Hortonworks Data Platform

Reference

1.0 (Nov 30, 2012)

Hortonworks Data Platform : Reference

1.0 (2012-11-30)

Hortonworks Data Platform (HDP) is a 100% open source data management platform based on Apache Hadoop. It allows you to load, store, process and manage data in virtually any format and at any scale. As the foundation for the next generation enterprise data architecture, HDP includes all of the necessary components to begin uncovering business insights from the quickly growing streams of data flowing into and throughout your business – TODO - verify

Table of Contents

List of Tables

1. Deploying HDP In Production Data Centers with Firewalls

1.1. Deployment Strategies for Data Centers with Firewalls

A typical Hortonworks Data Platform (HDP) install requires access to the Internet in order to fetch software packages from a remote repository. Since corporate networks typically have various levels of firewalls, these firewalls may limit or restrict Internet access, making it impossible for your cluster nodes to access the HDP repository during the install process.

The solution for this is to either:

- Create a local mirror repository inside your firewall hosted on a local mirror server inside your firewall; or
- Provide a trusted proxy server inside your firewall that can access the hosted repositories.

This document will cover these two options in detail, discuss the trade-offs, provide configuration guidelines, and will also provide recommendations for your deployment strategy.

In general, before installing Hortonworks Data Platform in a production data center, it is best to ensure that both the Data Center Security team and the Data Center Networking team are informed and engaged to assist with these aspects of the deployment.

1.1.1. Terminology

Table 1.1. Terminology

Item	Description
Yum Package Manager (yum)	A package management tool that fetches and installs software packages and performs automatic dependency resolution. See http://yum.baseurl.org/ for more information.
Local Mirror Repository	The yum repository hosted on your Local Mirror Server that will serve the HDP software.
Local Mirror Server	The server in your network that will host the Local Mirror Repository. This server must be accessible from all hosts in your cluster where you will install HDP.
HDP Repositories	A set of repositories hosted by Hortonworks that contains the HDP software packages. HDP software packages include the HDP Repository and the HDP-UTILS Repository.
HDP Repository Tarball	A tarball image that contains the complete contents of the HDP Repositories.

1.1.2. Options for Mirroring or Proxying

HDP uses yum to install software, and this software is obtained from the: HDP Repositories, and the Extra Packages for Enterprise Linux (EPEL) repository.

If your firewall prevents Internet access, it will be necessary to mirror and/or proxy both the HDP repository and the Extra Packages for Enterprise Linux (EPEL) repository. Many Data Centers already mirror or proxy the EPEL repository, so discuss with your Data Center team whether EPEL is already available from within your firewall.

Mirroring a repository involves copying the entire repository and all its contents onto a local server and enabling an HTTPD service on that server to serve the repository locally. Once the local mirror server setup is complete, the `*.repo` configuration files on every repository client (i.e. cluster nodes) must be updated, so that the given package names are associated with the local mirror server instead of the remote repository server.

There are three options for creating a local mirror server. Each of these options is explained in detail in a later section.

- **Option I:** Mirror server has no access to Internet at all

  Use a web browser on your workstation to download the HDP Repository Tarball, move the tarball to the selected mirror server using scp or an USB drive, and extract it to create the repository on the local mirror server.

- **Option II:** Mirror server has temporary access to Internet

  Temporarily configure a server to have Internet access, download a copy of the HDP Repository to this server using the **reposync** command, then reconfigure the server so that it is back behind the firewall.

- **Option III:** Mirror server has permanent access to Internet (modified form of Option II)

  Establish a “trusted host”, by permanently configuring a server to have Internet access, but still be accessible from within the firewall. Download a copy of the HDP Repository to this server using the **reposync** command.

  ### Note

  Option I is probably the least effort, and in some respects, is the most secure deployment option.

  Option III is best if you want to be able to update your Hadoop installation periodically from the Hortonworks Repositories.

  However, if you are considering Option III, you should also consider the fourth option, which is to proxy the HDP Repositories through a trusted proxy server. If you have a network administrator who has expertise in setting up proxies, and if the proxy option is acceptable within your Data Center Security policies, this can be the easiest of all the options.

