Hortonworks DataPlane Service

DPS Installation and Setup

(October 31, 2017)

docs.cloudera.com

Hortonworks DataPlane Service (DPS™): DPS Installation and Setup

Please visit the Hortonworks DataPlane Service page for more information on Hortonworks DPS technology. For more information on Hortonworks support services, please visit either the Support or Training page. Feel free to contact us directly to discuss your specific needs.

Hortonworks reserves the right to change any products described herein at any time, and without notice. Hortonworks assumes no responsibility or liability arising from the use of products described herein, except as expressly agreed to in writing by Hortonworks.

Disclaimer: EXCEPT AS EXPRESSLY PROVIDED IN A WRITTEN AGREEMENT WITH HORTONWORKS, HORTONWORKS DOES NOT MAKE OR GIVE ANY REPRESENTATION, WARRANTY, OR COVENANT OF ANY KIND, WHETHER EXPRESS OR IMPLIED, IN CONNECTION WITH HORTONWORKS TECHNOLOGY OR RELATED SUPPORT PROVIDED IN CONNECTION THEREWITH. HORTONWORKS DOES NOT WARRANT THAT DPS WILL OPERATE UNINTERRUPTED OR THAT IT WILL BE FREE FROM DEFECTS OR ERRORS, THAT IT WILL PROTECT YOUR DATA FROM LOSS, CORRUPTION OR UNAVAILABILITY, OR THAT DPS WILL MEET ALL OF CUSTOMER'S BUSINESS REQUIREMENTS. WITHOUT LIMITING THE FOREGOING, AND TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW, HORTONWORKS EXPRESSLY DISCLAIMS ANY AND ALL IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO IMPLIED WARRANTIES OF MERCHANTABILITY, QUALITY, NON-INFRINGEMENT, TITLE, AND FITNESS FOR A PARTICULAR PURPOSE AND ANY REPRESENTATION, WARRANTY, OR COVENANT BASED ON COURSE OF DEALING OR USAGE IN TRADE.

Open Source: Open source software licensing information for third party software used in connection with or included in the DataPlane Services software may be found in documentation accompanying the delivery of Hortonworks DataPlane Services software.

HORTONWORKS PROVIDES SEPARATELY LICENSED CODE, INCLUDING ALL OPEN SOURCE SOFTWARE, TO CUSTOMER WITHOUT WARRANTIES OF ANY KIND. HORTONWORKS DISCLAIMS ANY AND ALL EXPRESS AND IMPLIED WARRANTIES WITH RESPECT TO SEPARATELY LICENSED CODE, INCLUDING BUT NOT LIMITED TO IMPLIED WARRANTIES OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

Table of Contents

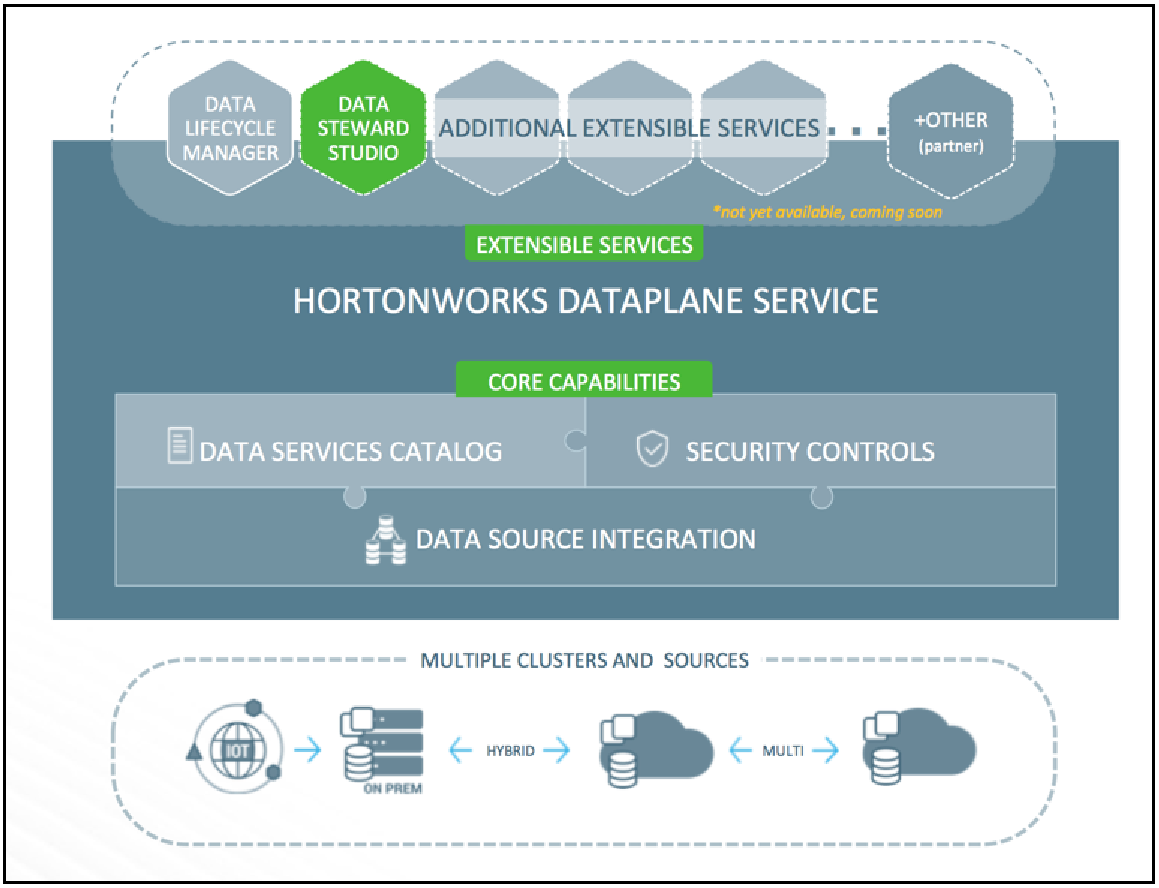
List of Figures

List of Tables

List of Examples

1. Overview

Hortonworks DataPlane Service (DPS™) is a common set of services to manage, secure, and govern data assets across multiple tiers and types. It does this for data at rest and in multiple clusters and tiers (on-premises, cloud, and to the point of origin (IOT)).



Services Available for Hortonworks DataPlane Service (DPS)

The following services are currently available as plugins to DPS. You can install and use one or all of these services in any combination. DPS Platform is required to implement any plugin service.

Data Lifecycle Manager (DLM)	Provides the ability to schedule replication across clusters and sources for HDFS and Hive data. See the DLM Administration guide for information about using DLM.
Data Steward Studio (DSS)	Provides a suite of capabilities that allows users to understand and govern data across enterprise data lakes. See the DSS Administration guide for information about using DSS.

1.1. Supported Configurations

See the following for information about supported configurations:

- Hortonworks Support Matrix

.

2. Planning for a DPS Installation

Prior to installing DPS, you must consider various aspects of your HDP environment and prepare them prior to DPS installation. This includes items such as operating system version, cluster security, the node configuration requirements for DPS Platform and associated services, etc.

Review the following information prior to starting your DPS installation to ensure your environment is properly configured for a successful DPS installation.

2.1. Support Matrix information

You can find the most current information about interoperability for this release on the Support Matrix. The Support Matrix tool provides information about:

- HDP and Ambari
- Operating Systems
- Databases
- Browsers
- JDKs

To access the tool, go to: https://supportmatrix.hortonworks.com.

2.2. Requirements for DataPlane Service Platform Host

Hortonworks DataPlane Service (DPS™) is composed of a platform (DPS Platform) and the services that plug into the platform (DLM, DSS, etc.), which are all installed on the same host node. DPS also includes engines and agents that are installed on the clusters used with DPS.

You should install the DPS Platform on a host remote to the cluster. The DPS Platform host must meet the requirements identified in the following sections.

All clusters registered with DPS must be managed by Apache Ambari.

2.2.1. DPS Support Matrix Information

See the Requirements for DataPlane Service Platform Host and the Hortonworks Support Matrix for details regarding supported operating systems, databases, software, etc.

2.2.2. Required Docker Versions

Docker 1.12.x

2.2.3. Other Software Requirements

On each DPS Platform host, ensure that the following software is available:

- `yum` and `rpm`
- `tar` and `wget`
- bash shell

2.2.4. Processing and Memory Requirements

The DPS Platform host requires the following:

- Multicore processor, with minimum 8 cores
- Minimum 16 GB RAM

See the HDP and Ambari Support Matrices for requirements.

2.2.5. Port and Network Requirements

Have the following ports available and open:

Port Number	Purpose	Required to be open?
80	Where DPS Platform runs.	Yes
443	For SSL-based communication.	Yes
8443	Where the Apache Knox instance for login runs.	Yes
8500	For debugging using Consul. This port must be available, but it is optional to have it open.	No

It is recommended that you use a DNS server to resolve host names. If resolving host names from an `/etc/hosts` file, you must add the names to the hosts files of each DPS container. Follow the instructions, Add Host Entries to /etc/hosts Files in *DPS Administration*.

2.2.6. LDAP and AD Support Requirements

To use LDAP and AD, you must use the same LDAP and AD instance across all HDP clusters managed by DataPlane, as well as for DataPlane Service itself.

2.2.7. HDP 2.6.3 Apache Component Requirements

The following additional Apache components are required for DPS Platform support:

Component	Purpose	Comments
Knox	User authentication with LDAP (SSO)	Knox must be enabled on clusters before you can register the clusters with DPS.

Component	Purpose	Comments
Ambari	Cluster registration in DPS	All clusters used with DPS must be using Ambari.

2.2.8. SmartSense Requirements

A SmartSense ID is required to install DPS Services (DLM and DSS).

You can retrieve the SmartSense ID from the Hortonworks Support Portal, under the Tools tab.

2.2.9. Additional DPS Requirements and Recommendations

Understanding the requirements and recommendations indicated below can help to avoid common issues during and after DPS installation.

- Prior to starting installation, you must have downloaded the required tarballs and MPacks from the customer portal, following the instructions provided as part of the product procurement process.
- You need to have root access to the nodes on which all DPS services will be installed.
- If you are using AWS, do not use the public DNS to access DPS.

  Use a public IP address or set up and use a DNS (Route 53) fully qualified domain name (FQDN).
- Every host name used with DPS must be resolvable by DNS or configured in the `/etc/hosts` file on the DPS container, so that host names can be resolved between all cluster nodes.

  Using a DNS server is the recommended method, unless you are using Amazon Web Services (AWS). But if the instances are added to `/etc/hosts`, you must explicitly register the cluster host names within the DPS Docker containers. It is not sufficient to have the host names included in the `/etc/hosts` file on the DPS Platform host. See the *DPS Platform Administration* guide for instructions.
- If you are not using the LDAP server packaged with DPS, you need the corporate LDAP settings to configure LDAP.

  Ensure you have the correct settings if using your own LDAP, as most of the settings cannot be changed in DPS after they are set.
- Use the default Knox user.

  If you choose a server to host Knox that is not the one the Ambari Server defaulted to, proxyuser rules will change and you will be prompted for a restart.
- When enabling DPS Platform and DLM, and installing Knox, follow the automated Ambari placement recommendations to avoid requiring a restart.
- **Important:** Do not edit the cluster name from Ambari after registering the cluster with DPS Platform.

