Hortonworks Data Platform

Automated Install with Ambari

(Apr 1, 2014)

docs.hortonworks.com

Hortonworks Data Platform : Automated Install with Ambari

Copyright © 2012, 2014 Hortonworks, Inc. Some rights reserved.

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

Unlike other providers of platforms built using Apache Hadoop, Hortonworks contributes 100% of our code back to the Apache Software Foundation. The Hortonworks Data Platform is Apache-licensed and completely open source. We sell only expert technical support, training and partner-enablement services. All of our technology is, and will remain free and open source.

Please visit the Hortonworks Data Platform page for more information on Hortonworks technology. For more information on Hortonworks services, please visit either the Support or Training page. Feel free to Contact Us directly to discuss your specific needs.

Except where otherwise noted, this document is licensed under
Creative Commons Attribution ShareAlike 3.0 License.
http://creativecommons.org/licenses/by-sa/3.0/legalcode

Table of Contents

1. Getting Ready 1
1.1. Determine Version Compatibility 1
1.2. Meet Minimum System Requirements 2
1.2.1. Hardware Recommendations 2
1.2.2. Operating Systems Requirements 2
1.2.3. Browser Requirements 2
1.2.4. Software Requirements 3
1.2.5. JDK Requirements 3
1.2.6. Database Requirements 4
1.2.7. File System Partitioning Recommendations 4
1.3. Collect Information 4
1.4. Prepare the Environment 5
1.4.1. Check Existing Installs 5
1.4.2. Set Up Password-less SSH 6
1.4.3. Set up Users and Groups 7
1.4.4. Enable NTP on the Cluster and on the Browser Host 7
1.4.5. Check DNS 7
1.4.6. Configuring iptables 8
1.4.7. Disable SELinux and PackageKit and check the umask Value 8
1.5. Optional: Configure Local Repositories 9
1.5.1. Setting Up a Local Server Having No Internet Access 10
1.5.2. Setting up a Local Server Having Limited Internet Access 14
2. Running the Ambari Installer 21
2.1. Set Up the Bits 21
2.1.1. RHEL/CentOS/Oracle Linux 5.x 21
2.1.2. RHEL/CentOS/Oracle Linux 6.x 22
2.1.3. SLES 11 22
2.2. Set Up the Server 22
2.2.1. Setup Options 23
2.3. Start the Ambari Server 24
3. Hadoop 1.x - Installing, Configuring, and Deploying the Cluster 25
3.1. Log into Apache Ambari 25
3.2. Welcome 25
3.3. Select Stack 25
3.4. Install Options 26
3.5. Confirm Hosts 27
3.6. Choose Services 27
3.7. Assign Masters 28
3.8. Assign Slaves and Clients 28
3.9. Customize Services 28
3.9.1. Service Users and Groups 29
3.9.2. Properties That Depend on Service Usernames/Groups 30
3.9.3. Recommended Memory Configurations for the MapReduce Service 31
3.10. Review 31
3.11. Install, Start and Test 31
3.12. Summary 32
4. Hadoop 2.x - Installing, Configuring, and Deploying the Cluster 33
4.1. Log into Apache Ambari 33

4.2. Welcome 33
4.3. Select Stack 33
4.4. Install Options 34
4.5. Confirm Hosts 35
4.6. Choose Services 35
4.7. Assign Masters 36
4.8. Assign Slaves and Clients 36
4.9. Customize Services 36
4.9.1. Service Users and Groups 37
4.9.2. Properties That Depend on Service Usernames/Groups 38
4.10. Review 39
4.11. Install, Start and Test 39
4.12. Summary 39

List of Tables

1.1. Deploying HDP - Option I 10
1.2. Options for *$os* parameter in repo URL 12
1.3. Options for *$os* parameter in repo URL 13
1.4. Deploying HDP - Option II 14
1.5. Options for *$os* parameter in repo URL 18
1.6. Options for *$os* parameter in repo URL 19
2.1. Download the repo 21
3.1. Service Users 29
3.2. Service Group 30
3.3. HDFS Settings: Advanced 30
3.4. MapReduce Settings: Advanced 30
4.1. Service Users 37
4.2. Service Group 38
4.3. HDFS Settings: Advanced 38
4.4. MapReduce Settings: Advanced 39

1. Getting Ready

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

- Determine Version Compatibility
- Meet Minimum System Requirements
- Collect Information
- Prepare the Environment
- Optional: Configure Local Repositories for Ambari

1.1. Determine Version Compatibility

Use this table to determine whether your Ambari and Hadoop stack versions are compatible.

Ambari	HDP 1.2.0	HDP 1.2.1.	HDP 1.3.0	HDP 1.3.2	HDP 1.3.3	HDP 2.0 [a]
1.5.0				X	X	X
1.4.4.23				X	X	X
1.4.3.38				X	X	X
1.4.2.104				X	X	X
1.4.1.61				X	X	X
1.4.1.25				X		X
1.2.5.17		X	X	X		
1.2.4.9	X	X	X			
1.2.3.7	X	X	X			
1.2.3.6	X	X				
1.2.2.5	X	X				
1.2.2.4	X	X				
1.2.2.3	X					
1.2.1.2	X					
1.2.0.1	X					

[a] HDP 2.0.6 stack (or later) patch releases.

For more information about the latest HDP patch releases, see HDP Documentation.

For more information about using Ambari and the Hadoop 1.x stack, see Hadoop 1.x-Deploying, Configuring, and Upgrading Ambari.

For more information about using Ambari and the Hadoop 2.x stack, see Hadoop 2.x-Deploying, Configuring, and Upgrading Ambari.

1.2. Meet Minimum System Requirements

To run Hadoop, your system must meet minimum requirements.

- Hardware Recommendations
- Operating Systems Requirements
- Browser Requirements
- Software Requirements
- JDK Requirements
- Database Requirements
- File System Partitioning Recommendations

1.2.1. Hardware Recommendations

There is no single hardware requirement set for installing Hadoop.

For more information on the parameters that may affect your installation, see Hardware Recommendations For Apache Hadoop.

1.2.2. Operating Systems Requirements

The following operating systems are supported:

- Red Hat Enterprise Linux (RHEL) v5.x or 6.x (64-bit)
- CentOS v5.x or 6.x (64-bit)
- Oracle Linux v5.x or 6.x (64-bit)
- SUSE Linux Enterprise Server (SLES) 11, SP1 (64-bit)

Important

The installer pulls many packages from the base OS repos. If you do not have a complete set of base OS repos available to all your machines at the time of installation you may run into issues.

If you encounter problems with base OS repos being unavailable, please contact your system administrator to arrange for these additional repos to be proxied or mirrored. For more information see Optional: Configure the Local Repositories

1.2.3. Browser Requirements

The Ambari Install Wizard runs as a browser-based Web app. You must have a machine capable of running a graphical browser to use this tool. The supported browsers are:

- Windows (Vista, 7)
  - Internet Explorer 9.0 and higher (for Vista + Windows 7)
  - Firefox latest stable release
  - Safari latest stable release
  - Google Chrome latest stable release
- Mac OS X (10.6 or later)
  - Firefox latest stable release
  - Safari latest stable release
  - Google Chrome latest stable release
- Linux (RHEL, CentOS, SLES, Oracle Linux)
  - Firefox latest stable release
  - Google Chrome latest stable release

1.2.4. Software Requirements

On each of your hosts:

- yum and rpm (RHEL/CentOS/Oracle Linux)
- zypper (SLES)
- scp, curl, and wget
- python (2.6 or later)

Important

The Python version shipped with SUSE 11, 2.6.0-8.12.2, has a critical bug that may cause the Ambari Agent to fail within the first 24 hours. If you are installing on SUSE 11, please update all your hosts to Python version 2.6.8-0.15.1.

1.2.5. JDK Requirements

The following Java runtimes are supported:

- Oracle JDK 1.6.0_31 64-bit
- Oracle JDK 1.7_45 64-bit (default)
- OpenJDK 7 64-bit

1.2.6. Database Requirements

Hive/HCatalog, Oozie, and Ambari all require their own internal databases.

- Hive/HCatalog: By default uses an Ambari-installed MySQL 5.x instance. With appropriate preparation, you can also use an existing MySQL 5.x or Oracle 11g r2 instance. See Using Non-Default Databases for more information on using existing instances.
- Oozie: By default uses an Ambari-installed Derby instance. With appropriate preparation, you can also use an existing MySQL 5.x or Oracle 11g r2 instance. See Using Non-Default Databases for more information on using existing instances.
- Ambari: By default uses an Ambari-installed PostgreSQL 8.x instance. With appropriate preparation, you can also use an existing Oracle 11g r2, or MySQL 5.x instance. See Using Non-Default Databases for more information on using existing instances.