- **Option IV:** Trusted proxy server

  Proxying a repository involves setting up a standard HTTP proxy on a local server to forward repository access requests to the remote repository server and route responses back to the original requestor. Effectively, the proxy server makes the repository server accessible to all clients, by acting as an intermediary.

Once the proxy is configured, change the `/etc/yum.conf` file on every repository client (i.e. cluster nodes), so that when the client attempts to access the repository during installation, the request will go through the local proxy server instead of going directly to the remote repository server.

1.1.3. Considerations for choosing a Mirror or Proxy solution

The following table lists some benefits provided by these alternative deployment strategies:

Table 1.2. Comparison - HDP Deployment Strategies

Advantages of repository mirroring (Options I, II, and III)	Advantages of creating a proxy(Options IV)
• Minimizes network access (after the initial investment of copying the repository to local storage). The install process is therefore faster, reliable, and more cost effective (reduced WAN bandwidth minimizes the data center costs). • Allows security-conscious data centers to qualify a fixed set of repository files. It also ensures that the remote server will not change these repository files. • Large data centers may already have existing repository mirror servers for the purpose of OS upgrades and software maintenance. You can easily add the HDP Repositories to these existing servers.	• Avoids the need for long term management of the repository files (including periodic updates for upgrades, new versions, and bug fixes). • Almost all data centers already have a setup of well-known proxies. In such cases, you can simply add the local proxy server to the existing proxies' configurations. This approach is easier compared to creating local mirror servers in data centers with no mirror server setup. • The network access is same as that required when using a mirror repository, but the source repository handles file management.

However, each of the above approaches are also known to have the following disadvantages:

- Mirrors have to be managed for updates, upgrades, new versions, and bug fixes.
- Proxy servers rely on the repository provider to not change the underlying files without notice.
- Caching proxies are necessary, because non-caching proxies do not decrease WAN traffic and do not speed up the install process.

1.2. Recommendations for Deploying HDP

This section provides information on the various components of the Apache Hadoop ecosystem.

In many data centers, the following deployment strategy may be optimal:

- Use a mirror for the HDP Repositories. The HDP Repositories are small and easily mirrored, thereby allowing secure control over the contents of the Hadoop packages accepted for use in your data center.
- Use a caching proxy for the EPEL repository, which is a well-known and trustworthy repository managed by the Fedora Project team, and which may be too large to mirror in your data center. If your data center already mirrors or proxies EPEL, use that mirror or proxy.

Note

The installer pulls many packages from the base OS repositories (repos). If you do not have a complete base OS available to all your machines at the time of installation, you may run into issues. For example, if you are using RHEL 6 your hosts must be able to access the “Red Hat Enterprise Linux Server 6 Optional (RPMs)” repo. If this repo is disabled, the installation is unable to access the rubygems package, which is necessary for HMC to operate.

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.

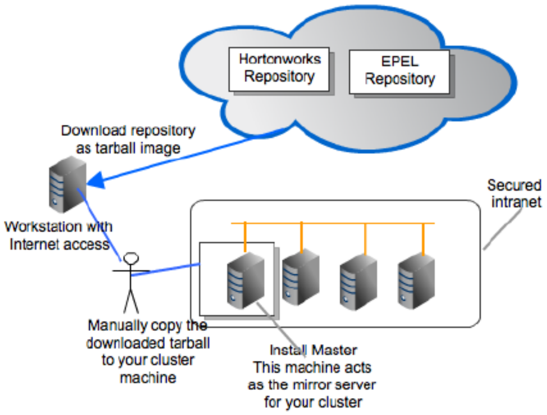
1.3. Detailed Instructions for Creating Mirrors and Proxies

In this section:

- Option I: Mirror server has no access to the Internet
- TODO add xml ids

1.3.1. Option I - Mirror server has no access to the Internet

The local mirror setup for Option I is shown in the following illustration:


Hortonworks Repository
EPEL Repository
Download repository as tarball image
Secured intranet
Workstation with Internet access
Manually copy the downloaded tarball to your cluster machine
Install Master This machine acts as the mirror server for your cluster

1.3.1.1. Prerequisites

Select a mirror server host with the following characteristics:

- This server runs on either CentOS (v5.x, v6.x) or RHEL (v5.x, v6.x) or SLES (v11) 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 allows all cluster nodes (the servers on which you want to install HDP) to access this server.