2.3. DPS Service Requirements for HDP Clusters

2.3.1. DPS Support Matrix Information

See the Requirements for DataPlane Service Platform Host and the Hortonworks Support Matrix for details regarding supported HDP configurations.

All clusters used with DPS must be managed by Ambari.

2.3.2. Configuring Cluster Security for DPS Services

Following are lists of the *minimum* required actions that you must perform on *each HDP cluster* as part of configuring security for DPS and onboarding clusters for each of the DPS services. You can perform any additional security-related tasks as appropriate for your environment and company policies.

Table 2.1. Minimum Security Requirements Checklist for DPS

Task	Instructions	Found in...	Comments
Enable Knox in Ambari	Install Knox	Apache Knox Gateway User's Guide	Services required in the Knox topology for DPS are Ambari, AmbariUI, JobTracker, NameNode, Ranger, RangerUI, and ResourceManager
Enable Ranger in Ambari	Installing Ranger Using Ambari	HDP Security guide	
Configure a reverse proxy with Knox	Configuring the Knox Gateway	HDP Security guide	The Knox Gateway is not required, but is recommended
Configure SSO topology	Form-based Identity Provider (IdP)	HDP Security guide	
Configure LDAP with Ambari	Configuring Ambari Authentication with LDAP or Active Directory Authentication	HDP Security guide	
Synchronize required LDAP users and groups with Ambari	Synchronizing LDAP Users and Groups	HDP Security guide	You must disable LDAP pagination; Users registering clusters in DPS must have Admin role in Ambari
Configure LDAP with Ranger	Configuring Ranger Authentication with UNIX, LDAP, or AD	HDP Security guide	Required for DSS and if using Ranger with DLM
Configure LDAP with Knox for proxy authentication	Setting Up LDAP Authentication	HDP Security guide	
Configure Knox for HA	Setting Up Knox Services for HA	HDP Security guide	Required only if clusters are configured for HA
Configure Knox SSO for Ambari	Setting up Knox SSO for Ambari	HDP Security guide	If done on an existing cluster, at login you will see a Knox page and must log in with your LDAP credentials

If you are performing Hive replication with the Data Lifecycle Manager (DLM) service, ensure that the following tasks were completed during cluster installation. You must configure Ambari Ranger on clusters used in replicating Hive databases.

Table 2.2. Minimum Security Requirements Checklist for DLM

Task	Instructions	Found in...	Comments
Configure LDAP with Ranger	Configuring Ranger Authentication with UNIX, LDAP, or AD	HDP Security guide	Required if using Ranger with DLM
Configure user synchronization for policy administration	Configure Ranger User Sync	HDP Security guide	Required only if using Ranger
Configure Ranger plugin for HDFS	Enabling Ranger Plugins: HDFS	HDP Security guide	Required only if using Ranger
Configure Ranger plugin for Hive	Enabling Ranger Plugins: Hive	HDP Security guide	Required only if using Ranger
Configure Ranger plugin for Knox	Enabling Ranger Plugins: Knox	HDP Security guide	Required only if using Ranger
Configure Ranger HDFS plugin for Kerberos	Ranger Plugins–Kerberos: HDFS	HDP Security guide	Required only if using Ranger
Configure Ranger Hive plugin for Kerberos	Ranger Plugins–Kerberos: Hive	HDP Security guide	Required only if using Ranger
Configure Ranger Knox plugin for Kerberos	Ranger Plugins–Kerberos: Knox	HDP Security guide	Required only if using Ranger
Configure Knox SSO for Ranger	Setting up Knox SSO for Ranger	HDP Security guide	

If you are using the Data Steward Studio (DSS) service, ensure that the following tasks were completed during cluster installation. You must configure Apache Atlas and Apache Knox SSO before you can use DSS.

Table 2.3. Minimum Security Requirements Checklist for DSS

Task	Instructions	Found in...	Comments
Enable Atlas in Ambari	Installing and Configuring Apache Atlas Using Ambari	HDP Data Governance guide	
Configure LDAP with Atlas	Customize Services	HDP Data Governance guide	Adapt the instructions for Ranger
Configure Ranger plugin for Atlas	Enabling Ranger Plugins: Atlas	HDP Security guide	
Configure Knox SSO for Atlas	Setting up Knox SSO for Atlas	HDP Security guide	
Configure Knox SSO for Ranger	Setting up Knox SSO for Ranger	HDP Security guide	

2.4. Data Lifecycle Manager (DLM) Installation Requirements and Recommendations

The clusters on which you install the Data Lifecycle Manager (DLM) Engine must meet the requirements identified in the following sections. After the DLM Engine is installed and properly configured on a cluster, the cluster can be used for DLM replication.

⚠ **Important**

Clusters used as source and destination in a DLM replication relationship must have exactly the same configurations for LDAP, Kerberos, Ranger, Knox, HA, etc.

2.4.1. DLM Support Matrix Information

See the Requirements for Clusters Used With Data Lifecycle Manager Engine and the Hortonworks Support Matrix for details regarding supported operating systems, databases, software, etc.

2.4.2. Port and Network Requirements

Have the following ports available and open:

Default Port Number	Purpose	Comments	Required to be open?
25968	Port for DLM Engine (Beacon) service on hosts.	Accessibility is required from all clusters. “Beacon” is the internal name for the DLM Engine. You will see the name Beacon in some paths, commands, etc.	Yes
8020	NameNode host		Yes
50010	All DataNode hosts		Yes
8080	Ambari server host		Yes
10000	HiveServer2 host	Binary mode port (Thrift)	Yes
10001	HiveServer2 host	HTTP mode port	Yes
2181	ZooKeeper hosts		Yes
6080	Ranger Port		Yes
8443	Knox Port		Yes
8050	Yarn Port		Yes

2.4.3. HDP 2.6.3 Apache Component Requirements

The following additional Apache components are required for DLM support:

Component	Purpose	Comments
Hive 1	For replicating Hive database content	Hive 2 queries are supported, but for replication, HiveServer 2 with Hive 1 is always used.
HDFS	For replicating HDFS data.	
Knox	Authentication federation from DPS	Knox must be enabled on clusters before you can register the clusters with DPS.
Ranger	Authorization on clusters during replication	Ranger is optional for HDFS replication, but required for Hive replication.

2.4.4. Additional DLM Requirements and Recommendations

Understanding the requirements and recommendations indicated below can help to avoid common issues during and after installation of the DLM service.

- Apache Hive should be installed during initial installation, unless you are certain you will not use Hive replication in the future.

  If you decide to install Hive after creating HDFS replication policies in Data Lifecycle Manager, all HDFS replication policies must be deleted and then recreated after adding Hive.

- Clusters used in DLM replication must have symmetrical configurations.

  That is, each cluster in a replication relationship must be configured exactly the same for Kerberos, LDAP, High Availability (HA), Apache Ranger, and so forth.

2.5. Data Steward Studio (DSS) Installation Requirements and Recommendations

The clusters on which you install the DSS Profiler Agent must meet the requirements identified in the following sections. After the Profiler Agent is installed and properly configured on a cluster, the cluster can be used by DSS.

Data Steward Studio (DSS) is provided as Evaluation Software with Hortonworks DPS 1.0. Evaluation Software is provided without charge and pursuant to your the DataPlane Service Terms of Use. Evaluation Software may only be used for internal business, evaluation, and non-production purposes. Feedback on Evaluation Software is welcomed and may be submitted through your regular support channels.

2.5.1. DSS Support Matrix Information

See the Requirements for Data Steward Studio Profiler and the Hortonworks Support Matrix for details regarding supported operating systems, databases, software, etc.

2.5.2. Other Software Requirements

DSS has no additional software requirements.

2.5.3. Port and Network Requirements

Have the following ports available and open:

Port Number	Purpose	Required?
21900	Profiler Web service runs on this port	This is required for DataPlane to access profiled data from the profiler datastore.
8999	Livy runs on this port	Livy is the observer for profilers and is required for submitting profiler jobs.
21000	Atlas	Required if you are installing in a different DMZ.
6080	Ranger	Required if you are installing in a different DMZ.
8443	Knox	Required if you are installing in a different DMZ.

Port Number	Purpose	Required?
8080	Ambari	Yes

2.5.4. HDP 2.6.3 Apache Component Requirements

The following additional Apache components are required for DSS support:

Component	Purpose	Comments
Atlas	For Hive Metadata availability and storage of univariate statistics	
Ranger	For access logs availability for usage profiling	
Spark 2	For Profiler computation – both univariate and Ranger profilers	
Livy Server 2	Job Server for Profilers	
HDFS	For registering and sharing Profiler .jars	Co-located on the Profiler Agent Node.
Hive	For column profiling	

3. Preparing Your External Database

Although DPS includes an embedded PostgreSQL database, the embedded database is intended for nonproduction use. You should use an external database for production environments. Currently, PostgreSQL database is supported. After installing the database following the instructions provided with the database software, you must set up the database for use with DPS.

Prerequisites

- The PostgreSQL database must have been installed and properly configured for remote access.
- A database must have been created.
- A database user must have been created and assigned permissions for the new database.

Steps

1. Install the PostgreSQL server.

   ```
   sudo yum install postgresql-server
   ```

2. Enable remote access by editing the following files:

   a. In the `postgresql.conf` file, set *`listen_addresses = '*'`*.

      This configures the server to listen for connections from client applications on all available IP interfaces.

   b. In the `pg_hba.conf` file, set *`host all all 0.0.0.0/0 md5`*.

      IP address range can be modified as required, if known.

      This provides additional security by locking down which client machines can access the PostgreSQL server.

3. Restart PostgreSQL.

   ```
   service postgresql restart
   ```

4. Log in as the default user `postgres`.

   ```
   su - postgres
   ```

5. Create a database.

   ```
   createdb <database_name>
   ```

6. Log in to the PostgreSQL server.

   ```
   psql -h <database_ip> <database_name>
   ```

7. Add a database user.

   ```
   CREATE USER <user_name> WITH PASSWORD '<password>';
   ```

8. Set the permissions on the new database to the user.

   For example:

   ```
   ALTER DATABASE <database_name> OWNER TO <user_name>;
   ```

9. Exit the `postgres` account.

   ```
   exit
   ```

   You are returned to the root user role.

10. Navigate to ${INSTALLER_HOME}/ and open the `config.env.sh` file for editing.

11. Modify the following settings to add the appropriate connection information.

    ```
    USE_EXT_DB="yes"
    DATABASE_URI="jdbc:postgresql://<host_name>:5432/<database_name>"
    DATABASE_USER="<user_name>"
    DATABASE_PASS="<password>"
    ```

Your external PostgreSQL is now configured so you can configure it during DPS installation.

4. Setting Up the Local Repository for Your DPS Installation

To install Hortonworks DPS, you must have previously downloaded tarballs from the customer portal, following the instructions provided as part of the product procurement process. DPS Platform and related services are installed as RPMs in a repository on the DPS host. The tarballs containing the RPMs should have been downloaded to the DPS host.

You must also download tarballs containing MPacks. These tarballs should be downloaded to the Ambari hosts on each DPS cluster. You install the DPS engines or agents on the clusters using the MPacks.

4.1. Prepare the Web Server for the Local Repository

Before setting up your local repository, you must properly configure an HTTP web server, on which you will create the repositories.

Prerequisites

You must have met the following requirements.

- Selected an existing server that is accessible from all hosts in your clusters.
- Enabled HTTP access from all hosts in your cluster to the mirror server (the server on which you are mirroring the repositories).

After meeting these requirements, you can prepare your local repository.

