Hortonworks DataFlow

Installing an HDF Cluster

(February 28, 2018)

docs.cloudera.com

Hortonworks DataFlow: Installing an HDF Cluster

Table of Contents

1. Installing Ambari

Perform the following tasks to install Ambari.

Note

This document describes how to install Ambari and HDF on Intel x86 hardware. To install Ambari and HDF on IBM Power Systems, review your deployment options using Planning Your Deployment for IBM Power Systems.

1.1. Getting Ready for an Ambari Installation

This section describes the information and materials you should get ready to install a cluster using Ambari. Ambari provides an end-to-end management and monitoring solution for your cluster. Using the Ambari Web UI and REST APIs, you can deploy, operate, manage configuration changes, and monitor services for all nodes in your cluster from a central point.

1.1.1. Reviewing System Requirements

Your first task in installing Ambari is to review the Hortonworks DataFlow (HDF) support matrices for system requirements, supported operating systems, component interoperability, and similar information.

- HDF Support Matrices

1.1.2. Set Up Password-less SSH

About This Task

To have Ambari Server automatically install Ambari Agents on all your cluster hosts, you must set up password-less SSH connections between the Ambari Server host and all other hosts in the cluster. The Ambari Server host uses SSH public key authentication to remotely access and install the Ambari Agent.

Note

You can choose to manually install an Ambari Agent on each cluster host. In this case, you do not need to generate and distribute SSH keys.

Steps

1. Generate public and private SSH keys on the Ambari Server host.

```
ssh-keygen
```

2. Copy the SSH Public Key (id_rsa.pub) to the root account on your target hosts.

   ```
   .ssh/id_rsa
   ```

   ```
   .ssh/id_rsa.pub
   ```

3. Add the SSH Public Key to the authorized_keys file on your target hosts.

   ```
   cat id_rsa.pub >> authorized_keys
   ```

4. Depending on your version of SSH, you may need to set permissions on the .ssh directory (to 700) and the authorized_keys file in that directory (to 600) on the target hosts.

   ```
   chmod 700 ~/.ssh
   ```

   ```
   chmod 600 ~/.ssh/authorized_keys
   ```

5. From the Ambari Server, make sure you can connect to each host in the cluster using SSH, without having to enter a password.

   ```
   ssh root@<remote.target.host>
   ```

   where `<remote.target.host>` has the value of each host name in your cluster.

6. If the following warning message displays during your first connection: `Are you sure you want to continue connecting (yes/no)?` Enter **Yes**.

7. Retain a copy of the SSH Private Key on the machine from which you will run the web-based Ambari Install Wizard.

> **Note**
>
> It is possible to use a non-root SSH account, if that account can execute `sudo` without entering a password.

1.1.3. Set Up Service User Accounts

Each service requires a service user account. The Ambari Cluster Install wizard creates new and preserves any existing service user accounts, and uses these accounts when configuring Hadoop services. Service user account creation applies to service user accounts on the local operating system and to LDAP/AD accounts.

1.1.4. Enable NTP on the Cluster and on the Browser Host

The clocks of all the nodes in your cluster and the machine that runs the browser through which you access the Ambari Web interface must be able to synchronize with each other.

To install the NTP service and ensure it's ensure it's started on boot, run the following commands on each host:

RHEL/CentOS/Oracle 6	`yum install -y ntp`
	`chkconfig ntpd on`

RHEL/CentOS/Oracle 7	`yum install -y ntp`
	`systemctl enable ntpd`

SLES

```
zypper install ntp
chkconfig ntp on
```

Ubuntu

```
apt-get install ntp
update-rc.d ntp defaults
```

Debian

```
apt-get install ntp
update-rc.d ntp defaults
```

1.1.5. Check DNS and NSCD

All hosts in your system must be configured for both forward and and reverse DNS.

If you are unable to configure DNS in this way, you should edit the /etc/hosts file on every host in your cluster to contain the IP address and Fully Qualified Domain Name of each of your hosts. The following instructions are provided as an overview and cover a basic network setup for generic Linux hosts. Different versions and flavors of Linux might require slightly different commands and procedures. Please refer to the documentation for the operating system(s) deployed in your environment.

Hadoop relies heavily on DNS, and as such performs many DNS lookups during normal operation. To reduce the load on your DNS infrastructure, it's highly recommended to use the Name Service Caching Daemon (NSCD) on cluster nodes running Linux. This daemon will cache host, user, and group lookups and provide better resolution performance, and reduced load on DNS infrastructure.

1.1.5.1. Edit the Host File

1. Using a text editor, open the hosts file on every host in your cluster. For example:

   ```
   vi /etc/hosts
   ```

2. Add a line for each host in your cluster. The line should consist of the IP address and the FQDN. For example:

   ```
   1.2.3.4 <fully.qualified.domain.name>
   ```

   #### Important

   Do **not** remove the following two lines from your hosts file. Removing or editing the following lines may cause various programs that require network functionality to fail.

   ```
   127.0.0.1 localhost.localdomain localhost
   ```

   ```
   ::1 localhost6.localdomain6 localhost6
   ```

1.1.5.2. Set the Hostname

1. Confirm that the hostname is set by running the following command:

   ```
   hostname -f
   ```

   This should return the <fully.qualified.domain.name> you just set.

2. Use the "hostname" command to set the hostname on each host in your cluster. For example:

```
hostname <fully.qualified.domain.name>
```

1.1.5.3. Edit the Network Configuration File

1. Using a text editor, open the network configuration file on every host and set the desired network configuration for each host. For example:

   ```
   vi /etc/sysconfig/network
   ```

2. Modify the HOSTNAME property to set the fully qualified domain name.

   ```
   NETWORKING=yes
   ```

   ```
   HOSTNAME=<fully.qualified.domain.name>
   ```

1.1.6. Configuring iptables

For Ambari to communicate during setup with the hosts it deploys to and manages, certain ports must be open and available. The easiest way to do this is to temporarily disable iptables, as follows:

RHEL/CentOS/Oracle Linux 6

```
chkconfig iptables off
/etc/init.d/iptables stop
```

RHEL/CentOS/Oracle Linux 7

```
systemctl disable firewalld
service firewalld stop
```

SLES

```
rcSuSEfirewall2 stop
chkconfig SuSEfirewall2_setup off
```

Ubuntu

```
sudo ufw disable
sudo iptables -X
sudo iptables -t nat -F
sudo iptables -t nat -X
sudo iptables -t mangle -F
sudo iptables -t mangle -X
sudo iptables -P INPUT ACCEPT
sudo iptables -P FORWARD ACCEPT
sudo iptables -P OUTPUT ACCEPT
```

Debian

```
sudo iptables -X
sudo iptables -t nat -F
sudo iptables -t nat -X
sudo iptables -t mangle -F
sudo iptables -t mangle -X
sudo iptables -P INPUT ACCEPT
sudo iptables -P FORWARD ACCEPT
sudo iptables -P OUTPUT ACCEPT
```

You can restart iptables after setup is complete. If the security protocols in your environment prevent disabling iptables, you can proceed with iptables enabled, if all required ports are open and available.

Ambari checks whether iptables is running during the Ambari Server setup process. If iptables is running, a warning displays, reminding you to check that required ports are open and available. The Host Confirm step in the Cluster Install Wizard also issues a warning for each host that has iptables running.

1.1.7. Disable SELinux and PackageKit and check the umask Value

1. You must disable SELinux for the Ambari setup to function. On each host in your cluster, enter:

   ```
   setenforce 0
   ```

   Note

   To permanently disable SELinux set SELINUX=disabled in `/etc/selinux/config` This ensures that SELinux does not turn itself on after you reboot the machine .

2. On an installation host running RHEL/CentOS with PackageKit installed, open `/etc/yum/pluginconf.d/refresh-packagekit.conf` using a text editor. Make the following change:

   ```
   enabled=0
   ```

   Note

   PackageKit is not enabled by default on Debian, SLES, or Ubuntu systems. Unless you have specifically enabled PackageKit, you may skip this step for a Debian, SLES, or Ubuntu installation host.

3. UMASK (User Mask or User file creation MASK) sets the default permissions or base permissions granted when a new file or folder is created on a Linux machine. Most Linux distros set 022 as the default umask value. A umask value of 022 grants read, write, execute permissions of 755 for new files or folders. A umask value of 027 grants read, write, execute permissions of 750 for new files or folders.

   Ambari, HDP, and HDF support umask values of 022 (0022 is functionally equivalent), 027 (0027 is functionally equivalent). These values must be set on all hosts.

   UMASK Examples:

   Setting the umask for your current login session:

   ```
   umask 0022
   ```

   Checking your current umask:

   ```
   umask 0022
   ```

   Permanently changing the umask for all interactive users:

   ```
   echo umask 0022 >> /etc/profile
   ```

1.2. Download the Ambari Repository

Follow the instructions in the section for the operating system that runs your installation host.

- RHEL/CentOS/Oracle Linux 6 [6]
- RHEL/CentOS/Oracle Linux 7 [7]
- SLES 12 [8]
- SLES 11 [9]
- Ubuntu 14 [10]
- Ubuntu 16 [11]
- Debian 7 [12]

Use a command line editor to perform each instruction.

1.2.1. RHEL/CentOS/Oracle Linux 6

On a server host that has Internet access, use a command line editor to perform the following:

Steps

1. Log in to your host as `root`.
2. Download the Ambari repository file to a directory on your installation host.

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

Important

Do not modify the `ambari.repo` file name. This file is expected to be available on the Ambari Server host during Agent registration.

