HDP Installing Ranger KMS 3

# Installing Apache Ranger KMS

**Date of Publish:** 2019-12-17

**https://docs.hortonworks.com**

# Contents

# Installing the Ranger Key Management Service

This section describes how to install the Ranger Key Management Service (KMS) using Ambari on a Kerberized cluster.

### Prerequisites

Ranger KMS requires HDFS and Ranger to be installed and running on the cluster.

To install Ranger using Ambari, refer to “Installing Ranger Using Ambari”. (For more information about the Ambari Add Service Wizard, see “Apache Ambari Operations > Adding a Service” in Apache Ambari Operations.)

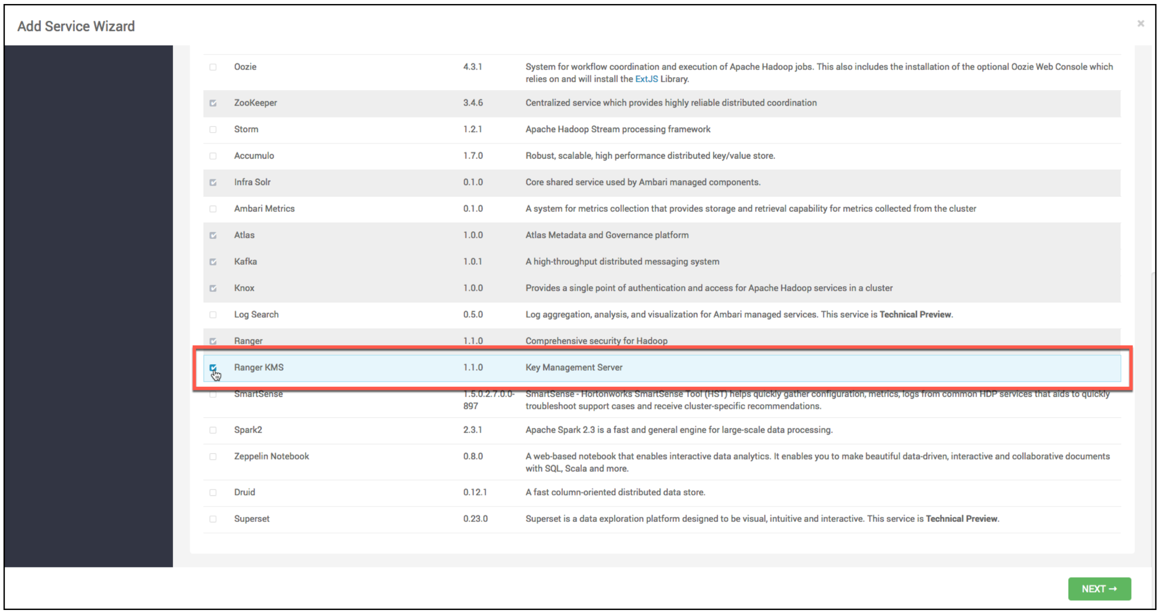
To use 256-bit keys, install the Java Cryptography Extension (JCE) Unlimited Strength Jurisdiction Policy File on all hosts in the cluster. For installation information, see “Install the JCE for Kerberos”. Make sure that the Java location is specified in the $PATH environment variable.

# Install Ranger KMS using Ambari (Kerberized Cluster)

To install Ranger KMS on a Kerberized cluster, complete the following steps.