Steps

1. Create an HTTP server:

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

   b. Activate the server.

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

   ### Note

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

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

   ```
   mkdir -p /var/www/html/
   ```

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

Next Steps

You next must set up your local repository.

More Information

Downloading the Apache HTTP Server

4.2. Set Up a Local Repository

Setting up a local repository involves moving the tarball to the selected mirror server and extracting the tarball to create the repository.

Prerequisites

Ensure that you have downloaded the required tarballs from the customer portal, following the instructions provided as part of the product procurement process.

You must have completed the procedure, Section 4.1, “Prepare the Web Server for the Local Repository” [13].

Steps

1. Copy the repository tarballs to the web server directory and expand (uncompress) the archive file:

   a. Navigate to the web server directory you previously created.

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

      All content in this directory is served by the web server.

   b. Move the tarballs to the current directory and expand each of the repository tarballs that you downloaded.

      Replace <*file-name*> with the actual name of the RPM tarball that you are expanding.

      ```
      tar zxvf <file-name>.tar.gz
      ```

      During expansion of the tarball, subdirectories are created in `/var/www/html/`, such as `DP/centos7`. These directories contain the repositories.

      Expanding the DPS tarball takes several seconds.

2. Confirm that you can browse to the newly created local repositories by using the base URLs:

   http://<*webserver-host-name*>/<*repo-name*>/<*OS*>/<*service-version-X*>

   - <webserver-host-name>

     This is the FQDN of the web server host.

   - <repo-name>

This is composed of the abbreviated name of the repository, such as DP, DLM, DSS.

- <OS>

  This is the operating system version.

- <service-version-X>

  This is the version number of the downloaded repository with an appended unique number.

Example 4.1. Base URL Examples

DPS Base URL

```
http://webserver.com/DP/centos7/1.0.0.0-X
```

DLM Base URL

```
http://webserver.com/DLM/centos7/1.0.0.0-X
```

Example 4.2. DSS Base URL

```
http://webserver.com/DSS/centos7/1.0.0.0-X
```

Be sure to record these Base URLs, because you need them when installing DPS Platform on the host, and installing the associated engines and agents on the clusters.

3. If you have multiple repositories configured in your environment, deploy the following plugin on all the nodes in your cluster.

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

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

   ```
   [main]
   ```

   ```
   enabled=1
   ```

   ```
   gpgcheck=0
   ```

The repositories for DPS Platform and all services are now prepared for installation.

Next Steps

Create the configuration file for the DPS Platform repository.

4.3. Create the Repository Configuration File

A repository configuration file must be created on the DPS host. The file is required to identify the path to the repository data, establish whether a GPG signature check should be performed on the repository packages, etc. Only one repository configuration file is needed.

Steps

1. Navigate to the repository directory.

   ```
   cd /etc/yum.repos.d/
   ```

2. Create a repository file.

   For example:

   ```
   vi dp.repo
   ```

   Alternatively, you can copy an existing repository file to edit.

3. Add or modify the following content in the repository file.

   ```
   #VERSION_NUMBER=<downloaded-version#>
   ```

   ```
   [<service-name-abbreviation>]
   ```

   This is composed of the service name abbreviation and version number (includes the build number). Example: DP-1.0.0-100

   ```
   name=<service-name-abbreviation> Version - <service-name-abbreviation>
   ```

   ```
   baseurl=http://<webserver-host-name>/<directory-containing-repo>
   ```

   <webserver-host-name> is the FQDN of the web server host that contains the repository.

   This is the same base URL that you used in the task to prepare the repositories.

   <directory-containing-repo> is the path expanded from the tarball.

   ```
   gpgcheck=1
   ```

   ```
   gpgkey=http://<webserver-host-name>/<directory-containing-repo>/RPM-GPG-KEY/
   RPM-GPG-KEY-Jenkins
   ```

   ```
   enabled=1
   ```

   ```
   priority=1
   ```

   #### Example 4.3. Example Repository File

   ```
   #VERSION_NUMBER=1.0.0.0-100
   [DP-1.0.0.0-100]
   name=DP Version - DP-1.0.0.0-100
   baseurl=http://my-repo-server/DP/centos7/1.0.0.0-100
   gpgcheck=1
   gpgkey=http://my-repo-server/DP/centos7/1.0.0.0-100/RPM-GPG-KEY/RPM-GPG-KEY-
   Jenkins
   enabled=1
   priority=1
   ```

4. If you have multiple repositories configured in your environment, edit the `/etc/yum/pluginconf.d/priorities.conf` file to add the following values:

   ```
   [main]
   ```

   ```
   enabled=1
   ```

```
gpgcheck=0
```

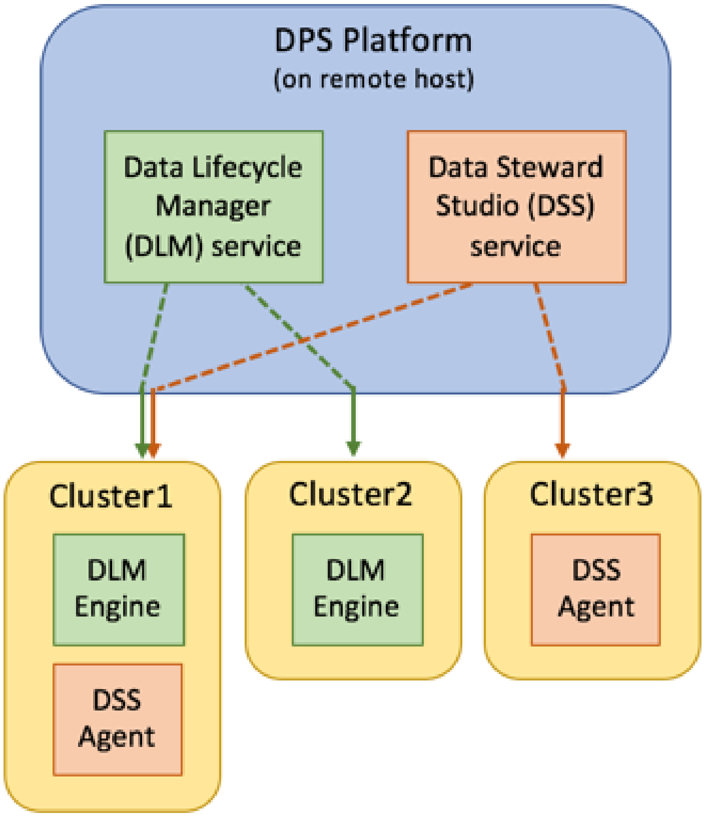
You are now ready to install the DPS Platform and associated UI services, such as DLM and DSS.

5. Installing DPS Services, Engines, and Agents

You install DPS Docker images on a host that is remote to all clusters. This allows DPS services to access all clusters, regardless of the status of any given cluster. The DPS Platform container and all services that plug into DPS Platform are installed on the same host node. You can implement one service or any combination of services with DPS Platform.

Any engines or agents associated with a DPS service, such as the DLM Engine or DSS Profiler Agent, must be installed on every cluster. They can be installed on any node that meets the hardware and software requirements for the engine or agent.

Figure 5.1. Diagram of a Sample Installation Configuration



More Information

Planning for a DPS Installation

Hortonworks Support Matrix

5.1. Install DPS Platform

DPS Platform and associated DPS Services need to be installed in a separate host that is not part of any cluster. DPS Platform is required for all DPS installations, but you can install any combination of DPS services with DPS Platform.

Prerequisites

- The host on which you install DPS services must meet the following requirements:
  - Running CentOS 7.0-7.3 or RHEL 7.0.1-7.3 operating system
  - Has the following TCP ports available: 80, 443, 8443, 8500, 9011
  - Must be a dedicated host that is not part of an existing HDP cluster

    This prevents potential port conflicts with other HDP services
- If you are using an external database, *which is recommended for production environments*, the following is required:
  - External PostgreSQL database version 9.2 or above must have been installed and properly configured for remote access
  - A database must have been created for DPS
  - A database user must have been created and assigned full privileges for the database, to act as DPS database admin
- During installation, you must have the following information available:
  - FQDN or IP address of the host on which DPS Platform will be installed
  - If using your own certificate, have the information available that you need to enter during installation.
  - If using an external database, have the host name, database name, user name, and password for the database

About This Task

- You need root user access to perform this task.
- As part of this task, you need to turn off SELinux to install and run Docker containers.

  This is due to an issue with SELinux and Docker.
- The DPS services only need to be installed on one remote host node.
- No default password is provided during installation of DPS. The DataPlane administrator creates a password during the installation process.
- The DPS Docker images are embedded in the binaries that you download, so you do not need internet access to complete the installation.

- Although DPS includes an embedded PostgreSQL database, the embedded database is intended for nonproduction use. You should use an external database for production environments. Currently, PostgreSQL database is supported. After installing the database following the instructions provided with the database software, you must modify a configuration file to use the database with DPS.

Steps

1. Log in as root to the host on which you set up the DPS repositories.

   ```
   sudo su
   ```

2. Disable Linux enforcement of permissions.

   Important

   If you do not disable SELinux before installing the DPS service, then DPS will not install and run properly, and you will have to destroy and reinstall the containers.

   ```
   setenforce 0    #A zero, not a letter
   sed -i 's/^SELINUX=.*/SELINUX=disabled/g' /etc/sysconfig/selinux
   ```

   The second command prevents SELinux from being automatically re-enabled after a reboot.

3. Install and start the Docker service, if not installed.

   ```
   yum install docker -y
   ```

   ```
   service docker start
   ```

4. Install the RPMs for DPS Platform and associated services.

   ```
   yum install dp-core
   yum install dlm-app
   ```

   A folder is created that contains the Docker image tarball files and a configuration script.

   If the **yum** commands fail, then the local repository was not set up correctly. Check the repository file `/etc/yum.repos.d/dp.repo` on the DPS host.

5. Navigate to the folder containing the DPS configuration script.

   ```
   cd /usr/dp/current/core/bin
   ```

6. Load all DPS Platform Docker images into the Docker repository of the local machine.

   ```
   ./dpdeploy.sh load
   ```

   Loading the DPS Platform images might take a while.

7. Edit configuration properties in the `config.env.sh` file in `/usr/dp/current/core/bin`.

   Modifying this file allows the properties to be automatically implemented. If you do not add the properties to this file, you need to enter them manually when the initialization script runs.

See “Installation Configuration Properties” for a list of the properties and descriptions.

- Uncomment the following property and enter the IP address to set the IP address of the host where DPS containers are launched:

  ```
  CONSUL_HOST="<ip-address>"
  ```

- *If you want to use the LDAP version included with DPS*, uncomment and set the following property:

  ```
  USE_TEST_LDAP="yes"
  ```

- *If using an external database* (recommended), uncomment and modify the following properties to add the appropriate information for DPS to access the PostgreSQL database:

  ```
  USE_EXT_DB="yes"
  DATABASE_URI="jdbc:postgresql://<host_name>:5432/<database_name>"
  DATABASE_USER="<user_name>"
  DATABASE_PASS="<password>"
  ```

- *If using your own certificate*, uncomment and modify the following properties:

  ```
  USE_TLS="true"
  USE_PROVIDED_CERTIFICATES="yes"
  PUBLIC_KEY_L="<absolute-path-of-public-key-file>"
  PRIVATE_KEY_L="<absolute-path-of-encrypted-private-key-file>"
  ```

- Configure any other properties, as appropriate for your environment.

