Hortonworks Data Platform

Administering Ambari

(Apr 13, 2015)

docs.hortonworks.com

Hortonworks Data Platform: Administering Ambari

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.

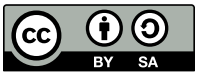


Table of Contents

1. Administering Ambari: Overview

Apache Ambari is a system to help you provision, manage and monitor Hadoop clusters. This guide is intended for Cluster Operators and System Administrators responsible for installing and maintaining Ambari and the Hadoop clusters managed by Ambari. Installing Ambari creates a default user with "Admin Admin" privilege, with the following username/password: `admin/admin`.

When you sign into Ambari as Ambari Admin, you can:

- Perform Ambari Admin Tasks
- Create and Manage a Cluster
- Manage Stack and Versions
- Manage Users and Groups
- Manage Views

For specific information about provisioning an HDP cluster, see Install, Configure, and Deploy an HDP Cluster.

1.1. Terms and Definitions

The following basic terms help describe the key concepts associated with Ambari Administration.

Term	Definition
Ambari Admin	Specific privilege granted to a user that enables the user to administer Ambari. The default user `admin` created by Ambari is flagged as an “Ambari Admin”. Users with the Ambari Admin privilege can grant, or revoke this privilege on other users.
Account	User name, password and privileges.
Cluster	Installation of a Hadoop cluster, based on a particular Stack, that is managed by Ambari.
Group	Unique group of users in Ambari.
Group Type	Local and LDAP. Local groups are maintained in the Ambari database. LDAP groups are imported (and synchronized) with an external LDAP (if configured).
Permissions	Represents the permission that can be granted to a principal (user or group) on a particular resource. For example, cluster resources support Operator and Read-Only permissions.
Principal	User or group that can be authenticated by Ambari.
Privilege	Represents the mapping of a principal to a permission and a resource. For example: the user `admin` is granted the permission Operator on cluster DevCluster.
Resource	Represents the resource available and managed in Ambari. Ambari supports two types of resources: cluster and view. An Ambari Admin assigns permissions for a resource for users and groups.
User	Unique user in Ambari.
User Type	Local and LDAP. Local users are maintained in the Ambari database and authentication is performed against the Ambari database. LDAP users are imported (and synchronized) with an external LDAP (if configured).
Version	Represents a Stack version, which includes a set of repositories to install that version on a cluster. For more information about Stack versions, see Manage Stack and Versions.
View	Defines a user interface component that is available to Ambari.

1.2. Logging in to Ambari

After installing Ambari, you can log in to Ambari as follows:

1. Enter the following URL in a web browser:

   `http://<your.ambari.server>:8080` where `<your.ambari.server>` is the hostname for your Ambari server machine and `8080` is the default HTTP port.

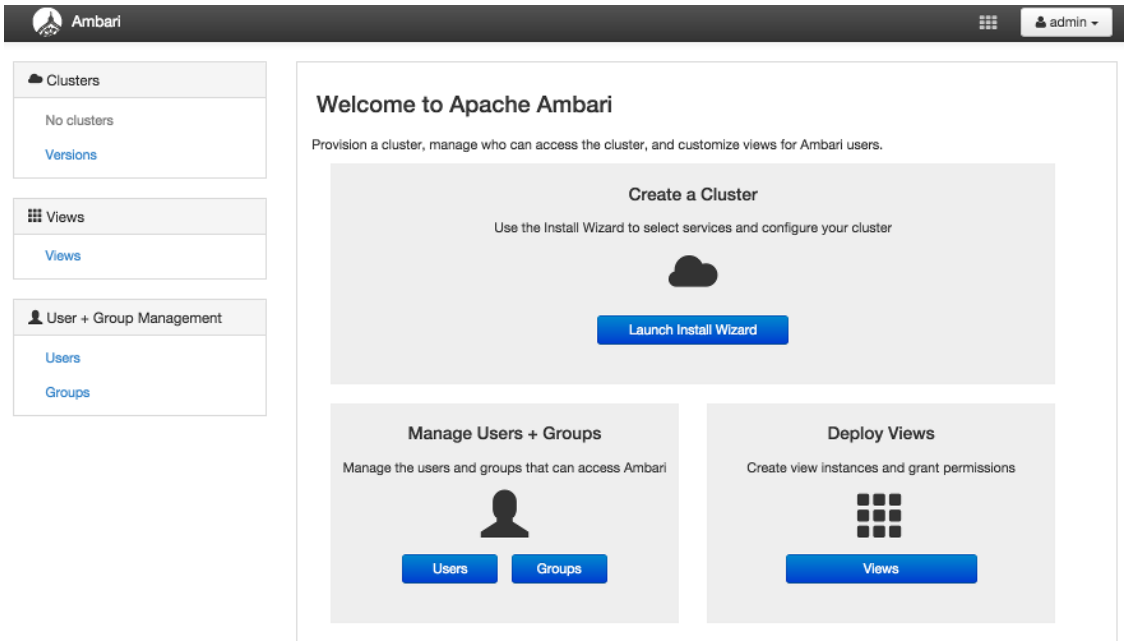
2. Enter the user account credentials for the default administrative user automatically created during install:

