Installing Streams Messaging Manager

Date of Publish: 2018-10-12

http://docs.hortonworks.com

Contents

Installation Overview

Streams Messaging Manager installation steps

You are encouraged to review this overview of Streams Messaging Manager (SMM) installation steps to get a thorough understanding of the requirements before you begin the installation process.

Procedure

1. Get your software.
   - Set up a local repository.
   - Create the repository configuration file.
2. Installing SMM Application on DPS.
   a. Set up DPS.
      - Install or upgrade to DPS 1.2.0.
      - Configure DPS.
   b. Install the SMM application.
3. Prepare your cluster for use with SMM.
   a. Install or upgrade Ambari.
   b. Install or upgrade HDF or HDP.
   c. Set up security on your cluster.
      - (Required) Set up Ambari with AD authentication.
      - (Required) Configure Knox SSO topology.
      - (Recommended) Install Ranger and set up permissions.
      - (Recommended) Set up Kerberos for your cluster.
   d. Install the SMM REST Agent.
      - Install a database for SMM.
      - Get the DPS Knox Cert key.
      - Download and install the SMM management pack.
      - Update base URLs.
      - Add the REST Server as an Ambari Service.
      - Configure TLS for SMM.
4. Integrate your cluster and DPS.
   a. Assign SMM users the Kafka DevOps role.
   b. Generate and upload your TLS public key.
   c. Set up Knox SSO for DPS.
   d. Add the HDP or HDF cluster and register it with DPS.
   e. Enable SMM Service for your target cluster.

Requirements for clusters used with SMM REST Server

The HDF or HDP cluster on which the SMM REST Server is installed has the following requirements. See the *Hortonworks Support Matrix* for information on the requirements for DP Platform and the SMM application.

Item	Versions
OS Versions	• RHEL/Centos 7 • Amazon Linux 2 • Debian 9 • SLES 12 • Ubuntu 12 • Ubuntu 14 • Ubuntu 16
HDP Versions	3.0.0, 3.0.1
HDF Versions	3.2
Ambari Versions	2.7.0, 2.7.1
Database	• Postgres 9.5 or later • MySQL 5.5 or later • MariaDB 10 • Oracle 12 and 11g Release 2
JDK	JDK 8 (Open JDK & Oracle JDK)