8. Run the DPS Platform initialization script.

   ```
   ./dpdeploy.sh init --all
   ```

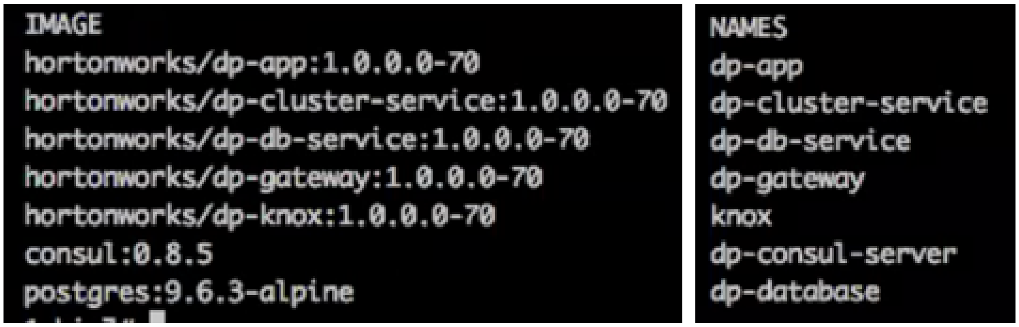
   During initialization, you are required to create a password for the administrative user and a master password for the system. Ensure that you remember these passwords, as they cannot be retrieved or reset by any user. If you forget the password, you must contact Hortonworks Support to reset the password.

9. Verify which DPS Platform UI containers are running.

   ```
   docker ps
   ```

   If using an external database, six containers should be running.

   If using the packaged database, seven containers should be running, similar to what is shown in the following example:



```
IMAGE                                        NAMES
hortonworks/dp-app:1.0.0.0-70                dp-app
hortonworks/dp-cluster-service:1.0.0.0-70    dp-cluster-service
hortonworks/dp-db-service:1.0.0.0-70         dp-db-service
hortonworks/dp-gateway:1.0.0.0-70            dp-gateway
hortonworks/dp-knox:1.0.0.0-70               knox
consul:0.8.5                                 dp-consul-server
postgres:9.6.3-alpine                        dp-database
```

If all of the containers are not running, you must destroy the containers and start over, as described in the troubleshooting section of this guide.

More Information

Preparing Your External Database

Installation Configuration Properties

Troubleshooting Installation Issues

5.2. Install the DPS Services

After installing the DPS Platform, install the services you intend to use, such as DLM, DSS, etc. The services are installed on the same host as DPS Platform. You can install one DPS service or any combination of DPS services with DPS Platform.

Tip

The following task uses "dlm" in many of the examples. Replace "dlm" with the appropriate short name used for the service you are installing.

1. Navigate to the directory containing the installation scripts for a DPS service, for example:

   ```
   cd /usr/dp/current/apps/dlm/bin
   ```

2. Load all DLM Docker images into the Docker repository of the local machine.

   ```
   ./dlmdeploy.sh load
   ```

   Loading the images might take a while.

3. Run the initialization script.

   ```
   ./dlmdeploy.sh init
   ```

4. If prompted for the host IP address (for Consul), provide the IP address of the DPS Platform host.

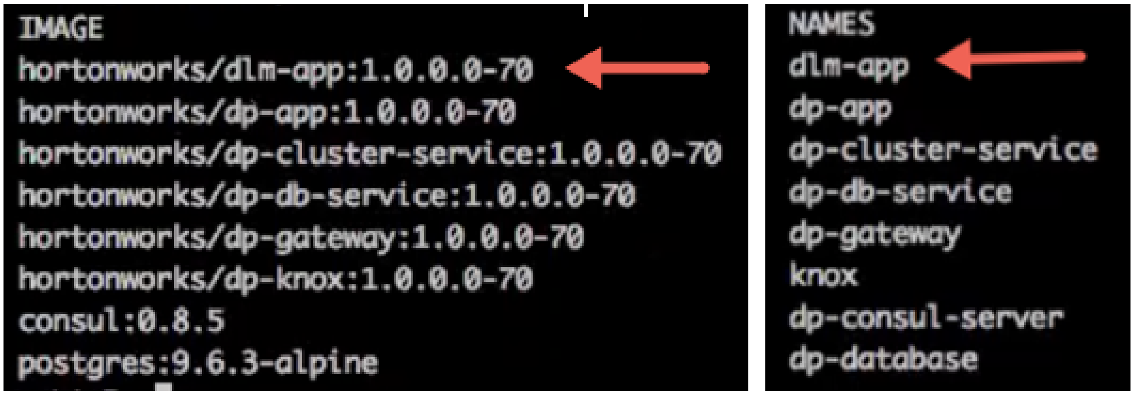
   The address must be the same one used when deploying the DPS Platform Docker container.

If you entered the configuration properties in the `config.env.sh` file, you are not prompted for the IP address.

5. Verify that the container you installed is running.

   ```
   docker ps
   ```

   The entry should be similar to the following:



```
IMAGE                                        NAMES
hortonworks/dlm-app:1.0.0.0-70               dlm-app
hortonworks/dp-app:1.0.0.0-70                dp-app
hortonworks/dp-cluster-service:1.0.0.0-70    dp-cluster-service
hortonworks/dp-db-service:1.0.0.0-70         dp-db-service
hortonworks/dp-gateway:1.0.0.0-70            dp-gateway
hortonworks/dp-knox:1.0.0.0-70               knox
consul:0.8.5                                 dp-consul-server
postgres:9.6.3-alpine                        dp-database
```

   If any containers are not running, you must destroy the containers and start over, as described in the troubleshooting section of this guide.

6. Log in to the DPS Platform UI to verify access to DPS.

   The first time you log in to DPS Platform, use the password that you set during installation.

   ⚠ **Important**

   Whatever URL you use the first time you log in is the URL you must continue to use, or log-in fails. For example, if you use the IP address of the DPS host at first login, then you will not be able to log in using the FQDN of the DPS host later, and vice versa.

   This is because Knox supports a whitelist of URLs to which it can validly redirect. DPS Platform automatically sets the whitelisted URL as the URL first used to access DPS. Typically, the FQDN for the host is used for the URL, because using the host FQDN simplifies administrative activities, such as replacing hardware.

7. Click **Services** in the navigation pane to ensure that the services you installed are discovered.

   At this point, you can only view the service icons on the Services page. You cannot enable the DPS services until you install and configure the service engines or agents on the clusters.

Next Steps

You can either configure the LDAP settings on the DPS host or install the service engines and agents on the clusters.

More Information

Setting Up the DPS Services (includes configuring LDAP)

Install the Engines and Agents on Ambari Clusters [24]

5.3. Install the Engines and Agents on Ambari Clusters

Some DPS services require that software be installed on the clusters used with the service. For example, Data Lifecycle Manager requires that the DLM Engine be installed on each cluster that is to be used in replication jobs. The engines and agents are installed on the Ambari host, using an Ambari MPack. They must be installed on each cluster that is to be used with DPS.

A management pack (MPack) bundles service definitions, stack definitions, and stack add-on service definitions so they do not need to be included with the Ambari core functionality and can be updated in between major releases.

Prerequisites

Ensure that you have downloaded the required MPacks from the customer portal to the Ambari Server host on each cluster you plan to use with DPS. Follow the download instructions provided as part of the product procurement process.

About This Task

You must have root access to the Ambari Server host node to perform this task.

This task must be completed on all clusters to be used with DPS.

“Beacon” is the internal name for the DLM Engine. If you install DLM, you will see the name Beacon in some paths, commands, etc.

Steps

1. Log in as root to an Ambari host on a cluster.

   ```
   ssh root@<ambari-ip-address>
   ```

2. Install the engine or agent MPack by running the following command.

   ```
   ambari-server install-mpack --mpack <mpack-file-name> --verbose
   ```

   Repeat this command for each service that you are installing.

3. Restart the Ambari server.

   ```
   ambari-server restart
   ```

4. Repeat this task on each cluster being used with DPS.

Next Steps

If installing DLM, configure the Beacon user as HDFS superuser.

Configure the services in Ambari.

More Information

Planning for a DPS Installation

Hortonworks Support Matrix

Configure the Beacon User if DLM Is Installed [25]

5.4. Configure the Beacon User if DLM Is Installed

The DLM Engine employs the default Beacon user to perform actions on the cluster nodes. The Beacon user is created during the DLM Engine installation, configured as a Hadoop Proxy superuser. However, the Beacon user must also be configured as an HDFS superuser, after completing the DLM Engine installation.

About This Task

This task must be completed on every NameNode on clusters used with DLM.

“Beacon” is the internal name for the DLM Engine. You will see the name Beacon in some paths, commands, etc.

Steps

1. Log into a NameNode host as *root*.
2. Assign the Beacon user to the HDFS superuser group.

   ```
   usermod -a -G hdfs beacon
   ```

3. Refresh configuration and mappings files.

   ```
   hdfs dfsadmin -refreshSuperUserGroupsConfiguration
   ```

   ```
   hdfs dfsadmin -refreshUserToGroupsMappings
   ```

4. Verify that Beacon was added as a user to the HDFS superuser group.

   ```
   hdfs groups beacon
   ```

   The output should display HDFS as one of the groups.

5. Repeat this process on every NameNode used with DLM for replication.

5.5. Configure the Engines and Agents on Existing Ambari-Managed Clusters

After the engines and agents for the DPS services are installed on the Ambari host, you must properly configure them. You can configure the engines and agents required for DPS services on existing clusters or when configuring newly created clusters. If configuring on new clusters, see "Configure the Engines and Agents on New Ambari Clusters [29]".

About This Task

You must have root access to the Ambari Server host node to perform this task.

This task must be completed on all clusters to be used with DPS.

“Beacon” is the internal name for the DLM Engine. If you install DLM, you will see the name Beacon in some paths, commands, etc.

DSS is available only as Evaluation Software.

Steps

1. Launch Ambari in a browser and log in.

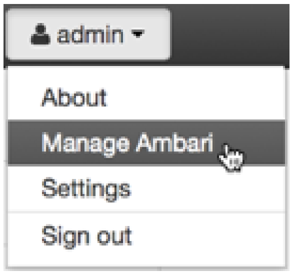
   `http://<ambari-server-host>:8080`

   Default credentials are shown below.

   Username: **`admin`**

   Password: **`admin`**

2. Click **Admin>Manage Ambari**.

3. Click **Versions**, and then do the following on the **Versions** page:

   a. Click the HDP version in the **Name** column.

   b. Change the **Base URL** path for each DPS service to point to the local repository, for example:

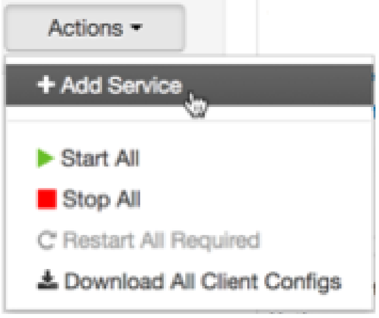
      ```
      http://webserver.com/DLM/centos7/1.0.0.0-X
      ```

      ```
      http://webserver.com/DSS/centos7/1.0.0.0-X
      ```

      URLs shown above are for example purposes only. Actual URLs might be different.

4. Click the Ambari logo to return to the main Ambari page.

5. In the Ambari Services navigation pane, click **Actions>Add Service**.



The Add Service Wizard displays.

6. On the **Choose Services** page of the Wizard, select the DPS service to install in Ambari, and then follow the on-screen instructions.

   Other required services are automatically selected.