   `username/password = admin/admin`

3. The Ambari Administration web page displays. From this page you can Manage Users and Groups, Manage Views, Manage Stack and Versions, and Create a Cluster.

1.3. About the Ambari Administration Interface

When you log in to the Ambari Administration interface with "Ambari Admin" privilege, a landing page displays links to the operations available. Plus, the operations are available from the left menu for clusters, views, users, and groups.



- Clusters displays a link to a cluster (if created) and links to manage access permissions for that cluster. See Creating a Cluster for more information.
- User and Group Management provides the ability create and edit users and groups. See Managing Users and Groups for more information.
- Views lets you to create and edit instances of deployed Views and manage access permissions for those instances. See Managing Views for more information.

- Versions provides the ability to manage the Stack versions that are available for the clusters. See Managing Stack and Versions for more information.

2. Ambari Admin Tasks

An "Ambari Admin" has administrator (or super-user) privilege. When logged into Ambari with the "Ambari Admin" privilege, you can:

- Change the Administrator Account Password
- Create a cluster
- Set access permissions for an existing cluster
- Create, delete, and edit view instances
- Manage permissions for view instances
- Create, edit, and delete users and user groups

For more information about creating Ambari users locally and importing Ambari LDAP users, see Managing Users and Groups.

2.1. Changing the Administrator Account Password

During install and setup, the Cluster Installer wizard automatically creates a default user with "Ambari Admin" privilege. You can change the password for this user (or other Local users in the system) from the Ambari Administration interface. You can change the password for the default `admin` user to create a unique administrator credential for your system.

To change the password for the default `admin` account:

1. Browse to the Users section.
2. Select the `admin` user.
3. Click the Change Password button.
4. Enter the current `admin` password and the new password twice.
5. Click OK to save the new password.

2.2. Creating a Cluster

As an Ambari Admin, you can launch the Cluster Install Wizard and create a cluster. To create a cluster, from the Ambari Administration interface:

1. Click `Install Cluster`. The Cluster Install Wizard displays.
2. Follow the steps in the wizard to install your cluster.

For more information about prerequisites and system requirements, see Getting Ready.

2.3. Setting Cluster Permissions

After you create a cluster, users with Admin Admin privileges automatically get Operator permission on the cluster. By default, no users have access to the cluster. You can grant permissions on the cluster to other users and groups from the Ambari Administration interface.

Ambari manages the following permissions for a cluster: `Operator` and `Read-Only`. Users and Groups with `Operator` permission are granted access to the cluster. Operator permission provides full control of the following services:

- Start
- Stop
- Restart
- Add New

  And The Following Configurations:

- Modify
- Revert

Users and Groups with `Read-Only` permission can only view, not modify, services and configurations.

Users with Ambari Admin privileges are implicitly granted `Operator` permission. Plus, Ambari Admin users have access to the Ambari Administration interface which allows them to control permissions for the cluster.

To modify user and group permissions for a cluster:

1. As an Ambari Admin, access the Ambari Administration interface.
2. Click Permissions, displayed under the cluster name.
3. The form showing the permissions `Operator` and `Read-Only` with users and groups is displayed.
4. Modify the users and groups mapped to each permission and save.

For more information about managing users and groups, see Managing Users and Groups.

Warning

Assigning permissions to a group having *no* members is possible.

Note

Verify user permissions, group membership, and group permissions to ensure that each user and group has appropriate permissions.

