Web**, browse to **Services > Ambari Metrics** and select **Stop** from the **Service Actions** menu.

2. Stop the Ambari Server. On **the host** running Ambari Server:

```
ambari-server stop
```

3. Stop all Ambari Agents. On **each host** in your cluster running an Ambari Agent:

```
ambari-agent stop
```

4. **ONLY IF migrating from IOP-4.2.0**, upgrade Ambari to version 2.5.2.

> ⚠️ **Important**
>
> If migrating from IOP-4.2.5, skip this step and proceed with step 5.

a. Fetch the Ambari-2.5.2 repo and replace the old repository file with the new repository file **on all hosts** in your cluster.

> ⚠️ **Important**
>
> 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 will be saved with a numeric extension, such as ambari.repo.1. Make sure that the version you copy is the new version.

Select the repository appropriate for your environment:

• **For RHEL/CentOS/Oracle Linux 6:**

```
wget -nv http://public-repo-1.hortonworks.com/ambari/centos6/2.x/
BUILDS/2.5.2.0/ambari.repo -O /etc/yum.repos.d/ambari.repo
```

• **For RHEL/CentOS/Oracle Linux 7:**

```
wget -nv http://public-repo-1.hortonworks.com/ambari/centos7/2.x/
BUILDS/2.5.2.0/ambari.repo -O /etc/yum.repos.d/ambari.repo
```

• **For IBM Power Systems:**

```
wget -nv http://public-repo-1.hortonworks.com/ambari/centos7-ppc/2.x/
updates/2.5.2.0/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 before you continue.

Note

Ambari Server does not automatically turn off `iptables`. After upgrading the server, you must either disable `iptables` manually or make sure that you have appropriate ports available on all cluster hosts.

b. Upgrade Ambari Server. On **the host** running Ambari Server:

```
yum clean all
```

```
yum info ambari-server
```

In the info output, visually validate that there is an available version containing "2.5.2"

```
yum upgrade ambari-server ambari-agent
```

c. Check for upgrade success by noting progress during the Ambari Server installation process you started in Step 5.

- 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-2.5.2.0-220.x86_64 Complete!
```

Note

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 other ambari-server*.jar files except for the ambari-server-2.5.2.jar to `/tmp` before proceeding with upgrade.

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

```
yum upgrade ambari-agent
```

Note

If you will immediately upgrade Ambari to version 2.6.0 , upgrading the agents to version 2.5.2 is optional.

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

```
rpm -qa | grep ambari-agent
```

f. Upgrade Ambari Server database schema. On **the host** running Ambari Server:

```
ambari-server upgrade
```

5. Upgrade Ambari to version 2.6.0.

a. Fetch the Ambari-2.6.0 repo and replace the old repository file with the new
repository file **on all hosts** in your cluster.

> ⚠️ **Important**
>
> 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 will
> be saved with a numeric extension, such as `ambari.repo.1`. Make sure
> that the version you copy is the new version.

Select the repository appropriate for your environment:

• **For RHEL/CentOS/Oracle Linux 6:**

```
wget -nv http://public-repo-1.hortonworks.com/ambari/centos6/2.x/
BUILDS/2.6.0.0/ambari.repo -O /etc/yum.repos.d/ambari.repo
```

• **For RHEL/CentOS/Oracle Linux 7:**

```
wget -nv http://public-repo-1.hortonworks.com/ambari/centos7/2.x/
BUILDS/2.6.0.0/ambari.repo -O /etc/yum.repos.d/ambari.repo
```

• **For IBM Power Systems:**

```
wget -nv http://public-repo-1.hortonworks.com/ambari/centos7-ppc/2.x/
updates/2.6.0.0/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 before you continue.

> 📝 **Note**
>
> Ambari Server does not automatically turn off `iptables`. After
> upgrading the server, you must either disable `iptables` manually or
> make sure that you have appropriate ports available on all cluster hosts.

b. Upgrade Ambari Server. On **the host** running Ambari Server:

```
yum clean all
```

```
yum info ambari-server
```

In the info output, visually validate that there is an available version containing "2.6.0"

```
yum upgrade ambari-server ambari-agent
```

c. Check for upgrade success by noting progress during the Ambari Server installation process you started in Step 5.

- 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-2.6.0.0-267.x86_64 Complete!
```

> **Note**
>
> 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 other ambari-server*.jar files except for the ambari-server-2.6.0.jar to `/tmp` before proceeding with upgrade.

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

```
yum upgrade ambari-agent
```

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

```
rpm -qa | grep ambari-agent
```

f. Upgrade Ambari Server database schema. On **the host** running Ambari Server:

```
ambari-server upgrade
```

6. If you have configured Ambari to authenticate using LDAP or Active Directory, you **must** re-run

```
ambari-server setup-ldap
```

7. If you have configured your cluster for Hive or Oozie with an external database (Oracle, MySQL or PostgreSQL), you **must** re-run

```
ambari-server setup --jdbc-db={database-type} --jdbc-driver={/jdbc/driver/
path}
```

to get the appropriate JDBC driver .jar file in place. For example:

```
ambari-server setup --jdbc-db=mysql --jdbc-driver=/usr/share/java/mysql-
connector-java.jar
```

8. If the original IOP cluster was configured for a non-root user, you must update `/etc/sudoers` file using the **visudo** command in all the nodes in your cluster, to remove

sudoer permissions for the non-root user added in IOP non-root configuration. You should refresh the `sudoers` file to the original settings, as described here.

9. Start the Ambari Server. On **the host** running Ambari Server:

```
ambari-server start
```

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

```
ambari-agent start
```

11 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.

12 Log in, using the Ambari administrator credentials that you have set up.

For example, the default name/password is **admin/admin**.

You will see a Restart indicator next to each service after upgrading. The Ambari upgrade has added to/adjusted the configuration properties of your cluster based on new configuration types and properties being made available for each service with Ambari-2.6.0. Review these changes by comparing the previous configuration with the latest version created by the Ambari upgrade.

> 📝 **Note**
>
> HDFS may fail to restart after upgrading Ambari on a NameNode HA-enabled, IOP cluster.
>
> To resolve this issue, see Troubleshooting Migration Issues.

Next Steps

Post-Upgrade Tasks *Mandatory*

> ⚠️ **Important**
>
> Completing post-upgrade tasks is mandatory.

More Information

Using a Local Repository

Configuring Network Port Numbers

Set Up LDAP or Active Directory Authentication

Upgrade Ambari Metrics

Add Grafana

5.3. *Mandatory* Post-Upgrade Tasks

Depending on the configuration of your cluster and your current Ambari version, you must upgrade any of the following features in your cluster, as described in the following topics:

Upgrading Ambari Metrics	If your cluster includes the Ambari Metrics System (AMS) service, you must upgrade the system along with Ambari.
Adding Grafana to Ambari Metrics	If you are upgrading from IOP-4.2.0 or earlier, your Ambari Metrics service does not contain the Grafana component and it will need to be added for full functionality.
Upgrading Configurations	Certain scenarios may require that you modify configurations that Ambari did not upgrade automatically.

> **Note**
>
> If you are upgrading from IOP-4.2.5, you must upgrade Ambari Infra and upgrade Ambari Log Search.

Disable Audit to Solr	*Only if* you are migrating from IOP 4.2.5 cluster with Ranger and Solr.