1.3.1.2. Instructions

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

Table 1.3. Deploying HDP - Option I

Cluster OS	HDP Repository Tarballs
RHEL/ CentOS 5.x	**http://public-repo-1.hortonworks.com/HDP-1.1.1.16/repos/centos5/HDP-1.1.1.16-centos5.tar.gz** **http://public-repo-1.hortonworks.com/HDP-UTILS-1.1.0.15/repos/centos5/HDP-UTILS-1.1.0.15-centos5.tar.gz**
RHEL/ CentOS 6.x	**http://public-repo-1.hortonworks.com/HDP-1.1.1.16/repos/centos6/HDP-1.1.1.16-centos6.tar.gz** **http://public-repo-1.hortonworks.com/HDP-UTILS-1.1.0.15/repos/centos6/HDP-UTILS-1.1.0.15-centos6.tar.gz**
SLES 11	**http://public-repo-1.hortonworks.com/HDP-1.1.1.16/repos/suse11/HDP-1.1.1.16-suse11.tar.gz** **http://public-repo-1.hortonworks.com/HDP-UTILS-1.1.0.15/repos/suse11/HDP-UTILS-1.1.0.15-suse11.tar.gz**

Note

The EPEL repository is not available as a tarball currently. Use one of Options II through IV to provide access to the EPEL repository.

Step 2: Create an HTTP server.

- On the mirror server, install an HTTP server (such as Apache httpd) using the instructions provided here.
- Activate this web server.
- 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.

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

- For example, from a shell window, type: **mkdir –p** ***/var/www/html/hdp/***
- If you are using a symlink, enable the **followsymlinks** on your web server.

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

Step 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/HDP-1.1.1.16/`
- You should see directory listing for all the HDP components along with the RPMs at the following location: `HDP-1.1.1.16/repos/`*`os`*.

Step 6: Configure the **yum** or **zypper** clients on all the nodes in your cluster.

- **For RHEL and CentOS:**

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

     ```
     http://yourwebserver>/hdp/HDP-1.1.1.16/repos/os/hdp.repo
     ```

     where *`os`* is either *centos5* or *centos6*.

  2. Store the `hdp.repo` file in a temporary location and edit it, changing the value of the **baseurl** property to the local mirror URL. For example, http://*`yourwebserver`*/hdp/HDP-1.1.1.16/repos/*`os`*

  3. 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:**

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

     ```
     http://yourwebserver>/hdp/HDP-1.1.1.16/repos/suse11/hdp.repo
     ```

  2. Store the `hdp.repo` file in a temporary location and edit it, changing the value of the **baseurl** property to the local mirror URL. For example, http://*`yourwebserver`*/hdp/HDP-1.1.1.16/repos/suse11.

  3. Store the hdp.repo file back into the repository location in the web server.

     ```
     cp /tmp/hdp.repo /var/www/html/hdp/HDP-1.1.1.16/suse11/
     ```

  4. On every node, invoke the following command:

     ```
     zypper addrepo -r http://yourwebserver/hdp/HDP-1.1.1.16/repos/suse11/hdp.
     repo
     ```

Step 7 [Conditional]: If your cluster runs CentOS or RHEL, and if you have multiple repositories configured in your environment, deploy the following plugin on all the nodes in your cluster.