1.2.7. File System Partitioning Recommendations

For information on setting up file system partitions on master and slave nodes in a HDP cluster, see File System Partitioning Recommendations.

1.3. Collect Information

To deploy your Hadoop installation, you need to collect the following information:

- The fully qualified domain name (FQDN) for each host in your system, and which components you want to set up on which host. The Ambari install wizard *does not* support using IP addresses. You can use `hostname -f` to check for the FQDN if you do not know it.

Note

While it is possible to deploy all of Hadoop on a single host, this is appropriate only for initial evaluation. In general you should use at least three hosts: one master host and two slaves.

- The base directories you want to use as mount points for storing:
  - NameNode data
  - DataNodes data
  - Secondary NameNode data
  - Oozie data
  - MapReduce data (Hadoop version 1.x)
  - YARN data (Hadoop version 2.x)
  - ZooKeeper data, if you install ZooKeeper
  - Various log, pid, and db files, depending on your install type

1.4. Prepare the Environment

To deploy your Hadoop instance, you need to prepare your deploy environment:

- Check Existing Installs
- Set up Password-less SSH
- Set up Users and Groups
- Enable NTP on the Cluster
- Check DNS
- Configure iptables
- Disable SELinux, PackageKit and Check umask Value

1.4.1. Check Existing Installs

Ambari automatically installs the correct versions of the files that are necessary for Ambari and Hadoop to run. Versions other than the ones that Ambari uses can cause problems in running the installer, so remove any existing installs that do not match the following lists.

	RHEL/CentOS/Oracle Linux v5	RHEL/CentOS/Oracle Linux v6	SLES 11
Ambari Server	• libffi 3.0.5-1.el5 • python26 2.6.8-2.el5 • python26-libs 2.6.8-2.el5 • postgresql 8.4.13-1.el6_3 • postgresql-libs 8.4.13-1.el6_3 • postgresql-server 8.4.13-1.el6_3	• postgresql 8.4.13-1.el6_3 • postgresql-libs 8.4.13-1.el6_3 • postgresql-server 8.4.13-1.el6_3	• postgresql 8.3.5-1 • postgresql-server 8.3.5-1 • postgresql-libs 8.3.5-1
Ambari Agent [a]	• libffi 3.0.5-1.el5 • python26 2.6.8-2.el5 • python26-libs 2.6.8-2.el5	None	None
Nagios Server[b]	• nagios 3.5.0-99 • nagios-devel 3.5.0-99 • nagios-www 3.5.0-99 • nagios-plugins 1.4.9-1	• nagios 3.5.0-99 • nagios-devel 3.5.0-99 • nagios-www 3.5.0-99 • nagios-plugins 1.4.9-1	• nagios 3.5.0-99 • nagios-devel 3.5.0-99 • nagios-www 3.5.0-99 • nagios-plugins 1.4.9-1
Ganglia Server[c]	• ganglia-gmetad 3.5.0-99 • ganglia-devel 3.5.0-99 • libganglia 3.5.0-99 • ganglia-web 3.5.7-99 • rrdtool 1.4.5-1.el5	• ganglia-gmetad 3.5.0-99 • ganglia-devel 3.5.0-99 • libganglia 3.5.0-99 • ganglia-web 3.5.7-99 • rrdtool 1.4.5-1.el6	• ganglia-gmetad 3.5.0-99 • ganglia-devel 3.5.0-99 • libganglia 3.5.0-99 • ganglia-web 3.5.7-99 • rrdtool 1.4.5-4.5.1
Ganglia Monitor[d]	• ganglia-gmond 3.5.0-99	• ganglia-gmond 3.5.0-99	• ganglia-gmond 3.5.0-99

	RHEL/CentOS/Oracle Linux v5	RHEL/CentOS/Oracle Linux v6	SLES 11
	• libganglia 3.5.0-99	• libganglia 3.5.0-99	• libganglia 3.5.0-99

[a]Installed on each host in your cluster. Communicates with the Ambari Server to execute commands.

[b]The host that runs the Nagios server

[c]The host that runs the Ganglia Server

[d]Installed on each host in the cluster. Sends metrics data to the Ganglia Collector.

1.4.2. Set Up Password-less SSH

To have Ambari Server automatically install Ambari Agents in all your cluster hosts, you must set up password-less SSH connections between the main installation (Ambari Server) host and all other machines. The Ambari Server host acts as the client and uses the key-pair to access the other hosts in the cluster to install the Ambari Agent.

Note

You can choose to install the Agents on each cluster host manually. In this case you do not need to setup SSH. See Appendix: Installing Ambari Agents Manually for more information.

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.

```
ssh root@{remote.target.host}
```

You may see this warning. This happens on your first connection and is normal.

```
Are you sure you want to continue connecting (yes/no)?
```

6. 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.

1.4.3. Set up Users and Groups

The Ambari cluster installer automatically creates user and group accounts for you. Ambari preserves any existing user and group accounts, and uses these accounts when configuring Hadoop services. User and group creation applies to user/group accounts on the local operating system and to LDAP/AD accounts.

To set up custom accounts before running the Ambari installer, see Service Users and Groups (for the 1.x stack) or Service Users and Groups (for the 2.x stack) for more information about customizing service users and groups.

1.4.4. 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 Ambari Web must be able to synchronize with each other.

1.4.5. Check DNS

All hosts in your system must be configured for DNS and Reverse DNS.

If you are unable to configure DNS and Reverse DNS, you must edit the hosts file on every host in your cluster to contain the address of each of your hosts and to set the Fully Qualified Domain Name hostname of each of those hosts. The following instructions cover basic hostname network setup for generic Linux hosts. Different versions and flavors of Linux might require slightly different commands. Please refer to your specific operating system documentation for the specific details for your system.

1.4.5.1. 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
```

Note

Do **not** remove the following two lines from your host file, or various programs that require network functionality may fail.

```
127.0.0.1 localhost.localdomain localhost
::1 localhost6.localdomain6 localhost6
```

1.4.5.2. Set the Hostname

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

```
hostname fully.qualified.domain.name
```

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

```
hostname -f
```

This should return the name you just set.

1.4.5.3. Edit the Network Configuration File

1. Using a text editor, open the network configuration file on every host. This file is used to 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
NETWORKING_IPV6=yes
HOSTNAME=fully.qualified.domain.name
```

1.4.6. 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`.

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

You can restart `iptables` after setup is complete.

If the security protocols at your installation do not allow you to disable `iptables`, you can proceed with them on, as long as all of the relevant ports are open and available.

During the Ambari Server setup process, Ambari checks to see if `iptables` is running. If it is, a warning prints to remind you to check that the necessary ports are open and available. The **Host Confirm** step of the Cluster Install Wizard will also issue a warning for each host that has `iptables` running.

Important

If you leave iptables enabled **and** do not set up the necessary ports, the cluster installation will fail.

1.4.7. Disable SELinux and PackageKit and check the umask Value

1. SELinux must be temporarily disabled for the Ambari setup to function. Run the following command on each host in your cluster:

```
setenforce 0
```

2. On the RHEL/CentOS installation host, if PackageKit is installed, open `/etc/yum/pluginconf.d/refresh-packagekit.conf` with a text editor and make this change:

```
enabled=0
```

Note

PackageKit is not enabled by default on SLES. Unless you have specifically enabled it, you do not need to do this step.

3. Make sure umask is set to 022.

1.5. Optional: Configure Local Repositories

If your cluster includes a firewall that prevents or limits Internet access, you can set up local repositories to deploy Ambari and HDP.

Before deploying HDP on a cluster having no or limited Internet access:

1. Review your deployment strategies.
2. Compare specific deployment options.
3. Write down the Base URL of the local mirror repository for each operating system. You select these Base URLs during the cluster install procedure.

   For example, if your system includes hosts running CentOS 6, pointed to a HDP 2.0.10.0 repository, your local repository Base URL would look something like this:

   http://{your.hosted.local.repository}/HDP-2.0.10.0/repos/centos6

Important

Each relative path must reference your local repository configuration. If your cluster includes multiple operating systems (for example, CentOS 6 and RHEL 6), you must configure a unique repository for each OS. A host in your cluster can retrieve software packages only from a repository configured for that operating system.

4. Choose a JDK version to deploy, how to download it, and where to install it on each host. See JDK Requirements for more information on supported JDKs.

   - If you have not already installed the JDK on all hosts, and plan to use Oracle JDK 1.7, download jdk-7u45-linux-x64.tar.gz to `/var/lib/ambari-server/resources`.
   - If you plan to use a JDK other than Oracle JDK 1.7, you **must** install that JDK on each host in your cluster.
   - Set the Java Home path on each host, using -j option when running `ambari-server setup`. See Setup Options for more information about using the -j option to set JAVA_Home.