2.4. Viewing the Cluster Dashboard

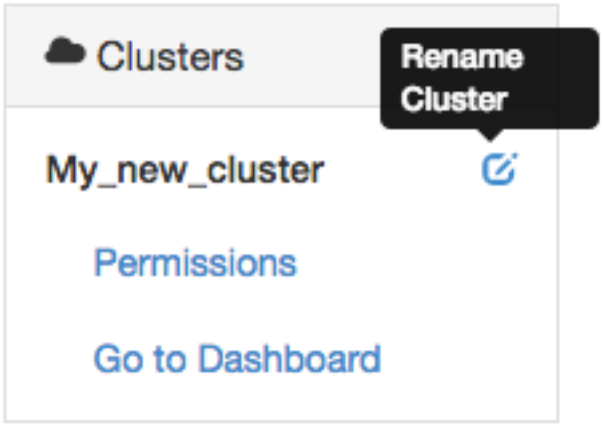
After you have created a cluster, select `Clusters > Go to Dashboard` to open the Dashboard view. For more information about using Ambari to monitor and manage your cluster, see Monitoring and Managing your HDP Cluster with Ambari.

2.5. Renaming a Cluster

A user with Admin Admin privileges can rename a cluster, using the Ambari Administration interface.

To rename a cluster:

1. In Clusters, click the Rename Cluster icon, next to the cluster name.



   The cluster name becomes write-able.

2. Enter alphanumeric characters as a cluster name.
3. Click the check mark.
4. Confirm.

3. Managing Users and Groups

An "Ambari Admin" can create and manage users and groups available to Ambari. An Ambari Admin can also import user and group information into Ambari from external LDAP systems. This section describes the specific tasks you perform when managing users and groups in Ambari.

- Local and LDAP User Types
- Ambari Admin Privileges
- Creating a Local User
- Setting User Status
- Setting the Ambari Admin Flag
- Changing the Password for a Local User
- Deleting a Local User
- Creating a Local Group
- Managing Group Membership
- Deleting a Local Group

3.1. Users and Groups Overview

Ambari supports two types of users and groups: Local and LDAP. The following topics describe how Ambari Administration supports managing Local and LDAP users and groups.

- Local and LDAP User and Group Types
- Ambari Admin Privileges

3.1.1. Local and LDAP User and Group Types

Local users are stored in and authenticate against the Ambari database. LDAP users have basic account information stored in the Ambari database. Unlike Local users, LDAP users authenticate against an external LDAP system.

Local groups are stored in the Ambari database. LDAP groups have basic information stored in the Ambari database, including group membership information. Unlike Local groups, LDAP groups are imported and synchronized from an external LDAP system.

To use LDAP users and groups with Ambari, you must configure Ambari to authenticate against an external LDAP system. For more information about running ambari-server setup-ldap, see Configure Ambari to use LDAP Server. A new Ambari user or group, created either locally or by synchronizing against LDAP, is granted no privileges by default. You, as an Ambari Admin, must explicitly grant each user permissions to access clusters or views.

3.1.2. Ambari Admin Privileges

As an Ambari Admin, you can create new users, delete users, change user passwords and edit user settings. You can control certain privileges for Local and LDAP users. The following table lists the privileges available and those not available to the Ambari Admin for Local and LDAP Ambari users.

Ambari Administrator Privileges for Ambari Local and LDAP Users

Administrator User Privilege	Local User	LDAP User
Change Password	Available	Not Available
Set Ambari Admin Flag	Available	Available
Change Group Membership	Available	Not Available
Delete User	Available	Not Available
Set Active / Inactive	Available	Available

3.2. Creating a Local User

To create a local user:

1. Browse to Users.
2. Click Create Local User.
3. Enter a unique user name.
4. Enter a password, then confirm that password.
5. Click Save.

3.3. Setting User Status

User status indicates whether the user is active and should be allowed to log into Ambari or should be inactive and denied the ability to log in. By setting the Status flag as Active or Inactive, you can effectively "disable" user account access to Ambari while preserving the user account information related to permissions.

To set user Status:

1. On the Ambari Administration interface, browse to Users.
2. Click the user name of the user to modify.
3. Click the Status control to toggle between Active or Inactive.
4. Choose OK to confirm the change. The change is saved immediately.

3.4. Setting the Ambari Admin Flag

You can elevate one or more users to have Ambari administrative privileges, by setting the Ambari Admin flag. You must be logged in as an account that is an Ambari Admin to set or remove the Ambari Admin flag.

To set the Ambari Admin Flag:

1. Browse to the Users section.
2. Click the user name you wish to modify.
3. Click on the Ambari Admin control.
4. Switch Yes to set, or No to remove the Admin flag.