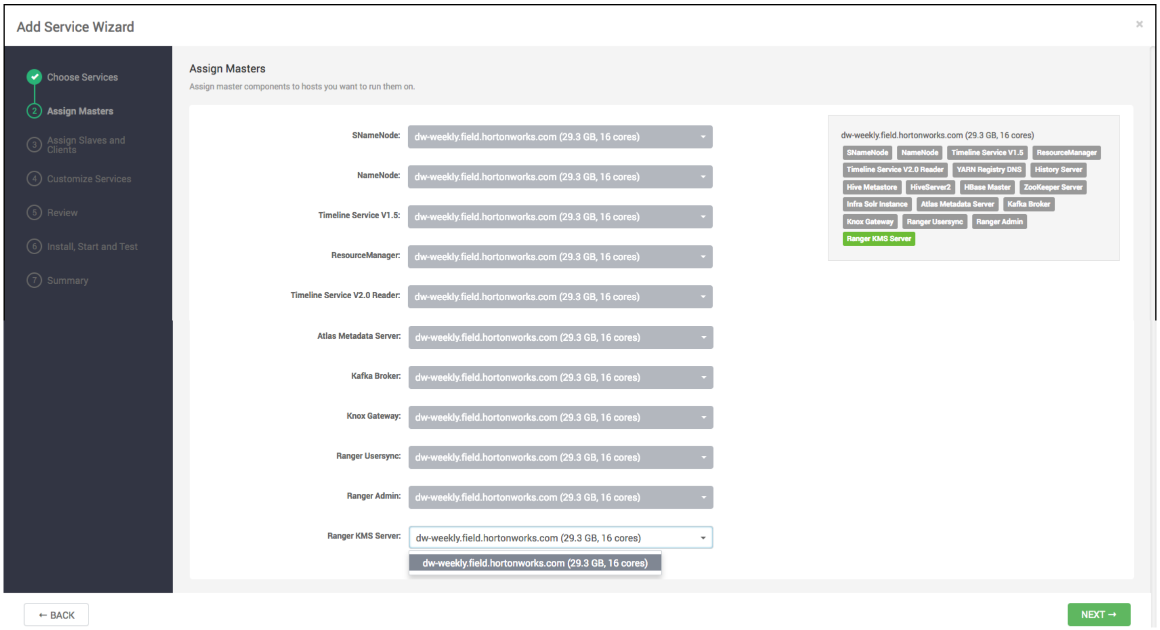
### Procedure

1. Go to the Ambari Web UI, http://<gateway-URL>:8080.
2. From the Ambari dashboard, go to the Actions menu. Choose Add Service.
3. On the next screen, check the box next to Ranger KMS:



| | | | |
|---|---|---|---|
| □ | Oozie | 4.3.1 | System for workflow coordination and execution of Apache Hadoop jobs. This also includes the installation of the optional Oozie Web Console which relies on and will install the ExtJS Library. |
| ☑ | ZooKeeper | 3.4.6 | Centralized service which provides highly reliable distributed coordination |
| □ | Storm | 1.2.1 | Apache Hadoop Stream processing framework |
| □ | Accumulo | 1.7.0 | Robust, scalable, high performance distributed key/value store. |
| ☑ | Infra Solr | 0.1.0 | Core shared service used by Ambari managed components. |
| □ | Ambari Metrics | 0.1.0 | A system for metrics collection that provides storage and retrieval capability for metrics collected from the cluster |
| ☑ | Atlas | 1.0.0 | Atlas Metadata and Governance platform |
| ☑ | Kafka | 1.0.1 | A high-throughput distributed messaging system |
| ☑ | Knox | 1.0.0 | Provides a single point of authentication and access for Apache Hadoop services in a cluster |
| □ | Log Search | 0.5.0 | Log aggregation, analysis, and visualization for Ambari managed services. This service is **Technical Preview**. |
| ☑ | Ranger | 1.1.0 | Comprehensive security for Hadoop |
| ☑ | Ranger KMS | 1.1.0 | Key Management Server |
| □ | SmartSense | 1.5.0.2.7.0.0-897 | SmartSense - Hortonworks SmartSense Tool (HST) helps quickly gather configuration, metrics, logs from common HDP services that aids to quickly troubleshoot support cases and receive cluster-specific recommendations. |
| □ | Spark2 | 2.3.1 | Apache Spark 2.3 is a fast and general engine for large-scale data processing. |
| □ | Zeppelin Notebook | 0.8.0 | A web-based notebook that enables interactive data analytics. It enables you to make beautiful data-driven, interactive and collaborative documents with SQL, Scala and more. |
| □ | Druid | 0.12.1 | A fast column-oriented distributed data store. |
| □ | Superset | 0.23.0 | Superset is a data exploration platform designed to be visual, intuitive and interactive. This service is **Technical Preview**. |

4. Then, choose Next.
5. (Optional) In Assign Masters, if you wish to override the default host setting, specify the Ranger KMS host address.