7. When prompted to confirm addition of dependent services, give a positive confirmation to all.

   This adds other required services.

8. On the **Assign Masters** page, you can choose the default settings.

9. On the **Customize Services** page, fill out all the required username and password fields that are highlighted.

   You can set credentials to whatever you want, such as admin/admin.

10. If doing Hive replication with DLM, navigate to **Customize Services** and enable Hive replication.

    a. Click **Hive** in the list of services.

    b. On the **Settings** tab, move the toggle to *off* for "Run as end user instead of Hive user".

    c. Click the **Advanced** tab and scroll to the **Custom hive-site** section.

    d. Verify that these properties have the following values, or set the properties as shown:

    ```
    hive.metastore.dlm.events=true
    hive.metastore.transactional.event.listeners=org.apache.hive.hcatalog.
    listener.DbNotificationListener
    hive.repl.cm.enabled=true
    hive.repl.cmrootdir=/apps/hive/cmroot
    hive.repl.rootdir=/apps/hive/repl
    ```

    e. Click **HDFS** in the list of services.

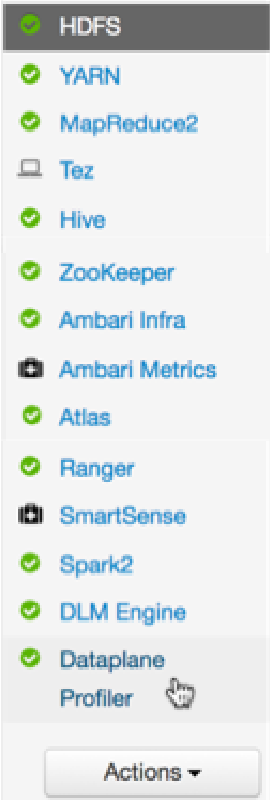
    f. Scroll to the **Custom core-site** section and modify the following parameter:

       hadoop.proxyuser.hive.hosts=*

11. If you added DataPlane Profiler, complete the following configuration steps.

a. Return to the Ambari main page.

b. In the Services pane, click **DataPlane Profiler**.



c. Click **Configs**>**Advanced**, and then scroll to **Custom core-site**.

d. Click **Add Property** and enter the following key-value pairs:

```
hadoop.proxyuser.livy.groups=*

hadoop.proxyuser.livy.hosts=*
```

12.Complete the remaining installation wizard steps and exit the wizard.

13.Ensure that all components required for your DPS services have started successfully.

See the support requirements for information about required Apache components.

14.Repeat this procedure on the Ambari hosts on all remaining clusters, for each DPS service engine and agent you installed.

More Information

Planning for a DPS Installation

Hortonworks Support Matrix

5.6. Configure the Engines and Agents on New Ambari Clusters

If you installed DPS on a new cluster, you can install the service engines and agents during the Install Wizard process. If configuring on existing clusters, see "Configure the Engines and Agents on Existing Ambari-Managed Clusters [25]"

About This Task

You must have root access to the Ambari Server host node to perform this task.

This task must be completed on all clusters to be used with DSS.

DSS is available only as Evaluation Software.

Steps

1. Launch Ambari in a browser and log in.

   `http://<ambari-server-host>:8080`

   Default credentials are shown below.

   Username: **`admin`**

   Password: **`admin`**

2. From the **Ambari Welcome** page, choose **Launch Install Wizard** and begin the wizard.

3. On the **Select Versions** page, change the **Base URL** path for each DPS service to point to the local repository, for example:

   ```
   http://webserver.com/DLM/centos7/1.0.0.0-X
   ```

   ```
   http://webserver.com/DSS/centos7/1.0.0.0-X
   ```

   URLs shown above are for example purposes only. Actual URLs might be different.

4. On the **Select Services** page of the Install Wizard, select the engine or agent you want to configure, and then follow the on-screen instructions.

   Other required services are automatically selected.

5. When prompted to confirm addition of dependent services, give a positive confirmation to all.

   This adds other required services.

6. On the **Assign Masters** page, you can choose the default settings.

7. On the **Customize Services** page, fill out all the required username and password fields that are highlighted.

   You can set credentials to whatever you want, such as admin/admin.

8. If doing Hive replication with DLM, from the **Customize Services** page, enable Hive replication.

    a. Click **Hive** in the list of services.

    b. On the **Settings** tab, move the toggle to *off* for "Run as end user instead of Hive user".

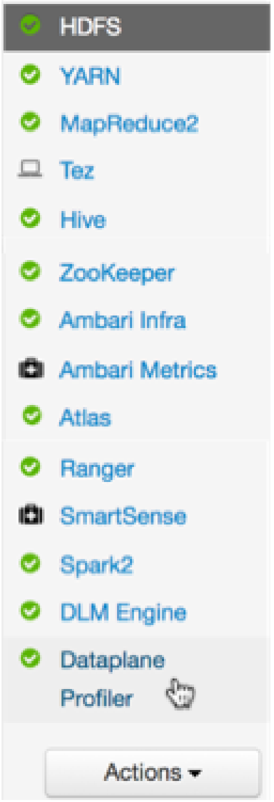
    c. Click **HDFS** in the list of services.

    d. Scroll to the **Custom core-site** section and modify the following parameter:

    hadoop.proxyuser.hive.hosts=*

9. If using DSS, configure the DataPlane Profiler.

    a. In the Services pane, click **DataPlane Profiler**.



    b. Click **Configs>Advanced**, and then scroll to **Custom core-site**.

    c. Click **Add Property** and enter the following key-value pairs:

    hadoop.proxyuser.livy.groups=*

    hadoop.proxyuser.livy.hosts=*

10.Complete the remaining installation wizard steps and close the wizard.

11.Ensure that all components required for your DPS services have started successfully.

See the support requirements for information about required Apache components.

12.Repeat this procedure on the Ambari hosts on all remaining clusters, for each DPS service engine and agent you installed.

More Information

Planning for a DPS Installation

Hortonworks Support Matrix

6. Configuring DPS for Secure Clusters

Before installing Hortonworks DataPlane Service (DPS), you should configure your clusters for the security options you plan to use.

Task Overview

Following is a high-level overview of the process for setting up security for DPS:

- Copy the public key from the `ssl-cert.pem` certificate.
- Create a `token.xml` file and add the public key.
- Add or copy the `token.xml` file, with the same configuration, to all other Knox nodes in each cluster.
- Configure Ranger for use with DPS.

Prerequisites

- Clusters used with DPS must have DPS version 2.6.3 installed and be managed by Apache Ambari 2.6.0 or later.
- You need to have your corporate LDAP settings available.

6.1. Enabling DPS to Interact with HDP Clusters

DPS services, such as DLM, must communicate with services on the HDP cluster like Ambari, DLM Engine, Atlas, Ranger, etc. The services perform actions on behalf of the user logged into DPS Platform, using the Knox token service. Therefore, you must use the Knox SSO service to provide federated single sign-on authentication between DPS Platform and all clusters enabled in DPS.

Prerequisites

Knox host IP mapping must be in the `/etc/hosts` file on DPS service container.

About This Task

To perform this task, you must have root user privileges on the DPS host and on all nodes that have Knox enabled.

To set up SSO for DPS, you must create a new topology file for Knox on all cluster nodes that have Knox enabled. This new topology file must be configured to contain the DPS public key, which is used to establish a trust relationship between DPS and the HDP Knox instance.

Steps

1. In a terminal, SSH to the DPS host.
2. Navigate to `$DP_INSTALL_HOME/certs/`.

   For example:

```
cd /usr/dp/current/core/bin/certs/
```

3. Display the content of the `ssl-cert.pem` file.

   For example:

   ```
   cat ssl-cert.pem
   ```

4. Copy and retain the DPS public key displayed in the certificate between “Begin Certificate” and “End Certificate”, because you need it in a succeeding step.

   The public key looks similar to the following:

   ```
   -----BEGIN CERTIFICATE-----
   NIICzTCCAaKjAwIBAgIIVJzHWfmsfP8wDQYJK0ZIhvcNAQEFCQAwXzELMAkGA1UE
   BhMCVVMxDTALBgMVBAgTBFRlc4QxDTALBgNVBAcTBFRlc3QxFzANBgNVBAoTBkhH
   ZG9vcDETMAsGA1WECxMEVGvzdDESMBAGA4UEAxMJbG9jYWxob3N0MB2XDTE3MDcx
   MjEzMTUxMVoXDTE0MDcxMjEzMTUxMVoWxzELMAkGA1UCBhMCVVMxDTALBgNVBAgT
   BFRlc9QxDTALBgNVBCcTBFRlc3QxFzANBgNVBAoTBkhhZG9vcDENNAsGA5UECxMF
   VGvzdDESMBBGA1UEAxMJbG8jYWxob3N2MIGfMA0GCSqGSIb3DQLBAQUAA4GNAKCB
   iQKBgQcYLhQDwCcQa12BZ2+v1gWICsFxOplW+EA6tBCJtMJDs5sNSV/XiomPxY2D
   8OU3oY68DiLs/U+la4K2mHp+gvh5+91EuMvXHtkui++7zDtD+cfBmsY5peAFwZ6g
   2NBwIjyMsKSiJWtT4syKMnAB5yv2p8xp3Z6c+0GCmZ+EeguWVQyDAQABMA0GCSqG
   zIb3DQEBbQUAA9GBAJAeMEFZY1Q4mK+RFq6wbshUOSQR+wB8zDkxAtgPfQINR9tK
   5MA8Iy6J90/eBUqGvAoN8PbEnTHU5VsL6m3J0vPmJ4EzFqCwI5VjeWdIMdoPPB/b
   QfmRZb0bpriGv6TrNdr9SKDTlchxW2tBbB1PaiR5yi3oEsuAaNKsi7GeT2wa
   -----END CERTIFICATE-----
   ```

5. Create a `token.xml` topology file.

   ```
   vi /etc/knox/conf/topologies/token.xml
   ```

6. Add the required content to the `token.xml` file:

   a. Add the basic topology content.

      You can copy and paste the following content into the file.

      ```
      <?xml version="1.0" encoding="UTF-8"?>
      <topology>
         <uri>https://$knox-hostname-FQDN:8443/gateway/token</uri>
         <name>token</name>
         <gateway>
            <provider>
               <role>federation</role>
               <name>SSOCookieProvider</name>
               <enabled>true</enabled>
               <param>
                  <name>sso.authentication.provider.url</name>
                  <value>https://$knox-hostname-FQDN:8443/gateway/knoxsso/api/
      v1/websso</value>
               </param>
               <param>
                  <name>sso.token.verification.pem</name>
                  <value>
                      $ADD-THE-PUBLIC-KEY-HERE
                  </value>
      ```

```
         </param>
      </provider>
      <provider>
         <role>identity-assertion</role>
         <name>HadoopGroupProvider</name>
         <enabled>true</enabled>
      </provider>

   </gateway>
   <service>
      <role>KNOXTOKEN</role>
      <param>
         <name>knox.token.ttl</name>
         <value>500000</value>
      </param>
      <param>
         <name>knox.token.client.data</name>
         <value>cookie.name=hadoop-jwt</value>
      </param>
   </service>
</topology>
```

The HadoopGroupProvider provider enables the Hadoop user-group mapping, to identify the groups to which users belong.

b. Replace *`$knox-hostname-FQDN`* with the fully qualified domain name of the host.

c. In the **`sso.token.verification.pem`** parameter, paste in the public key value that you copied in a previous step, replacing *`$ADD-THE-PUBLIC-KEY-HERE`*.

d. If using Ranger, also add the following content after the public key, within the `gateway` parameter:

```

      <provider>
         <role>authorization</role>
         <name>XASecurePDPKnox</name>
         <enabled>true</enabled>
      </provider>
```

This parameter enables Ranger authorization with the token topologies in Knox.