Related Information

Hortonworks Support Matrix

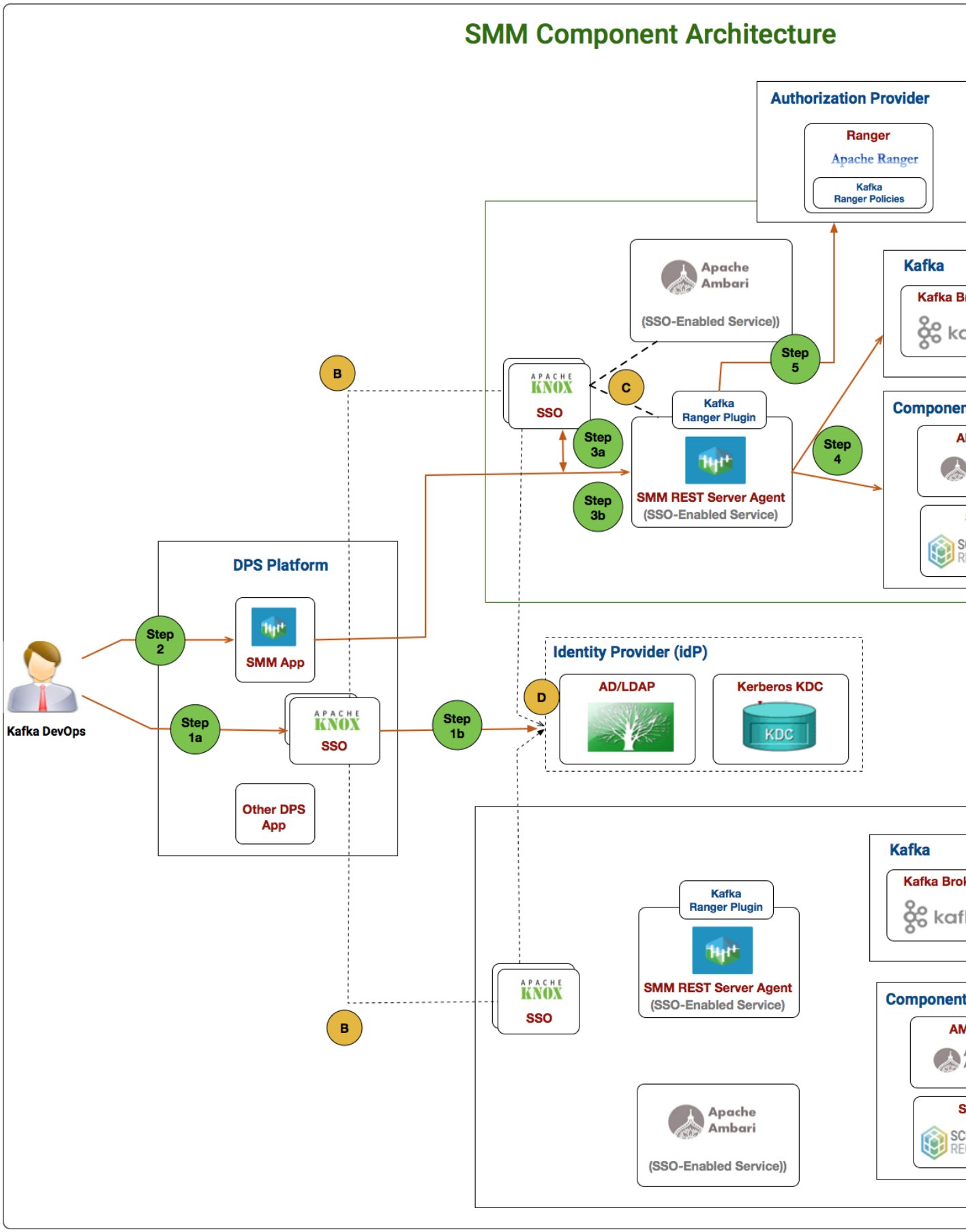
HDF / HDP target cluster service requirements

These are the component services required for your SMM REST Server Agent target cluster.

Component	Purpose	Comment
SMM Rest Server Agent	Provides REST endpoints that power the SMM DP Platform Application to manage and monitor.	Installed using a separate Ambari Management Pack.
Apache Kafka	The Kafka cluster that you are managing with SMM.	Installed using HDF or HDP.
Apache ZooKeeper	ZooKeeper contains information about Kafka metadata.	
Knox	Provides SSO services from the DP Platform Application in DP Platform, to the SMM REST Server on the HDP/HDF cluster. The logged in user in DP Platform using Knox SSO connects to SMM REST Services.	Installed using HDP or HDF.
AMS	Ambari Metrics Server is a time Series and Graph database built on top of HBase. It houses all the time series Kafka Metrics that power SMM.	Installed using HDP or HDF.
AD/LDAP	Identity Provider Store that houses user and group information for users that access the system. DataPlane and HDP/HDF must use the same LDAP instance.	

SMM component architecture

Before you get started with SMM, it is helpful to review this information about SMM component architecture.


SMM Component Architecture
Authorization Provider
Ranger
Apache Ranger
Kafka
Ranger Policies
Apache
Ambari
(SSO-Enabled Service))
Kafka
Kafka B
Step
5
B
APACHE
KNOX
SSO
C
Kafka
Ranger Plugin
Step
3a
Step
4
Step
3b
SMM REST Server Agent
(SSO-Enabled Service)
Componer
DPS Platform
Step
2
SMM App
Identity Provider (idP)
D
AD/LDAP
Kerberos KDC
KDC
Kafka DevOps
Step
1a
APACHE
KNOX
SSO
Step
1b
Other DPS
App
Kafka
Kafka Brok
Kafka
Ranger Plugin
SMM REST Server Agent
(SSO-Enabled Service)
APACHE
KNOX
SSO
Component
B
Apache
Ambari
(SSO-Enabled Service))

The below table outlines key components of SMM as described in the above diagram.