Assign Masters

Assign master components to hosts you want to run them on.

SNameNode: dw-weekly.field.hortonworks.com (29.3 GB, 16 cores)

NameNode: dw-weekly.field.hortonworks.com (29.3 GB, 16 cores)

Timeline Service V1.5: dw-weekly.field.hortonworks.com (29.3 GB, 16 cores)

ResourceManager: dw-weekly.field.hortonworks.com (29.3 GB, 16 cores)

Timeline Service V2.0 Reader: dw-weekly.field.hortonworks.com (29.3 GB, 16 cores)

Atlas Metadata Server: dw-weekly.field.hortonworks.com (29.3 GB, 16 cores)

Kafka Broker: dw-weekly.field.hortonworks.com (29.3 GB, 16 cores)

Knox Gateway: dw-weekly.field.hortonworks.com (29.3 GB, 16 cores)

Ranger Usersync: dw-weekly.field.hortonworks.com (29.3 GB, 16 cores)

Ranger Admin: dw-weekly.field.hortonworks.com (29.3 GB, 16 cores)

Ranger KMS Server: dw-weekly.field.hortonworks.com (29.3 GB, 16 cores)

dw-weekly.field.hortonworks.com (29.3 GB, 16 cores)

dw-weekly.field.hortonworks.com (29.3 GB, 16 cores)

SNameNode NameNode Timeline Service V1.5 ResourceManager Timeline Service V2.0 Reader YARN Registry DNS History Server Hive Metastore HiveServer2 HBase Master ZooKeeper Server Infra Solr Instance Atlas Metadata Server Kafka Broker Knox Gateway Ranger Usersync Ranger Admin Ranger KMS Server

**6.** In Customize Services, set required values (marked in red). Review other configuration settings, and determine whether you'd like to change any of the default values. (For more information about these properties, see “Ranger KMS Properties”.)

a) Provide the required settings, marked in red.

**Note:**

If do not wish to provide system Database Administrator (DBA) account details to the Ambari Ranger installer, you can use the dba_script.py Python script to create Ranger DB database users without exposing DBA account information to the Ambari Ranger installer. For more information, see “Set up Database Users Without Sharing DBA Credentials”.

b) Confirm if the following properties are present in Custom kms-site. If not, add values for the following properties in the "Custom kms-site" section. These properties allow the specified system users (hive, oozie, and others) to proxy on behalf of other users when communicating with Ranger KMS. This helps individual services (such as Hive) use their own keytabs, but retain the ability to access Ranger KMS as the end user (use access policies associated with the end user).

- hadoop.kms.proxyuser.hive.users
- hadoop.kms.proxyuser.oozie.users
- hadoop.kms.proxyuser.HTTP.users
- hadoop.kms.proxyuser.ambari.users
- hadoop.kms.proxyuser.yarn.users
- hadoop.kms.proxyuser.hive.hosts
- hadoop.kms.proxyuser.oozie.hosts
- hadoop.kms.proxyuser.HTTP.hosts
- hadoop.kms.proxyuser.ambari.hosts
- hadoop.kms.proxyuser.yarn.hosts

c) Add the following properties to the Custom KMS-site section of the configuration. These properties use the REPOSITORY_CONFIG_USERNAME specified in the first step in this section.