Next Steps

Restart services, <u>only</u> after you complete all applicable post-upgrade tasks.

5.3.1. Upgrading Ambari Metrics

Prerequisites

Upgrade to Ambari 2.6.0 and perform needed post-upgrade checks. Make sure all services are up and healthy.

Steps

1. Make sure Ambari Metrics service is 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:

```
yum clean all
```

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

3. Execute the following command on all hosts running the Metrics Collector:

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

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

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

5. Start the Ambari Metrics Service.

   From **Ambari Web**, browse to **Services > Ambari Metrics** select **Service Actions** then choose **Start**.

   Updated Ambari Metrics Sink jars will be installed on all hosts and you must restart each service to pick up the latest sink implementations.

   Please wait to restart all services until after you have completed all applicable post-upgrade tasks, for example: HDFS, YARN, Kafka, HBase, Flume, Storm.

Next Steps

- Restart services, <u>only</u> after you complete all applicable, post-upgrade tasks.

  Note

  New Ambari Metrics Sinks will not be activated until all services are restarted.

- If you are upgrading from IOP 4.2.0 or earlier, your Ambari Metrics service does not contain Grafana. Proceed to add Grafana to Ambari Metrics.

More Information

Add Grafana to Ambari Metrics

5.3.2. Adding Grafana to Ambari Metrics

As of Ambari 2.4, Grafana is included as a component of Ambari Metrics. This section walks you through how to add Grafana to the system and install Grafana on a host in the cluster.

Note

When using the API commands below, be sure to replace the **ambari.server** with the Ambari Server hostname, **cluster.name** with your cluster name and **host.name** with the host where you will run Grafana. This can be the same host that is running the Metrics Colllector.

1. Upgrade to Ambari 2.6.0 and perform needed post-upgrade checks. Make sure all services are up and healthy.

2. Add the METRICS_GRAFANA component to Ambari:

```
curl -u admin:admin -H "X-Requested-By:ambari" -i -X POST
```

```
http://ambari.server:8080/api/v1/clusters/cluster.name/services/
AMBARI_METRICS/components/METRICS_GRAFANA
```

3. Add METRICS_GRAFANA to a host in the cluster.

```
curl -u admin:admin -H "X-Requested-By:ambari" -i -X POST -d
'{"host_components":[{"HostRoles":{"component_name":"METRICS_GRAFANA"}}]}'
http://ambari.server:8080/api/v1/clusters/cluster.name/hosts?Hosts/
host_name=host.name
```

4. From **Ambari Web**, browse to **Services > Ambari Metrics** and you will see Grafana is in the **Install Pending...** state.

   You need to complete the configuration of Grafana before installing and starting.

5. To complete the configuration, click on **Services > Ambari Metrics > Configs** and enter the default Grafana Admin Password in the **General** section. Click **Save**.

6. Browse to **Hosts > host.name** (the **host.name** used in the API call where you added Grafana). You will see the Grafana component is in an **Install Pending...** state. Use the **Install Pending**... action button and select **Re-install**.

7. Once the install operation completes, select **Start** to start Grafana.

8. To access Grafana, browse to **Services > Ambari Metrics**, select **Quick Links** and then click **Grafana**.

   More Information

   Using Grafana

5.3.3. Upgrading Configurations

This section describes potential cluster configuration updates that may be required.

Upgrading Kerberos krb5.conf [16]

5.3.3.1. Upgrading Kerberos krb5.conf

Ambari has added support for handling more than one KDC host . Only one kadmin host is supported by the Kerberos infrastructure. This required modifications for the **krb5.conf** template. In order for Ambari to properly construct the krb5.conf configuration file, make the following configuration change if your cluster meets all of these criteria:

• Kerberos is enabled and Ambari is configured for automated setup, and

• Ambari is managing the krb5.conf, and

• You **have modified** the krb5.conf template content from the default content. If you have not modified the default content, Ambari will automatically update the template

content as part of upgrade and these configuration updates do not need to be applied manually.

If you meet all of the above criteria, you must update the **krb5.conf** template content found in **Services > Kerberos > Advanced**:

Original Template Entry	Updated Template Entry
admin_server = {{admin_server_host|default(kdc_host, True)}}	admin_server = {{admin_server_host|default(kdc_host_list[0]|trim(), True)}}
kdc = {{kdc_host}}	{% for kdc_host in kdc_host_list %} kdc = {{kdc_host|trim()}} {%- endfor -%}

More Information

Configure Ambari for Automated Setup

5.3.3.2. Switch from IBM Open JDK to Oracle JDK

If you are migrating from an IOP cluster with the IBM Open JDK on x86 hardware, we recommend that you modify your cluster to use the Oracle JDK (the default JDK for HDP clusters).

1. Verify current settings in the /etc/ambari-server/conf/ambari.properties file:

```
cat /etc/ambari-server/conf/ambari.properties | grep java
java.home=/usr/jdk64/java-1.8.0-openjdk-1.8.0.121-0.b13.el6_8.x86_64
java.releases=jdk1.8,jdk1.7
java.releases.ppc64le=
openjdk1.8.re=(java.*)/jre
```

2. Download and install Oracle JDK8 on all hosts in the cluster.

   ### Note

   For IOP clusters operating on IBM Power Systems, continue to use OpenJDK. You should:

   a. Upgrade OpenJDK to version 8

   b. Install the Java Cryptography Extension (JCE)

   For a Kerberized cluster, you should also:

   a. Regenerate Keytabs

   b. Restart Services

3. Modify the java.home variable in the /etc/ambari-server/conf/ ambari.properties file to point to the downloaded Oracle Open JDK home, example:

```
java.home=/usr/java/jdk1.8.0_112
```

4. Restart Ambari server.

```
ambari-server restart
```

When all services are re-started as a part of the Express Upgrade, the Oracle JDK will be used.

5.3.4. Disable Audit to Solr

If you are migrating from IOP 4.2.5 cluster with Ranger and Solr, make sure to:

1. Disable Audit to Solr, on the Ranger Service Configs page, as shown in the following example:

To allow for Ranger/Solr plugin migration.

2. Then, perform the restart required operation for this change to take effect.

6. Clean Up IBM IOP value-add services

Please see the IBM Knowledge Center topic:

Cleaning Up IBM IOP value-add services.

7. Removing Deprecated Components and Services

Please see the IBM Knowledge Center topic:

Removing deprecated components and services.

8. Upgrade Stack

This document describes the procedure for upgrading a Ambari-managed IOP cluster to an Ambari-managed HDP cluster. The source IOP version could be IOP-4.2.0 or IOP-4.2.5. The target HDP version is HDP-2.6.3. The target Ambari version used for the upgrade is 2.6.0. Only the Express upgrade method is supported.

8.1. Prepare to Upgrade the Stack

Make sure that you have review and completed all prerequisites described in previous sections.

You must perform backups of your databases before beginning this upgrade.