Note

If you have already installed the JDK on all your hosts, you **must** include the -j option when running `ambari-server setup`.

5. Set up local repositories for Ambari, and HDP based on your preferred strategy, as described in one of the following sections:
   - Local Server Having no Internet Access
   - Local Server Having Limited Internet Access
6. Set up the Ambari server.

1.5.1. Setting Up a Local Server Having No Internet Access

Complete the following instructions to set up a mirror server that has no access to the Internet:

- Check Your Prerequisites
- Install the Repos

1.5.1.1. Check Your Prerequisites

Select a mirror server host with the following characteristics:

- This server runs on either CentOS (v5.x, v6.x), RHEL (v5.x, v6.x), Oracle Linux(v5.x, v6.x), or SLES 11 and has several GB of storage available.
- This server and the cluster nodes are all running the same OS.

Note

To support repository mirroring for heterogeneous clusters requires a more complex procedure than the one documented here.

- The firewall lets all cluster nodes (the servers on which you want to install HDP) to access this server.
- Ensure that the mirror server has **yum** installed.
- Add the **yum-utils** and **createrepo** packages on the mirror server.

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

1.5.1.2. Install the Repos

1. Use a workstation with access to the Internet and download the tarball image of the appropriate Hortonworks yum repository.

Table 1.1. Deploying HDP - Option I

Cluster OS	HDP Repository Tarballs

RHEL/ CentOS/ Oracle Linux 5.x	`wget http://public-repo-1.hortonworks.com/HDP/centos5/HDP-2.0.10.0-centos5-rpm.tar.gz`
	
	`wget http://public-repo-1.hortonworks.com/HDP-UTILS-1.1.0.16/repos/centos5/HDP-UTILS-1.1.0.16-centos5.tar.gz`

Cluster OS	HDP Repository Tarballs
	`wget http://public-repo-1.hortonworks.com/ambari/centos5/ambari-1.5.0-centos5.tar.gz`

RHEL/ CentOS/ Oracle Linux 6.x	`wget http://public-repo-1.hortonworks.com/HDP/centos6/HDP-2.0.10.0-centos6-rpm.tar.gz`
	
	`wget http://public-repo-1.hortonworks.com/HDP-UTILS-1.1.0.16/repos/centos6/HDP-UTILS-1.1.0.16-centos6.tar.gz`
	
	`wget http://public-repo-1.hortonworks.com/ambari/centos6/ambari-1.5.0-centos6.tar.gz`

SLES 11	`wget http://public-repo-1.hortonworks.com/HDP/suse11/HDP-2.0.10.0-suse11-rpm.tar.gz`
	
	`wget http://public-repo-1.hortonworks.com/HDP-UTILS-1.1.0.16/repos/suse11/HDP-UTILS-1.1.0.16-suse11.tar.gz`
	
	`wget http://public-repo-1.hortonworks.com/ambari/suse11/ambari-1.5.0-suse11.tar.gz`

2. Create an HTTP server.

   a. On the mirror server, install an HTTP server (such as Apache httpd) using the instructions provided here.

   b. Activate this web server.

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

   Note

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

3. On your mirror server, create a directory for your web server.

   - For example, from a shell window, type:

     - **For RHEL/CentOS/Oracle:**

       ```
       mkdir –p /var/www/html/hdp/
       ```

     - **For SLES:**

       ```
       mkdir –p /srv/www/htdocs/rpms
       ```

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

4. Copy the HDP Repository Tarball to the directory created in step 3, and untar it.

5. Verify the configuration.

   - The configuration is successful, if you can access the above directory through your web browser.

     To test this out, browse to the following location: `http://$yourwebserver/hdp/$os/HDP-2.0.10.0/`.

You should see directory listing for all the HDP components along with the RPMs at: *$os*/HDP-2.0.10.0.

Note

If you are installing a 2.x.0 release, use: `http://$yourwebserver/hdp/$os/2.x/GA`

If you are installing a 2.x.x release, use:`http://$yourwebserver/hdp/$os/2.x/updates`

where

- *$os* can be `centos5`, `centos6`, or `suse11`, . Use the following options table for *$os* parameter:

Table 1.2. Options for *$os* parameter in repo URL

Operating System	Value
CentOS 5	centos5
RHEL 5	
Oracle Linux 5	
CentOS 6	centos6
RHEL 6	
Oracle Linux 6	
SLES 11	suse11

6. Configure the **yum** clients on all the nodes in your cluster.

   a. Fetch the yum configuration file from your mirror server.

   ```
   http://<$yourwebserver>/hdp/$os/2.x/updates/2.0.10.0/hdp.repo
   ```

   b. Store the `hdp.repo` file to a temporary location.

   c. Edit `hdp.repo` file changing the value of the **baseurl** property to point to your local repositories based on your cluster OS.

   ```
   [HDP-2.x]
   name=Hortonworks Data Platform Version - HDP-2.x
   baseurl=http://$yourwebserver/HDP/$os/2.x/GA
   gpgcheck=1
   gpgkey=http://public-repo-1.hortonworks.com/HDP/$os/RPM-GPG-KEY/RPM-GPG-
   KEY-Jenkins
   enabled=1
   priority=1

   [HDP-UTILS-1.1.0.16]
   name=Hortonworks Data Platform Utils Version - HDP-UTILS-1.1.0.16
   baseurl=http://$yourwebserver/HDP-UTILS-1.1.0.16/repos/$os
   gpgcheck=1
   gpgkey=http://public-repo-1.hortonworks.com/HDP/$os/RPM-GPG-KEY/RPM-GPG-
   KEY-Jenkins
   ```

```
enabled=1
priority=1

[HDP-2.0.10.0]
name=Hortonworks Data Platform HDP-2.0.10.0
baseurl=http://$yourwebserver/HDP/$os/2.x/updates/2.0.10.0
gpgcheck=1
gpgkey=http://public-repo-1.hortonworks.com/HDP/$os/RPM-GPG-KEY/RPM-GPG-
KEY-Jenkins
enabled=1
priority=1
```

where

- *$yourwebserver* is FQDN of your local mirror server.
- *$os* can be centos5, centos6, or suse11 . Use the following options table for *$os* parameter:

Table 1.3. Options for *$os* parameter in repo URL

Operating System	Value
CentOS 5	centos5
RHEL 5	
Oracle Linux 5	
CentOS 6	centos6
RHEL 6	
Oracle Linux 6	
SLES 11	suse11

- Use **scp** or **pdsh** to copy the client yum configuration file to /etc/yum.repos.d/ directory on every node in the cluster.

d. [Conditional]: If you have multiple repositories configured in your environment, deploy the following plugin on all the nodes in your cluster.

i. Install the plugin.

- **For RHEL and CentOs v5.x**

```
yum install yum-priorities
```

- **For RHEL and CentOs v6.x**

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

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

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

1.5.2. Setting up a Local Server Having Limited Internet Access

Complete the following instructions to set up a mirror server that has limited, or temporary access to the Internet:

1. Check Your Prerequisites
2. Install the Repos

1.5.2.1. Check Your Prerequisites

The local mirror server host must have the following characteristics:

- This server runs on either CentOS/RHEL/Oracle Linux 5.x or 6.x, or SLES 11 and has several GB of storage available.
- The local mirror server and the cluster nodes must have the same OS. If they are not running CentOS or RHEL, the mirror server must not be a member of the Hadoop cluster.

Note

To support repository mirroring for heterogeneous clusters requires a more complex procedure than the one documented here.

- The firewall allows all cluster nodes (the servers on which you want to install HDP) to access this server.
- Ensure that the mirror server has **yum** installed.
- Add the **yum-utils** and **createrepo** packages on the mirror server.

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

1.5.2.2. Install the Repos

- Temporarily reconfigure your firewall to allow Internet access from your mirror server host.
- Execute the following command to download the appropriate Hortonworks yum client configuration file and save it in `/etc/yum.repos.d/` directory on the mirror server host.

Table 1.4. Deploying HDP - Option II

Cluster OS	HDP Repository Tarballs
RHEL/ CentOS/ Oracle Linux 5.x	`wget http://public-repo-1.hortonworks.com/HDP/centos5/2.x/updates/2.0.10.0/hdp.repo -O /etc/yum.repos.d/hdp.repo` `wget http://public-repo-1.hortonworks.com/ambari/centos5/1.x/updates/1.5.0/ambari.repo -O /etc/yum.repos.d/ambari.repo`
RHEL/ CentOS/ Oracle Linux 6.x	`wget http://public-repo-1.hortonworks.com/HDP/centos6/2.x/updates/2.0.10.0/hdp.repo -O /etc/yum.repos.d/hdp.repo`

Cluster OS	HDP Repository Tarballs
	`wget http://public-repo-1.hortonworks.com/ambari/centos6/1.x/updates/1.5.0/ambari.repo -O /etc/yum.repos.d/ambari.repo`

SLES 11	`wget http://public-repo-1.hortonworks.com/HDP/suse11/2.x/updates/2.0.10.0/hdp.repo -O /etc/zypp/hdp.repo`
	
	`wget http://public-repo-1.hortonworks.com/ambari/suse11/1.x/updates/1.5.0/ambari.repo -O /etc/zypp/ambari.repo`

- Create an HTTP server.