Component Name / Version	Version Supported by SMM	Description	Where it is deployed	Binary Location	Required/Optional
DP Platform	1.2.0	A platform that provides an extensible set of services for HDP and HDF clusters.	DPS_HOST	Customer Support Portal	Required
SMM Application	1.1.0	A DP Platform application that provides a monitoring platform for Kafka clusters.	DPS_HOST	Customer Support Portal	Required
DP Platform Knox	N/A	DP Platform Knox instance is used for SSO between DP Platform and any HDP/HDF Clusters that it is managing. This Knox instances needs to be synced with the same LDAP store as the target clusters.	DPS_HOST	N/A (Part of DP Platform)	Required
HDP / HDF	HDP and/or HDF	SMM supports monitoring Kafka version (1.1.1) that are part of HDP or HDF.	HDP/HDF Nodes	Hortonworks Public Downloads Page	Required (Either HDP or HDF)
SMM REST Server Agent	1.1.0	Provides REST endpoints that power the SMM Application to manage and monitor Kafka clusters. Knox SSO must be enabled for this service. Note that enabling TLS for SMM REST Server Agent is strongly recommended for any production setup.	HDP or HDF Cluster	Customer Support Portal	Required
HDP/HDF Knox	N/A	Provides SSO services from the SMM Application in DP Platform to the SMM REST Server on the HDP/ HDF cluster. The logged in user in DP Platform using Knox SSO connects to SMM REST Services, Ambari Services, and other services.	HDP or HDF Cluster	N/A (Part of HDP/ HDF)	Required
Ambari	2.7.1	HDP or HDF Cluster must be provisioned by Ambari and SSO must be enabled for Ambari and SMM Rest Server.	HDP/HDF Cluster	Hortonworks Public Downloads Page	Required

Kafka	1.1.1 (installed as part of HDP / HDF)	Kafka 1.1.1 that is installed via HDP or HDF is required for SMM.	HDP/HDF Cluster	N/A (Part of HDP/ HDF)	Required
AMS	Part of HDP 3.0 and HDF 3.2)	Ambari Metrics Server is a time Series and Graph database built on top of HBase. It houses all the time-series Kafka Metrics that power SMM.	HDP/HDF Cluster	N/A (Part of HDP/ HDF)	Required
Zookeeper	Part of HDP and HDF)	ZooKeeper contains information about Kafka metadata.	HDP/HDF Cluster	N/A (Part of HDP/ HDF)	Required
AD/LDAP		Identity Provider Store that houses user and group information for users that access the system. Data Plane and HDP/HDF must use the same LDAP instance.	Up to the Organization	Organization specific	Required for Knox SSO integration with HDP/HDF cluster and DP Platform
Kerberos / KDC		System used for strong authentication and identity propagation for both user and services in an HDP/HDF Cluster.	HDP/HDF Cluster	Organization Specific	Optional, but strongly recommended)
Ranger	Part of HDP 3.0 and HDF 3.2)	The Kafka Ranger plugin is used by SMM Rest Server to ensure that the logged in user only has access to the topics the user is configured to see based on the configured Ranger policies.	HDP/HDF Cluster	N/A (Part of HDP/ HDF)	Optional, but strongly recommended)
Schema Registry	Part of HDF	A central repository that houses schema information including serializers/ deserializers for a given Kafka topics. SMM uses SR for the Topic Visualizer capability.	HDF Cluster	N/A (Part of HDF)	Optional, but recommended)
Atlas	Part of HDP	SMM provides links to Kafka entities in Atlas.	HDP cluster	N/A (Part of HDP Cluster)	Optional
Grafana	Part of HDP and HDF)	SMM provides links to broker or topic dashboards in Grafana for more detailed metrics.	HDP/HDF Cluster	N/A (Part of HDP/ HDF)	Optional

The below table outlines some of the key integration points between the different SMM components. These integration points are marked in the diagram above with yellow circles.