1. Install the plugin.

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

     ```
     yum install yum-priorities
     ```

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

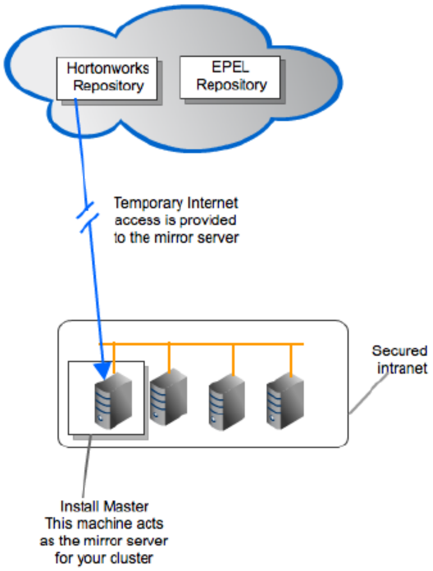
```
yum install yum-plugin-priorities
```

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

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

1.3.2. Option II - Mirror server has temporary access to the Internet

The local mirror setup for Option II is shown in the following illustration:


Hortonworks Repository
EPEL Repository
Temporary Internet access is provided to the mirror server
Secured intranet
Install Master This machine acts as the mirror server for your cluster

1.3.2.1. Prerequisites

Select a local mirror server host with the following characteristics:

- This server runs on either CentOS (v5.x, v6.x) or RHEL (v5.x, v6.x) 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
```

1.3.2.2. Instructions

Step 1: Temporarily reconfigure your firewall to allow Internet access from your mirror server host.

Step 2: 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 5.x	**wget http://public-repo-1.hortonworks.com/HDP-1.1.1.16/repos/centos5/hdp.repo -O /etc/yum.repos.d/hdp.repo**
RHEL/ CentOS 6.x	**wget http://public-repo-1.hortonworks.com/HDP-1.1.1.16/repos/centos6/hdp.repo -O /etc/yum.repos.d/hdp.repo**
SLES 11	**wget http://public-repo-1.hortonworks.com/HDP-1.1.1.16/repos/suse11/hdp.repo -O /etc/yum.repos.d/hdp.repo**

Step 3: Create an HTTP server.

- On the mirror server, install an HTTP server (such as Apache **httpd**) using the instructions provided http://httpd.apache.org/download.cgi
- Activate this web server.
- 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.

Step 4: On your mirror server, create a directory for your web server.

- For example, from a shell window, type: **mkdir –p** ***`/var/www/html/hdp/`***
- If you are using a symlink, enable the **followsymlinks** on your web server.

Step 5: 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:

```
cd /var/www/html/hdp
reposync -r HDP-1.1.1.16
reposync -r HDP-UTILS-1.1.0.15
```

You should now see both an `HDP-1.1.1.16` directory and an `HDP-UTILS-1.1.0.15` directory, each with several subdirectories.

Step 6: Generate appropriate metadata.

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

