Security 3

Enabling Kerberos

Date of Publish: 2018-11-15

https://docs.hortonworks.com/

Contents

Enabling Kerberos 3
Installing and Configuring the KDC 3
Use an Existing MIT KDC 3
Use an Existing Active Directory 3
Use Manual Kerberos Setup 4
(Optional) Install a new MIT KDC 4
Installing the JCE 6
Install the JCE 6
Enabling Kerberos on Ambari 7
Cluster Component Configuration Updates 8

Enabling Kerberos

To enable Kerberos on Ambari, complete the following steps:

Installing and Configuring the KDC

Ambari is able to configure Kerberos in the cluster to work with an existing MIT KDC, or existing Active Directory installation. This section describes the steps necessary to prepare for this integration.

Note:

If you do not have an existing KDC (MIT or Active Directory), Install a new MIT KDC. Installing a KDC on a cluster host after installing the Kerberos client may overwrite the krb5.conf file generated by Ambari.

You can choose to have Ambari connect to the KDC and automatically create the necessary Service and Ambari principals, generate and distribute the keytabs (“Automated Kerberos Setup”). Ambari also provides an advanced option to manually configure Kerberos. If you choose this option, you must create the principals, generate and distribute the keytabs. Ambari will not do this automatically (“Manual Kerberos Setup”).

For convenience, use the instructions to (Optional) Install a new MIT KDC if you do not have an existing KDC available.

Use an Existing MIT KDC

To use an existing MIT KDC for the cluster, you must prepare the following:

- Ambari Server and cluster hosts have network access to both the KDC and KDC admin hosts.
- KDC administrative credentials are on-hand.

Note:

You will be prompted to enter the KDC Admin Account credentials during the Kerberos setup so that Ambari can contact the KDC and perform the necessary principal and keytab generation. By default, Ambari will not retain the KDC credentials unless you have configured Ambari for encrypted passwords.

Use an Existing Active Directory

To use an existing Active Directory domain for the cluster with Automated Kerberos Setup, you must prepare the following:

- Ambari Server and cluster hosts have network access to, and be able to resolve the DNS names of, the Domain Controllers.
- Active Directory secure LDAP (LDAPS) connectivity has been configured.
- Active Directory User container for principals has been created and is on-hand. For example, "OU=Hadoop,OU=People,dc=apache,dc=org"
- Active Directory administrative credentials with delegated control of “Create, delete, and manage user accounts” on the previously mentioned User container are on-hand.

Note:

You will be prompted to enter the KDC Admin Account credentials during the Kerberos setup so that Ambari can contact the KDC and perform the necessary principal and keytab generation. By default, Ambari will not retain the KDC credentials unless you have configured Ambari for encrypted passwords.

Note:

If Centrify is installed and being used on any of the servers in the cluster, it is critical that you refer to Centrify's integration guide before attempting to enable Kerberos Security on your cluster. The

documentation can be found in the Centrify Server Suite documentation library, with a direct link to the Hortonworks specific PDF here.

Use Manual Kerberos Setup

To perform Manual Kerberos Setup, you must prepare the following:

- Cluster hosts have network access to the KDC.
- Kerberos client utilities (such as kinit) have been installed on every cluster host.
- The Java Cryptography Extensions (JCE) have been setup on the Ambari Server host and all hosts in the cluster.
- The Service and Ambari Principals will be manually created in the KDC before completing this wizard.
- The keytabs for the Service and Ambari Principals will be manually created and distributed to cluster hosts before completing this wizard.

(Optional) Install a new MIT KDC

The following gives a very high level description of the KDC installation process.

Note:

Because Kerberos is a time-sensitive protocol, all hosts in the realm must be time-synchronized, for example, by using the Network Time Protocol (NTP). If the local system time of a client differs from that of the KDC by as little as 5 minutes (the default), the client will not be able to authenticate.

Install the KDC Server

1. Install a new version of the KDC server:

   RHEL/CentOS/Oracle Linux

   yum install krb5-server krb5-libs krb5-workstation

   SLES

   zypper install krb5 krb5-server krb5-client

   Ubuntu/Debian

   apt-get install krb5-kdc krb5-admin-server
2. Using a text editor, open the KDC server configuration file, located by default here:

   vi /etc/krb5.conf
3. Change the [realms] section of this file by replacing the default “kerberos.example.com” setting for the kdc and admin_server properties with the Fully Qualified Domain Name of the KDC server host. In the following example, “kerberos.example.com” has been replaced with “my.kdc.server”.