• Ambari database

• Hive Metastore database

• Oozie Server database

• Ranger Admin database

• Ranger Audit database

> **Important**
>
> If you use MySQL 5.6, you must use the Inno DB engine.
>
> If your current MySQL engine is MyISAM, you must migrate to Inno DB before upgrading to Ambari 2.5.
>
> • Export your current, MySQL (MyISAM) data.
>
> • Drop the MySQL (MyISAM) database schema.
>
> • Create a MySQL (Inno DB) database.
>
> • Import your data into the MySQL (Inno DB) database instance.
>
> For example:

```
mysqldump -U ambari -p ambari > /tmp/ambari.original.mysql

cp /tmp/ambari.original.mysql /tmp/ambari.innodb.mysql

sed -ie 's/MyISAM/INNODB/g' /tmp/ambari.innodb.mysql

mysql -u ambari -p ambari

DROP DATABASE ambari;

CREATE DATABASE ambari;

mysql -u ambari "-pbigdata" --force ambari < /tmp/ambari.innodb.
mysql
```

Please contact Hortonworks customer support if you have issues running your cluster using MySQL 5.6.

Turn off Service Auto-Start, as follows:

In Ambari Web UI, go to **Admin > Service Auto Start**, toggle off the Service Auto Start feature and save the change.

Checkpoint HDFS

1. Perform the following steps on the NameNode host. If you are configured for NameNode HA, perform the following steps on the Active NameNode. You can locate the Active NameNode from **Ambari Web > Services > HDFS** in the **Summary** area.

2. Check the NameNode directory to ensure that there is no snapshot of any prior HDFS upgrade. Specifically, using Ambari Web, browse to **Services > HDFS > Configs**, and examine the dfs.namenode.name.dir in the NameNode Directories property. Make sure that only a `/current` directory and no `/previous` directory exists on the NameNode host.

3. Create the following log and other files. Creating these logs allows you to check the integrity of the file system after the Stack upgrade.

   As the HDFS user,

   ```
   "su -l <HDFS_USER>"
   ```

   run the following (where <HDFS_USER> is the HDFS Service user, for example, hdfs):

   • Run fsck with the following flags and send the results to a log. The resulting file contains a complete block map of the file system. You use this log later to confirm the upgrade.

     ```
     hdfs fsck / -files -blocks -locations > dfs-old-fsck-1.log
     ```

   • Create a list of all the DataNodes in the cluster.

     ```
     hdfs dfsadmin -report > dfs-old-report-1.log
     ```

   • **Optional**: Capture the complete namespace of the file system. The following command does a recursive listing of the root file system:

     ```
     hdfs dfs -ls -R / > dfs-old-lsr-1.log
     ```

   • **Optional**: Copy all unrecoverable data stored in HDFS to a local file system or to a backup instance of HDFS.

4. Save the namespace. As the HDFS user, " su -l <HDFS_USER> ", you must put the cluster in Safe Mode.

   ```
   hdfs dfsadmin -safemode enter
   ```

   ```
   hdfs dfsadmin -saveNamespace
   ```

> **Note**
>
> In a highly-available NameNode configuration, the command
>
> ```
> hdfs dfsadmin -saveNamespace
> ```
>
> sets a checkpoint in the first NameNode specified in the configuration, in `dfs.ha.namenodes.[nameserviceID]`.
>
> You can also use the
>
> ```
> dfsadmin -fs
> ```
>
> option to specify which NameNode to connect. For example, to force a checkpoint in NameNode2:
>
> ```
> hdfs dfsadmin -fs hdfs://namenode2-hostname:namenode2-port -
> saveNamespace
> ```

5. Copy the checkpoint files located in `${dfs.namenode.name.dir}/current` into a backup directory.

   > **Note**
   >
   > In a highly-available NameNode configuration, the location of the checkpoint depends on where the
   >
   > ```
   > saveNamespace
   > ```
   >
   > command is sent, as defined in the preceding step.

6. Store the layoutVersion for the NameNode located at

   `${dfs.namenode.name.dir}/current/VERSION`, into a backup directory where

   `${dfs.namenode.name.dir}` is the value of the config parameter NameNode directories.

   This file will be used later to verify that the layout version is upgraded.

7. As the HDFS user, " su -l <HDFS_USER> ", take the NameNode out of Safe Mode.

   ```
   hdfs dfsadmin -safemode leave
   ```

8. Finalize any prior HDFS upgrade, if you have not done so already. As the HDFS user, " su -l <HDFS_USER> ", run the following:

   ```
   hdfs dfsadmin -finalizeUpgrade
   ```

   > **Important**
   >
   > In HDP-2.5, Audit to DB is no longer supported in Apache Ranger. If you are upgrading from HDP-2.3 or HDP-2.4 to HDP-2.5, you should use Apache Solr for Ranger audits. You should also migrate your audit logs from DB to Solr.

Next Steps

More Information

8.2. Register and Install Target Version

This section describes the steps to register the software repositories with Ambari for the new target version (i.e. the version you will be upgrading to) and how to install the software on all hosts before performing the upgrade.

Register Target Version

Steps

1. Log in to Ambari.

2. Browse to **Admin > Stack and Versions**.

3. Click the **Versions** tab. You see the IOP version currently running (either 4.2.0 or 4.2.5), marked as **Current**.

4. Click **Manage Versions**.

5. Proceed to register a new version by clicking **+ Register Version**.

6. Select the software version and method of delivery for your cluster.

   a. **Choose HDP Stack**.

   The available HDP 2.6.3 version appears in a tab. When you select the tab, Ambari attempts to discover the specific version of HDP Stack available. That list is shown in a drop-down. For that specific version, the available Services are displayed, with their Versions shown in the table.

   b. **Choose HDP Version**.

   If Ambari has access to the Internet, the specific Versions will be listed as options in the drop-down. If you have a Version Definition File for a version that is not listed, you can click **Add Version...** and upload the VDF file. In addition, a **Default Version Definition** is also included in the list if you do not have Internet access or are not sure which specific version to install. If you choose the **Default Version Definition**, you must enter a "two-digit Version Number" in the **Name** input field.

    c. **Choose Repository Delivery Method**.

    Using a Public Repository requires Internet connectivity. Using a Local Repository requires you have configured the software in a repository available in your network. To use the public software repositories, see the list of available HDP Repositories for each OS. Or, if you are using a local repository, enter the Base URLs for the local repository you have created.

7. Click **Save**.

8. Click **Go To Dashboard**, and browse back to **Stack and Versions > Versions**.

9. You will see the IOP version currently running, marked **Current**, and the version you just registered **HDP-2.6.3.0** displaying an **Install** button.