3. Confirm that the repository is configured by checking the repo list.

```
yum repolist
```

You should see values similar to the following for Ambari repositories in the list.

```
repo id              repo name                               status
ambari-2.6.1.0-143  ambari Version - ambari-2.6.1.0-143    12
base                 CentOS-6 - Base                         6,696
extras               CentOS-6 - Extras                       64
updates              CentOS-6 - Updates                      974
 repolist: 7,746
```

Version values vary, depending on the installation.

Note

When deploying a cluster having limited or no Internet access, you should provide access to the bits using an alternative method.

Ambari Server by default uses an embedded PostgreSQL database. When you install the Ambari Server, the PostgreSQL packages and dependencies must be available for install. These packages are typically available as part of your Operating System repositories. Please confirm you have the appropriate repositories available for the postgresql-server packages.

Next Step

- Install the Ambari Server [13]
- Set Up the Ambari Server [19]

1.2.2. RHEL/CentOS/Oracle Linux 7

On a server host that has Internet access, use a command line editor to perform the following

Steps

1. Log in to your host as `root`.
2. Download the Ambari repository file to a directory on your installation host.

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

Important

Do not modify the `ambari.repo` file name. This file is expected to be available on the Ambari Server host during Agent registration.

3. Confirm that the repository is configured by checking the repo list.

```
yum repolist
```

You should see values similar to the following for Ambari repositories in the list.

```
repo id                    repo name
 status
ambari-2.6.1.0-143        ambari Version - ambari-2.6.1.0-143            12
epel/x86_64                Extra Packages for Enterprise Linux 7 - x86_64
 11,387
ol7_UEKR4/x86_64           Latest Unbreakable Enterprise Kernel Release 4
                            for Oracle Linux 7Server (x86_64)   295
ol7_latest/x86_64          Oracle Linux 7Server Latest (x86_64)
 18,642
puppetlabs-deps/x86_64     Puppet Labs Dependencies El 7 - x86_64
 17
puppetlabs-products/x86_64 Puppet Labs Products El 7 - x86_64
 225
 repolist: 30,578
```

Version values vary, depending on the installation.

Note

When deploying a cluster having limited or no Internet access, you should provide access to the bits using an alternative method.

Ambari Server by default uses an embedded PostgreSQL database. When you install the Ambari Server, the PostgreSQL packages and dependencies must be available for install. These packages are typically available as part of your Operating System repositories. Please confirm you have the appropriate repositories available for the postgresql-server packages.

Next Step

- Install the Ambari Server [13]
- Set Up the Ambari Server [19]

1.2.3. SLES 12

On a server host that has Internet access, use a command line editor to perform the following:

Steps

1. Log in to your host as `root`.
2. Download the Ambari repository file to a directory on your installation host.

```
wget -nv http://public-repo-1.hortonworks.com/ambari/sles12/2.x/updates/2.6.
1.0/ambari.repo -O /etc/zypp/repos.d/ambari.repo
```

Important

Do not modify the `ambari.repo` file name. This file is expected to be available on the Ambari Server host during Agent registration.

3. Confirm the downloaded repository is configured by checking the repo list.

```
zypper repos
```

You should see the Ambari repositories in the list.

```
# | Alias                    | Name                                 |
 Enabled | Refresh
--+------------------------- +--------------------------------------
+---------+--------
1 | ambari-2.6.1.0-143      | ambari Version - ambari-2.6.1.0-143 | Yes
 | No
2 | http-demeter.uni         | SUSE-Linux-Enterprise-Software
     -regensburg.de-c997c8f9 |  -Development-Kit-12-SP1 12.1.1-1.57 | Yes
   | Yes
3 | opensuse                 | OpenSuse                             | Yes
   | Yes
```

Version values vary, depending on the installation.

Note

When deploying a cluster having limited or no Internet access, you should provide access to the bits using an alternative method.

Ambari Server by default uses an embedded PostgreSQL database. When you install the Ambari Server, the PostgreSQL packages and dependencies must be available for install. These packages are typically available as part of your Operating System repositories. Please confirm you have the appropriate repositories available for the postgresql-server packages.

Next Step

- Install the Ambari Server [13]
- Set Up the Ambari Server [19]

1.2.4. SLES 11

On a server host that has Internet access, use a command line editor to perform the following

Steps

1. Log in to your host as `root`.

2. Download the Ambari repository file to a directory on your installation host.

```
wget -nv http://public-repo-1.hortonworks.com/ambari/suse11/2.x/updates/2.6.
1.0/ambari.repo -O /etc/zypp/repos.d/ambari.repo
```

Important

Do not modify the `ambari.repo` file name. This file is expected to be available on the Ambari Server host during Agent registration.

3. Confirm the downloaded repository is configured by checking the repo list.

```
zypper repos
```

You should see the Ambari repositories in the list.

```
# | Alias                     | Name                                 |
 Enabled | Refresh
--+---------------------------+--------------------------------------
+---------+--------
1 | ambari-2.6.1.0-143       | ambari Version - ambari-2.6.1.0-143 | Yes
 | No
2 | http-demeter.uni          |SUSE-Linux-Enterprise-Software
     -regensburg.de-c997c8f9 | -Development-Kit-11-SP3 12.1.1-1.57  | Yes
   | Yes
3 | opensuse                  | OpenSuse                             | Yes
   | Yes
```

Version values vary, depending on the installation.

Note

When deploying a cluster having limited or no Internet access, you should provide access to the bits using an alternative method.

Ambari Server by default uses an embedded PostgreSQL database. When you install the Ambari Server, the PostgreSQL packages and dependencies must be available for install. These packages are typically available as part of your Operating System repositories. Please confirm you have the appropriate repositories available for the postgresql-server packages.

Next Step

- Install the Ambari Server [13]
- Set Up the Ambari Server [19]

1.2.5. Ubuntu 14

On a server host that has Internet access, use a command line editor to perform the following:

Steps

1. Log in to your host as `root`.
2. Download the Ambari repository file to a directory on your installation host.

   ```
   wget -O /etc/apt/sources.list.d/ambari.list http://public-repo-1.
   hortonworks.com/ambari/ubuntu14/2.x/updates/2.6.1.0/ambari.list
   ```

   ```
   apt-key adv --recv-keys --keyserver keyserver.ubuntu.com B9733A7A07513CAD
   ```

   ```
   apt-get update
   ```

   ### Important

   Do not modify the `ambari.list` file name. This file is expected to be available on the Ambari Server host during Agent registration.

3. Confirm that Ambari packages downloaded successfully by checking the package name list.

   ```
   apt-cache showpkg ambari-server
   ```

   ```
   apt-cache showpkg ambari-agent
   ```

   ```
   apt-cache showpkg ambari-metrics-assembly
   ```

   You should see the Ambari packages in the list.

Note

When deploying a cluster having limited or no Internet access, you should provide access to the bits using an alternative method.

Ambari Server by default uses an embedded PostgreSQL database. When you install the Ambari Server, the PostgreSQL packages and dependencies must be available for install. These packages are typically available as part of your Operating System repositories. Please confirm you have the appropriate repositories available for the postgresql-server packages.

Next Step

- Install the Ambari Server [13]
- Set Up the Ambari Server [19]

1.2.6. Ubuntu 16

On a server host that has Internet access, use a command line editor to perform the following:

Steps

1. Log in to your host as `root`.

2. Download the Ambari repository file to a directory on your installation host.

```
wget -O /etc/apt/sources.list.d/ambari.list http://public-repo-1.
hortonworks.com/ambari/ubuntu16/2.x/updates/2.6.1.0/ambari.list
```

```
apt-key adv --recv-keys --keyserver keyserver.ubuntu.com B9733A7A07513CAD
```

```
apt-get update
```

Important

Do not modify the `ambari.list` file name. This file is expected to be available on the Ambari Server host during Agent registration.

3. Confirm that Ambari packages downloaded successfully by checking the package name list.

```
apt-cache showpkg ambari-server
```

```
apt-cache showpkg ambari-agent
```

```
apt-cache showpkg ambari-metrics-assembly
```

You should see the Ambari packages in the list.

Note

When deploying a cluster having limited or no Internet access, you should provide access to the bits using an alternative method.

Ambari Server by default uses an embedded PostgreSQL database. When you install the Ambari Server, the PostgreSQL packages and dependencies must be available for install. These packages are typically available as part of your Operating System repositories. Please confirm you have the appropriate repositories available for the postgresql-server packages.

Next Step

- Install the Ambari Server [13]
- Set Up the Ambari Server [19]

More Information

Using a Local Repository

1.2.7. Debian 7

On a server host that has Internet access, use a command line editor to perform the following:

Steps

1. Log in to your host as `root`.

2. Download the Ambari repository file to a directory on your installation host.

   ```
   wget -O /etc/apt/sources.list.d/ambari.list http://public-repo-1.
   hortonworks.com/ambari/debian7/2.x/updates/2.6.1.0/ambari.list
   ```

   ```
   apt-key adv --recv-keys --keyserver keyserver.ubuntu.com B9733A7A07513CAD
   ```

   ```
   apt-get update
   ```

   Important

   Do not modify the `ambari.list` file name. This file is expected to be available on the Ambari Server host during Agent registration.

3. Confirm that Ambari packages downloaded successfully by checking the package name list.

   ```
   apt-cache showpkg ambari-server
   ```

   ```
   apt-cache showpkg ambari-agent
   ```

   ```
   apt-cache showpkg ambari-metrics-assembly
   ```

   You should see the Ambari packages in the list.

Note

When deploying a cluster having limited or no Internet access, you should provide access to the bits using an alternative method.

Ambari Server by default uses an embedded PostgreSQL database. When you install the Ambari Server, the PostgreSQL packages and dependencies must be available for install. These packages are typically available as part of your Operating System repositories. Please confirm you have the appropriate repositories available for the postgresql-server packages.