If you are using an account other than keyadmin to access Ranger KMS, replace “keyadmin” with the configured user for the Ranger KMS repository in Ranger admin:

- hadoop.kms.proxyuser.keyadmin.groups=*
- hadoop.kms.proxyuser.keyadmin.hosts=*
- hadoop.kms.proxyuser.keyadmin.users=*

d) Confirm settings of the following values in the "advanced kms-site" group:

- hadoop.kms.authentication.type=kerberos
- hadoop.kms.authentication.kerberos.keytab=/etc/security/keytabs/spnego.service.keytab
- hadoop.kms.authentication.kerberos.principal=*

7. Then, choose Next.
8. Review the default values on the Configure Identities screen. Determine whether you'd like to change any of the default values. Then, choose Next.
9. In Review, make sure the configuration values are correct. Ranger KMS will be listed under Services.
10. Then, choose Deploy.
11. Monitor the progress of installing, starting, and testing the service. When the service installs and starts successfully, choose Next.
12. The Summary screen displays the results. Choose Complete.
13. Restart the Ranger and Ranger KMS services.

# Set up Database Users Without Sharing DBA Credentials

How to create Ranger DB users using the dba_script.py script, without sharing DBA credentials.

### About this task

If do not wish to provide system Database Administrator (DBA) account details to the Ambari Ranger installer, you can use the dba_script.py Python script to create Ranger DB database users without exposing DBA account information to the Ambari Ranger installer. You can then run the normal Ambari Ranger installation without specify a DBA user name and password.

### Procedure

1. Download the Ranger rpm using the yum install command.yum install ranger-kms.
2. You should see one file named dba_script.py in the /usr/hdp/current/ranger-admin directory.
3. Get the script reviewed internally and verify that your DBA is authorized to run the script.
4. Execute the script by running the following command: python dba_script.py.
5. Pass all values required in the argument. These should include db flavor, JDBC jar, db host, db name, db user, and other parameters.
   - If you would prefer not to pass runtime arguments via the command prompt, you can update the /usr/hdp/current/ranger-admin/install.properties file and then run: python dba_script.py -q
   - When you specify the -q option, the script will read all required information from the install.properties file
   - You can use the -d option to run the script in "dry" mode. Running the script in dry mode causes the script to generate a database script: python dba_script.py -d /tmp/generated-script.sql

     Anyone can run the script, but it is recommended that the system DBA run the script in dry mode. In either case, the system DBA should review the generated script, but should only make minor adjustments to the script, for example, change the location of a particular database file. No major changes should be made that substantially alter the script -- otherwise the Ranger install may fail.
   - The system DBA must then run the generated script.
6. Log in to the host where KMS is to be installed. Run the following commands to back up files:

```
cp /var/lib/ambari-agent/cache/common-services/RANGER_KMS/0.5.0.2.3/
package/scripts/kms.py /var/lib/ambari-agent/cache/common-services/
RANGER_KMS/0.5.0.2.3/package/scripts/kms.py.bak
```

```
cp /var/lib/ambari-server/resources/common-services/RANGER_KMS/0.5.0.2.3/
package/scripts/kms.py /var/lib/ambari-server/resources/common-services/
RANGER_KMS/0.5.0.2.3/package/scripts/kms.py.bak
```

7. In both of the kms.py files copied in the previous step, find and comment out the following line (shown here commented out): #Execute(dba_setup, environment=env_dict, logoutput=True, user=params.kms_user).
8. Run the Ranger Ambari install procedure, but set Setup Database and Database User to No in the Ranger Admin section of the Customize Services screen.

# Configure HDFS Encryption to use Ranger KMS Access

If you plan to use Ranger KMS for HDFS data at rest encryption, complete the following steps.