Next Steps

Install Target Version

Steps

1. Log in to Ambari.

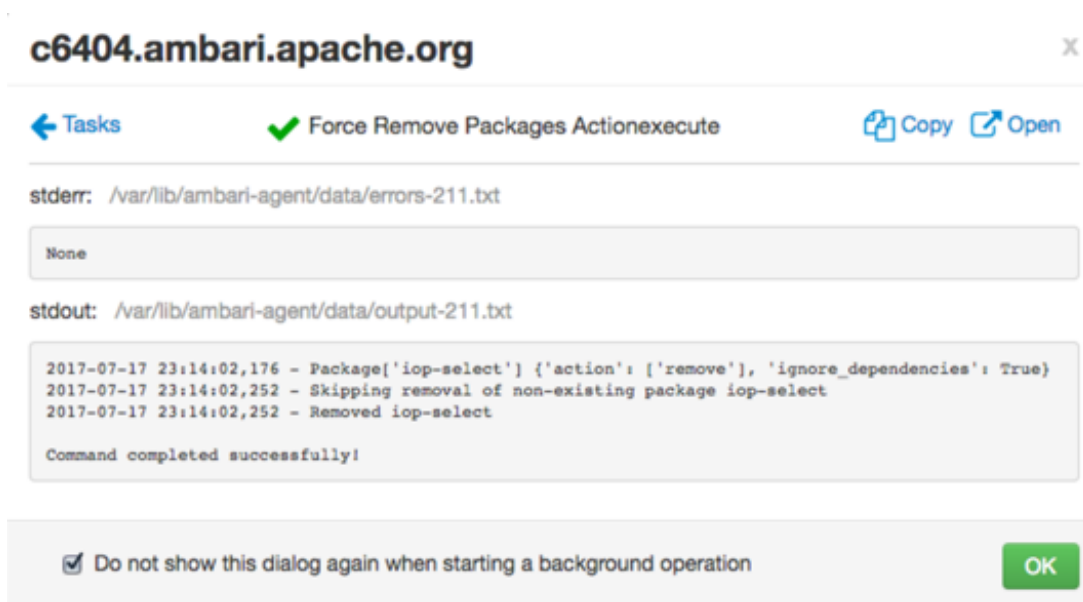2. Browse to **Admin > Stack and Versions**.

3. Click the **Versions** tab.

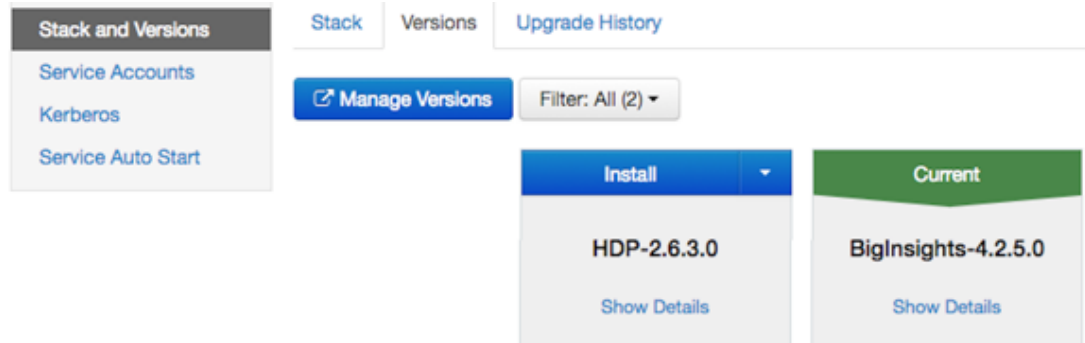4. In the HDP-2.6.3.0 Install drop-down, choose Remove IOP select, as shown in the following figure.
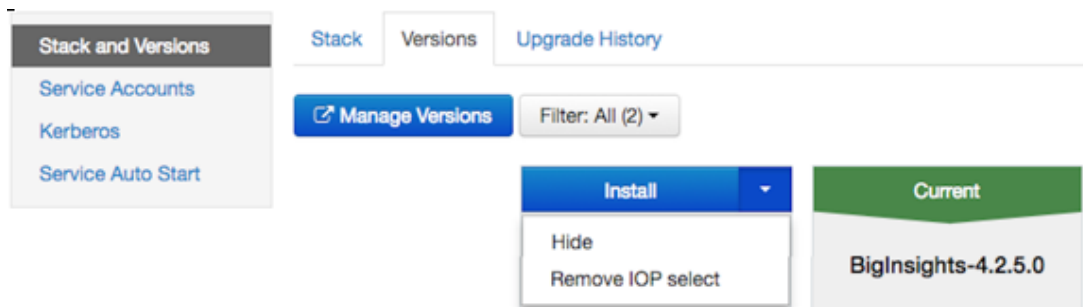
5. In **Backgroud Ops,**click the Remove IOP Select operation.
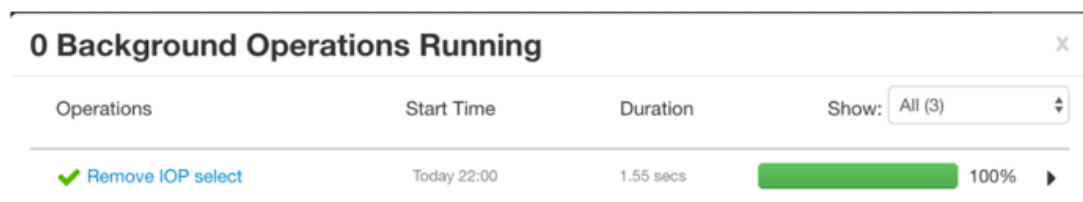
6. Verify that removal of IOP Select completed successfully, then click **OK**.

7. On the **Versions** tab, in HDP-2.6.3.0, click **Install**.

8. In **Confirmation**, click **Skip Dependency Check** and then click **OK**. to confirm.

⚠️ **Important**

For GA: the HDP version is: HDP-2.6.3.0-xxx. The following is an example only.

The Install version operation starts. This installs the target version on all hosts in the cluster. You can monitor the progress of the install by clicking the **Installing** link.

📝 **Note**

If you have Apache Ranger on your cluster, confirm that the `ambari-infra-solr-client` package is installed on the Ranger Admin host. The **Ambari Infra** service contains the default Solr instance used by Apache Ranger.

When the install completes, the **Upgrade** button replaces the **Install** button.

Next Steps

Start Express Upgrade.

More Information

Managing Versions

Using a Local Repository

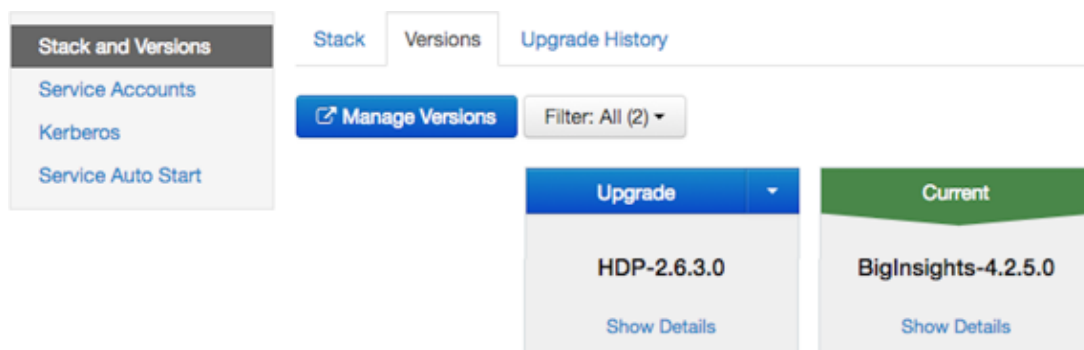# 8.3. Start Express Upgrade

To upgrade your IOP stack to HDP-2.6.3:

Steps

1. Log in to Ambari.

2. Browse to **Admin > Stack and Versions**.

3. Click the **Versions** tab.

   The registered and installed target HDP version displays.

4. Click **Upgrade** on the target version.

Based on your current IOP version and the HDP target version, Ambari performs a set of prerequisite checks to determine whether the Express Upgrade meets all the necessary criteria to be started.

5. Select the **Express Upgrade** method.

Proceed with Express Upgrade. the Skip All Service Check failures and Skip all Slave Component failures is optional.