Next Step

- Install the Ambari Server [13]
- Set Up the Ambari Server [19]

1.3. Install the Ambari Server

Follow the instructions in the section for the operating system that runs your installation host.

- **RHEL/CentOS/Oracle Linux 6** [13]
- **RHEL/CentOS/Oracle Linux 7** [14]
- **SLES 12** [15]
- **SLES 11** [16]
- **Ubuntu 14** [17]
- **Ubuntu 16** [18]
- **Debian 7** [18]

Use a command line editor to perform each instruction.

1.3.1. RHEL/CentOS/Oracle Linux 6

On a server host that has Internet access, use a command line editor to perform the following:

Steps

1. Install the Ambari bits. This also installs the default PostgreSQL Ambari database.

   ```
   yum install ambari-server
   ```

2. Enter **y** when prompted to confirm transaction and dependency checks.

   A successful installation displays output similar to the following:

   ```
   Installing : postgresql-libs-8.4.20-6.el6.x86_64         1/4
   Installing : postgresql-8.4.20-6.el6.x86_64              2/4
   Installing : postgresql-server-8.4.20-6.el6.x86_64       3/4
   Installing : ambari-server-2.6.1.0-1050.x86_64           4/4
   Verifying  : ambari-server-2.6.1.0-1050.x86_64           1/4
   Verifying  : postgresql-8.4.20-6.el6.x86_64              2/4
   Verifying  : postgresql-server-8.4.20-6.el6.x86_64       3/4
   Verifying  : postgresql-libs-8.4.20-6.el6.x86_64         4/4

   Installed:
     ambari-server.x86_64 0:2.6.1.0-1050
   ```

```
Dependency Installed:
 postgresql.x86_64 0:8.4.20-6.el6
 postgresql-libs.x86_64 0:8.4.20-6.el6
 postgresql-server.x86_64 0:8.4.20-6.el6
 Complete!
```

Note

Accept the warning about trusting the Hortonworks GPG Key. That key will be automatically downloaded and used to validate packages from Hortonworks. You will see the following message:

```
Importing GPG key 0x07513CAD: Userid: "Jenkins (HDP
Builds) <jenkin@hortonworks.com>" From : http://
s3.amazonaws.com/dev.hortonworks.com/ambari/centos6/RPM-
GPG-KEY/RPM-GPG-KEY-Jenkins
```

Note

When deploying a cluster having limited or no Internet access, you should provide access to the bits using an alternative method.

Ambari Server by default uses an embedded PostgreSQL database. When you install the Ambari Server, the PostgreSQL packages and dependencies must be available for install. These packages are typically available as part of your Operating System repositories. Please confirm you have the appropriate repositories available for the postgresql-server packages.

Next Step

Set Up the Ambari Server [19]

1.3.2. RHEL/CentOS/Oracle Linux 7

On a server host that has Internet access, use a command line editor to perform the following

Steps

1. Install the Ambari bits. This also installs the default PostgreSQL Ambari database.

   ```
   yum install ambari-server
   ```

2. Enter **y** when prompted to confirm transaction and dependency checks.

   A successful installation displays output similar to the following:

   ```
   Installing : postgresql-libs-9.2.18-1.el7.x86_64         1/4
   Installing : postgresql-9.2.18-1.el7.x86_64              2/4
   Installing : postgresql-server-9.2.18-1.el7.x86_64       3/4
   Installing : ambari-server-2.6.1.0-1050.x86_64           4/4
   Verifying  : ambari-server-2.6.1.0-1050.x86_64           1/4
   Verifying  : postgresql-9.2.18-1.el7.x86_64              2/4
   ```

```
Verifying  : postgresql-server-9.2.18-1.el7.x86_64        3/4
Verifying  : postgresql-libs-9.2.18-1.el7.x86_64          4/4

Installed:
  ambari-server.x86_64 0:2.6.1.0-1050

Dependency Installed:
 postgresql.x86_64 0:9.2.18-1.el7
 postgresql-libs.x86_64 0:9.2.18-1.el7
 postgresql-server.x86_64 0:9.2.18-1.el7
Complete!
```

Note

Accept the warning about trusting the Hortonworks GPG Key. That key will be automatically downloaded and used to validate packages from Hortonworks. You will see the following message:

```
Importing GPG key 0x07513CAD: Userid: "Jenkins (HDP
Builds) <jenkin@hortonworks.com>" From : http://
s3.amazonaws.com/dev.hortonworks.com/ambari/centos6/RPM-
GPG-KEY/RPM-GPG-KEY-Jenkins
```

Note

When deploying a cluster having limited or no Internet access, you should provide access to the bits using an alternative method.

Ambari Server by default uses an embedded PostgreSQL database. When you install the Ambari Server, the PostgreSQL packages and dependencies must be available for install. These packages are typically available as part of your Operating System repositories. Please confirm you have the appropriate repositories available for the postgresql-server packages.

Next Step

Set Up the Ambari Server [19]

1.3.3. SLES 12

On a server host that has Internet access, use a command line editor to perform the following:

Steps

1. Install the Ambari bits. This also installs the default PostgreSQL Ambari database.

   ```
   zypper install ambari-server
   ```

2. Enter **y** when prompted to confirm transaction and dependency checks.

   A successful installation displays output similar to the following:

```
 Retrieving package postgresql-libs-8.3.5-1.12.x86_64 (1/4), 172.0 KiB (571.
0 KiB unpacked)
Retrieving: postgresql-libs-8.3.5-1.12.x86_64.rpm [done (47.3 KiB/s)]
Installing: postgresql-libs-8.3.5-1.12 [done]
Retrieving package postgresql-8.3.5-1.12.x86_64 (2/4), 1.0 MiB (4.2 MiB
 unpacked)
Retrieving: postgresql-8.3.5-1.12.x86_64.rpm [done (148.8 KiB/s)]
Installing: postgresql-8.3.5-1.12 [done]
Retrieving package postgresql-server-8.3.5-1.12.x86_64 (3/4), 3.0 MiB (12.6
 MiB unpacked)
Retrieving: postgresql-server-8.3.5-1.12.x86_64.rpm [done (452.5 KiB/s)]
Installing: postgresql-server-8.3.5-1.12 [done]
Updating etc/sysconfig/postgresql...
Retrieving package ambari-server-2.6.1.0-143.noarch (4/4), 99.0 MiB (126.3
 MiB unpacked)
Retrieving: ambari-server-2.6.1.0-143.noarch.rpm [done (3.0 MiB/s)]
Installing: ambari-server-2.6.1.0-143 [done]
 ambari-server 0:off 1:off 2:off 3:on 4:off 5:on 6:off
```

Note

When deploying a cluster having limited or no Internet access, you should provide access to the bits using an alternative method.

Ambari Server by default uses an embedded PostgreSQL database. When you install the Ambari Server, the PostgreSQL packages and dependencies must be available for install. These packages are typically available as part of your Operating System repositories. Please confirm you have the appropriate repositories available for the postgresql-server packages.

Next Step

Set Up the Ambari Server [19]

1.3.4. SLES 11

On a server host that has Internet access, use a command line editor to perform the following

Steps

1. Install the Ambari bits. This also installs the default PostgreSQL Ambari database.

   ```
   zypper install ambari-server
   ```

2. Enter **y** when prompted to to confirm transaction and dependency checks.

   A successful installation displays output similar to the following:

```
 Retrieving package postgresql-libs-8.3.5-1.12.x86_64 (1/4), 172.0 KiB (571.
0 KiB unpacked)
Retrieving: postgresql-libs-8.3.5-1.12.x86_64.rpm [done (47.3 KiB/s)]
Installing: postgresql-libs-8.3.5-1.12 [done]
Retrieving package postgresql-8.3.5-1.12.x86_64 (2/4), 1.0 MiB (4.2 MiB
 unpacked)
Retrieving: postgresql-8.3.5-1.12.x86_64.rpm [done (148.8 KiB/s)]
Installing: postgresql-8.3.5-1.12 [done]
Retrieving package postgresql-server-8.3.5-1.12.x86_64 (3/4), 3.0 MiB (12.6
 MiB unpacked)
Retrieving: postgresql-server-8.3.5-1.12.x86_64.rpm [done (452.5 KiB/s)]
Installing: postgresql-server-8.3.5-1.12 [done]
Updating etc/sysconfig/postgresql...
Retrieving package ambari-server-2.6.1.0-143.noarch (4/4), 99.0 MiB (126.3
 MiB unpacked)
Retrieving: ambari-server-2.6.1.0-143.noarch.rpm [done (3.0 MiB/s)]
Installing: ambari-server-2.6.1.0-143 [done]
ambari-server 0:off 1:off 2:off 3:on 4:off 5:on 6:off
```

Note

When deploying a cluster having limited or no Internet access, you should provide access to the bits using an alternative method.

Ambari Server by default uses an embedded PostgreSQL database. When you install the Ambari Server, the PostgreSQL packages and dependencies must be available for install. These packages are typically available as part of your Operating System repositories. Please confirm you have the appropriate repositories available for the postgresql-server packages.

Next Step

Set Up the Ambari Server [19]

1.3.5. Ubuntu 14

On a server host that has Internet access, use a command line editor to perform the following:

Steps

1. Install the Ambari bits. This also installs the default PostgreSQL Ambari database.

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

Note

When deploying a cluster having limited or no Internet access, you should provide access to the bits using an alternative method.

Ambari Server by default uses an embedded PostgreSQL database. When you install the Ambari Server, the PostgreSQL packages and dependencies must be available for install. These packages are typically available as part of your Operating System repositories. Please confirm you have the appropriate repositories available for the postgresql-server packages.