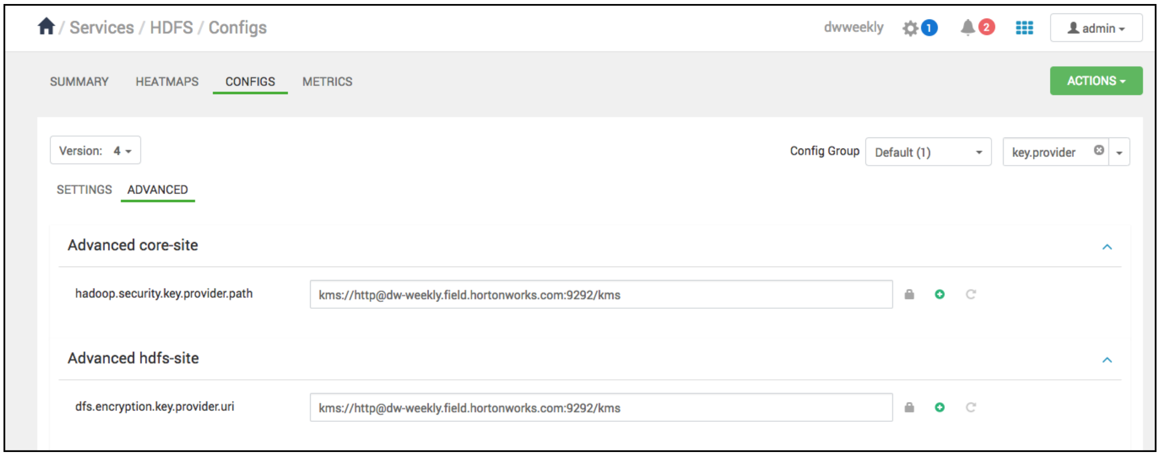
### Before you begin

At this point, Ranger KMS should already be installed and running.

### Procedure

1. Create a link to /etc/hadoop/conf/core-site.xml under /etc/ranger/kms/conf: sudo ln -s /etc/hadoop/conf/core-site.xml /etc/ranger/kms/conf/core-site.xml.
2. Configure HDFS to access Ranger KMS.
   a) In the left panel of the Ambari main menu, choose HDFS.
   b) Choose the Configs tab at the top of the page, and then choose the Advanced tab partway down the page.
   c) Specify the provider path (the URL where the Ranger KMS server is running) in the following two properties, if the path is not already specified:

      - In "Advanced core-site", specify hadoop.security.key.provider.path
      - In "Advanced hdfs-site", specify dfs.encryption.key.provider.uri



Version: 4

Config Group Default (1)

key.provider

Advanced core-site

hadoop.security.key.provider.path kms://http@dw-weekly.field.hortonworks.com:9292/kms

Advanced hdfs-site

dfs.encryption.key.provider.uri kms://http@dw-weekly.field.hortonworks.com:9292/kms

The Ranger KMS host is where Ranger KMS is installed. The Ranger KMS host name should have the following format:

kms://http@<kmshost>:9292/kms

3. Under Custom core-site.xml, set the value of the hadoop.proxyuser.kms.groups property to * or service user.
4. Restart the Ranger KMS service and the HDFS service.

# Use a Kerberos Principal for the Ranger KMS Repository

To manage access policies for Ranger KMS, a repository is needed with Ranger for the Ranger KMS service. Ambari creates the repository automatically using the repository config user and password provided. The repository config user also needs to be created as a principal in Kerberos with a password. Use the following steps to use a Kerberos principal for the Ranger KMS repository.

### About this task

In Ranger, all access policies are configured within a repository for each service..

### Procedure

1. Create system user keyadmin which should be sync in User Tabs in Ranger Admin.
2. Create principal keyadmin@EXAMPLE.COM with password keyadmin: kadmin.local -q 'addprinc -pw keyadmin keyadmin'.
3. On the Add Service wizard Customize Services page, set the required values (marked in red).
4. Under ranger-kms-properties, set the principal and password in the REPOSITORY_CONFIG_USERNAME and REPOSITORY_CONFIG_PASSWORD fields.
5. To check logs, select Audit to DB under Advanced ranger-kms-audit.
6. Click Next to continue with the Ranger KMS Add Service wizard.