  1. On the mirror server, install an HTTP server (such as Apache `httpd`) using the instructions provided http://httpd.apache.org/download.cgi
  2. Activate this web server.
  3. Ensure that the firewall settings (if any) allow inbound HTTP access from your cluster nodes to your mirror server.

     Note

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

  4. Optional - If your mirror server uses SLES, modify the `default-server.conf` file to enable the docs root folder listing.

     ```
     sed -e "s/Options None/Options Indexes MultiViews/ig" /etc/apache2/
     default-server.conf > /tmp/tempfile.tmp
     mv /tmp/tempfile.tmp /etc/apache2/default-server.conf
     ```

1.5.2.3. Setting up a Local Repository Having Temporary Internet Access

After completing the Getting Started Setting up a Local Repository procedure, finish setting up your repository by completing the following steps:

1. Put the repository configuration files for Ambari and the Stack in place on the host. For options, see Obtaining the Repositories.
2. Confirm the repositories are available.

   For **RHEL/CentOS/Oracle Linux:**

   ```
   yum repolist
   ```

   For **SLES:**

   ```
   zypper repos
   ```

3. Browse to the web server directory.

   For **RHEL/CentOS/Oracle Linux:**

   ```
   cd /var/www/html
   ```

   For **SLES:**

   ```
   cd /srv/www/htdocs/rpms
   ```

4. Synchronize the repository contents to your mirror server.

   - For Ambari, create `ambari` directory and reposync.

     ```
     mkdir -p ambari/{$os}
     cd ambari/{$os}
     reposync -r Updates-ambari-1.5.0
     ```

   - For HDP Stack Repositories, create `hdp` directory and reposync.

     ```
     mkdir -p hdp/{$os}
     cd hdp/{$os}
     reposync -r HDP-{$latest}
     reposync -r HDP-UTILS-{$version}
     ```

5. On your mirror server, create a directory for your web server.

   - For example, from a shell window, type:

     - **For RHEL/CentOS/Oracle:**

       ```
       mkdir –p /var/www/html/hdp/
       ```

     - **For SLES:**

       ```
       mkdir –p /srv/www/htdocs/rpms
       ```

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

6. Copy the contents of entire HDP repository for your desired OS from the remote yum server to your local mirror server.

   - Continuing the previous example, from a shell window, type:

     - **For RHEL/CentOS/Oracle:**

       ```
       cd /var/www/html/hdp
       ```

     - **For SLES:**

       ```
       cd /srv/www/htdocs/rpms
       ```

     Then for all hosts, type:

     - HDP Repository

       ```
       reposync -r HDP
       reposync -r HDP-2.0.10.0
       reposync -r HDP-UTILS-1.1.0.16
       ```

       You should see both an `HDP-2.0.10.0` directory and an `HDP-UTILS-1.1.0.16` directory, each with several subdirectories.

     - Optional - Ambari Repository

       ```
       reposync –r ambari-1.x
       reposync –r $release_type-ambari-1.5.0
       ```

7. Generate appropriate metadata.

This step defines each directory as a yum repository. From a shell window, type:

- For RHEL/CentOS/Oracle:
  - HDP Repository:

```
createrepo /var/www/html/hdp/HDP-2.0.10.0
createrepo /var/www/html/hdp/HDP-UTILS-1.1.0.16
```

  - Optional - Ambari Repository:

```
createrepo /var/www/html/hdp/ambari-1.x
createrepo /var/www/html/hdp/$release_type-ambari-1.5.0
```

- For SLES:
  - HDP Repository:

```
createrepo /srv/www/htdocs/rpms/hdp/HDP
```

  - Optional - Ambari Repository:

```
createrepo /srv/www/htdocs/rpms/hdp/ambari-1.x
createrepo /srv/www/htdocs/rpms/hdp/$release_type-ambari-1.5.0
```

You should see a new folder called `repodata` inside both HDP directories.

8. Verify the configuration.
   - The configuration is successful, if you can access the above directory through your web browser.

     To test this out, browse to the following location:
     - HDP: **http://*$yourwebserver*/hdp/HDP-2.0.10.0/**
     - Optional - Ambari Repository: **http://*$yourwebserver*/hdp/ambari/*$os*/1.x/updates/1.5.0**
   - You should now see directory listing for all the HDP components.
9. At this point, you can disable external Internet access for the mirror server, so that the mirror server is again entirely within your data center firewall.
10. Depending on your cluster OS, configure the **yum** clients on all the nodes in your cluster
    a. Edit the repo files, changing the value of the baseurl property to the local mirror URL.
       - Edit the `/etc/yum.repos.d/hdp.repo` file, changing the value of the `baseurl` property to point to your local repositories based on your cluster OS.

```
[HDP-2.x]
name=Hortonworks Data Platform Version - HDP-2.x
baseurl=http://$yourwebserver/HDP/$os/2.x/GA
gpgcheck=1
gpgkey=http://public-repo-1.hortonworks.com/HDP/$os/RPM-GPG-KEY/RPM-
GPG-KEY-Jenkins
```

```
enabled=1
priority=1

[HDP-UTILS-1.1.0.16]
name=Hortonworks Data Platform Utils Version - HDP-UTILS-1.1.0.16
baseurl=http://$yourwebserver/HDP-UTILS-1.1.0.16/repos/$os
gpgcheck=1
gpgkey=http://public-repo-1.hortonworks.com/HDP/$os/RPM-GPG-KEY/RPM-
GPG-KEY-Jenkins
enabled=1
priority=1

[HDP-2.0.10.0]
name=Hortonworks Data Platform HDP-2.0.10.0
baseurl=http://$yourwebserver/HDP/$os/2.x/updates/2.0.10.0
gpgcheck=1
gpgkey=http://public-repo-1.hortonworks.com/HDP/$os/RPM-GPG-KEY/RPM-
GPG-KEY-Jenkins
enabled=1
priority=1
```

where

- *$yourwebserver* is FQDN of your local mirror server.
- *$os* can be centos5, centos6, or suse11. Use the following options table for *$os* parameter:

Table 1.5. Options for *$os* parameter in repo URL

Operating System	Value
CentOS 5	centos5
RHEL 5	
Oracle Linux 5	
CentOS 6	centos6
RHEL 6	
Oracle Linux 6	
SLES 11	suse11

- Edit the /etc/yum.repos.d/ambari.repo file, changing the value of the baseurl property to point to your local repositories based on your cluster OS.

```
[ambari-1.x]
name=Ambari 1.x
baseurl=http://$yourwebserver/hdp/ambari/$os/1.x/updates/ambari.repo
gpgcheck=0
gpgkey=http://public-repo-1.hortonworks.com/ambari/$os/RPM-GPG-KEY/RPM-
GPG-KEY-Jenkins
enabled=1
priority=1

[HDP-UTILS-1.1.0.16]
name=Hortonworks Data Platform Utils Version - HDP-UTILS-1.1.0.16
baseurl=http://$yourwebserver/HDP-UTILS-1.1.0.16/repos/$os
gpgcheck=0
gpgkey=http://public-repo-1.hortonworks.com/ambari/$os/RPM-GPG-KEY/RPM-
GPG-KEY-Jenkins
```

```
enabled=1
priority=1

[$release_type-ambari-1.5.0
name=ambari-1.5.0 - updates
baseurl=http://$yourwebserver/ambari/$os/1.x/updates/1.5.0
gpgcheck=0
gpgkey=http://public-repo-1.hortonworks.com/ambari/$os/RPM-GPG-KEY/RPM-
GPG-KEY-Jenkins
enabled=1
priority=1
```

- *$yourwebserver* is FQDN of your local mirror server.
- *$os* can be centos5, centos6, or suse11. Use the following options table for *$os* parameter:

Table 1.6. Options for *$os* parameter in repo URL

Operating System	Value
CentOS 5	centos5
RHEL 5	
Oracle Linux 5	
CentOS 6	centos6
RHEL 6	
Oracle Linux 6	
SLES 11	suse11

b. Copy the yum/zypper client configuration file to all nodes in your cluster.

- RHEL/CentOS/Oracle Linux:

  Use **scp** or **pdsh** to copy the client yum configuration file to **/etc/yum.repos.d/** directory on every node in the cluster.