Next Step

Set Up the Ambari Server [19]

1.3.6. Ubuntu 16

On a server host that has Internet access, use a command line editor to perform the following:

Steps

1. Install the Ambari bits. This also installs the default PostgreSQL Ambari database.

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

Note

When deploying a cluster having limited or no Internet access, you should provide access to the bits using an alternative method.

Ambari Server by default uses an embedded PostgreSQL database. When you install the Ambari Server, the PostgreSQL packages and dependencies must be available for install. These packages are typically available as part of your Operating System repositories. Please confirm you have the appropriate repositories available for the postgresql-server packages.

Next Step

Set Up the Ambari Server [19]

1.3.7. Debian 7

On a server host that has Internet access, use a command line editor to perform the following:

Steps

1. Install the Ambari bits. This also installs the default PostgreSQL Ambari database.

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

Note

When deploying a cluster having limited or no Internet access, you should provide access to the bits using an alternative method.

Ambari Server by default uses an embedded PostgreSQL database. When you install the Ambari Server, the PostgreSQL packages and dependencies must be available for install. These packages are typically available as part of your Operating System repositories. Please confirm you have the appropriate repositories available for the postgresql-server packages.

Next Step

Set Up the Ambari Server [19]

1.4. Set Up the Ambari Server

Before starting the Ambari Server, you **must** set up the Ambari Server. Setup configures Ambari to talk to the Ambari database, installs the JDK and allows you to customize the user account the Ambari Server daemon will run as. The

```
ambari-server setup
```

command manages the setup process. Run the following command on the Ambari server host to start the setup process. You may also append Setup Options to the command.

```
ambari-server setup
```

Respond to the setup prompt:

1. If you have *not* temporarily disabled SELinux, you may get a warning. Accept the default (**y**), and continue.
2. By default, Ambari Server runs under `root`. Accept the default **`(n)`** at the `Customize user account for ambari-server daemon` prompt, to proceed as `root`. If you want to create a different user to run the Ambari Server, or to assign a previously created user, select **y** at the `Customize user account for ambari-server daemon` prompt, then provide a user name.
3. If you have not temporarily disabled `iptables` you may get a warning. Enter **y** to continue.
4. Select a JDK version to download. Enter 1 to download Oracle JDK 1.8. Alternatively, you can choose to enter a Custom JDK. If you choose Custom JDK, you must manually install the JDK on all hosts and specify the Java Home path.

   ### Note

   JDK support depends entirely on your choice of Stack versions. By default, Ambari Server setup downloads and installs Oracle JDK 1.8 and the accompanying Java Cryptography Extension (JCE) Policy Files.
5. Accept the Oracle JDK license when prompted. You must accept this license to download the necessary JDK from Oracle. The JDK is installed during the deploy phase.
6. Review the GPL license agreement when prompted. To explicitly enable Ambari to download and install LZO data compression libraries, you must answer **y**. If you enter **n**, Ambari will not automatically install LZO on any new host in the cluster. In this case, you must ensure LZO is installed and configured appropriately. Without LZO being installed and configured, data compressed with LZO will not be readable. If you do not want Ambari to automatically download and install LZO, you must confirm your choice to proceed.
7. Select **n** at `Enter advanced database configuration` to use the default, embedded PostgreSQL database for Ambari. The default PostgreSQL database name is `ambari`. The default user name and password are `ambari/bigdata`. Otherwise, to use an existing PostgreSQL, MySQL/MariaDB or Oracle database with Ambari, select **y**.

   - If you are using an existing PostgreSQL, MySQL/MariaDB, or Oracle database instance, use one of the following prompts:

Important

You must prepare an existing database instance, before running setup and entering advanced database configuration.

Important

Using the **Microsoft SQL Server** or **SQL Anywhere** database options are not supported.

- To use an existing Oracle instance, and select your own database name, user name, and password for that database, enter **2**.

  Select the database you want to use and provide any information requested at the prompts, including host name, port, Service Name or SID, user name, and password.

- To use an existing MySQL/MariaDB database, and select your own database name, user name, and password for that database, enter **3**.

  Select the database you want to use and provide any information requested at the prompts, including host name, port, database name, user name, and password.

- To use an existing PostgreSQL database, and select your own database name, user name, and password for that database, enter **4**.

  Select the database you want to use and provide any information requested at the prompts, including host name, port, database name, user name, and password.

8. At `Proceed with configuring remote database connection properties [y/n]` choose **y**.

9. Setup completes.

Note

If your host accesses the Internet through a proxy server, you must configure Ambari Server to use this proxy server.

1.4.1. Setup Options

The following options are frequently used for Ambari Server setup.

-j (or –java-home)

Specifies the JAVA_HOME path to use on the Ambari Server and all hosts in the cluster. By default when you do not specify this option, Ambari Server setup downloads the Oracle JDK 1.8 binary and accompanying Java Cryptography Extension (JCE) Policy Files to /var/lib/ambari-server/resources. Ambari Server then installs the JDK to /usr/jdk64.

Use this option when you plan to use a JDK other than the default Oracle JDK 1.8. If you are using an alternate

JDK, you must manually install the JDK on all hosts and specify the Java Home path during Ambari Server setup. If you plan to use Kerberos, you must also install the JCE on all hosts.

This path must be valid on all hosts. For example:

```
ambari-server setup –j /usr/java/default
```

–jdbc-driver

Should be the path to the JDBC driver JAR file. Use this option to specify the location of the JDBC driver JAR and to make that JAR available to Ambari Server for distribution to cluster hosts during configuration. Use this option with the –jdbc-db option to specify the database type.

–jdbc-db

Specifies the database type. Valid values are: [postgres | mysql | oracle] Use this option with the –jdbc-driver option to specify the location of the JDBC driver JAR file.

-s (or –silent)

Setup runs silently. Accepts all the default prompt values, such as:

- User account "root" for the ambari-server
- Oracle 1.8 JDK (which is installed at /usr/jdk64). This can be overridden by adding the -j option and specifying an existing JDK path.
- Embedded PostgreSQL for Ambari DB (with database name "ambari")

Important

By choosing the silent setup option and by not overriding the JDK selection, Oracle JDK will be installed and you will be agreeing to the Oracle Binary Code License agreement.

Do not use this option if you do not agree to the license terms.

If you want to run the Ambari Server as non-root, you must run setup in interactive mode. When prompted to customize the ambari-server user account, provide the account information.

–enable-lzo-under-gpl-license

Use this option to download and install LZO compression, subject to the General Public License.

-v (or –verbose)

Prints verbose info and warning messages to the console during Setup.

-g (or –debug)	Prints debug info to the console during Setup.

1.5. Start the Ambari Server

- Run the following command on the Ambari Server host:

  ```
  ambari-server start
  ```

- To check the Ambari Server processes:

  ```
  ambari-server status
  ```

- To stop the Ambari Server:

  ```
  ambari-server stop
  ```

Note

If you plan to use an existing database instance for Hive or for Oozie, you must prepare to use an existing database **before** installing your Hadoop cluster.

On Ambari Server start, Ambari runs a database consistency check looking for issues. If any issues are found, Ambari Server **start will abort** and display the following message: DB `configs consistency check failed`. Ambari writes more details about database consistency check results to the`/var/log/ambari-server/ambari-server-check-database.log` file.

You can force Ambari Server to start by skipping this check with the following option:

```
ambari-server start --skip-database-check
```

If you have database issues, by choosing to skip this check, **do not make any changes to your cluster topology or perform a cluster upgrade until you correct the database consistency issues**. Please contact Hortonworks Support and provide the `ambari-server-check-database.log` output for assistance.

2. Installing Databases

Schema Registry, SAM, Druid, and Superset require a relational data store to store metadata. You can use either MySQL, Postgres, or Oracle for this. This chapter describes how to install either MySQL, Postgres, and Oracle and how create a databases for SAM and Registry. If you are installing on an existing HDP cluster by using Superset, you can skip the installation instructions, because MySQL was installed with Druid. In this case, configure the databases.

Note

You should install either Postgres, Oracle or MySQL; both are not necessary. It is recommended that you use MySQL.

Warning

If you are installing Postgres, you must install Postgres 9.5 or later for SAM and Schema Registry. Ambari does not install Postgres 9.5, so you must perform a manual Postgres installation.

Installing and Configuring MySQL

- Installing MySQL [23]
- Configuring SAM and Schema Registry Metadata Stores in MySQL [24]
- Configuring Druid and Superset Metadata Stores in MySQL [25]

Installing and Configuring Postgres

- Install Postgres [25]
- Configure Postgres to Allow Remote Connections [26]
- Configure SAM and Schema Registry Metadata Stores in Postgres [27]
- Configure Druid and Superset Metadata Stores in Postgres [27]

Using an Oracle Database

- Section 2.8, “Specifying an Oracle Database to Use with SAM and Schema Registry” [28]
- Section 2.9, “Switching to an Oracle Database After Installation” [28]

2.1. Installing MySQL

About This Task

You can install MySQL 5.5 or later.

Before You Begin

On the Ambari host, install the JDBC driver for MySQL, and then add it to Ambari:

```
yum install mysql-connector-java* \
sudo ambari-server setup --jdbc-db=mysql \
--jdbc-driver=/usr/share/java/mysql-connector-java.jar
```

Steps

1. Log in to the node on which you want to install the MySQL metastore to use for SAM, Schema Registry, and Druid.

2. Install MySQL and the MySQL community server, and start the MySQL service:

   ```
   yum localinstall \
   https://dev.mysql.com/get/mysql57-community-release-el7-8.noarch.rpm

   yum install mysql-community-server

   systemctl start mysqld.service
   ```

3. Obtain a randomly generated MySQL root password:

   ```
   grep 'A temporary password is generated for root@localhost' \
   /var/log/mysqld.log |tail -1
   ```

4. Reset the MySQL root password. Enter the following command. You are prompted for the password you obtained in the previous step. MySQL then asks you to change the password.