6. Click **Proceed**.

7. In Confirmation, click Yes.

After removing IOP select, you may see warnings, which acknowledge that IOP select is removed, when the Express Upgrade process starts.

Warnings

Correcting the warnings is not required but is **recommended**.

```
CRITICAL: Ambari Agent Distro/Conf Select Versions: natr66-kods-iop420tofnsec-r6-1.openstacklocal
CRITICAL: Ambari Agent Distro/Conf Select Versions: natr66-kods-iop420tofnsec-r6-2.openstacklocal
CRITICAL: Ambari Agent Distro/Conf Select Versions: natr66-kods-iop420tofnsec-r6-4.openstacklocal
CRITICAL: Ambari Agent Distro/Conf Select Versions: natr66-kods-iop420tofnsec-r6-5.openstacklocal
CRITICAL: Ambari Agent Distro/Conf Select Versions: natr66-kods-iop420tofnsec-r6-6.openstacklocal
CRITICAL: Ambari Agent Distro/Conf Select Versions: natr66-kods-iop420tofnsec-r6-7.openstacklocal
CRITICAL: Ambari Agent Distro/Conf Select Versions: natr66-kods-iop420tofnsec-r6-8.openstacklocal
CRITICAL: Ambari Agent Distro/Conf Select Versions: natr66-kods-iop420tofnsec-r6-9.openstacklocal
CRITICAL: Ambari Agent Distro/Conf Select Versions: natr66-kods-iop420tofnsec-r6-10.openstacklocal
CRITICAL: Ambari Agent Distro/Conf Select Versions: natr66-kods-iop420tofnsec-r6-11.openstacklocal
Failed on: ambarihcube
```

Next Steps

Complete Express Upgrade.

8.4. Complete Express Upgrade

1. Ambari checks that your cluster meets prerequisites. A dialog displays the results:

   a. If any *required* prerequisites are not met, the result displays an error.

      You cannot proceed with the upgrade until you make the appropriate corrections and return to Perform Upgrade again.

   b. If any *optional* prerequisites are not met, the result displays a warning.

      You may proceed with the upgrade.

   c. Ambari displays a list of configuration changes that occur during the upgrade.

2. When the prerequisite checks complete, the upgrade starts. The time required to perform the upgrade depends on many factors. As part of the upgrade process, each component in the cluster restarts in a serial fashion. The stop/start times contribute to the total upgrade time.
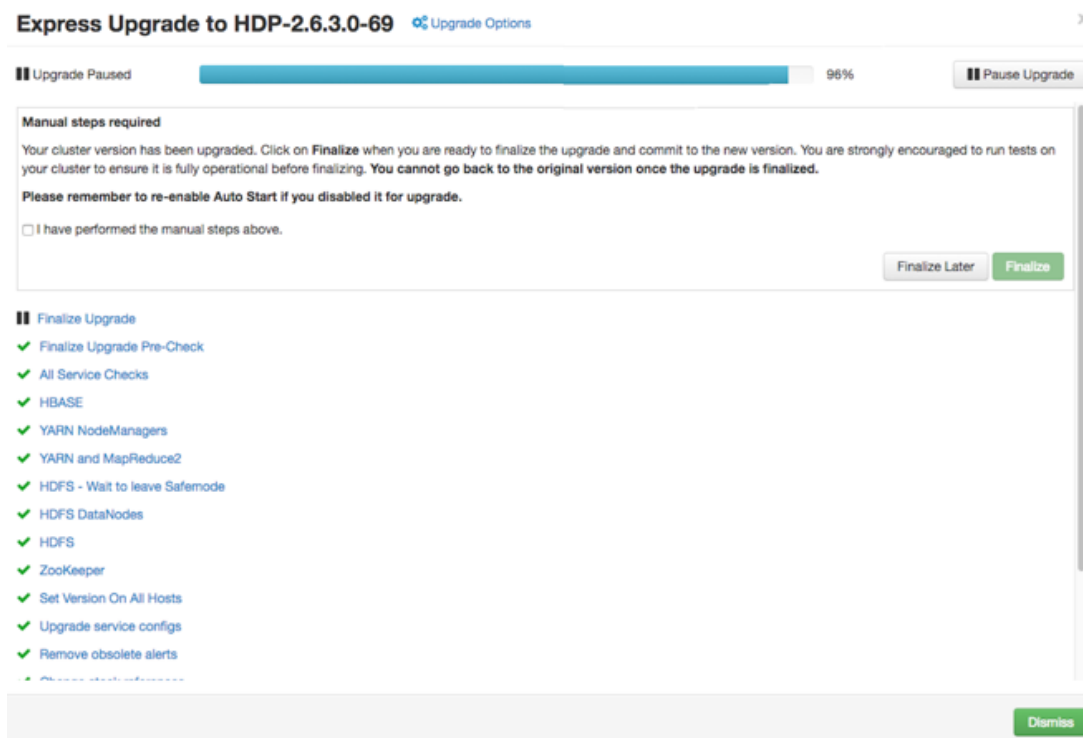
3. The upgrade process includes the following stages. Some stages require that you complete an action during normal operation.

If any stage fails, the upgrade stops and prompts you for action.

Stage	Description	Action Required
Prepare Upgrade	You should stop all YARN queues, all long-running applications on Slider, and deactivate & kill all running Storm topologies.	Perform the actions to prepare for the upgrade.
Stop Components for High-Level Services	This will stop all components for High-Level Services. This includes all master components **except** those of HDFS, HBase, ZooKeeper and Ranger.	None
Perform Backups	This step prompts you to confirm that you have taken proper backups before proceeding.	You must acknowledge the prompt for database backups.
Stop Components for Core Service	Stops all components with HDFS, HBase, ZooKeeper and Ranger.	None
Update Target Stack	Updates the stack version in Ambari to the target version. There is no downgrade past this point.	None
Update Service Configs	Updates (i.e. transfers or replaces) any configurations that are necessary for the upgrade.	None
Restart Components	Restarts all core components such as ZooKeeper, Ranger, HDFS, YARN, MapReduce2 and various Clients (Tez, Pig, Sqoop).	None
All Service Checks	All Service Checks are performed against the cluster.	Any service check that fails prompts you to **Ignore and Continue**, or **Retry**. If you selected the **Skip all Service Check**

Stage	Description	Action Required
		failures option, you will only be prompted when all Service Checks complete.
Restart Components	Restarts the remaining components such as Oozie, Falcon, Hive, Spark and others.	None
Set Version on All Hosts	Sets the HDP version on all hosts to the target HDP version.	None
Finalize Upgrade Pre-Check	Checks if any hosts were not upgraded, either because the host was in Maintenance Mode, or one or more components on the host failed to upgrade (and were skipped).	Click the list that displays **# hosts** for details on the hosts (and their components) that are not upgraded. You can **Pause Upgrade**, delete the hosts and return to finalize.
Finalize Upgrade	The component upgrades are complete. You are presented the option to Finalize, which when selected, completes the upgrade process + saves the cluster state.	Prompted to Finalize or Finalize Later.

4. When the upgrade stages complete, you may choose to **Finalize** the upgrade, to **Finalize Later** Finalizing later gives you a chance to perform more validation on the cluster. **You cannot downgrade back to the previous version**.