- For SLES:

  On every node, invoke the following command:

  - HDP Repository: **zypper addrepo -r http://$yourwebserver/hdp/HDP/suse11/2.x/updates/2.0.10.0/hdp.repo**
  - Optional - Ambari Repository: **zypper addrepo -r http://$yourwebserver/hdp/ambari/suse11/1.x/updates/1.5.0/ambari.repo**
  - If using Ambari, verify the configuration by deploying Ambari server on one of the cluster nodes. **yum install ambari-server**

11. If your cluster runs CentOS, Oracle, or RHEL and if you have multiple repositories configured in your environment, deploy the following plugin on all the nodes in your cluster.

    a. Install the plugin.

- **For RHEL and CentOs v5.x**

```
yum install yum-priorities
```

- **For RHEL and CentOs v6.x**

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

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

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

2. Running the Ambari Installer

This section describes the process for installing Apache Ambari. Ambari manages installing and deploying Hadoop.

2.1. Set Up the Bits

1. Log into the machine that is to serve as the Ambari Server as `root`. You may login and `sudo` as `su` if this is what your environment requires. This machine is the main installation host.

2. Download the the Ambari repository file and copy it to your repos.d.

Table 2.1. Download the repo

Platform	Access
RHEL, CentOS, and Oracle Linux 5	wget http://public-repo-1.hortonworks.com/ambari/centos5/1.x/updates/1.5.0/ambari.repo cp ambari.repo /etc/yum.repos.d
RHEL, CentOS and Oracle Linux 6	wget http://public-repo-1.hortonworks.com/ambari/centos6/1.x/updates/1.5.0/ambari.repo cp ambari.repo /etc/yum.repos.d
SLES 11	wget http://public-repo-1.hortonworks.com/ambari/suse11/1.x/updates/1.5.0/ambari.repo cp ambari.repo /etc/zypp/repos.d

Note

If your cluster does not have access to the Internet, or you are creating a large cluster and you want to conserve bandwidth, you need to provide access to the bits using an alternative method. For more information, see Optional: Configure the Local Repositories section.

When you have the software, continue your installation based on your base platform.

2.1.1. RHEL/CentOS/Oracle Linux 5.x

1. Confirm the repository is configured by checking the repo list.

```
yum repolist
```

You should see the Ambari and HDP utilities repositories in the list. The version values vary depending the installation.

```
  repo id                 repo name
| AMBARI-1.x             | Ambari 1.x
| HDP-UTILS-1.1.0.16     | Hortonworks Data Platform Utils
```

2. Install the Ambari bits using yum. This also installs PostgreSQL.

```
yum install ambari-server
```

2.1.2. RHEL/CentOS/Oracle Linux 6.x

1. Confirm the repository is configured by checking the repo list.

   ```
   yum repolist
   ```

   You should see the Ambari and HDP utilities repositories in the list. The version values vary depending the installation.

   ```
     repo id                 repo name
   | AMBARI-1.x            | Ambari 1.x
   | HDP-UTILS-1.1.0.16    | Hortonworks Data Platform Utils
   ```

2. Install the Ambari bits using yum. This also installs PostgreSQL.

   ```
   yum install ambari-server
   ```

2.1.3. SLES 11

1. Verify that `php_curl` is installed before you begin.

   ```
   zypper se 'php_curl*'
   ```

   If `php_curl` is not installed, install it:

   ```
   zypper in php_curl
   ```

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

   ```
   zypper repos
   ```

   You should see the Ambari and HDP utilities in the list. The version values vary depending the installation.

   ```
   # | Alias                | Name
   1 | AMBARI.dev-1.x       | Ambari 1.x
   2 | HDP-UTILS-1.1.0.16   | Hortonworks Data Platform Utils
   ```

3. Install the Ambari bits using zypper. This also installs PostgreSQL.

   ```
   zypper install ambari-server
   ```

2.2. Set Up the Server

The `ambari-server` command manages the setup process. Run the following command and respond to the prompts:

```
ambari-server setup
```

1. If you have *not* temporarily disabled SELinux, you may get a warning. Enter y to continue.

2. By default, Ambari Server runs under `root`. If you want to create a different user to run the Ambari Server instead, or to assign a previously created user, select `y` at **Customize user account for ambari-server daemon** and give the prompt the username you want to use.

3. If you have not temporarily disabled `iptables` you may get a warning. Enter `y` to continue.

4. Agree to the Oracle JDK license when asked. You must accept this license to be able to download the necessary JDK from Oracle. The JDK is installed during the deploy phase.

Note

By default, Ambari Server setup will download and install Oracle JDK 1.7. If you plan to download this JDK and install on all your hosts, or plan to use a different version of the JDK, skip this step and see Setup Options for more information

5. At **Enter advanced database configuration**:

   - To use the default PostgreSQL database, named `ambari`, with the default username and password (`ambari/bigdata`), enter `1`.

Important

If you are using an existing Oracle or MySQL database instance, you must prepare the instance using the steps detailed in Using Non-Default Databases **before** running the installer.

   - To use an existing Oracle 11g r2 instance, and select your own database name, username and password for that database, enter 2.

     Select the database you want to use and provide any information required by the prompts, including hostname, port, Service Name or SID, username, and password.

   - To use an existing MySQL 5.x database, and select your own database name, username and password for that database, enter 3.

     Select the database you want to use and provide any information required by the prompts, including hostname, port, database name, username, and password.

6. Setup completes.

Note

If your host accesses the Internet through a proxy server, you must configure Ambari Server to use this proxy server. See Configure Ambari Server for Internet Proxy for more information.

2.2.1. Setup Options

The following table describes options frequently used for Ambari Server setup.

Option	Description
-j –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, Setup downloads the Oracle JDK 1.7 binary to `/var/lib/ambari-server/resources` on the Ambari Server and installs the JDK to `/usr/jdk64`. Use this option when you plan to use a JDK other than the default Oracle JDK 1.7. See JDK Requirements for more information on the supported JDKs. 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. This path must be valid on all hosts. For example. `ambari-server setup –j /usr/java/default`
-s –silent	Setup runs silently. Accepts all default prompt values.
-v –verbose	Prints verbose info and warning messages to the console during Setup.
-i –jdk-location	Use specified JDK file in local filesystem instead of downloading
-g –debug	Start Ambari Server in debug mode

2.3. Start the Ambari Server

- To start the Ambari Server:

```
ambari-server start
```

- To check the Ambari Server processes:

```
ps -ef | grep Ambari
```

- To stop the Ambari Server:

```
ambari-server stop
```

3. Hadoop 1.x - Installing, Configuring, and Deploying the Cluster

This section describes using the Ambari install wizard in your browser to complete your installation, configuration and deployment of Hadoop.

3.1. Log into Apache Ambari

Once you have started the Ambari service, you can access the Ambari Install Wizard through your browser.

1. Point your browser to `http://{main.install.hostname}:8080`.
2. Log in to the Ambari Server using the default username/password: admin/admin. You can change this later to whatever you want.

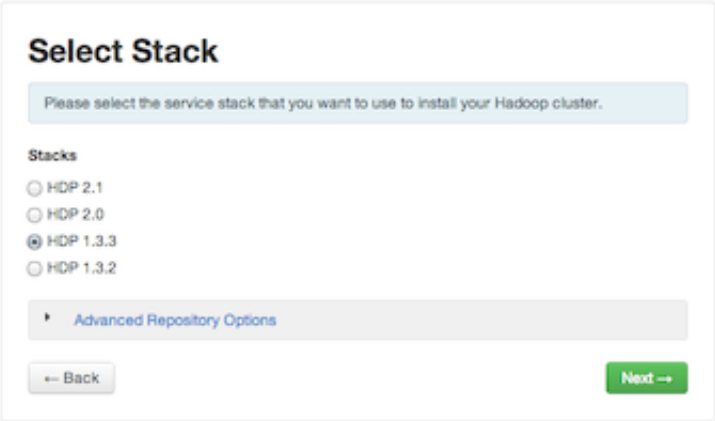
3.2. Welcome

The first step creates the cluster name.