   ```
   /usr/bin/mysql_secure_installation
   ```

2.2. Configuring SAM and Schema Registry Metadata Stores in MySQL

Steps

1. Launch the MySQL monitor:

   ```
   mysql -u root -p
   ```

2. Create the database for the Registry and SAM metastore:

   ```
   create database registry;
   create database streamline;
   ```

3. Create Schema Registry and SAM user accounts, replacing the last string with your password:

   ```
   CREATE USER 'registry'@'%' IDENTIFIED BY 'R12$%34qw';
   CREATE USER 'streamline'@'%' IDENTIFIED BY 'R12$%34qw';
   ```

4. Assign privileges to the user account:

   ```
   GRANT ALL PRIVILEGES ON registry.* TO 'registry'@'%' WITH GRANT OPTION ;
   GRANT ALL PRIVILEGES ON streamline.* TO 'streamline'@'%' WITH GRANT OPTION ;
   ```

5. Commit the operation:

```
commit;
```

2.3. Configuring Druid and Superset Metadata Stores in MySQL

About This Task

Druid and Superset require a relational data store to store metadata. To use MySQL for this, install MySQL and create a database for the Druid metastore.

Steps

1. Launch the MySQL monitor:

   ```
   mysql -u root -p
   ```

2. Create the database for the Druid and Superset metastore:

   ```
   CREATE DATABASE druid DEFAULT CHARACTER SET utf8;
   CREATE DATABASE superset DEFAULT CHARACTER SET utf8;
   ```

3. Create `druid` and `superset` user accounts, replacing the last string with your password:

   ```
   CREATE USER 'druid'@'%' IDENTIFIED BY '9oNio)ex1ndL';
   CREATE USER 'superset'@'%' IDENTIFIED BY '9oNio)ex1ndL';
   ```

4. Assign privileges to the `druid` account:

   ```
   GRANT ALL PRIVILEGES ON *.* TO 'druid'@'%' WITH GRANT OPTION;
   GRANT ALL PRIVILEGES ON *.* TO 'superset'@'%' WITH GRANT OPTION;
   ```

5. Commit the operation:

   ```
   commit;
   ```

2.4. Install Postgres

Before You Begin

If you have already installed a MySQL database, you may skip these steps.

Warning

You must install Postgres 9.5 or later for SAM and Schema Registry. Ambari does not install Postgres 9.5, so you must perform a manual Postgres installation.

Steps

1. Install RPM according to the requirements of your operating system:

   ```
   yum install https://yum.postgresql.org/9.6/redhat/rhel-7-x86_64/pgdg-
   redhat96-9.6-3.noarch.rpm
   ```

2. Install 9.5+ Postgres database:

   ```
   yum install postgresql96-server postgresql96-contrib postgresql96
   ```

3. Initialize the database:

   - For CentOS 7, use the following syntax:

     ```
     /usr/pgsql-9.6/bin/postgresql96-setup initdb
     ```

   - For CentOS 6, use the following syntax:

     ```
     sudo service postgresql initdb
     ```

4. Start Postgres.

   For example, if you are using CentOS 7, use the following syntax:

   ```
   systemctl enable postgresql-9.6.service
   systemctl start postgresql-9.6.service
   ```

5. Verify that you can log in:

   ```
   sudo su postgres
   psql
   ```

2.5. Configure Postgres to Allow Remote Connections

About This Task

It is critical that you configure Postgres to allow remote connections before you deploy a cluster. If you do not perform these steps in advance of installing your cluster, the installation fails.

Steps

1. Open /var/lib/pgsql/9.6/data/pg_hba.conf and update to the following

   ```

   # "local" is for Unix domain socket connections only
   local all all trust


   # IPv4 local connections:
   host all all 0.0.0.0/0 trust


   # IPv6 local connections:
   host all all ::/0 trust
   ```

2. Open /var/lib//pgsql/9.6/data/postgresql.conf and update to the following:

   ```
   listen_addresses = '*'
   ```

3. Restart Postgres:

4.
   ```
   systemctl stop postgresql-9.6.service
   systemctl start postgresql-9.6.service
   ```

2.6. Configure SAM and Schema Registry Metadata Stores in Postgres

About This Task

If you have already installed MySQL and configured SAM and Schema Registry metadata stores using MySQL, you do not need to configure additional metadata stores in Postgres.

Steps

1. Log in to Postgres:

   ```
   sudo su postgres
   psql
   ```

2. Create a database called `registry` with the password `registry`:

   ```
   create database registry;
   CREATE USER registry WITH PASSWORD 'registry';
   GRANT ALL PRIVILEGES ON DATABASE "registry" to registry;
   ```

3. Create a database called `streamline` with the password `streamline`:

   ```
   create database streamline;
   CREATE USER streamline WITH PASSWORD 'streamline';
   GRANT ALL PRIVILEGES ON DATABASE "streamline" to streamline;
   ```

2.7. Configure Druid and Superset Metadata Stores in Postgres

About This Task

Druid and Superset require a relational data store to store metadata. To use Postgres for this, install Postgres and create a database for the Druid metastore. If you have already created a data store using MySQL, you do not need to configure additional metadata stores in Postgres.

Steps

1. Log in to Postgres:

   ```
   sudo su postgres
   psql
   ```

2. Create a database, user, and password, each called `druid`, and assign database privileges to the user `druid`:

   ```
   create database druid;
   CREATE USER druid WITH PASSWORD 'druid';
   GRANT ALL PRIVILEGES ON DATABASE  "druid" to druid;
   ```

3. Create a database, user, and password, each called `superset`, and assign database privileges to the user `superset`:

   ```
   create database superset;
   CREATE USER superset WITH PASSWORD 'superset';
   GRANT ALL PRIVILEGES ON DATABASE "superset" to superset;
   ```

2.8. Specifying an Oracle Database to Use with SAM and Schema Registry

About This Task

You may use an Oracle database with SAM and Schema Registry. Oracle databases 12c and 11g Release 2 are supported

Prerequisites

You have an Oracle database installed and configured.

Steps

1. Register the Oracle JDBC driver jar.

   ```
   sudo ambari-server setup --jdbc-db=oracle --jdbc-driver=/usr/share/java/
   ojdbc.jar
   ```

2. From the SAM an Schema Registry configuration screen, select Oracle as the database type and provide the necessary Oracle Server JDBC credentials and connection string.

2.9. Switching to an Oracle Database After Installation

About This Task

If you want to use an Oracle database with SAM or Schema Registry after you have performed your initial HDF installation or upgrade, you can switch to an Oracle database. Oracle databases 12c and 11g Release 2 are supported

Prerequisites

You have an Oracle database installed and configured.

Steps

1. Log into Ambari Server and shut down SAM or Schema Registry.
2. From the configuration screen, select Oracle as the database type and provide Oracle credentials, the JDBC connection string and click **Save**.
3. From the command line where Ambari Server is running, register the Oracle JDBC driver jar:

```
sudo ambari-server setup --jdbc-db=oracle --jdbc-driver=/usr/share/java/
ojdbc.jar
```

4. From the host where SAM or Schema Registry are installed, copy the JDBC jar to the following location, depending on which component you are updating.

   ```
    cp ojdbc6.jar /usr/hdf/current/registry/bootstrap/lib/.
    cp ojdbc6.jar /usr/hdf/current/streamline/bootstrap/lib/.
   ```

5. From the host where SAM or Schema Registry are installed, run the following command to create the required schemas for SAM or Schema Registry.

   ```
   export JAVA_HOME=/usr/jdk64/jdk1.8.0_112 ; source /usr/hdf/current/
   streamline/conf/streamline-env.sh ; /usr/hdf/current/streamline/bootstrap/
   bootstrap-storage.sh create

   export JAVA_HOME=/usr/jdk64/jdk1.8.0_112 ; source /usr/hdf/current/registry/
   conf/registry-env.sh ; /usr/hdf/current/registry/bootstrap/bootstrap-
   storage.sh create
   ```

   ### Note

   You only this command run once, from a single host, to prepare the database.

6. Confirm that new tables are created in the Oracle database.

7. From Ambari, restart SAM or Schema Registry.

8. If you are specifying an Oracle database for SAM, run the following command after you have restarted SAM.

   ```
   export JAVA_HOME=/usr/jdk64/jdk1.8.0_112 ; source /usr/hdf/current/
   streamline/conf/streamline-env.sh ; /usr/hdf/current/streamline/bootstrap/
   bootstrap.sh
   ```

9. Confirm that Sam or Schema Registry are available and turn off maintenance mode.

3. Installing the HDF Management Pack

About This Task

A management pack (mpack) 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.

Warning

If you are installing an HDF cluster only, begin the installation with a fresh Ambari instance. Do not install the HDF management pack on a system where HDP is already installed.

Steps

1. Back up your Ambari resources folder:

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

2. Download the Hortonworks HDF management pack. You can find the download location for your operating system in the *HDF Release Notes*.

3. Copy the bundle to `/tmp` on the node where you installed Ambari.

4. Install the management pack:

   ```
   ambari-server install-mpack \
   --mpack=/tmp/hdf-ambari-mpack-<version>.tar.gz \
   --purge
   --verbose
   ```

5. Restart the Ambari server:

   ```
   ambari-server restart
   ```

More Information

HDF Release Notes

4. Install an HDF Cluster Using Ambari

About This Task

After you start the Ambari service, you can open Ambari web in a browser and launch the Install wizard to prepare for installing an HDF cluster.