Important

To prevent you from accidently locking yourself out of the Ambari Administration user interface, Ambari prevents setting the Ambari Admin flag for your own Ambari Admin account to No.

3.5. Changing the Password for a Local User

An Ambari Administrator can change local user passwords. LDAP passwords are not managed by Ambari since LDAP users authenticate to external LDAP. Therefore, LDAP user passwords cannot be changed from Ambari.

To change the password for a local user:

1. Browse to the user.
2. Click `Change password`.
3. Enter YOUR administrator password to confirm that you have privileges required to change a local user password.
4. Enter a password, then confirm that password.
5. Click Save.

3.6. Deleting a Local User

Deleting a local user removes the user account from the system, including all privileges associated with the user. You can reuse the name of a local user that has been deleted. To delete a local user:

1. Browse to the User.
2. Click `Delete User`.
3. Confirm.

Note

If you want to disable user log in, set the user Status to Inactive.

3.7. Creating a Local Group

To create a local group:

1. Browse to Groups.
2. Click Create Local Group.
3. Enter a unique group name.
4. Click Save.

3.8. Managing Group Membership

You can manage group membership of Local groups by adding or removing users from groups.

- Adding a User to a Group
- Modifying Group Membership

3.8.1. Adding a User to a Group

To add a user to group:

1. Browse to Groups.
2. Click a name in the Group Name list.
3. Choose the `Local Members` control to edit the member list.
4. In the empty space, type the first character in an existing user name.
5. From the list of available user names, choose a user name.
6. Click the check mark to save the current, displayed members as group members.

3.8.2. Modifying Group Membership

To modify Local group membership:

1. In the Ambari Administration interface, browse to Groups.
2. Click the name of the Group to modify.
3. Choose the `Local Members` control to edit the member list.
4. Click in the Local Members text area to modify the current membership.
5. Click the X to remove a user.
6. To save your changes, click the checkmark. To discard your changes, click the `x`.

3.9. Deleting a Local Group

Deleting a local group removes all privileges associated with the group. To delete a local group:

1. Browse to the Group.
2. Click `Delete Group`.
3. Confirm. The group is deleted and the associated group membership information is removed.

4. Managing Views

The Ambari Views Framework offers a systematic way to plug in UI capabilities to surface custom visualization, management and monitoring features in Ambari Web. The development and use of Views allows you to extend and customize Ambari Web to meet your specific needs.

A View extends Ambari to let third parties plug in new resource types along with APIs, providers, and UIs to support them. A View is deployed into the Ambari Server and Ambari Admins can create View instances and set the privileges on access to users and groups.

The following sections cover the basics of Views and how to deploy and manage View instances in Ambari:

4.1. Terminology

The following are Views terms and concepts you should be familiar with:

Term	Description
Views Framework	The core framework that is used to develop a View. This is very similar to a Java Web App.
View Definition	Describes the View resources and core View properties such as name, version and any necessary configuration properties. On deployment, the View definition is read by Ambari.
View Package	Packages the View client and server assets (and dependencies) into a bundle that is ready to deploy into Ambari.
View Deployment	Deploying a View into Ambari. This makes the View available to Ambari Admins for creating instances.
View Name	Unique identifier for a View. A View can have one or more versions of a View. The name is defined in the View Definition (created by the View Developer) that is built into the View Package.
View Version	Specific version of a View. Multiple versions of a View (uniquely identified by View name) can be deployed into Ambari.
View Instance	Instantiation of a specific View version. Instances are created and configured by Ambari Admins and must have a unique View instance name.
View Instance Name	Unique identifier of a specific instance of View.
Framework Services	View context, instance data, configuration properties and events are available from the Views Framework.

4.2. Basic Concepts

Views are basically Web applications that can be “plugged into” Ambari. Just like a typical web application, a View can include server-side resources and client-side assets. Server-side

resources, which are written in Java, can integrate with external systems (such as cluster services) and expose REST end-points that are used by the view. Client-side assets, such as HTML/JavaScript/CSS, provide the UI for the view that is rendered in the Ambari Web interface.

Ambari Views Framework Ambari exposes the Views Framework as the basis for View development. The Framework provides the following:

- Method for describing and packaging a View
- Method for deploying a View
- Framework services for a View to integrate with Ambari
- Method for managing View versions, instances, and permissions

The Views Framework is separate from Views themselves. The Framework is a core feature of Ambari and Views build on that Framework. Although Ambari does include some Views out-of-the-box, the feature of Ambari is the Framework to enable the development, deployment and creation of views.