Integration Point	Description

	• HDP and HDF clusters can be registered within DP Platform, if they meet the DP Platform prerequisites. Clusters are registered by providing Ambari Endpoint URL. • The DP Platform user that registers a cluster must also be an “Admin” user within that Ambari instance. • If these registered clusters have the prerequisite services that SMM requires (see below sections on these prerequisites), each of these clusters can be enabled with the SMM Application. • The SMM Application can be used to monitor the Kafka clusters in each of the enabled HDP/HDF clusters.
	• This is the integration point between DP Platform Knox and each registered cluster's Knox instance. • This integration point is through the configuration of the SSO token topology on the target cluster as documented in the DP Platform Installation Guide. • This allows SMM Application to retrieve an SSO token from the target cluster’s Knox instance and use that token to make SMM REST Server requests.
	• This is the integration point between services in the target cluster (Ambari, SMM Rest Server) and the target instance’s Knox service. • This integration point allows the service to be SSO enabled. • This is configured through the SSO configuration for that service by configuring the Knox’s public key within knox-sso-cert.pem.
	• This integration point is between DP Platform and all the registered clusters to use the same LDAP instance for authentication. • Note in this diagram, the DP Platform Knox instance and the Knox instance in Cluster 1 and Cluster 2 all share the same LDAP instance.

The below table outlines the flow of when a SMM Kafka DevOps users named gjvett logs into DP Platform and uses SMM to monitor and manage a Kafka cluster. These integration points are marked in the diagram above with green circles.

	User gjvett logs into DP Platform with credentials.
	• DP Platform Knox talks to configured LDAP instance to validate the user credentials. • If credentials are valid, DP Platform fetches the configured roles for user gjvett (roles are managed in DP Platform).
	• If gjvett has DP Platform Role “Kafka DevOps”, then the user sees the SMM Application in DP Platform. • User gjvett clicks the SMM Application icon and selects a cluster to manage and monitor with SMM.

	For the cluster selected, SMM Application talks to the target cluster's Knox instance to fetch SSO token for user gjvett to be able to talk to SMM REST Server Agent on the target cluster.
	SMM Application then makes calls to SMM REST Server Agent as gjvett with SSO token stored in cookie header named hadoop-jwt.
	SMM REST Server Agent then queries a series of components like Kafka internal consumer group topics, Ambari AMS, Schema Registry, etc to fetch data required for user's action in the SMM Application.
	The information returned to SMM Application is filtered based on the ACL permissions associated for user gjvett (only information on topics that user gjvett has permission to see is returned.)

Obtaining necessary SMM software

SMM installation artifacts

Prior to starting the installation, you must download the SMM repository tarballs and management pack from the Hortonworks Customer Portal following the instructions provided as part of the subscription fulfillment process. For SMM, you need to get an RPM tarball for the SMM application, and an RPM tarball and an Ambari management pack for the SMM Rest Server Agent

The general steps you take to access the SMM installation artifacts are:

1. Log into the Hortonworks Customer Portal. Access instructions are provided as part of your subscription fulfillment process.
2. Click the Streams Messaging Manager link at the bottom of the portal landing page to access the SMM Download Page.
3. Download the following artifacts appropriate to your operating system:
    - Repository Tar Ball for the SMM App
    - Repository Tar Ball for the SMM REST Server (Cluster Agent)
    - Ambari Management Pack Tar Ball for the SMM Rest Server (Cluster Agent)

SMM Artifact Name	Artifact Type	How is it Installed

SMM App	RPM tarball that user needs to create a local repository from	• user needs to create local repository • from that, then do their yum install. • One tarball per OS, and in this case, RHEL/CentOS/OEL 7 to match DPS Platform.
SMM Rest Server Agent (Cluster Agent)	Two pieces: **1.** RPM tarball from which you can create a local repository **2.** Ambari Management Pack tar ball Need #1 and #2 per OS that you will support for clusters.	Ambari management pack scripts

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. You can use the same local repository for DPS, the SMM application, and the SMM REST Server, or you can create unique local repositories for each.