Steps

1. Navigate to http://<your.ambari.server>:8080, where <your.ambari.server> is the name of your Ambari server host.

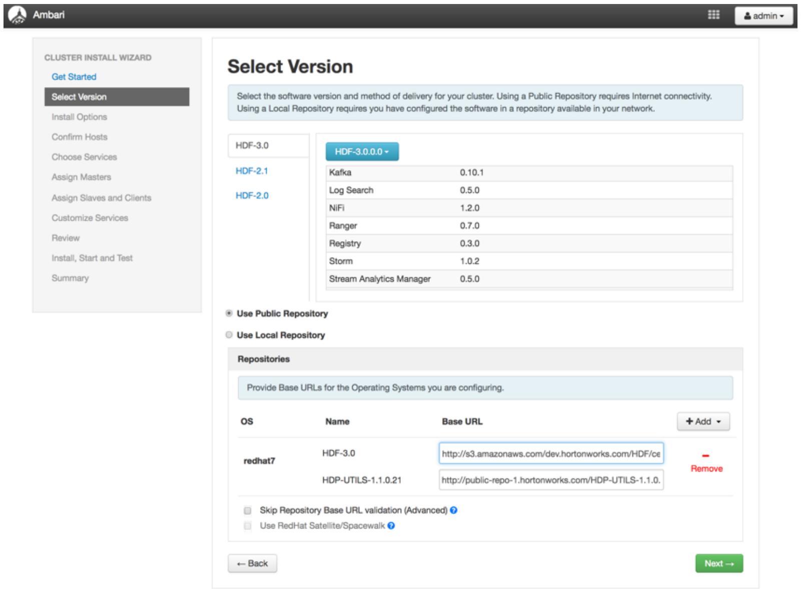
2. Log in to the Ambari server by using the default user name and password: admin and admin. You can change these credentials later.

3. In the **Ambari Welcome** page, choose **Launch Install Wizard**.

4. In the **Get Started** step, specify a name for your cluster.

5. In the **Select Version** page, remove all repositories except the one appropriate for your operating system. Change the Base URL for HDF to the base URL appropriate for your operating system. Find the HDF Base URLs in the HDF Release Notes.

   The wizard page should look similar to the following example:



Select Version

Select the software version and method of delivery for your cluster. Using a Public Repository requires Internet connectivity. Using a Local Repository requires you have configured the software in a repository available in your network.

Kafka	0.10.1
Log Search	0.5.0
NiFi	1.2.0
Ranger	0.7.0
Registry	0.3.0
Storm	1.0.2
Stream Analytics Manager	0.5.0

● Use Public Repository

○ Use Local Repository

Repositories

Provide Base URLs for the Operating Systems you are configuring.

OS	Name	Base URL
redhat7	HDF-3.0	http://s3.amazonaws.com/dev.hortonworks.com/HDF/ce
	HDP-UTILS-1.1.0.21	http://public-repo-1.hortonworks.com/HDP-UTILS-1.1.0.

□ Skip Repository Base URL validation (Advanced)

□ Use RedHat Satellite/Spacewalk

5. Configure HDF Components

You can customize your Hortonworks DataFlow (HDF) component configurations either during or after installation. During installation, you customize HDF component configurations in the **Customize Services** page of the installation wizard. After installation, you can navigate to Services > Configs in the Ambari dashboard.

5.1. Configure Schema Registry

About This Task

The main Schema Registry configuration task you have is to establish a connection between Schema Registry and the database you want to use as the metadata store.

Steps

1. In the **Customize Services** step, navigate to the **REGISTRY CONFIG** section of the **Registry** tab.

2. Select **Jar Storage Type** and then the storage type that you want to use.

   If you plan to enable HA for Schema Registry on this cluster, you must select **HDFS**.

3. If you selected **HDFS** as the **Jar Storage Type**, configure **Jar Storage HDFS URL**. This specifies the HDFS location where you want the jars to be stored. For example, `hdfs://<<NN_HOST:8020:/hdfs/registry`.

4. Configure **jar.storage** to the directory in which you want to store `.jar` files for serializers and deserializers.

5. Configure the **REGISTRY STORAGE** configurations based on the database you created to use as the Schema Registry metadata store.

6. Ensure that the registry storage connector URL has the fully qualified name of the host on which the database was installed and the connector url and default port for the database selected.

Example

```
MYSQL example:

jdbc:mysql://FQDN_MYSQL:3306/registry


Postgres Example:

jdbc:postgresql://FQDN_POSTGRES:5432/registry
```

5.2. Configure SAM

About This Task

When you configure Hortonworks Streaming Analytics Manager (SAM), you must provide information about the metadata store database, configure a connection with Schema Registry, and establish the URL for the Druid Supersets.

Steps

1. In the **Customize Services** step, navigate to the **STREAMLINE CONFIG** section of the **Streaming Analytics Manager** tab.

2. Select **Jar Storage Type**. If you plan to enable HA for SAM on this cluster, you must select **HDFS**.

3. If you selected **HDFS** as the **Jar Storage Type**, configure **Jar Storage HDFS URL**. This specifies the HDFS location where you want the jars to be stored. For example, `hdfs://<<NN_HOST:8020:/hdfs/registry`.

4. Configure **jar.storage** to the directory on which you want to store `.jar` files for custom processors.

5. Set the **streamline.dashboard.url** to the Superset URL which you can access using **Quick Links** for Druid.

6. Configure **registry.url** to the REST API Endpoint URL for the Registry.

   The format should be `http://$FQDN_REGISTRY_HOST:$REGISTRY_PORT/api/v1`, where

   - $FQDN_REGISTRY_HOST specifies the host on which you are running Schema Registry and

   - $REGISTRY_PORT specifies the Schema Registry port number, as in the following example:

     http://FQDN_REGISTRY_HOST:7788/api/v1

You can find the Schema Registry port in the **REGISTRY_CONFIG** section of the **Registry** tab.

7. Configure the **STREAMLINE STORAGE** configurations based on the database you created to use as a SAM metadata store.
8. Ensure that the registry storage connector URL has the fully qualified name of the host on which the database was installed and the connector url and default port for the database selected.

Example

```
MYSQL example:

jdbc:mysql://FQDN_MYSQL:3306/streamline


Postgres Example:

jdbc:postgresql://FQDN_POSTGRES:5432/streamline
```

5.3. Configure NiFi

About This Task

You use the **NiFi** tab in the **Customize Services** step to configure Apache NiFi. Generally, you can accept the defaults during initial installation. However, there are some settings that you must set before proceeding.

Steps

1. From **Advanced-nifi-ambari-config**, specify the **Encrypt Configuration Master Key Passwords**.

   This password is used when you generate the master key for sensitive properties encryption in the NiFi properties file when it is written to disk. It must contain at least 12 characters.

2. From **Advanced-nifi-ambari-config**, provide the **Sensitive property values encryption password**.

   This is the password used when you encrypt any sensitive property values that are configured in processors. For enhanced security, it should contain at least 10 characters.

5.4. Configure NiFi for Atlas Integration

About This Task

You can integrate NiFi with Apache Atlas to take advantage of robust dataset and application lineage support. You do this by configuring the NiFi ReportLineageToAtlas Reporting Task once you have NiFi configured and running.

Before You Begin

If NiFi is installed on an HDP cluster, you must be running HDP 2.6.4. If NiFi is installed on an HDF cluster managed by a separate Ambari instance, you must be running HDP 2.6.1 or later, and Apache Atlas 0.8.0 or later.

Steps

1. From the Global Menu located in NiFi’s upper right corner, select **Controller Services** and click the **Reporting Tasks** tab.

2. Click the **Add** (+) icon to launch the **Add Reporting Task** dialog.

3. Select ReportLineageToAtlas and click **Add**.

4. Click the **Edit** icon to launch the **Configure Reporting Task** dialog. The following Properties are required:

   - Atlas URLs – a comma-separated list of Atlas Server URLs. Once you have started reporting, you cannot modify an existing Reporting Task to add a new Atlas Server. When you need to add a new Atlas Server, you must create a new reporting task.

   - Atlas Authentication Method – Specifies how to authenticate the Reporting Task to the Atlas Server. Basic authentication is the default.

   - NiFi URL for Atlas – Specifies the NiFi cluster URL

   - NiFi Lineage Strategy – Specifies the level of granularity for your NiFi dataflow reporting to Atlas. Once you have started reporting, you should not switch between simple and complete lineage reporting strategies.

   - Provenance Record Start Position – Specifies where in the Provenance Events stream the Reporting Task should start.

   - Provenance Record Batch Size – Specifies how many records you want to send in a single batch

   - Create Atlas Configuration File – If enabled, the `atlas-application-properties` file and the `Atlas Configuration Directory` are automatically created when the Reporting Task starts.

   - Kafka Security Protocol – Specifies the protocol used to communicate with Kafka brokers to send Atlas hook notification messages. This value should match Kafka's security.protocol property value.

Result

Once you have ReportLineageToAtlas up and running, you may view dataset level lineage graphs in the Atlas UI.

Note

The default time interval for the Reporting Task to start sending data to an Atlas Server is 5 minutes so do not expect to see immediate lineage graphs.

You can change the default time interval in the Reporting Task property configuration.

More Information

For complete information, see the help included with the Reporting Task.

5.5. Configure Kafka

About This Task

You can configure Apache Kafka from the **Kafka** tab in the **Customize Services** step.

Steps

1. For your initial installation, accept the default values set by Apache Ambari.
2. If Ambari prompts you with Some configurations need your attention before you can proceed, review the list of properties and provide the required information.
3. Review the *Apache Kafka Component Guide* for information about configuring Apache Storm to meet your operational objectives.