```
createrepo /var/www/html/hdp/HDP-1.1.1.16
createrepo /var/www/html/hdp/HDP-UTILS-1.1.0.15
```

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

Step 7: 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/HDP-1.1.1.16/**
- You should now see directory listing for all the HDP components.

Step 8: At this point, it is okay to disable external Internet access for the mirror server, so that the mirror server is once again entirely within your data center firewall.

Step 9: Depending on your cluster OS, configure the **yum** or **zypper** clients on all the nodes in your cluster

- For RHEL and CentOS:

  1. Edit the `hdp.repo` file (downloaded here: `/etc/yum.repos.d/`), changing the value of the `baseurl` property to the local mirror URL. For example, **http://*yourwebserver*/hdp/HDP-1.1.1.16/*os*replaceable**

     where *os* is either **centos5** or **centos6**

  2. 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:

  1. Edit the `hdp.repo` file (downloaded here: `/etc/yum.repos.d/`), changing the value of the `baseurl` property to the local mirror URL. For example, **http://*yourwebserver*/hdp/HDP-1.1.1.16/**

  2. On every node, invoke the following command:

     ```
     zypper addrepo -r http://yourwebserver/hdp/HDP-1.1.1.16/hdp.repo
     ```

Step 10 [Conditional]: If your cluster runs CentOS or RHEL, and if you have multiple repositories configured in your environment, deploy the following plugin on all the nodes in your cluster.

1. Install the plugin.

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

     ```
     yum install yum-priorities
     ```

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

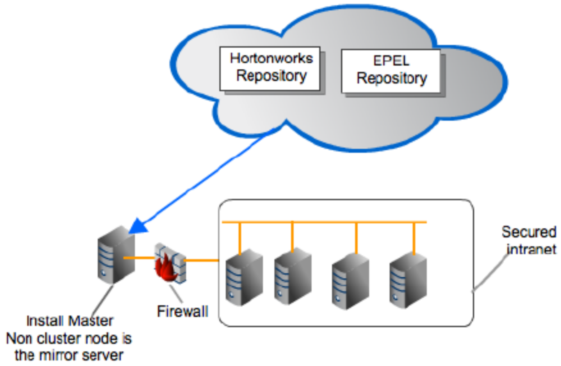
```
yum install yum-plugin-priorities
```

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

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

1.3.3. Option III - Mirror server has permanent access to the Internet

The local mirror setup for Option III is shown in the following illustration:



1.3.3.1. Prerequisites

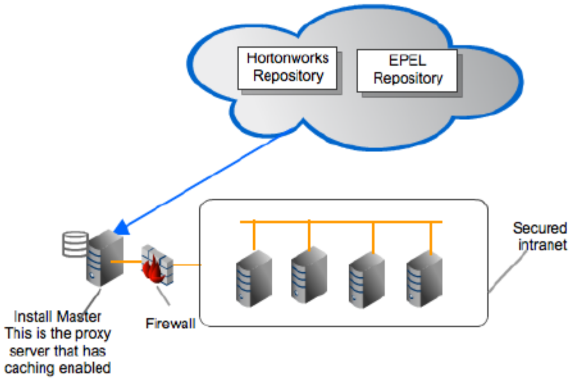
Same as Option II.

1.3.3.2. Instructions

Same as Option II but Step 1 and Step 8 are unnecessary and may be skipped.

1.3.4. Option IV - Trusted proxy server

The local mirror setup for Option IV is shown in the following illustration:



1.3.4.1. Prerequisites

Select a mirror server host with the following characteristics:

- This server runs on either CentOS (v5.x, v6.x) or RHEL (v5.x, v6.x) and has several GB of storage available.
- The firewall allows all cluster nodes (the servers on which you want to install HDP) to access this server, and allows this server to access the Internet (at least those Internet servers for the repositories to be proxied).

1.3.4.2. Instructions

Step 1: Create a caching HTTP PROXY server on the selected host.

1. It is beyond the scope of this document to show how to set up an HTTP PROXY server, given the many variations that may be required, depending on your data center's network security policy. If you choose to use the Apache HTTPD server, it starts by installing **httpd**, using the instructions provided here, and then adding the **mod_proxy** and **mod_cache modules**, as stated here. Please engage your network security specialists to correctly set up the proxy server.
2. Activate this proxy server and configure its cache storage location.
3. Ensure that the firewall settings (if any) allow inbound HTTP access from your cluster nodes to your mirror server, and outbound access to the desired repo sites, including **public-repo-1.hortonworks.com**.

   Note

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

Step 2: Depending on your cluster OS, configure the **yum** or **zypper** clients on all the nodes in your cluster.

- **For RHEL and CentOS:**

  Note

  The following description is taken from the CentOS documentation here.

  1. On each cluster node, add the following lines to the **/etc/yum.conf** file.

     (As an example, the settings below will enable **yum** to use the proxy server **mycache.mydomain.com**, connecting to port **3128**, with the following credentials **yum-user/qwerty**.

     ```
     # proxy server:port number
     proxy=http://mycache.mydomain.com:3128
     ```

     ```
     # account details for secure yum proxy connections
     proxy_username=yum-user
     proxy_password=qwerty
     ```

2. Once all nodes have their **/etc/yum.conf** file updated with appropriate configuration info, you can proceed with the HDP installation just as though the nodes had direct access to the Internet repositories.

3. If this proxy configuration does not seem to work, try adding a **/** at the end of the proxy URL. For example:

   ```
   proxy=http://mycache.mydomain.com:3128/
   ```

- **For SLES:**

1. Configure the zypper clients on all the nodes in your cluster, to use the proxy server.

2. Please consult with your system administrator to do this correctly in your environment.