> **Note**
>
> If you choose to finalize later, both versions will be listed on the Stack and Versions tab with the starting version displaying as Current. It is not until you finalize that Ambari makes the target version the current version. Also, until you finalize, you will not be able to perform operational changes to the cluster (such as move components, change configurations, etc).

> **Important**
>
> Downgrade is not supported for an IOP-HDP migration.

5. Click **Finalize** to complete the express upgrade process.

Next Steps

Post-upgrade Tasks [32]

8.5. Post-upgrade Tasks

1. **Remove ambari-log4j** package that is no longer used from the ambari-server host.

```
yum remove ambari-log4j
```

2. **Restart Ambari server after the Express Upgrade has been finalized and migration is complete.**

```
ambari-server restart
```

3. **Upgrade bigtop-tomcat for Oozie**

> ⚠ **Important**
>
> If upgrading from IOP- 4.2.0, Oozie start will fail due to the incompatible C version 6.0.36-3.el6 installed from the IOP-4.2.0 repo.
>
> You must upgrade bigtop-tomcat to version 6.0.48-1 provided by HDP-2.6.3 repo.
>
> Perform upgrade of bigtop-tomcat on all hosts where Oozie server is installed.
>
> ```
> yum upgrade bigtop-tomcat
> ```

4. **Regenerate Keytabs for All Services**

> ⚠ **Important**
>
> *If the cluster is kerberized*, regenerate keytabs for all services after the upgrade.

5. **(Optional) Add Grafana**

   If upgrading from IOP- 4.2.0, and if you did not add Grafana after upgrading Ambari, Grafana does not exist as a component in your cluster.

   You can add Grafana using the following set of instructions:

   Add Grafana to Ambari Metrics

8.6. Solr Migration

After HDP 2.6 Upgrade completes successfully, there are 2 options to consider:

1. IOP SOLR is being actively used by the user outside of the use by Ranger and thereby SOLR has custom collections that need to be migrated. This requires installation of the IOP SOLR management pack after the upgrade.

> 📝 **Note**
>
> If Solr was used for Ranger/Atlas as well, you can go ahead with the mpack install or use option 2.

2. There are no custom collections and is it ok to lose Ranger audit data (if Solr was used with Ranger before the upgrade). In that case, installing the IOP SOLR management pack is not required. After upgrading to HDP 2.6.3, the only required step is to use the **Add Service** wizard to install **Ambari Infra** service.

If Option 1 above is true, install the IOP SOLR using management pack using the following instructions:

1. Download the IOP Solr Management Pack.

```
http://s3.amazonaws.com/dev.hortonworks.com/IOPSOLR/centos7/1.x/BUILDS/1.0.
0.0-24/tars/iopsolr/iop-solr-mpack-6.3.0.1.0.0.0-24.tar.gz
```

2. Stop ambari server.

```
ambari-server stop
```

3. Install the IOP Solr Management Pack.

```
ambari-server install-mpack --mpack=/PATH/TO/iop-solr-mpack-VERSION.tar.gz
```

   Replace the path with the absolute path to the IOP Solr Management Pack.

4. Restart ambari-server.

```
ambari-server restart
```

5. If you removed Solr before and created a backup (see: Backup SOLR), use
   migration_helper.py script to apply the old configurations. Before running the migration
   helper script, it's required to create a json file for the secrets (usually keystore/truststore
   passwords). (as we cannot get those values through the rest api), you can find a sample
   in the utilities directory (after you untar the mpack tar.gz file),

   • http://public-repo-1.hortonworks.com/IOPSOLR/centos6/1.x/updates/1.0.0.0/
     IOPSOLR-1.0.0.0-centos6-rpm.tar.gz

   • http://public-repo-1.hortonworks.com/IOPSOLR/centos6/1.x/updates/1.0.0.0/
     IOPSOLR-1.0.0.0-centos6-tars-tarball.tar.gz

   • http://public-repo-1.hortonworks.com/IOPSOLR/centos7/1.x/updates/1.0.0.0/
     IOPSOLR-1.0.0.0-centos7-rpm.tar.gz

   • http://public-repo-1.hortonworks.com/IOPSOLR/centos7/1.x/updates/1.0.0.0/
     IOPSOLR-1.0.0.0-centos7-tars-tarball.tar.gz

   If you did not use SSL for Solr before the upgrade, you can use the default values, also
   the same is true for the other ranger properties if you did not used ranger before. Then
   you can run the script:

```
/usr/bin/python migration_helper.py --action configure
--backup-location /tmp/solr-backup --ranger-solr-secrets=secret_defaults.
json
--cluster <cluster-name> -H <ambari-server-host> --username
<ambari-admin-username> --password <ambari-admin-password> --port
<ambari-server-port>
# use -s if ambari uses ssl
```

6. If the last command finished successfully, you can install IOP Solr Service with the
   migration_helper.py:

```
/usr/bin/python migration_helper.py --action install --cluster <cluster-
name> -H
<ambari-server-host> --username <ambari-admin-username> --password <ambari-
admin-password>
--port <ambari-server-port>
# use -s if ambari uses ssl
```

7. You have 2 options here:

- You can start your Solr service using Ambari Web UI (you can check the configuration on the UI before starting the Solr instances)

- You can use the migration_helper.py script to start the Solr service; just make sure the install has already finished:

```
/usr/bin/python migration_helper.py --action start --cluster <cluster-
name> -H
<ambari-server-host> --username <ambari-admin-username> --password
 <ambari-admin-password>
--port <ambari-server-port>
# use -s if ambari uses ssl
```

> **Note**
>
> - if something goes wrong, you can remove the service with –action stop and –action remove commands (use the same way as –action install or –action start)
>
> - The –action configure command will generate json files in the backup folder for every config type (with that name pattern: config-SOLR-<config_type>.json), those will be used as inputs to create the new configs and the host components.

8. Post Upgrade steps to enable audits to solr in Ranger using IOP-mpack:

- Change the following configurations from Ranger Audit Tab in Audit to Solr section:

   a. Enable Audit to Solr (xasecure.audit.destination.solr/ranger-env)

   b. Enable SolrCloud (is_solrCloud_enabled/ranger/env)

   c. Enable External SolrCloud(is_external_solrCloud_enabled/ranger-env)

   d. **For secure cluster only** Enable External SolrCloud kerberos (is_external_solrCloud_kerberos/ranger-env)

- When External SolrCloud is enabled, Ambari will recommend to use /ranger_audits znode, change it to the znode setup by IOP-SOLR mpack (default znode is /solr) in ranger.audit.solr.zookeepers config property.