7. Perform a secure copy of the `token.xml` topology file to a Knox-enabled node on the HDP cluster.

8. Verify that Knox has picked up the files:

   - Log in to the Knox-enabled node.
   - Ensure that a directory called `token.topo.<number>` is present in the path `/var/lib/knox/data-<version>/deployments/`.

     If the file is not present, verify that the content in the `token.xml` file is correct. You can check the Knox gateway logs for error information.

9. Log in to each additional cluster used with DPS and repeat Step 5 (create a `token.xml` file) through Step 8 (verify copy of the file).

Next Steps

Configure Ranger to Restrict Access to DPS [35]

More Information

Add Host Entries to the /etc/hosts File on the Container

Defining Cluster Topologies

Overview of Knox Integration with DPS [35]

Security guide, Defining Cluster Topologies

6.2. Configure Ranger to Restrict Access to DPS

You must configure a Ranger policy for the new Knox topology, in order to restrict access to only authorized users of DPS.

1. Navigate to the Ranger UI.
2. Click **Access Manager**, and then click the Knox repository link (<*cluster-name*>_knox).

   <*cluster-name*> Policies.
3. Click **Add New Policy**, and then enter the following values:

Parameter	Value
Policy Type	Access
Knox Topology	token
Knox Service	*

4. Enter groups or user names in **Select Group** or **Select User**.
5. Optional: Under Policy Conditions click **Add Condition** and enter the IP addresses of the DPS host.

   This adds an IP-based filter to ensure that only known DPS Core hosts can access cluster services through the token topology.
6. Under Permissions, click **Add Permission** and select **Allow**.

6.3. Overview of Knox Integration with DPS

DPS enables multi-cluster data management and monitoring capabilities using several different services, such as DLM and DSS, that plug into the DPS Platform. In addition, DPS Platform provides a set of functions for all of the services that plug into DPS. This set of functions includes centralized authentication, HDP cluster registration and service endpoint discovery, and communication to data lake (cluster) services. Knox provides users and services with simplified and consistent access to clusters, data, and other services.

DPS authenticates users against a centralized identity provider in the organization, such as an LDAP or AD store. Knox ensures that those users and services are authorized to perform specific actions on the respective clusters, and propagates the identity of the user or service from DPS to the cluster services. DPS services include a deployment of Knox that is independent of all HDP clusters.

DPS and associated clusters can be in different domains, so DPS KnoxSSO cookies must be exchanged for KnoxToken responses that contain tokens. The tokens can subsequently be used as cookies for API calls to the services (Ambari, Ranger, Atlas) that accept SSO cookies but do not support trusted proxy access.

After Knox SSO configuration is completed successfully, when you log into DPS Platform, you are redirected to a Knox login screen. After you have successfully logged in through Knox, you are redirected to the URL of the SSO-enabled DPS Platform service.

Important

Whatever URL you use the first time you log in is the URL you must continue to use, or log-in fails. For example, if you use the IP address of the DPS host first to log in, then you will not be able to log in using the FQDN of the DPS host later, and vice versa.

This is because Knox supports a whitelist of URLs to which it can validly redirect. DPS Platform automatically sets the whitelisted URL as the URL first used to access DPS. Typically, the host name is used for the URL, because using the host name simplifies administrative activities, such as replacing hardware.

7. Setting Up the DPS Services

After you complete the installation and configuration of DPS Platform, the service plugins you selected, and any engines or agents associated with the services, then you can access DPS Platform. At first login to DPS Platform, you need to set up the LDAP server, register clusters, create users, and enable the services you want to use.

Before starting setup, you should have gathered the information identified in the following sections.

7.1. Preparing for DPS Service Setup

There is some information you need to provide during set up of DPS Platform and associated services, such as the LDAP server details, your SmartSense ID, the Ambari UI URL, and so forth.

7.1.1. Locate the Required IP Address if Using Packaged LDAP Server (Non-Production Only)

You can either use the LDAP server that is packaged with Apache Knox, to be used for testing purposes, or you can use an actual LDAP server. You should use your own LDAP server for production environments. If you choose to use the packaged LDAP server, you need to identify and copy the IP address of the packaged LDAP server. You must have the IP address to enter in the LDAP settings when you set up DPS Platform.

Important

The LDAP server packaged with Knox is recommended ONLY for non-production use.

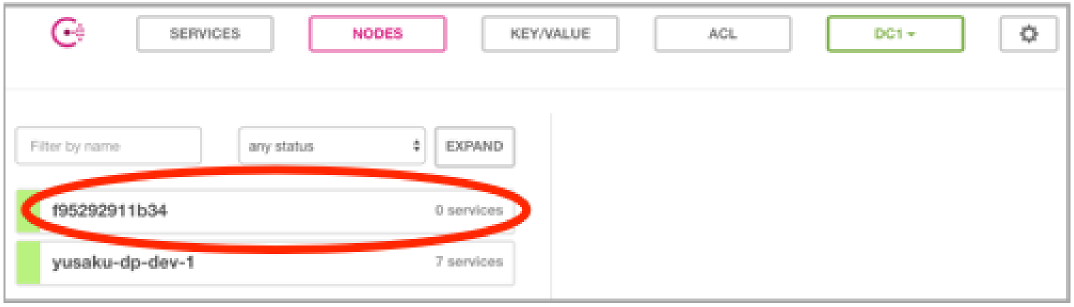
Steps

1. Navigate to the **Consul UI**.

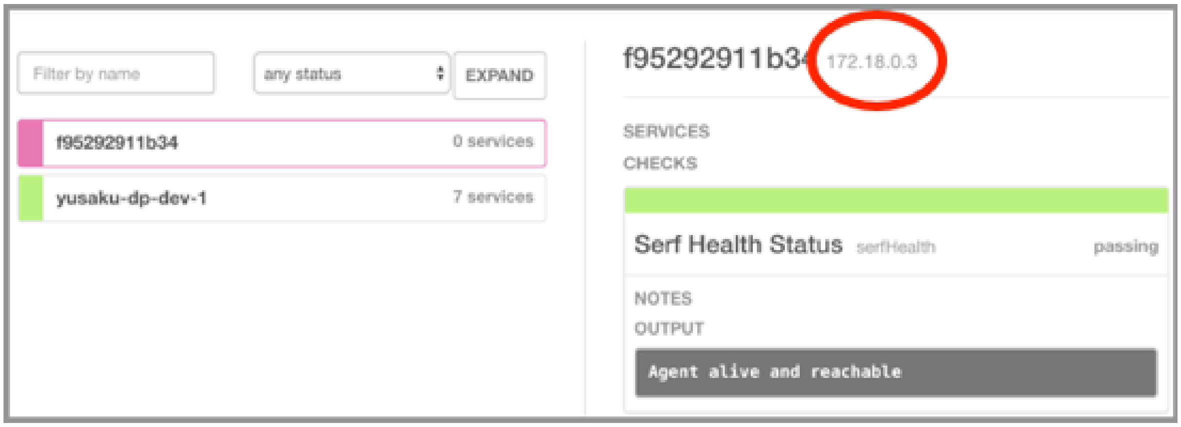
   http://<dataplane-host>:8500

2. Click on the **Nodes** tab.

3. Click the node that is identified with a hexadecimal string and **0 services**.



4. Make note of the IP address from the information panel that appears for the selected node.



```
Agent alive and reachable
```

Retain this IP address. You need it to complete the configuration of the packaged LDAP server that you can use for testing purposes.

7.1.2. Gather LDAP Server Details if Using Corporate LDAP

If you prefer to use a corporate LDAP server, you need to provide the required configuration information. You can use the table below to collect that information.

Ensure you use the correct LDAP settings. Only the LDAP URL and Bind DN (DPS Admin) user and password can be modified in DPS after LDAP properties are set. If any of the unalterable settings are incorrect, you have to destroy and reinstall the containers.

LDAP Properties	**Corporate Settings**
LDAP URL	
User Search Base	
User Search Attribute	
Group Search Base	
Group Search Attribute	
Group Object Class	
Group Member Attribute Name	
Administrator Bind DN	
Administrator Password	

7.1.3. Gather Additional Setup Information

Collect the following information prior to configuring DPS.

SmartSense ID	You must have a SmartSense ID to activate DPS services. Retrieve the ID from the Hortonworks Support Portal under the Tools tab.
Ambari URL	You need the Ambari URL to connect DPS Platform with Ambari-managed clusters.
LDAP users or groups to which you will assign roles in DPS.	You must assign appropriate roles to any users or user groups that will access DPS Platform and DPS plugin services.

7.2. Accessing DPS Platform for the First Time

The first time you log in to DPS Platform, you must configure LDAP, so you can set up users and groups in DPS. You must also register Ambari-managed clusters with DPS.

Before you can use any of the DPS services, such as DLM or DSS, you must enable the services, and then enable the clusters that have been configured for each service.

7.2.1. Configure LDAP for DPS

The first time you log into DPS (DataPlane) Platform, you see a Welcome page that directs you to configure the LDAP server and to add clusters to DPS Platform.

Prerequisities

If using the packaged LDAP server, ensure that you completed the task Locate the Required IP Address if Using Packaged LDAP Server.

About This Task

DPS comes with a pre-packaged "admin" user with permissions to perform initial bootstrapping actions.

Steps

1. In a browser, enter the FQDN of the DPS Platform host.

   `http://<DPS-host-FQDN>`.

   You can also use the IP address of the DPS Platform instance, but the FQDN is recommended.

   ### Important

   You must use a consistent method for accessing DPS Platform (either the FQDN, host name, or IP address) to avoid potential conflict issues with Apache Knox.

   ### Tip

   If a screen appears stating "Welcome to NGINX", the underlying services have not all started. Wait a moment and try again.

2. Log in as the DPS Super Admin:

   Username: `admin`

   Password: Use the password specified during DPS install

3. The DPS Platform Welcome page displays, showing the two-step setup process.

4. Click **Get Started**.

The Setup Authentication page displays the LDAP Configuration settings.

5. Do one of the following:

   - Enter the settings for your own LDAP server.

     Use the corporate LDAP values that you collected during the preparatory steps, Section 7.1.2, “Gather LDAP Server Details if Using Corporate LDAP” [38].

   - If using the LDAP server packaged with DPS, enter the following LDAP settings.

     The URL you enter includes the IP address you previously located in the Consul UI.

URL	ldap://<IP-from-Consul-UI>:33389
User Search Base	ou=people,dc=hadoop,dc=apache,dc=org
User Search Attribute	uid
Group Search Base	ou=groups,dc=hadoop,dc=apache,dc=org
Group Search Attribute	cn
Group Object Class	groupofnames
Group Member Attribute Name	member
Administrator Bind DN	uid=admin,ou=people,dc=hadoop,dc=apache,dc=org
Administrator Password	admin-password

Important

Ensure you enter the correct LDAP information before clicking Next. Only the LDAP URL and Bind DN (DPS Admin) user and password can be modified in DPS after LDAP properties are set. If any of the unalterable settings are incorrect, you have to destroy and reinstall the containers.

6. Click **Next**.

   A success message displays on the page.