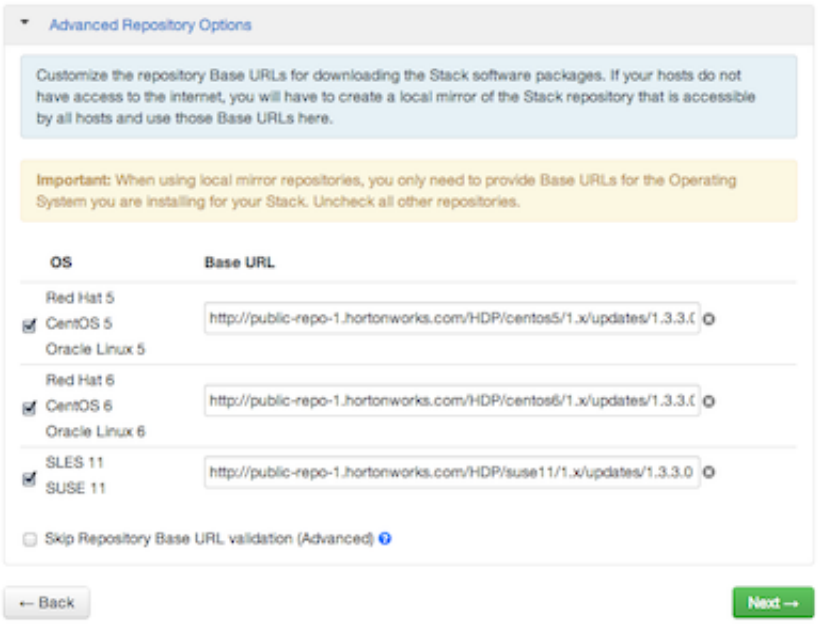
1. At the **Welcome** page, type a name for the cluster you want to create in the text box. No whitespaces or special characters can be used in the name.
2. Click the **Next** button.

3.3. Select Stack

The Service Stack (or simply the Stack) is a coordinated and tested set of Hadoop components. Use the radio button to select the Stack version you want to install. To install a Hadoop 1 stack, select HDP 1.3.3 under Stacks.



Under Advanced Repository Options, you can specify the Base URLs of your local repositories for each Operating System you plan to use in your cluster. You should have configured the Base URLs for your local repositories in Optional: Configure Ambari for Local Repositories.



3.4. Install Options

In order to build up the cluster, the install wizard needs to know general information about how you want to set up your cluster. You need to supply the FQDN of each of your hosts. The wizard also needs to access the private key file you created in Set Up Password-less SSH. It uses these to locate all the hosts in the system and to access and interact with them securely.

1. Use the **Target Hosts** text box to enter your list of host names, one per line. You can use ranges inside brackets to indicate larger sets of hosts. For example, for <domain>host01.<domain> through <domain>host10.<domain> use `<domain>host[01-10].<domain>`

   Note

   If you are deploying on EC2, use the **internal Private DNS** hostnames.

2. If you want to let Ambari automatically install the Ambari Agent on all your hosts using SSH, select **Provide your SSH Private Key** and either use the **Choose File** button in the **Host Registration Information** section to find the private key file that matches the public key you installed earlier on all your hosts or cut and paste the key into the text box manually.

   Note

   If you are using IE 9, the **Choose File** button may not appear. Use the text box to cut and paste your private key manually.

   Fill in the username for the SSH key you have selected. If you do not want to use `root`, you must provide the username for an account that can execute `sudo` without entering a password.

3. If you do not want Ambari to automatically install the Ambari Agents, select **Perform manual registration**. See Appendix: Installing Ambari Agents Manually for more information.

4. Click the **Register and Confirm** button to continue.

3.5. Confirm Hosts

This screen lets you confirm that Ambari has located the correct hosts for your cluster and to check those hosts to make sure they have the correct directories, packages, and processes to continue the install.

If any hosts were selected in error, you can remove them by selecting the appropriate checkboxes and clicking the grey **Remove Selected** button. To remove a single host, click the small white **Remove** button in the Action column.

At the bottom of the screen, you may notice a yellow box that indicates some warnings were encountered during the check process. For example, your host may have already had a copy of `wget` or `curl`. Select **Click here to see the warnings** to see a list of what was checked and what caused the warning. On the same page you can get access to a python script that can help you clear any issues you may encounter and let you run **Rerun Checks**.

Important

If you are deploying HDP using Ambari 1.4 or later on RHEL 6.5 you will likely see Ambari Agents fail to register with Ambari Server during the "Confirm Hosts" step in the Cluster Install wizard. Click the "Failed" link on the Wizard page to display the Agent logs. The following log entry indicates the SSL connection between the Agent and Server failed during registration:

```
INFO 2014-04-02 04:25:22,669 NetUtil.py:55 - Failed
to connect to https://<ambari-server>:8440/cert/ca due
to [Errno 1] _ssl.c:492: error:100AE081:elliptic curve
routines:EC_GROUP_new_by_curve_name:unknown group
```

For more information about this issue, see the Ambari Troubleshooting Guide.

When you are satisfied with the list of hosts, click **Next**.

3.6. Choose Services

Hortonworks Data Platform is made up of a number of services. You must at minimum install HDFS. You can decide which of the other services to install.

1. Select **all** to preselect all items or **minimum** to preselect only HDFS.

2. Use the checkboxes to unselect (if you have used **all)** or select (if you have used **minimum**) to arrive at your desired list of components.

Note

If you want to use Ambari for monitoring your cluster, make sure you select **Nagios** and **Ganglia.** If you do not select them, you get a warning popup

when you finish this section. If you are using other monitoring tools, you can ignore the warning.

3. When you have made your selections, click **Next**.

3.7. Assign Masters

The Ambari install wizard attempts to assign the master nodes for various services you have selected to appropriate hosts in your cluster. The right column shows the current service assignments by host, with the hostname and its number of CPU cores and amount of RAM indicated.

1. To change locations, click the dropdown list next to the service in the left column and select the appropriate host.

2. To remove a ZooKeeper instance, click the green minus icon next to the host address you want to remove.

3. When you are satisfied with the assignments, click the **Next** button.

3.8. Assign Slaves and Clients

The Ambari install wizard attempts to assign the slave components (DataNodes, NodeManagers, and RegionServers) to appropriate hosts in your cluster. It also attempts to select hosts for installing the appropriate set of clients.

1. Use **all** or **none** to select all of the hosts in the column or none of the hosts, respectively.

   If a host has a red asterisk next to it, that host is also running one or more master components. Hover your mouse over the asterisk to see which master components are on that host.

2. Fine-tune your selections by using the checkboxes next to specific hosts.

Note

As an option you can start the HBase REST server manually after the install process is complete. It can be started on any host that has the HBase Master or the Region Server installed. If you attempt to start it on the same host as the Ambari server, however, you need to start it with the -p option, as its default port is 8080 and that conflicts with the Ambari Web default port.

```
/usr/lib/hbase/bin/hbase-daemon.sh start rest -p
 <custom_port_number>
```

3. When you are satisfied with your assignments, click the **Next** button.

3.9. Customize Services

The **Customize Services** screen presents you with a set of tabs that let you manage configuration settings for Hadoop components. The wizard attempts to set reasonable

defaults for each of the options here, but you can use this set of tabs to tweak those settings. and you are strongly encouraged to do so, as your requirements may be slightly different. Pay particular attention to the directories suggested by the installer.

Note

In **HDFS Services Configs General**, make sure to enter an integer value, in bytes, that sets the HDFS maximum edit log size for checkpointing. A typical value is 500000000.

Hover your mouse over each of the properties to see a brief description of what it does. The number of tabs you see is based on the type of installation you have decided to do. In a complete installation there are nine groups of configuration properties and other related options, such as database settings for Hive and Oozie and admin name/password and alert email for Nagios.

The install wizard sets reasonable defaults for all properties except for those related to databases in the Hive tab and the Oozie tab, and two related properties in the Nagios tab. These four are marked in red and are the only ones you *must* set yourself.

Note

If you decide to use an existing database instance for Hive/HCatalog or for Oozie, you must have completed the preparations described in Using Non-Default Databases **prior** to running the install wizard.

Click the name of the group in each tab to expand and collapse the display.

3.9.1. Service Users and Groups

The individual services in Hadoop are each run under the ownership of a corresponding Unix account. These accounts are known as service users. These service users belong to a special Unix group. In addition there is a special service user for running smoke tests on components during installation and on-demand using the Management Header in the **Services** View of the Ambari Web GUI. Any of these users and groups can be customized using the **Misc** tab of the **Customize Services** step.

If you choose to customize names, Ambari checks to see if these custom accounts already exist. If they do not exist, Ambari creates them. The default accounts are always created during installation whether or not custom accounts are specified. These default accounts are not used and can be removed post-install.

Note

All new service user accounts, and any existing user accounts used as service users, must have a UID >= 1000.