Audit to Solr

Audit to Solr

ON

SolrCloud

ON

External SolrCloud

ON

External SolrCloud kerberos

ON
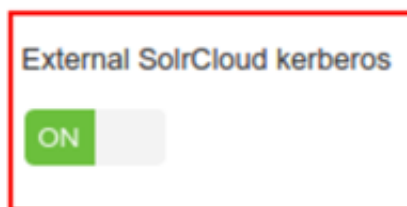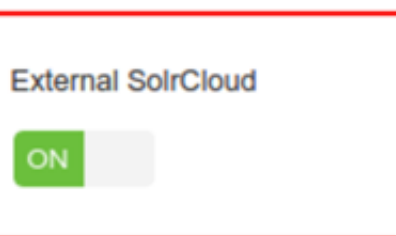
Enable this only for
secure cluster

ranger.audit.solr.zookeepers

c6401.ambari.apache.org:2181/solr
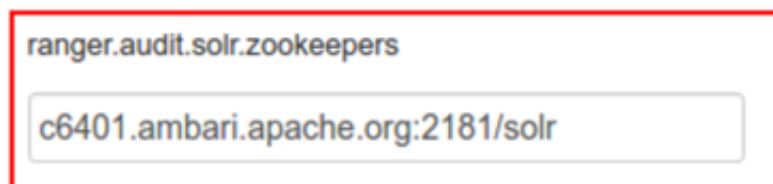
9. Ranger Solr plug-in migration steps:

   Verify that the `solr-solrj-5.5.x jar` is available in the follwing path:

   ```
   /usr/iopsolr/current/iop-solr/server/solr-webapp/webapp/WEB-INF/
   lib/ranger-solr-plugin-impl/.
   ```

   If not, copy it from `/usr/hdp/current/ranger-admin/ews/webapp/WEB-INF/
   lib/`.

10. Different migration steps are appropriate for each of the two, following existing setups.
    Perform the step(s) appropriate for your environment.

Existing setup	Migration steps	Notes
IOP 4.2.0 Ranger audits in DB	Add Ambari Infra for storing audits.	For Secure cluster: After upgrade to HDP-2.6.3, you must follow the steps for Ranger with Kerberos, in Post-upgrade Tasks for Ranger with Kerberos.
IOP 4.2.5 Ranger audits in Ambari Infra	You must upgrade Ambari Infra. No further actions required	After stack migration.

8.7. Adding SmartSense

After upgrading Ambari in an IOP-HDP migration, you must add the SmartSense service, using the Ambari Web UI.

More Information

Adding SmartSense

8.8. Troubleshooting Migration Issues

_____

Issue: HDFS restart fails after Ambari upgrade on a NameNode HA-enabled, IOP cluster.

Cause: The package hadoop-hdfs-zkfc is not supported by this version of the stack-select tool.

```
Error: ZooKeeperFailoverController restart fails with the following errors
 shown in the Ambari operation log about hadoop-hdfs-zkfc is not a supported
 package.
File "/usr/lib/python2.6/site-packages/resource_management/libraries/
functions/stack_select.py",
 line 109, in get_package_name package =
 get_packages(PACKAGE_SCOPE_STACK_SELECT, service_name, component_name)
File "/usr/lib/python2.6/site-packages/resource_management/libraries/
functions/stack_select.py",
 line 234, in get_packages raise Fail("The package {0} is not supported by
 this version of the stack-select tool.".format(package))
resource_management.core.exceptions.Fail: The package hadoop-hdfs-zkfc is not
 supported by this version of the stack-select tool.
```

Resolution:

On each ZKFailoverController:

1. Edit `/usr/bin/iop-select.` Insert `"hadoop-hdfs-zkfc": "hadoop-hdfs",` after line 33.

    > **Note**
    >
    > the "," is also part of the insert

2. Run one the following command to create the symbolic link.

If your cluster is IOP 4.2.5, run the following command:

```
ln -sfn /usr/iop/4.2.5.0-0000/hadoop-hdfs /usr/iop/current/hadoop-hdfs-zkfc
```

If your cluster is IOP 4.2.0, run the following command:

```
ln -sfn /usr/iop/4.2.0.0/hadoop-hdfs /usr/iop/current/hadoop-hdfs-zkfc
```

3. Restart ZKFailoverController via Ambari web UI.

_____

Issue: UI does not come up after migration to HDP

Error in logs:

```
Ambari-server log: Caused by: java.lang.RuntimeException: Trying to create a
ServiceComponent not recognized in stack info, clusterName=c1, serviceName=
HBASE,
componentName=HBASE_REST_SERVER, stackInfo=HDP-2.6
```

Cause: HBASE_REST_SERVER component was not deleted before migration, as described here.

Resolution: Delete the HBASE_REST_SERVER component from the Ambari Server database, using the following steps:

```
DELETE FROM hostcomponentstate
WHERE service_name = 'HBASE'
  AND component_name = 'HBASE_REST_SERVER';

DELETE FROM hostcomponentdesiredstate
WHERE service_name = 'HBASE'
  AND component_name = 'HBASE_REST_SERVER';

DELETE FROM servicecomponent_version
WHERE component_id IN (SELECT
    id
  FROM servicecomponentdesiredstate
  WHERE service_name = 'HBASE'
  AND component_name = 'HBASE_REST_SERVER');

DELETE FROM servicecomponentdesiredstate
WHERE service_name = 'HBASE'
  AND component_name = 'HBASE_REST_SERVER';
```

_____

Issue: Ambari Metrics does not work after migration

Error in logs:

```
2017-08-03 22:07:25,524 INFO org.apache.zookeeper.ClientCnxn: Opening socket
 connection to server eyang-2.openstacklocal/172.26.111.18:61181. Will not
 attempt to authenticate using SASL (unknown error)
2017-08-03 22:07:25,524 WARN org.apache.zookeeper.ClientCnxn: Session
 0x15daa1cd6780004 for server null, unexpected error, closing socket
 connection and attempting reconnect
```

```
java.net.ConnectException: Connection refused
 at sun.nio.ch.SocketChannelImpl.checkConnect(Native Method)
 at sun.nio.ch.SocketChannelImpl.finishConnect(SocketChannelImpl.java:717)
 at org.apache.zookeeper.ClientCnxnSocketNIO.doTransport(ClientCnxnSocketNIO.
java:361)
 at org.apache.zookeeper.ClientCnxn$SendThread.run(ClientCnxn.java:1081)
```

Cause: Ambari Metrics data is incompatible between IOP and HDP release.

Resolution: If AMS mode is embedded remove the transient ZooKeeper data on the collector node:

```
rm -rf /var/lib/ambari-metrics-collector/hbase-tmp/zookeeper/zookeeper_0/
version-2/*
```

_____

Issue: Spark History server failed to start and displays the following message:

```
raise ExecutionFailed(err_msg, code, out, err)
resource_management.core.exceptions.ExecutionFailed: Execution of
'/usr/bin/kinit -kt /etc/security/keytabs/spark.headless.keytab
spark-hey@IBM.COM; ' returned 1. kinit: Key table file
'/etc/security/keytabs/spark.headless.keytab' not found while getting initial
credentials
```

If Kerberos is enabled, AND both SPARK and SPARK2 are installed AND SPARK and SPARK2 have the same service user names, then make sure the following properties are the same:

Config Type	Property	Value
spark-defaults	spark.history.kerberos.keytab	/etc/security/keytabs/ spark2.headless.keytab
spark-defaults	spark.history.kerberos.keytab	/etc/security/keytabs/ spark2.headless.keytab

This can be addressed during add service wizard as well by setting the correct value on the Configure Identities section:

Spark user keytab	${keytab_dir}/spark2.headless.keytab
Spark2 user keytab	${keytab_dir}/spark2.headless.keytab

_____

Issue: Oozie start fails after upgrade from IOP 4.2.0.