The development and delivery of a View follows this process flow:

- Develop the View (similar to how you would build a Web application)
- Package the View (similar to a WAR)
- Deploy the View into Ambari (using the Ambari Administration interface)
- Create and configure instances of the View (performed by Ambari Admins)

Considering the above, it is important to understand the different personas involved. The following table describes the three personas:

Persona	Description
View Developer	Person who builds the front-end and back-end of a View and uses the Framework services available during development. The Developer created the View, resulting in a View Package that is delivered to an Ambari Admin.
Ambari Admin	Ambari user that has Ambari Admin privilege and uses the Views Management section of the Ambari Administration interface to create and managing instances of Views. Ambari Admin also deploys the View Packages delivered by the View Developer.
View User	Ambari user that has access to one or more Views in Ambari Web. Basically, this is the end user.

Important

This document covers the tasks related to an Ambari Admin using and making Views available to users in their Ambari deployment. This document does not cover View development and packaging. See Additional Information for more information on where to obtain information about developing Views.

4.2.1. Ambari Views Versions and Instances

After Views are developed, views are identified by unique a view name. Each View can have one or more View versions. Each View name + version combination is deployed as a

single View package. Once a View package is deployed, the Ambari Admin can create View instances, where each instance is identified by a unique View instance name. The Ambari Admin can then set access permissions for each View instance.

4.2.2. Deploying a View

Deploying a View involves obtaining the View Package and making the View available to the Ambari Server. Each View deployed has a unique name. Multiple versions of a View can be deployed at the same time. You can configure multiple versions of a View for your users, depending on their roles, and deploy these versions at the same time.

1. Obtain the View package. For example, `files-0.1.0.jar`.
2. On the Ambari Server host, browse to the views directory.

   ```
   cd /var/lib/ambari-server/resources/views
   ```

3. Copy the View package into place.
4. Restart Ambari Server.

   ```
   ambari-server restart
   ```

5. The View is extracted, registered with Ambari, and displays in the Ambari Administration interface as available to create instances.

Note

`/var/lib/ambari-server/resources/views` is the default directory into which Views are deployed. You can change the default location by editing the `views.dir` property in `ambari.properties`.

For more information about building Views, see the Apache Ambari Wiki page.

4.3. Creating View Instances

To create a View instance:

1. Browse to a View and expand.
2. Click the “Create Instance” button.
3. Provide the following information:

Item	Required	Description
View Version	Yes	Select the version of the View to instantiate.
Instance Name	Yes	Must be unique for a given View.
Display Label	Yes	Readable display name used for the View instance when shown in Ambari Web.
Description	Yes	Readable description used for the View instance when shown in Ambari Web.
Visible	No	Designates whether the View is visible or not visible to the end-user in Ambari web. Use this property to temporarily hide a view in Ambari Web from users.
Properties	Maybe	Depends on the View. If the View requires certain configuration properties, you are prompted to provide the required information.

4.4. Setting View Permissions

After a view instance has been created, an Ambari Admin can set which users and groups can access the view by setting the Use permission. By default, after view instance creation, no permissions are set on a view.

To set permissions on a view:

1. Browse to a view and expand. For example, browse to the Slider or Jobs view.
2. Click on the view instance you want to modify.
3. In the Permissions section, click the Users or Groups control.
4. Modify the user and group lists as appropriate.
5. Click the check mark to save changes.

Note

The Framework provides a way for view developers to specify custom permissions, beyond just the default Use permission. If custom permissions are are specified, they will show up in the Ambari Administration interface and the Ambari Admin can set users and groups on these permissions. See Additional Information for more information on developing with the Views framework.

4.5. Additional Information

To learn more about developing views and the views framework itself, refer to the following resources:

Resource	Description	Link
Views Wiki	Learn about the Views Framework and Framework services available to views developers.	https://cwiki.apache.org/confluence/display/AMBARI/Viewsche.org/confluence/display/AMBARI/Views
Views API	Covers the Views REST API and associated framework Java classes.	https://github.com/apache/ambari/blob/trunk/ambari-views/docs/index.md
Views Examples	Code for example views that hover different areas of the framework and framework services.	https://github.com/apache/ambari/tree/trunk/ambari-views/examples
View Contributions	Views that are being developed and contributed to the Ambari community.	https://github.com/apache/ambari/tree/trunk/contrib/views