Table 3.1. Service Users

Service	Component	Default User Account
HDFS	NameNode	hdfs

Service	Component	Default User Account
	SecondaryNameNode DataNode	
MapReduce	JobTracker HistoryServer TaskTracker	mapred
Hive	Hive Metastore HiveServer2	hive
HCat	HCatalog Server	hcat
WebHCat	WebHCat Server	hcat
Oozie	Oozie Server	oozie
HBase	MasterServer RegionServer	hbase
ZooKeeper	ZooKeeper	zookeeper
Ganglia	Ganglia Server Ganglia Collectors	nobody
Nagios	Nagios Server	nagios[a]
Smoke Test[b]	All	ambari-qa

[a]If you plan to use an existing user account named "nagios", that "nagios" account must be in a group named "nagios". If you customize this account, that account will be created and put in a group "nagios".

[b]The Smoke Test user performs smoke tests against cluster services as part of the install process. It also can perform these on-demand from the Ambari Web GUI.

Table 3.2. Service Group

Service	Components	Default Group Account
All	All	hadoop

3.9.2. Properties That Depend on Service Usernames/ Groups

Some properties must be set to match specific service usernames or service groups. If you have set up non-default, customized service usernames for the HDFS or HBase service or the Hadoop group name, you must edit the following properties:

Table 3.3. HDFS Settings: Advanced

Property Name	Value
dfs.permissions.supergroup	The same as the HDFS username. The default is "hdfs"
dfs.cluster.administrators	A single space followed by the HDFS username.
dfs.block.local-path-access.user	The HBase username. The default is "hbase".

Table 3.4. MapReduce Settings: Advanced

Property Name	Value
mapreduce.tasktracker.group	The Hadoop group name. The default is "hadoop".
mapreduce.cluster.administrators	A single space followed by the Hadoop group name.

3.9.3. Recommended Memory Configurations for the MapReduce Service

The following recommendations can help you determine appropriate memory configurations based on your usage scenario:

- Make sure that there is enough memory for all the processes. Remember that system processes take around 10% of the available memory.
- For co-deploying an HBase RegionServer and MapReduce service on the same node, reduce the RegionServer's heap size (use the **HBase Settings: RegionServer**: `HBase Region Servers maximum Java heap size` property to modify the RegionServer heap size).
- For co-deploying an HBase RegionServer and the MapReduce service on the same node, or for memory intensive MapReduce applications, modify the map and reduce slots as suggested in the following example:

EXAMPLE: For co-deploying an HBase RegionServer and the MapReduce service on a machine with 16GB of available memory, the following would be a recommended configuration:

2 GB: system processes

8 GB: MapReduce slots. 6 Map + 2 Reduce slots per 1 GB task

4 GB: HBase RegionServer

1 GB: TaskTracker

1 GB: DataNode

To change the number of Map and Reduce slots based on the memory requirements of your application, use the following properties:

MapReduce Settings: TaskTracker: `Number of Map slots per node`

MapReduce Settings: TaskTracker: `Number of Reduce slots per node`

3.10. Review

The assignments you have made are displayed. Check to make sure everything is correct. If you need to make changes, use the left navigation bar to return to the appropriate screen.

To print your information for later reference, click **Print.**

When you are satisfied with your choices, click the **Deploy** button.

3.11. Install, Start and Test

The progress of the install is shown on the screen. Each component is installed and started and a simple test is run on the component. You are given an overall status on the process in the progress bar at the top of the screen and a host by host status in the main section.

To see specific information on what tasks have been completed per host, click the link in the **Message** column for the appropriate host. In the **Tasks** pop-up, click the individual task to see the related log files. You can select filter conditions by using the **Show** dropdown list. To see a larger version of the log contents, click the **Open** icon or to copy the contents to the clipboard, use the **Copy** icon.

Depending on which components you are installing, the entire process may take 40 or more minutes. Please be patient.

When **Successfully installed and started the services** appears, click **Next**.

3.12. Summary

The Summary page gives you a summary of the accomplished tasks. Click **Complete**. You are taken to the Ambari Web GUI.

4. Hadoop 2.x - Installing, Configuring, and Deploying the Cluster

This section describes using the Ambari install wizard in your browser to complete your installation, configuration and deployment of Hadoop.

4.1. Log into Apache Ambari

Once you have started the Ambari service, you can access the Ambari Install Wizard through your browser.

1. Point your browser to `http://{your.ambari.server}:8080`.

2. Log in to the Ambari Server using the default username/password: admin/admin. You can change this later to whatever you want.

4.2. Welcome

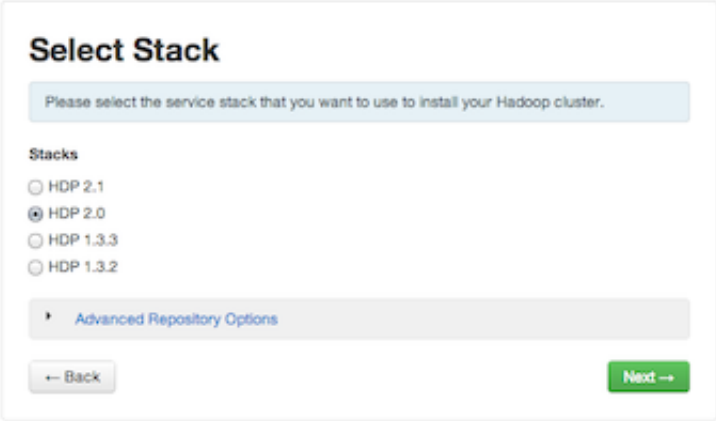
The first step creates the cluster name.

1. At the **Welcome** page, type a name for the cluster you want to create in the text box. No whitespaces or special characters can be used in the name.

2. Click the **Next** button.

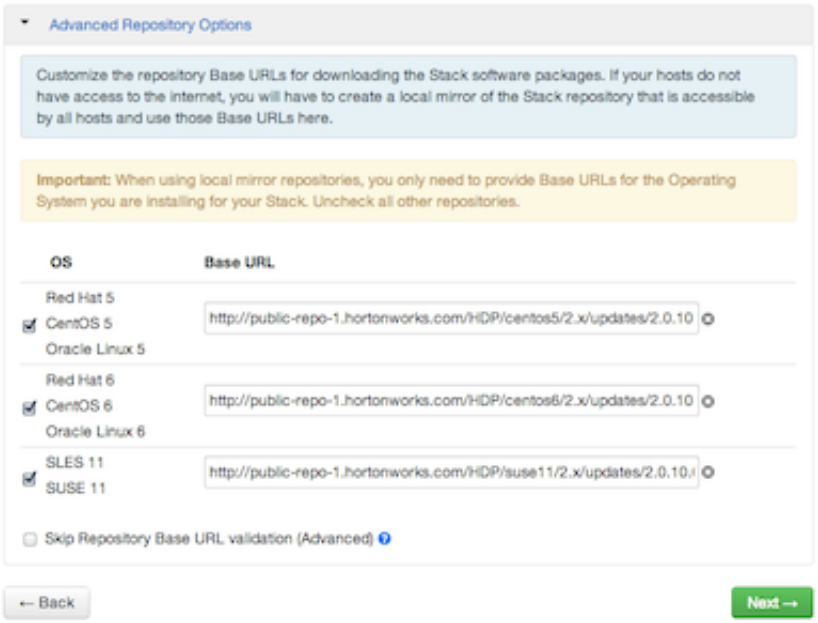
4.3. Select Stack

The Service Stack (or simply the Stack) is a coordinated and tested set of Hadoop components. Use the radio button to select the Stack version you want to install. To install a Hadoop 2 stack, select HDP 2.0.6, in Stacks.



Under Advanced Repository Options, you can specify the Base URLs of your local repositories for each Operating System you plan to use in your cluster. You should have

configured the Base URLs for your local repositories in Optional: Configure Ambari for Local Repositories.



4.4. Install Options

In order to build up the cluster, the install wizard needs to know general information about how you want to set it up. You need to supply the FQDN of each of your hosts. The wizard also needs to access the private key file you created in Set Up Password-less SSH. It uses these to locate all the hosts in the system and to access and interact with them securely.

1. Use the **Target Hosts** text box to enter your list of host names, one per line. You can use ranges inside brackets to indicate larger sets of hosts. For example, for host01.domain through host10.domain use `host[01-10].domain`

   Note

   If you are deploying on EC2, use the **internal Private DNS** hostnames.

2. If you want to let Ambari automatically install the Ambari Agent on all your hosts using SSH, select **Provide your SSH Private Key** and either use the **Choose File** button in the **Host Registration Information** section to find the private key file that matches the public key you installed earlier on all your hosts or cut and paste the key into the text box manually.

   Note

   If you are using IE 9, the **Choose File** button may not appear. Use the text box to cut and paste your private key manually.

   Fill in the username for the SSH key you have selected. If you do not want to use `root`, you must provide the username for an account that can execute `sudo` without entering a password.

3. If you do not want Ambari to automatically install the Ambari Agents, select **Perform manual registration**. See Appendix: Installing Ambari Agents Manually for more information.