More Information

Configuring Kafka for Production Environments

5.6. Configure Storm

About This Task

You can configure Storm from the **Storm** tab in the **Customize Services** step.

Steps

1. For your initial installation, accept the default values set by Ambari.
2. If Ambari prompts you with:

   Some configurations need your attention before you can proceed.

   Review the list of properties and provide the required information.
3. Review the *Apache Storm Component Guide* for information about configuring storm to meet your operational objectives.

More Information

Configuring Storm for Production Environments

5.7. Configure Log Search

About This Task

To ensure that you can view logs in the new SAM Log Search, you can manually review and adjust Log Search Settings for storm_worker and storm_worker_event.

Steps

1. From the left-hand navigation pane, select **Log Search | Configs**.
2. Manually set the Log Feeder Log Levels Filter for storm_worker and storm_worker_event to include **Info**, **Debug**, and **Trace**.

5.8. Deploy the Cluster Services

After you finish the wizard and deploy the cluster, some services might fail to start. If this is the case, you can start those services individually by launching them from the Ambari dashboard Services pane.

Steps

1. From Ambari's left-hand **Services** pane, click the service you want.
2. From the **Quick Links** drop-down, select the UI option.
3. Find links for the SAM UI under **Streaming Analytics Manager** and for the Schema Registry UI under **Registry.**

Result

The UI for your HDF service opens in a new window.

5.9. Access the UI for Deployed Services

About This Task

Once you have deployed your Ambari-managed cluster, you can launch the UI for any of the services from Ambari.

Steps

1. From Ambari's left-hand **Services** pane, click the service you want.
2. From the **Quick Links** drop-down, select the UI option.
3. Find links for the SAM UI under **Streaming Analytics Manager** and for the Schema Registry UI under **Registry.**

Result

The UI for your HDF service opens in a new window.

6. Configuring Schema Registry and SAM for High Availability

About This Task

You can configure Schema Registry and SAM for high availability.

Steps for Configuring SAM for HA

1. Install two or more instances of SAM on unique nodes.
2. From the **Services** pane, select **Streaming Analytics Manager** and click the **Configs** tab.
3. In the **Jar Storage Type** drop down, select **HDFS** or **Database**.

Note

If you are using a MySQL database, ensure that you make adjustments to the database configuration as well. `max_allowed_packet` must be greater than the maximum file size of any custom processor or user defined function that will be uploaded.

Steps for Configuring Schema Registry for HA

1. Install two or more instances of Schema Registry on unique nodes.
2. From the **Services** pane, select **Schema Registry** and click the **Configs** tab.
3. In the **Jar Storage Type** drop down, select **HDFS**.

7. Install the Storm Ambari View

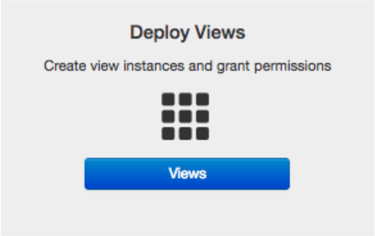
About This Task

The Storm Ambari view provides you a number of different troubleshooting and debugging tools.

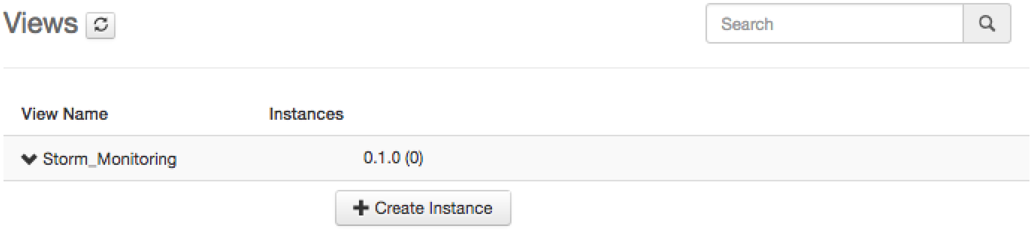
Steps

1. From the **admin** drop-down, select **Manage Ambari**.

2. Click the **Views** button.

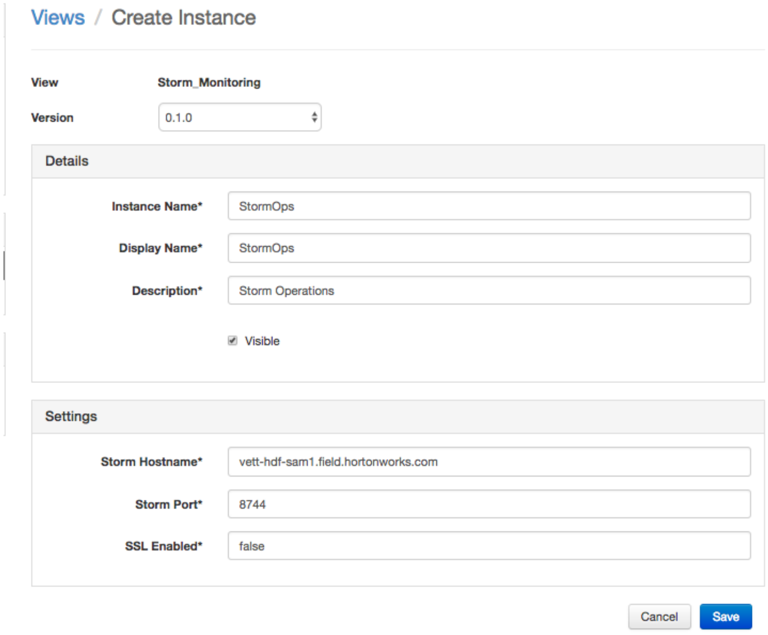


3. From the list of available Views, expand **Storm_Monitoring** and click + **Create Instance**.



View Name	Instances
Storm_Monitoring	0.1.0 (0)

4. Configure the Storm Ambari View.



View	Storm_Monitoring
Version	0.1.0
Instance Name*	StormOps
Display Name*	StormOps
Description*	Storm Operations
	☑ Visible
Storm Hostname*	vett-hdf-sam1.field.hortonworks.com
Storm Port*	8744
SSL Enabled*	false

a. **Instance Name** and **Display Name** may not have an spaces.

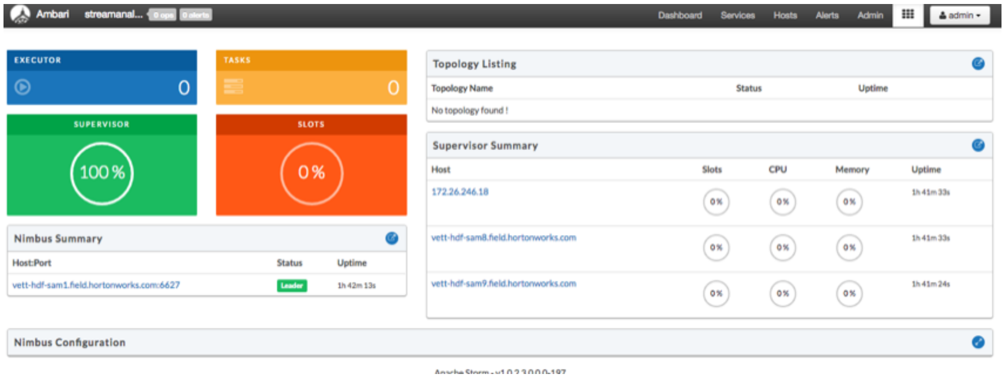
b. The **Storm Hostname** refers to the host where the Storm UI Server is deployed.

c. The Storm port is the Storm UI port server (keep it as default 8744 if you have not changed it).

d. Click **Save**.

Result

After saving it, you should see a menu item for the Storm Ambari View.



Nimbus Summary

Host:Port	Status	Uptime
vett-hdf-sam1.field.hortonworks.com:6627	Leader	1h 42m 13s

Topology Listing

Topology Name	Status	Uptime
No topology found !		

Supervisor Summary

Host	Slots	CPU	Memory	Uptime
172.26.246.18	0%	0%	0%	1h 41m 33s
vett-hdf-sam8.field.hortonworks.com	0%	0%	0%	1h 41m 33s
vett-hdf-sam9.field.hortonworks.com	0%	0%	0%	1h 41m 24s

8. Using a Local Repository

If your enterprise clusters have limited outbound Internet access, you should consider using a local repository, which enables you to benefit from more governance and better installation performance. You can also use a local repository for routine post-installation cluster operations such as service start and restart operations. Using a local repository includes obtaining public repositories, setting up the repository using either no internet access or limited internet access, and preparing the Apache Ambari repository configuration file to use your new local repository.

- Obtain Public Repositories from the HDF Release Notes
- Set up a local repository having:
  - Setting Up a Local Repository with No Internet Access [44]
  - Setting up a Local Repository with Temporary Internet Access [42]
- Preparing the Ambari Repository Configuration File to Use the Local Repository [46]

8.1. Setting Up a Local Repository

Based on your Internet access, choose one of the following options:

- No Internet Access

  This option involves downloading the repository tarball, moving the tarball to the selected mirror server in your cluster, and extracting the tarball to create the repository.

- Temporary Internet Access

  This option involves using your temporary Internet access to synchronize (using reposync) the software packages to your selected mirror server to create the repository.

  Both options proceed in a similar, straightforward way. Setting up for each option presents some key differences, as described in the following sections:

  - Preparing to Set Up a Local Repository [41]
  - Setting Up a Local Repository with No Internet Access [44]
  - Setting up a Local Repository with Temporary Internet Access [42]

8.1.1. Preparing to Set Up a Local Repository

Before setting up your local repository, you must have met certain requirements.

- Selected an existing server, in or accessible to the cluster, that runs a supported operating system.
- Enabled network access from all hosts in your cluster to the mirror server.