If oozie server fails with following message in `catalina.out` log file:

Error in logs:

```
org.apache.jasper.compiler.JDTCompiler$1 findType
SEVERE: Compilation error
org.eclipse.jdt.internal.compiler.classfmt.ClassFormatException
   at org.eclipse.jdt.internal.compiler.classfmt.ClassFileReader.
<init>(ClassFileReader.java:372)
   at org.apache.jasper.compiler.JDTCompiler$1.findType(JDTCompiler.java:206)
```

Cause: This is because the IOP bigtop-tomcat version is older than what is required by Oozie in HDP 2.6.3. The Express Upgrade process did not upgrade it to the specific minor version required because it is not possible to do a side-by-side install of this dependency.

Resolution:

Fix this problem by upgrading to big-tomcat 6.0.48-1

```
yum upgrade bigtop-tomcat
```

_____

Issue: Ambari upgrade results in db consistency check warning message

After ambari server upgrade, it is quite likely that a warning message would be thrown during start operation regarding database consistency check, example:

```
2017-08-15 07:46:18,903  INFO - Checking for configs that are not mapped to
 any service
2017-08-15 07:46:18,964  WARN - You have config(s):
wd-hiveserver2-config-version1500918446970,spark-metrics-properties-
version1493664394129,spark
-javaopts-properties-version1493664394129,spark-env-version1493752183508 that
 is(are)
not mapped (in serviceconfigmapping table) to any service!
```

This indicates that there were service(s) that were deleted from Ambari and the orphaned config associations exist in the database. These do not affect the cluster operation and therefore are listed as Warnings.

This message can be safely ignored with `--skip-database-check` or you can clear these warnings by following the steps mentioned in the description of this Apache Jira:

https://issues.apache.org/jira/browse/AMBARI-20875

_____

Issue: Ambari server log shows consistency warning after service delete

After deleting a service and restarting ambari server, Ambari server log shows warning messages if there exists any ConfigGroup(s) for that service which are not deleted by Ambari.

```
ERROR [ambari-hearbeat-monitor] HostImpl:1085 - Config inconsistency exists:
 unknown configType=solr-site
```

These messages are benign and can be safely ignored. However, the problem of the logs filling up with these messages remains.

Cause: https://issues.apache.org/jira/browse/AMBARI-21784, can be used to track the fix for this issue.

Resolution: This warning can be fixed by deleting ConfigGroups that belong to the deleted services using Ambari API(s).

Get all ConfigGroups to delete for service (tag = service-name). This GET call can be performed from the browser:

```
http://<ambari-server-host>:<port>/api/v1/clusters/<cluster-name>/
config_groups?ConfigGroup/tag=<service-name>&fields=ConfigGroup/id
```

Delete the ConfigGroups using the following delete API call (Use the <id> of the
ConfigGroup(s) obtained from the previous call):

```
curl -u <ambari-admin-username>:<ambari-admin-password> -H "X-Requested-
By:ambari" -i
-X DELETE
http://<ambari-server-host>:<port>/api/v1/clusters/<cluster-name>/
config_groups/<id>
```

_____

Issue: Kerberos cluster - Hive service check failed post migration

This is not directly related to either the Ambari or the stack upgrade but is related to the
local accounts on the hosts of the cluster.

The Hive service check will fail with an impersonation issue if the local ambari-qa user is
not part of the expected group; which, by default is "users". The expected groups can be
seen by viewing the value of the core-site/hadoop.proxyuser.HTTP.groups in the HDFS
configurations or via Ambari's REST API.

Error in logs:

The error seen in the STDERR of the service check operation will be as follows:

```
resource_management.core.exceptions.ExecutionFailed:
 Execution of '/var/lib/ambari-agent/tmp/templetonSmoke.sh c6402.ambari.
apache.org ambari-qa 50111
 idtest.ambari-qa.1503419582.05.pig /etc/security/keytabs/smokeuser.headless.
keytab true
 /usr/bin/kinit ambari-qa-c1@EXAMPLE.COM /var/lib/ambari-agent/tmp' returned
 1. Templeton Smoke Test (ddl cmd): Failed.
 : {"error":"java.lang.reflect.UndeclaredThrowableException"}http_code <500>
```

Looking at the /var/log/hive/hivemetastore.log file, the following error can be seen:

```
2017-08-22 16:33:03,183 ERROR [pool-7-thread-54]: metastore.RetryingHMSHandler
(RetryingHMSHandler.java:invokeInternal(203)) - MetaException(message:User:
HTTP/c6402.ambari.apache.org@EXAMPLE.COM is not allowed to impersonate ambari-
qa)
```

Resolution:

To fix the issue, either:

• Add the ambari-qa user to an expected group

    Example: **usermod -a -G users ambari-qa**

    Example:

```
[root@c6402 hive]# groups ambari-qa
ambari-qa : hadoop
[root@c6402 hive]# usermod -a -G users ambari-qa
[root@c6402 hive]# groups ambari-qa
ambari-qa : hadoop users
```

OR

• Add one (or more) of the ambari-qa groups to the HDFS configuration at:

core-site/hadoop.proxyuser.HTTP.groups

_____

Issue: Only post-upgrade Spark jobs display in Spark History Server.

Cause: IOP's default `spark.eventLog.dir` has been changed to a custom value that includes the string `/iop/apps` but does not match the default value (/iop/apps/4.2.0.0/spark/logs/history-server).

An example custom value would be `/iop/apps/custom/dir`.In this case, that custom value will be changed to `/hdp/apps/custom/dir` during the stack upgrade.

Resolution:

To see pre-upgrade Spark jobs, change the setting back to the custom value established before upgrading, in this case: `/iop/apps/custom/dir`

_____

Issue: Oozie Hive job failed with NoClassDefFoundError

If Oozie HA is enabled and the Oozie Hive job fails with the following error in the yarn application log:

```
<<< Invocation of Main class completed <<<

Failing Oozie Launcher, Main class [org.apache.oozie.action.hadoop.HiveMain],
 main() threw exception, org/apache/hadoop/hive/shims/ShimLoader
java.lang.NoClassDefFoundError: org/apache/hadoop/hive/shims/ShimLoader
 at org.apache.hadoop.hive.conf.HiveConf$ConfVars.<clinit>(HiveConf.java:400)
 at org.apache.hadoop.hive.conf.HiveConf.<clinit>(HiveConf.java:109)
 at sun.misc.Unsafe.ensureClassInitialized(Native Method)
```

Resolution:

• Run the following command as the oozie user:

```
oozie admin -oozie http://<oozie-server-host>:11000/oozie -sharelibupdate
```

• Rerun the oozie hive job to verify.

_____

Issue: Hive Server Interactive (HSI) fails to start in a kerberized cluster.

HSI is not present in IOP clusters, adding Hive and/or enabling HSI after migrating a kerberized cluser to HDP may result in HSI failing to start.

Cause: HSI conditional service logic not present in IOP cluster results in Keytabs not created in YARN/kerberos.json and therefore keytabs do not exist on all NameNodes.

Resolution: Manually regenerate keytabs from Ambari before enabling HSI, so that keytabs are distributed across all Node Managers.

9. BigSQLv5 Upgrade

Please see the IBM Knowledge Center topic:

Upgrading and finalizing Big SQL.

10. DSM Upgrade

Please see the IBM Knowledge Center topic:

Upgrading IBM Data Server Manager (DSM).