7. Add users or groups.

   You can look up the users and groups from the LDAP server that you previously added, and add them as DPS Admin users. The DPS Admin should be used for all future logins, rather than the local admin superuser, under which you are currently logged in.

   For test LDAP scenarios, you can use the predefined LDAP users listed in “Add Users and Assign Roles for Services”.

8. Click **Save & Login**.

9. Log in as DPS Admin, using your SSO credentials.

   Any users who have been assigned the DPS Admin role can now login with their username and password.

   The Welcome page displays, showing the setup actions you perform in DPS Platform.

More Information

Add Users and Assign Roles for Services [41]

7.2.2. Add Users and Assign Roles for Services

After you set up the LDAP configuration for DPS Platform, you need to add users for the services you plan to enable. During LDAP configuration, you added users and groups that can log in as DPS Admin. You must now assign roles to users and groups, which allow users to access the services that plug into DPS. Each service has a predefined role associated with it. You can select one or more roles for each user.

Prerequisites

User accounts must already exist within your corporate LDAP prior to adding the user to DPS Platform. (This does not apply to the predefined users included with the LDAP server packaged with DPS.)

About This Task

The DPS Admin role is required to perform this task.

The LDAP server packaged with Apache Knox includes several predefined users and groups. You can assign roles to any of these users from DPS Platform. Only the "admin" (super admin) user has been granted permissions, by default, to perform all actions in DPS Platform.

The predefined LDAP users are:

- admin (super admin)

  Password = admin-password
- guest

  Password = guest-password
- sam

  Password = sam-password
- tom

  Password = tom-password

The predefined LDAP groups are:

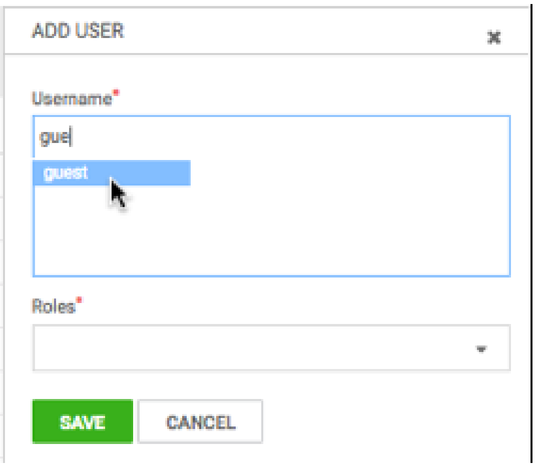
- analyst: Contains users “sam” and “tom”.
- scientist: Contains user “sam”.

Tip

Settings assigned to a user override the settings in any groups the user belongs to.

Steps

1. On the Welcome page, click **Getting Started>Users**.
2. Click **Add User**.
3. Enter the name of the user.

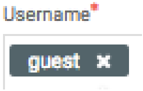


   If using your own LDAP server, the user must already exist within your corporate LDAP.

   If using the packaged LDAP, enter one of the predefined users (guest, sam, tom). The name autopopulates as you type.

   ### Tip

   You must click the name of the user when it displays and ensure it appears in the Username field on a dark background.



   If the name appears on a white background, it means the name is not recognized and the action fails.

4. Select one or more of the following roles to assign to the user:

DataPlane Admin	Can perform all actions in DPS (DataPlane) Platform, and can access and perform all actions in the UI of enabled services.
Infra Admin	Can perform all actions in the Data Lifecycle Manage (DLM) service UI, and can manage DLM-enabled clusters in DPS Platform.
Data Steward	Can perform all actions in the Data Steward Studio (DSS) service UI, and can manage DSS-enabled clusters in DPS Platform.

5. Click **Save**.

   The new user displays in the list on the Users page.

6. Optional: Add groups.

Note that user-level assignments override group-level assignments.

a. Click the **Groups** tab above the list of users.

b. Click **Add Group**.

c. Enter the group name and select one or more roles.

   The group must already exist within your corporate LDAP, if you are not using a predefined group.

d. Click **Save**.

   The new group displays in the list on the Groups page.

More Information

Required Roles

7.3. Register Clusters with DPS

You must register clusters with DPS before you can view or manage data on the clusters.

Prerequisites

All clusters used with DPS must be managed by Ambari.

All clusters must meet the requirements identified in Planning for a DPS Installation and Configuring DPS for Secure Clusters.

Steps

1. On the Welcome page, click **Getting Started**>**Clusters**.

   The Add Cluster page displays.

2. In a browser, enter the URL of the Ambari host for the cluster you want to add.

   Enter the URL using the format `http://<FQDN>:<port>`.

   You can also enter the IP address instead of the FQDN, but the FQDN is recommended.

Important

DPS Platform host must be able to resolve and reach the Ambari URL, whether you are using the FQDN or the IP Address. That is, you should be able to use **curl** or **wget** to access the Ambari URL from the DPS Platform host. If this requirement is not met, cluster registration fails.

If host names are resolved from `/etc/hosts`, you should explicitly register the cluster host names on the DPS container before the cluster is registered with DPS.

3. Click **Go**.

   Information about the cluster displays.

4. Select an item from the **Cluster Location** field.

   If you start typing the name of a city or country in the field, it autocompletes the text.

5. In the **Data Center** field, enter the name of the data center associated with the cluster.

   This property cannot be changed after it is set.

6. Optional: Add text strings in the **Tags** or **Description** fields and click **Enter**.

7. Click **Add**.

   The Clusters page appears. The cluster you added displays in the list and as a marker on the map.

8. To add more clusters, click the (Clusters) icon and complete the form for the new cluster.

More Information

Add Host Entries to the /etc/hosts File on the Container

Planning for a DPS Installation

Hortonworks Support Matrix

Troubleshooting Installation Issues

7.4. Enable Services

You must enable, through DPS Platform, any DPS service you want to use. Before enabling a service, you must have properly installed and configured the service UI on the DPS host, as well as the management pack for the service engine or agent on each cluster.

Prerequisites

You must have a SmartSense ID available.

About This Task

The DPS Admin role is required to perform this task.

Steps

1. Click the ⊞ (Services) icon in the DPS Platform navigation pane.

   The Services page displays. Services listed in the table have been enabled. Services identified by a tile icon are available to be enabled.

2. Move the cursor over the tile for the service you want to enable and click the **Enable** button that appears.

A verification page displays.

3. Enter the SmartSense ID and click **Verify**.

   The ID is case-sensitive.

   You can retrieve the SmartSense ID from the Hortonworks Support Portal under the Tools tab.

4. Click **Next**.

   The enabled service displays in the Enabled list on the Services page.

7.5. Enable Clusters for a Service

Each DPS service has specific configuration requirements that a cluster must meet before it can be used with the service. When you enable a service, a check is run to determine if the required service engine, such as DLM Engine, has been installed on any clusters. If the engine is installed but some configuration is still required, the cluster displays on the Services page with the action button Enable. If the cluster meets all requirements for the service it is automatically enabled, and the enabled cluster can only be viewed on the Services page by selecting the Show All Clusters action for the service.

Prerequisites

Clusters must be managed by Apache Ambari and registered with DPS Platform.

Before you can enable a cluster for a specific service, you must have enabled that service in DPS Platform.

About This Task

- The DPS Admin role is required to perform this task.
- Data Lifecycle Manager requires a minimum of two enabled clusters to perform replication jobs.

Steps

1. Click the ⊞ (Services) icon in the DPS Platform navigation pane.

   The Services page displays. Services listed in the table have been enabled. Services identified by a tile icon are available to be enabled.

2. Click on the row for a service.

   A list displays of any clusters that have the required service engine installed but have not yet been configured for use with the service.

   If no clusters display for the service, verify that the clusters you expect to see have been registered with DPS Platform, and that the proper service engine has been installed on the clusters.

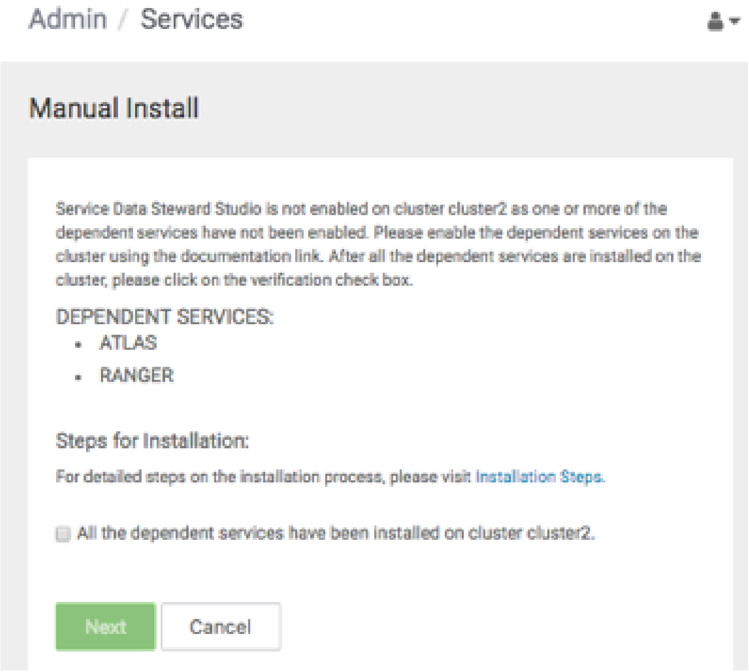
3. Click **Enable** for the clusters you want to use with the service.

   A check is run to determine what configuration is required on the cluster for the service you selected. For example, a required service such as Apache Ranger might have been installed on the clusters, but has not been enabled in Apache Ambari.

   The Manual Install page displays, indicating what you need to configure on the cluster to make the cluster usable by the service.

   a. Perform the actions stated on the Manual Install page.

      The required actions often involve enabling a service from Ambari. For example:



Manual Install

Service Data Steward Studio is not enabled on cluster cluster2 as one or more of the dependent services have not been enabled. Please enable the dependent services on the cluster using the documentation link. After all the dependent services are installed on the cluster, please click on the verification check box.

DEPENDENT SERVICES:
- ATLAS
- RANGER

Steps for Installation:

For detailed steps on the installation process, please visit Installation Steps.

□ All the dependent services have been installed on cluster cluster2.

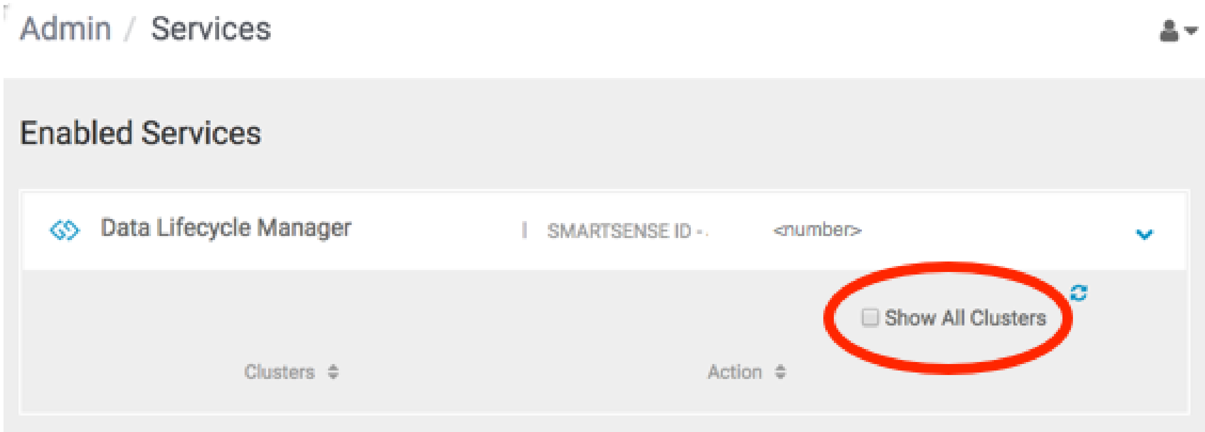
   b. When you complete the required actions, go to step 4.