- Ensured that the mirror server has a package manager installed such as yum (for RHEL, CentOS, or Oracle Linux), zypper (for SLES), or apt-get (for Debian and Ubuntu).
- **Optional:** If your repository has temporary Internet access, and you are using RHEL, CentOS, or Oracle Linux as your OS, installed yum utilities:

  ```
  yum install yum-utils createrepo
  ```

After meeting these requirements, you can take steps to prepare to set up your local repository.

Steps

1. Create an HTTP server:
   a. On the mirror server, install an HTTP server (such as Apache httpd) using the instructions provided on the Apache community website.

   b. Activate the server.

   c. Ensure that any firewall settings allow inbound HTTP access from your cluster nodes to your mirror server.

   ### Note

   If you are using Amazon EC2, make sure that SELinux is disabled.

2. On your mirror server, create a directory for your web server.
   - For example, from a shell window, type:

For RHEL/CentOS/Oracle Linux:	`mkdir -p /var/www/html/`
For SLES:	`mkdir -p /srv/www/htdocs/rpms`
For Debian/Ubuntu:	`mkdir -p /var/www/html/`

   - If you are using a symlink, enable the `followsymlinks` on your web server.

Next Steps

You next must set up your local repository, either with no Internet access or with temporary Internet access.

More Information

httpd.apache.org/download.cgi

8.1.2. Setting up a Local Repository with Temporary Internet Access

Prerequisites

You must have completed the Getting Started Setting up a Local Repository procedure.

Steps

1. Install the repository configuration files for Ambari and the Stack on the host.

2. Confirm repository availability;

For RHEL, CentOS, or Oracle Linux:	`yum repolist`
For SLES:	`zypper repos`
For Debian and Ubuntu:	`dpkg-list`

3. Synchronize the repository contents to your mirror server:

   - Browse to the web server directory:

For RHEL, CentOS, or Oracle Linux:	`cd /var/www/html`
For SLES:	`cd /srv/www/htdocs/rpms`
For Debain and Ubuntu:	`cd /var/www/html`

   - For Ambari, create the `ambari` directory and reposync:

     ```
     mkdir -p ambari/<OS>
     ```

     ```
     cd ambari/<OS>
     ```

     ```
     reposync -r Updates-Ambari-2.6.1.0
     ```

     In this syntax, the value of <OS> is centos6, centos7, sles11, sles12, ubuntu14, ubuntu16, or debian7.

   - For Hortonworks Data Platform (HDP) stack repositories, create the hdp directory and reposync:

     ```
      mkdir -p hdp/<OS>
     ```

     ```
     cd hdp/<OS>
     ```

     ```
     reposync -r HDP-<latest.version>
     ```

     ```
     reposync -r HDP-UTILS-<version>
     ```

   - For HDF Stack Repositories, create an `hdf` directory and reposync.

     ```
      mkdir -p hdf/<OS>
     ```

     ```
     cd hdf/<OS>
     ```

     ```
     reposync -r HDF-<latest.version>
     ```

4. Generate the repository metadata:

For Ambari:	`createrepo <web.server.directory>/ambari/<OS>/Updates-Ambari-2.6.1.0`
For HDP Stack Repositories:	`createrepo <web.server.directory>/hdp/<OS>/HDP-<latest.version>`

```
createrepo <web.server.directory>/hdp/<OS>/
HDP-UTILS-<version>
```

For HDF Stack Repositories:

```
createrepo <web.server.directory>/hdf/<OS>/
HDF-<latest.version>
```

5. Confirm that you can browse to the newly created repository:

Ambari Base URL	http://<web.server>/ambari/<OS>/Updates-Ambari-2.6.1.0
HDF Base URL	http://<web.server>/hdf/<OS>/HDF-<latest.version>
HDP Base URL	http://<web.server>/hdp/<OS>/HDP-<latest.version>
HDP-UTILS Base URL	http://<web.server>/hdp/<OS>/HDP-UTILS-<version>

Where:

- <web.server> – The FQDN of the web server host
- <version> – The Hortonworks stack version number
- <OS> – centos6, centos7, sles11, sles12, ubuntu14, ubuntu16, or debian7

Important

Be sure to record these Base URLs. You will need them when installing Ambari and the Cluster.

6. Optional. If you have multiple repositories configured in your environment, deploy the following plug-in on all the nodes in your cluster.

   a. Install the plug-in.

   For RHEL and CentOS 7:

   ```
   yum install yum-plugin-priorities
   ```

   For RHEL and CentOS 6:

   ```
   yum install yum-plugin-priorities
   ```

   b. Edit the `/etc/yum/pluginconf.d/priorities.conf` file to add the following:

   ```
   [main]
   ```

   ```
   enabled=1
   ```

   ```
   gpgcheck=0
   ```

8.1.3. Setting Up a Local Repository with No Internet Access

Prerequisites

You must have completed the Getting Started Setting up a Local Repository procedure.

Steps

1. Obtain the compressed tape archive file (tarball) for the repository you want to create.

2. Copy the repository tarball to the web server directory and uncompress (untar) the archive:

   a. Browse to the web server directory you created.

For RHEL/CentOS/Oracle Linux:	`cd /var/www/html/`
For SLES:	`cd /srv/www/htdocs/rpms`
For Debian/Ubuntu:	`cd /var/www/html/`

   b. Untar the repository tarballs and move the files to the following locations, where <web.server>, <web.server.directory>, <OS>, <version>, and <latest.version> represent the name, home directory, operating system type, version, and most recent release version, respectively:

Ambari Repository	Untar under <web.server.directory>.
HDF Stack Repositories	Create a directory and untar it under `<web.server.direcotry>`/hdf.
HDP Stack Repositories	Create a directory and untar it under <web.server.directory>/hdp.

3. Confirm that you can browse to the newly created local repositories, where <web.server>, <web.server.directory>, <OS>, <version>, and <latest.version> represent the name, home directory, operating system type, version, and most recent release version, respectively:

Ambari Base URL	http://<web.server>/Ambari-2.6.1.0/<OS>
HDF Base URL	http://<web.server>/hdf/HDF/<OS>/3.x/updates/<latest.version>
HDP Base URL	http://<web.server>/hdp/HDP/<OS>/2.x/updates/<latest.version>
HDP-UTILS Base URL	http://<web.server>/hdp/HDP-UTILS-<version>/repos/<OS>

   ### Important

   Be sure to record these Base URLs. You will need them when installing Ambari and the cluster.

4. Optional: If you have multiple repositories configured in your environment, deploy the following plug-in on all the nodes in your cluster.

a. **For RHEL and CentOS 7:**	`yum install yum-plugin-priorities`
For RHEL and CentOS 6:	`yum install yum-plugin-priorities`

   b. Edit the `/etc/yum/pluginconf.d/priorities.conf` file to add the following values:

```
[main]
enabled=1
gpgcheck=0
```

8.2. Preparing the Ambari Repository Configuration File to Use the Local Repository

Steps

1. Download the `ambari.repo` file from the public repository:

   ```
   http://public-repo-1.hortonworks.com/ambari/<OS>/2.x/updates/2.6.1.0/ambari.repo
   ```

   In this syntax, <OS> is centos6, centos7, sles11, sles12, ubuntu14, ubuntu16, or debian7.

2. Edit the `ambari.repo` file and replace the Ambari Base URL `baseurl` obtained when setting up your local repository.

   ```
   [Updates-Ambari-2.6.1.0]
   name=Ambari-2.6.1.0-Updates
   baseurl=INSERT-BASE-URL
   gpgcheck=1
   gpgkey=http://public-repo-1.hortonworks.com/ambari/centos6/RPM-GPG-KEY/RPM-GPG-KEY-Jenkins
   enabled=1
   priority=1
   ```

   ### Note

   You can disable the GPG check by setting gpgcheck =0. Alternatively, you can keep the check enabled but replace gpgkey with the URL to GPG-KEY in your local repository.

   Base URL for a Local Repository

Built with Repository Tarball (No Internet Access)	http://<web.server>/Ambari-2.6.1.0/<OS>
Built with Repository File (Temporary Internet Access)	http://<web.server>/ambari/<OS>/Updates-Ambari-2.6.1.0

   where <web.server> = FQDN of the web server host, and <OS> is centos6, centos7, sles11, sles12, ubuntu12, ubuntu14, or debian7.

3. Place the `ambari.repo` file on the host you plan to use for the Ambari server:

For RHEL/CentOS/Oracle Linux:	`/etc/yum.repos.d/ambari.repo`

For SLES:	`/etc/zypp/repos.d/ambari.repo`
For Debain/Ubuntu:	`/etc/apt/sources.list.d/ambari.list`

4. Edit the `/etc/yum/pluginconf.d/priorities.conf` file to add the following values:

```
[main]
```

```
enabled=1
```

```
gpgcheck=0
```

9. Navigating the HDF Library

To navigate the Hortonworks DataFlow (HDF) documentation library, begin by deciding your current goal.

If you want to...	See this document...
Install or upgrade an HDF cluster using Apache Ambari	• Release Notes • Support Matrix • Planning Your Deployment • Ambari Upgrade • MiNiFi Java Agent Quick Start
Get started with HDF	• Getting Started with Apache NiFi • Getting Started with Stream Analytics
Use and administer HDF Flow Management capabilities	• Apache NiFi User Guide • Apache NiFi Administration Guide • Apache NiFi Developer Guide • Apache NiFi Expression Language Guide • MiNiFi Java Agent Administration Guide
Use and administer HDF Stream Analytics capabilities	• Streaming Analytics Manager User Guide • Schema Registry User Guide • Apache Storm Component Guide • Apache Kafka Component Guide