4. Click the **Register and Confirm** button to continue.

4.5. Confirm Hosts

This screen lets you confirm that Ambari has located the correct hosts for your cluster and to check those hosts to make sure they have the correct directories, packages, and processes to continue the install.

If any hosts were selected in error, you can remove them by selecting the appropriate checkboxes and clicking the grey **Remove Selected** button. To remove a single host, click the small white **Remove** button in the Action column.

At the bottom of the screen, you may notice a yellow box that indicates some warnings were encountered during the check process. For example, your host may have already had a copy of `wget` or `curl`. Click **Click here to see the warnings** to see a list of what was checked and what caused the warning. On the same page you can get access to a python script that can help you clear any issues you may encounter and let you run **Rerun Checks**.

Important

If you are deploying HDP using Ambari 1.4 or later on RHEL 6.5 you will likely see Ambari Agents fail to register with Ambari Server during the "Confirm Hosts" step in the Cluster Install wizard. Click the "Failed" link on the Wizard page to display the Agent logs. The following log entry indicates the SSL connection between the Agent and Server failed during registration:

```
INFO 2014-04-02 04:25:22,669 NetUtil.py:55 - Failed
to connect to https://<ambari-server>:8440/cert/ca due
to [Errno 1] _ssl.c:492: error:100AE081:elliptic curve
routines:EC_GROUP_new_by_curve_name:unknown group
```

For more information about this issue, see the Ambari Troubleshooting Guide.

When you are satisfied with the list of hosts, click **Next**.

4.6. Choose Services

Hortonworks Data Platform is made up of a number of services. You must at minimum install HDFS. You can decide which of the other services to install.

1. Select **all** to preselect all items or **minimum** to preselect only HDFS.

2. Use the checkboxes to unselect (if you have used **all)** or select (if you have used **minimum**) to arrive at your desired list of components.

Note

If you want to use Ambari for monitoring your cluster, make sure you select **Nagios** and **Ganglia.** If you do not select them, you get a warning popup

when you finish this section. If you are using other monitoring tools, you can ignore the warning.

3. When you have made your selections, click **Next**.

4.7. Assign Masters

The Ambari install wizard attempts to assign the master nodes for various services you have selected to appropriate hosts in your cluster. The right column shows the current service assignments by host, with the hostname and its number of CPU cores and amount of RAM indicated.

1. To change locations, click the dropdown list next to the service in the left column and select the appropriate host.

2. To remove a ZooKeeper instance, click the green minus icon next to the host address you want to remove.

3. When you are satisfied with the assignments, click the **Next** button.

4.8. Assign Slaves and Clients

The Ambari install wizard attempts to assign the slave components (DataNodes, NodeManagers, and RegionServers) to appropriate hosts in your cluster. It also attempts to select hosts for installing the appropriate set of clients.

1. Use **all** or **none** to select all of the hosts in the column or none of the hosts, respectively.

   If a host has a red asterisk next to it, that host is also running one or more master components. Hover your mouse over the asterisk to see which master components are on that host.

2. Fine-tune your selections by using the checkboxes next to specific hosts.

Note

As an option you can start the HBase REST server manually after the install process is complete. It can be started on any host that has the HBase Master or the Region Server installed. If you attempt to start it on the same host as the Ambari server, however, you need to start it with the -p option, as its default port is 8080 and that conflicts with the Ambari Web default port.

```
/usr/lib/hbase/bin/hbase-daemon.sh start rest -p
 <custom_port_number>
```

3. When you are satisfied with your assignments, click the **Next** button.

4.9. Customize Services

The **Customize Services** screen presents you with a set of tabs that let you manage configuration settings for Hadoop components. The wizard attempts to set reasonable

defaults for each of the options here, but you can use this set of tabs to tweak those settings. and you are strongly encouraged to do so, as your requirements may be slightly different. Pay particular attention to the directories suggested by the installer.

Note

In **HDFS Services Configs General**, make sure to enter an integer value, in bytes, that sets the HDFS maximum edit log size for checkpointing. A typical value is 500000000.

Hover over each of the properties to see a brief description of what it does. The number of tabs you see is based on the type of installation you have decided to do. In a complete installation there are ten groups of configuration properties and other related options, such as database settings for Hive/HCat and Oozie, and admin name/password and alert email for Nagios.

The install wizard sets reasonable defaults for all properties except for those related to databases in the Hive and the Oozie tabs, and two related properties in the Nagios tab. These four are marked in red and are the only ones you *must* set yourself.

Note

If you decide to use an existing database instance for Hive/HCatalog or for Oozie, you must have completed the preparations described in Using Non-Default Databases **prior** to running the install wizard.

Click the name of the group in each tab to expand and collapse the display.

4.9.1. Service Users and Groups

The individual services in Hadoop are each run under the ownership of a corresponding Unix account. These accounts are known as service users. These service users belong to a special Unix group. In addition there is a special service user for running smoke tests on components during installation and on-demand using the Management Header in the **Services** View of the Ambari Web GUI. Any of these users and groups can be customized using the **Misc** tab of the **Customize Services** step.

If you choose to customize names, Ambari checks to see if these custom accounts already exist. If they do not exist, Ambari creates them. The default accounts are always created during installation whether or not custom accounts are specified. These default accounts are not used and can be removed post-install.

Note

All new service user accounts, and any existing user accounts used as service users, must have a UID >= 1000.

Table 4.1. Service Users

Service	Component	Default User Account
HDFS	NameNode SecondaryNameNode	hdfs

Service	Component	Default User Account
	DataNode	
MapReduce2	HistoryServer	mapred
YARN	NodeManager ResourceManager	yarn
Hive	Hive Metastore HiveServer2	hive
HCat	HCatalog Server	hcat
WebHCat	WebHCat Server	hcat
Oozie	Oozie Server	oozie
HBase	MasterServer RegionServer	hbase
ZooKeeper	ZooKeeper	zookeeper
Ganglia	Ganglia Server Ganglia Monitors	nobody
Ganglia	RRDTool (with Ganglia Server)	rrdcahed[a]
Ganglia	Apache HTTP Server	apache[b]
PostgreSQL	PostgreSQL (with Ambari Server)	postgres[c]
Nagios	Nagios Server	nagios[d]
Smoke Test[e]	All	ambari-qa

[a]Created as part of installing RRDTool, which is used to store metrics data collected by Ganglia.

[b]Created as part of installing Apache HTTP Server, which is used to serve the Ganglia web UI.

[c]Created as part of installing the default PostgreSQL database with Ambari Server. If you are not using the Ambari PostgreSQL database, this user is not needed.

[d]If you plan to use an existing user account named “nagios”, that “nagios” account must either be in a group named “nagios” or you must customize the Nagios Group.

[e]The Smoke Test user performs smoke tests against cluster services as part of the install process. It also can perform these on-demand from the Ambari Web GUI.

Table 4.2. Service Group

Service	Components	Default Group Account
All	All	hadoop
Nagios	Nagios Server	nagios
Ganglia	Ganglia Server Ganglia Monitor	nobody

4.9.2. Properties That Depend on Service Usernames/ Groups

Some properties must be set to match specific service usernames or service groups. If you have set up non-default, customized service usernames for the HDFS or HBase service or the Hadoop group name, you must edit the following properties:

Table 4.3. HDFS Settings: Advanced

Property Name	Value
dfs.permissions.superusergroup	The same as the HDFS username. The default is "hdfs"

Property Name	Value
dfs.cluster.administrators	A single space followed by the HDFS username.
dfs.block.local-path-access.user	The HBase username. The default is "hbase".

Table 4.4. MapReduce Settings: Advanced

Property Name	Value
mapreduce.cluster.administrators	A single space followed by the Hadoop group name.

4.10. Review

The assignments you have made are displayed. Check to make sure everything is correct. If you need to make changes, use the left navigation bar to return to the appropriate screen.

To print your information for later reference, click **Print.**

When you are satisfied with your choices, click the **Deploy** button.

4.11. Install, Start and Test

The progress of the install is shown on the screen. Each component is installed and started and a simple test is run on the component. You are given an overall status on the process in the progress bar at the top of the screen and a host by host status in the main section.

To see specific information on what tasks have been completed per host, click the link in the **Message** column for the appropriate host. In the **Tasks** pop-up, click the individual task to see the related log files. You can select filter conditions by using the **Show** dropdown list. To see a larger version of the log contents, click the **Open** icon or to copy the contents to the clipboard, use the **Copy** icon.

Depending on which components you are installing, the entire process may take 40 or more minutes. Please be patient.

When **Successfully installed and started the services** appears, click **Next**.

4.12. Summary

The Summary page gives you a summary of the accomplished tasks. Click **Complete**. You are taken to the Ambari Web GUI.