Before you begin

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

Procedure

1. Copy the SMM Application and SMM REST Server (Cluster Agent) 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 SMM/centos7. These directories contain the repositories.

Expanding the app 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.

  For the SMM, the repository name is SMM-APP.
- <OS>

  This is the operating system version.
- <service-version-X>

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

Base URL Examples

Base URL for the SMM Application:

```
http://webserver.com:port/SMM-APP/centos7/1.0.0.0-X
```

Base URL for the SMM REST Server (Cluster Agent):

```
 http://webserver.com/SMM/centos7/1.0.0.0-155/
```

Be sure to record these Base URLs, because you need them when installing the application on the host.

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
```

Results

The repositories are now prepared for installation.

What to do next

Create the configuration file for the repository.

Create the Repository Configuration File

A repository configuration file must be created for the Streams Messaging Manager Service 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. A unique repository configuration file is required for DPS and the SMM application. No configuration files is required for the SMM Rest Server.

Procedure

1. Navigate to the repository directory.
   cd /etc/yum.repos.d/
2. Create a repository file.
   vi smm-app.repo

   Alternatively, you can copy an existing repository file to edit.
3. Add 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: DSS-APP-1.0.0.0-59

```
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

Example Repository File

```
#VERSION_NUMBER=1.0.0.0-59
[SMM-APP-1.0.0.0-59]
name=SMM-APP Version - SMM-APP-1.0.0.0-59
baseurl=http://<your_webserver>:port/SMM-APP/centos7/1.0.0.0
gpgcheck=1
gpgkey=http://<your_webserver>:port/SMM-APP/centos7/1.0.0.0/RPM-GPG-KEY/RPM-GPG-KEY-Jenkins
enabled=1
priority=1
```

Installing Your SMM Application on DP Platform

Set up DP Platform

To get started installing SMM, you must first install or upgrade to DP Platform 1.2.0, and configure it according to the instructions provided in the DP Platform installation documentation.

Procedure

1. Install or upgrade to DP Platform 1.2.0.
2. Configure DP Platform.

Related Information

Installing DP Platform

Install the Streams Messaging Manager Application

Follow the instructions to install the SMM application.

Before you begin

You have successfully installed DP Platform and DP Platform is running.

Procedure

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

```
sudo su
```

2. Install the RPMs for the service application.

```
yum install smm-app
```

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

If the yum command fails, then the local repository was not set up correctly. Check the repository file /etc/yum.repos.d/smm-app.repoon the host.

3. Navigate to the directory containing the installation scripts for the service, for example:
cd /usr/smm-app/current/streams-messaging-manager//bin
4. Load the Docker images and initialize the environment.

```
./smmdeploy.sh init
```

Loading the images might take a while.

Note:

If you run into errors while deploying, you must destroy the deployment using ./smmdeploy.sh destroy command and re-install the app. To check the logs of the container, you can use the command ./smmdeploy.sh logs.

5. Verify that the container you installed is running.
./smmdeploy.sh ps

Make sure that the container is running.

Prepare Your Cluster for Use with SMM

Install or Upgrade Ambari, HDF, and HDP

Get started with the SMM installation by preparing your target cluster. Your cluster must be managed by Ambari 2.7.0, and it can be an HDF 3.2 cluster, an HDP 3.0 cluster, or an HDP 3.0 cluster with HDF 3.2 services installed.

Procedure

1. Install or upgrade to Ambari 2.7.0.0.
2. Install or upgrade your HDF 3.2.0 or HDP 3.0.0 cluster. You can also install HDF services on an HDP cluster.

Review the *Target cluster service requirements* information to understand which services are required for SMM 1.0.0.