4. On the Manual Install page, select the item "All the dependent services have been installed..." and click **Next**.

   Another configuration check is run and if all requirements are met, a verification message displays, indicating that the cluster meets requirements for the service.

5. Click the name of the service for which you enabled the cluster, then enable Show All Clusters.

   The new cluster displays in the list on the Clusters page.



7.6. Navigating Between Services

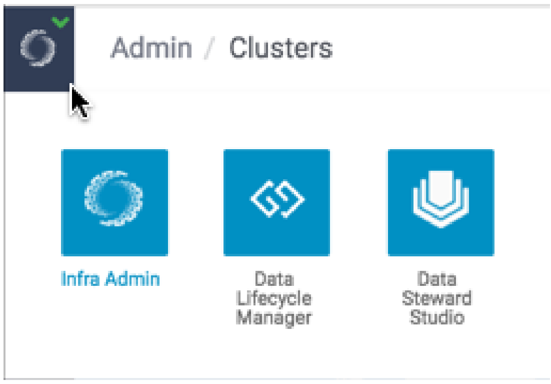
You can access any service for which you have been assigned the proper role. The DPS Admin has access to all DPS services.

Prerequisites

The DPS Admin must have assigned you the required role for any service you want to access.

Steps

1. Click the (Service Navigation) icon in the upper left corner of any page in DPS Platform.
2. Click the tile for the service you want.



If the service you want to access is not displayed, either the service is not enabled or you have not been assigned the role required to access the service. The DPS Admin can enable services and assign roles.

8. Reference

8.1. Installation Configuration Properties

During installation, you can access the DPS `/usr/dp/current/core/bin/config.env.sh` file to modify configuration properties.

The following list describes the installation configuration properties that you can set.

Configuration Item	Description	Default Value
`USE_EXT_DB`	Set to `yes` for pointing to an external Postgres instance, `no` otherwise	`no`
`DATABASE_URI`	If USE_EXT_DB is `yes`, this must point to the external Database URI	
`DATABASE_USER`	If USE_EXT_DB is `yes`, this must point to the DataPlane Admin user name of the external Database URI	
`DATABASE_PASS`	If USE_EXT_DB is `yes`, this must point to the DataPlane Admin password of the external Database URI	
`SEPARATE_KNOX_CONFIG`	Set to `true` if a separate Knox instance is setup on HDP clusters for handling DPS traffic, false otherwise	`false`
`KNOX_CONFIG_USING_CREDS`	If `SEPARATE_KNOX_CONFIG` is `true`, when a cluster is registered, we must provide additional information to discover it. This is either using Ambari credentials or explicitly specifying the URL. Set to `true` if you want to use Ambari credentials, `false` for URL	`true`
`CONSUL_HOST`	Set to the IP address of the host where DPS containers are launched	
`USE_TEST_LDAP`	Specifies whether to use an external LDAP instance or connect to a test LDAP instance that comes with the DataPlane Knox container	
`USE_TLS`	Set to `true` to enable TLS / HTTPS (SSL)	
`USE_PROVIDED_CERTIFICATES`	Set to `yes` if you have public-private key-pair already generated/issued. Setting to `no` automatically generates a key-pair for you.	
`PUBLIC_KEY_L`	If `USE_PROVIDED_CERTIFICATES` is `yes`, this must point to the absolute path of public key file	
`PRIVATE_KEY_L`	If `USE_PROVIDED_CERTIFICATES` is `yes`, this must point to the absolute path of encrypted private key file	

8.2. dpdeploy.sh Script Command Reference

You use the script **dpdeploy.sh** to deploy and configure DPS Platform during installation.

To use the tool, change to the directory to `/usr/dp/<version>/core/bin` and execute as **./dpdeploy.sh <*command*>**

The following table shows the actions that are supported by the script.

Command	Options	Default Value
init	`[ --all ]`	Initialize and start all containers for the first time

Command	Options	Default Value
migrate		Reset database to its pristine state and run schema migrations on it
utils add-host	`<ip> <host>`	Append a single entry to `/etc/hosts` file of the container interacting with HDP clusters
utils update-user	`[ ambari / atlas / ranger ]`	Update user credentials for services that DataPlane will use to connect to clusters
utils reload-apps		Restart all containers other than database, Consul and Knox
start	`[ --all ]`	Start all containers (existing data is retained)
stop	`[ --all ]`	Stop all containers (existing data is retained)
ps		List the status of associated docker containers
logs	`<container_name>`	Logs of supplied container id or name
destroy	`[ --all ]`	Kill all containers and remove them. Needs to start from init again. Deletes all data in the container
load		Load all images from `../lib` directory into docker daemon
upgrade	`--from <old_setup_directory>`	Upgrade existing `dp-core` to current version
version		Print the version of DPS

9. Examples

Before installing Hortonworks DataPlane Service (DPS), you should configure your clusters for the security options you plan to use.

9.1. Example `token.xml` Topology File with Ranger Enabled

```
<?xml version="1.0" encoding="UTF-8"?>
<topology>
   <uri>https://$knox-hostname-FQDN:8443/gateway/token</uri>
   <name>token</name>
   <gateway>
      <provider>
         <role>federation</role>
         <name>SSOCookieProvider</name>
         <enabled>true</enabled>
         <param>
            <name>sso.authentication.provider.url</name>
            <value>https://$knox-hostname-FQDN:8443/gateway/knoxsso/api/v1/
websso</value>
         </param>
         <param>
            <name>sso.token.verification.pem</name>
            <value>
                $ADD-THE-PUBLIC-KEY-HERE
            </value>
         </param>
      </provider>
      <provider>
         <role>authorization</role>
         <name>XASecurePDPKnox</name>
         <enabled>true</enabled>
      </provider>
      <provider>
         <role>identity-assertion</role>
         <name>HadoopGroupProvider</name>
         <enabled>true</enabled>
      </provider>

   </gateway>
   <service>
      <role>KNOXTOKEN</role>
      <param>
         <name>knox.token.ttl</name>
         <value>500000</value>
      </param>
      <param>
         <name>knox.token.client.data</name>
         <value>cookie.name=hadoop-jwt</value>
      </param>
   </service>
</topology>
```

10. Troubleshooting Installation Issues

Following are some common issues you might encounter during installation or setup of DPS.

To verify that your environment meets the requirements for DPS, see Planning for a DPS Installation and the Hortonworks Support Matrix.

10.1. Logging in Using the DPS Local Admin Role

When you log in as the local DPS Admin, you bypass Knox. The local admin role allows you to perform administrative activities and troubleshoot problems when access through LDAP and Knox is not available. The local admin is also the role you use to log in to DPS the first time, before LDAP is configured in DPS for SSO.

About This Task

For login, the default username is “admin”. The password you use to log in is set during the installation process.

Steps

1. Log in by appending `/sign-in` to the DPS login URL, for example:

   ```
   http://dataplane-host-name/sign-in
   ```

10.2. wget Command is Not Available

Use the command **yum install wget** to install the `wget` tool.

10.3. Delete and Clean Up Docker Containers

If you have problems with your installation or want to update a DPS container, you can delete the Docker containers and then install the new images.

About This Task

Important

Performing this task deletes all of your DPS Platform database content, so you will have to reconfigure the LDAP and cluster registration settings after reinstalling the Docker containers.

Steps

1. **cd /usr/dp/current/apps/dlm/bin**
2. **./dlmdeploy.sh destroy**
3. **cd /usr/dp/current/core/bin**

4. **./dpdeploy.sh destroy –all**

5. **docker ps** #this ensures that no containers are running. If you see any, kill them with `docker kill`.

6. Go to the step "Initialize all the DPS Platform Docker containers" in Install DPS and run the original DPS deployment commands starting with "dpdeploy.sh init –all".

See the dpdeploy.sh Script Command Reference.

10.4. Cluster Registration Error Messages

Following are errors you might encounter while registering a cluster in DPS on the Add Your Cluster page. Some possible causes and possible resolutions are also included.

10.4.1. Cluster is not reachable

This error indicates that the DPS containers are not able to resolve a provided hostname or use the IP address to connect to the machine.

Sample Message:

```
Failed: This is not a valid Ambari URL.
```

Possible Causes:

- DNS resolution is not setup.
- There are firewall or other networking restrictions that are preventing access.

Possible Resolutions:

- Verify that the specified hostname or IP address is valid and reachable from the DPS host machine. If it is reachable, try adding the hostname resolution to the DPS container using the **./dpdeploy.sh utils add-host `<ip> <host>`** command.
- Verify if network connectivity settings, such as firewalls, are configured correctly.

See the dpdeploy.sh Script Command Reference.

10.4.2. Knox is not set up on the HDP cluster, or Ambari credentials are incorrect for 'seeded user' mode

This error occurs when the cluster is reachable, but authentication is failing.

Sample Message:

```
Unable to connect, please retry. DataPlane could not retrieve
cluster information.
```

Possible Causes:

- Knox is not set up on the cluster.

- The user wants to use the less secure 'seeded user' mode, but the credentials of the seeded user (user name or password) are not setup in DPS.

Possible Resolutions:

- Validate that Knox is configured correctly as per documentation.
- If seeded user mode is being used (for evaluation purposes), add the correct credentials to DPS using **./dpdeploy.sh utils update-user ambari**.

See the dpdeploy.sh Script Command Reference.

10.4.3. Knox setup is incorrect on the HDP cluster

This error indicates that the cluster being registered has Knox enabled, but the communication from DPS to Knox is failing.

Sample Message:

```
Failed: There was an error fetching information from Ambari.
```

Possible Causes:

- The Knox token service is not properly configured.
- The public key of DPS is not set up correctly in the Knox topology.

Possible Resolutions:

- Validate that the Knox configuration for the token topology is done correctly, following the instructions Configuring DPS for Secure Clusters.

10.4.4. Cannot register a cluster, other causes

If you cannot register a cluster with DPS and none of the errors mentioned above are the cause, the following might apply.

Possible Causes:

- The Knox hostname is not reachable from the DPS host.
- The user who is logged in does not have Admin role in Ambari.

Possible Resolutions:

- Verify that the hostname where Knox is running is valid and reachable from the DPS host machine. If it is reachable, try adding the hostname resolution to the DPS container using the **./dpdeploy.sh utils add-host <*ip*> <*host*>** command.
- Verify that network connectivity settings, such as firewalls, are correctly configured.
- Verify that the username is an Ambari Admin on the cluster. If not, make the user an Ambari Admin user, by logging into Ambari, selecting the user, and providing Admin privileges.

See Planning for a DPS Installation and the Hortonworks Support Matrix for supported configurations.

10.5. Cluster status displays incorrectly on Details page

On the Cluster Details page, sometimes the Status of a cluster displays in gray, instead of red or green.

This generally indicates a timeout issue, in which DPS is not able to refresh the cluster details correctly. Manually refreshing the cluster information should fix the problem.

1. Click the Actions icon at the end of the row.

2. Click Refresh.

   Refresh of the cluster status might take several seconds.