Related Information

Installing Ambari
Upgrading Ambari
Installing HDP
Upgrading HDP
Installing HDF
Upgrading HDF

Security Setup

Mandatory security configuration

Procedure

1. Set up Ambari with AD authentication.
2. Configure Knox SSO topology for Ambari.

Related Information

Configuring Ambari Authentication with LDAP/AD
Configuring Knox SSO between DP Platform and Your Cluster

Recommended security configuration

Procedure

1. Set up Kerberos for your cluster.
2. Install Ranger and set up permissions.
3. Set up TLS.

Related Information

Configuring Authentication with Kerberos
Installing Apache Ranger
Providing Authorization with Apache Ranger

Install the SMM REST admin server

Configure your SMM database

About this task

You can use the following databases with SMM:

- Postgres
- MySQL
- Oracle
- MariaDB

Procedure

1. To configure a MySQL database:

```
create database streamsmsgmgr;
CREATE USER 'streamsmsgmgr'@'localhost' IDENTIFIED BY 'streamsmsgmgr';
GRANT ALL PRIVILEGES ON streamsmsgmgr.* TO 'streamsmsgmgr'@'localhost'
 WITH GRANT OPTION;
CREATE USER 'streamsmsgmgr'@'%' IDENTIFIED BY 'streamsmsgmgr';
GRANT ALL PRIVILEGES ON streamsmsgmgr.* TO 'streamsmsgmgr'@'%' WITH GRANT
 OPTION;
```

2. To configure a Postgres database:

```
# Log in database
sudo su postgres
psql

# Setup databases and users
create database streamsmsgmgr;
CREATE USER streamsmsgmgr WITH PASSWORD streamsmsgmgr;
GRANT ALL PRIVILEGES ON DATABASE "streamsmsgmgr" to streamsmsgmgr;
```

Configure Knox for SMM integration

Procedure

1. From the Ambari UI **Advanced streams-messaging-manager-sso-config**, verify that **Authentication.provider.url** is accurate.
2. Generate your **public.key.pem**.
   a. From **Knox | Configs | Advanced knoxsso-toplogy**, add the following:

```
<name>main.ldapRealm.userDnTemplate</name>
<value>CN=admin1,CN=Users,DC=HWQE,DC=HORTONWORKS,DC=COM</value>

<name>main.ldapRealm.contextFactory.url</name>
<value>ldap://ad-nano.qe.hortonworks.com:389</value>

<name>knoxsso.redirect.whitelist.regex</name>
<value>.*;^/.*$;https?://localhost*$;^http.*$</value>
```

   b. Save this change and restart Knox.
   c. From the command line, generate your pem key, using the default keystore password admin when prompted:

```
/usr/jdk64/jdk1.8.0_112/bin/keytool
-export
-alias gateway-identity
-rfc
-file /root/knox-sso-cert.pem -keystore /usr/hdf/current/knox-server/
data/security/keystores/gateway.jks
```

3. Open knox-sso-cert.pem from /root and remove any new line characters or special characters.
4. From the Ambari UI **Advanced streams-messaging-manager-sso-config**, upload the **Public.key.pem**.

Install the SMM management pack

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.

Before you begin

You have obtained the management pack and public repository locations from the Hortonworks . Customer Portal following the instructions provided as part of the subscription fulfillment process.

Procedure

1. Back up your Ambari resources folder:

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

2. Copy the bundle to /tmp on the node where you installed Ambari.
3. Install the SMM management pack.

```
ambari-server install-mpack
--mpack=tmp/smm-ambari-mpack-1.0.0.0-<version>.tar.gz
--verbose
```

4. Restart the Ambari Server:

```
ambari-server restart
```

5. Once Ambari server is restarted, update public repo URL, Go to Stack and Versions Section on Ambari, click Versions -> Manage Versions, Click HDP or HDF version link. Update SMM link with its public repo link.

Related Information

SMM installation artifacts

Update the SMM Base URL

Adding the base URL tells Ambari where to look for the SMM repository. The base URL will be included in the customer support portal, where you get the repository. This step is necessary because you are using an existing Ambari instance that is already managing an HDP or HDF cluster.

About this task

Update the base URL with the base URL you created for the SMM REST Server (Cluster Agent) management pack when you created the local repo. For example:

```
http://webserver.com/SMM/centos7/1.0.0.0-155/
```

Procedure

1. From the Ambari menu, click the **admin** drop-down on the top right of your Ambari Dashboard. Then select **Manage Ambari**.
2. From the **Clusters** view on the left, click**Versions**. Then select the **HDP versions** or **HDF versions** link, depending on your cluster.
3. Add the SMM Base URL appropriate for your operating system.
4. Click **Save**.

Add the SMM REST Server as a Service

Once you have installed the SMM management pack and updated the Base URL in Ambari, you are ready to add the SMM Rest Server as a service in your Ambari-managed HDF or HDP cluster.

Procedure

1. From the Ambari UI, launch the **Add Service Wizard**.
2. In Step 1, **Choose Services**, select Streams Messaging Manager.
3. In Step 2, **Assign Masters**, add the host on which you want to install the SMM REST Server.
4. In Step 4, **Customize Services**, select the Streams Messaging Manager tab and add the configuration information for the SMM REST Server.
5. Click through the rest of the **Add Service Wizard** and click **OK**.

Results

You can verify the SMM installation by launching the following:

- http://{your hostname}:8585/swagger
- http://{your hostname}:8585/api/v1/admin/brokers

In both cases the default user name is admin and the default password is Horton!#works.

Integrating Your Cluster with DP Platform

Assign users the Kafka DevOps role

Users must be assigned the **Kafka DevOps** role before they can access SMM from the DP Platform UI.

Procedure

1. From the DataPlane left navigation pane, go to **Users**.
2. Identify the user to which you want to give SMM access, and select **Edit** from the **Actions** icon on the right-hand side.
3. Add **Kafka DevOps** to the roles for the user and click **Save**.

Results

That use is now able to access SMM from the DataPlane UI.

Upload your TLS public key

If the SMM REST endpoint on your HDF or HDP cluster is TLS enabled, you must upload the TLS public key to DPS.

Procedure

1. From the command line, enter:

```
openssl s_client -showcerts -connect <cluster-name:cluster-port>
```

2. Copy the certificate, including the begin and end lines, and save to a file named SMM.pem.
3. From the DataPlane left navigation pane, click **Settings**.
4. Click **Upload** to upload your SMM.pem file

Enable SMM Service for Target Cluster Registered in DP Platform

Procedure

1. From the DataPlane UI left navigation pane, go to **Services** and click the SMM application.
2. Click **Enable** for each cluster you want to use with SMM.

Updating Ranger Users

Before you can launch SMM, you must manually add a user to Ranger, add the user to Ranger Policies for the Kafka service, and add the SMM user to the Ranger Policy for Kafka. Additionally, if the SSL is enabled for Ranger, you must add the Ranger plugin SSL CLName configuration value.

Procedure

1. Add a User to Ranger.
   a. From the Ranger UI, go to Settings, then Users/Groups, and ensure that the Users tab is selected.
   b. Click Add New User.
   c. Provide the user name. This user name is derived from the streams_messaging_manger_principal_name you set during the Ambari Kerberos configuration. For example: streamsmsgmgr-cluster-smm.
   d. For the Role, select User. For the Group, select hadoop, streamsmsgmgr, and ranger.
   e. Click **Save**.
2. Add user to Ranger Policy for Kafka Service.
   a. From Ranger UI, Service Manager, in the Kafka service pane, click the hyperlink (**cluster-name_kafka**).
   b. Add the SMM user to both policies. Select the edit policies icon, and from Allow Conditions, add the SMM user to the Select User field. Also add sreamsmsgmgr user, if it does not already exist.
3. Add SMM user to Ranger Policy for Kafka.
   a. From the Ranger UI, Services Manager, and select the Edit icon for the Kafka service.
   b. Add the streamsmsgmgr-cluster-smm user name to the following two configuration values:
      - policy.download.auth.users
      - tag.download.auth.users
4. (If SSL is enabled for Ranger) Update the Ranger plugin SSL CLName. Go to Config Properties | Ranger plugin SSL CLName. For example: Kafka Client. The CLName is the value you set up when generating your Ranger Admin SSL certificate.