   ```
   [realms]
    EXAMPLE.COM = {
      kdc = my.kdc.server
      admin_server = my.kdc.server
   }
   ```
4. Some components such as HUE require renewable tickets. To configure MIT KDC to support them, ensure the following settings are specified in the libdefaults section of the /etc/krb5.conf file.

   renew_lifetime = 7d

Note:

For Ubuntu/Debian, the setup of the default realm for the KDC and KDC Admin hostnames is performed during the KDC server install. You can re-run setup using dpkg-reconfigure krb5-kdc. Therefore, Steps 2 and 3 above are not needed for Ubuntu/Debian.

Create the Kerberos Database

- Use the utility kdb5_util to create the Kerberos database.

RHEL/CentOS/Oracle Linux

kdb5_util create -s

SLES

kdb5_util create -s

Ubuntu/Debian

krb5_newrealm

Start the KDC

- Start the KDC server and the KDC admin server.

  RHEL/CentOS/Oracle Linux 6

  /etc/rc.d/init.d/krb5kdc start

  /etc/rc.d/init.d/kadmin start

  RHEL/CentOS/Oracle Linux 7

  systemctl start krb5kdc

  systemctl start kadmin

  SLES

  rckrb5kdc start

  rckadmind start

  Ubuntu/Debian

  service krb5-kdc restart

  service krb5-admin-server restart

  Important:

  When installing and managing your own MIT KDC, it is very important to set up the KDC server to auto-start on boot . For example:

  RHEL/CentOS/Oracle Linux 6

  chkconfig krb5kdc on

  chkconfig kadmin on

  RHEL/CentOS/Oracle Linux 7

  systemctl enable krb5kdc

  systemctl enable kadmin

  SLES

  chkconfig rckrb5kdc on

  chkconfig rckadmind on

Create a Kerberos Admin

Kerberos principals can be created either on the KDC machine itself or through the network, using an “admin” principal. The following instructions assume you are using the KDC machine and using the kadmin.local command line administration utility. Using kadmin.local on the KDC machine allows you to create principals without needing to create a separate "admin" principal before you start.

Note:

You will need to provide these admin account credentials to Ambari when enabling Kerberos. This allows Ambari to connect to the KDC, create the cluster principals and generate the keytabs.

1. Create a KDC admin by creating an admin principal.

   kadmin.local -q "addprinc admin/admin"
2. Confirm that this admin principal has permissions in the KDC ACL. Using a text editor, open the KDC ACL file:

   RHEL/CentOS/Oracle Linux

   vi /var/kerberos/krb5kdc/kadm5.acl

   SLES

   vi /var/lib/kerberos/krb5kdc/kadm5.acl

   Ubuntu/Debian

   vi /etc/krb5kdc/kadm5.acl
3. Ensure that the KDC ACL file includes an entry so to allow the admin principal to administer the KDC for your specific realm. When using a realm that is different than EXAMPLE.COM, be sure there is an entry for the realm you are using. If not present, principal creation will fail. For example, for an admin/admin@HADOOP.COM principal, you should have an entry:

   */admin@HADOOP.COM *
4. After editing and saving the kadm5.acl file, you must restart the kadmin process.

   RHEL/CentOS/Oracle Linux 6

   /etc/rc.d/init.d/kadmin restart

   RHEL/CentOS/Oracle Linux 7

   systemctl restart kadmin

   SLES

   rckadmind restart

   Ubuntu/Debian

   service krb5-admin-server restart

Installing the JCE

Before enabling Kerberos in the cluster, you must deploy the Java Cryptography Extension (JCE) security policy files on the Ambari Server and on all hosts in the cluster.

Important:

If you are using Oracle JDK, you must distribute and install the JCE on all hosts in the cluster, including the Ambari Server. Be sure to restart Ambari Server after installing the JCE. If you are using OpenJDK, some distributions of the OpenJDK come with unlimited strength JCE automatically and therefore, installation of JCE is not required.

Install the JCE

1. On the Ambari Server, obtain the JCE policy file appropriate for the JDK version in your cluster.
   - For Oracle JDK 1.8:

     http://www.oracle.com/technetwork/java/javase/downloads/jce8-download-2133166.html
   - For Oracle JDK 1.7:

     http://www.oracle.com/technetwork/java/javase/downloads/jce-7-download-432124.html
2. Save the policy file archive in a temporary location.

3. On Ambari Server and on each host in the cluster, add the unlimited security policy JCE jars to $JAVA_HOME/jre/lib/security/.

   For example, run the following to extract the policy jars into the JDK installed on your host:

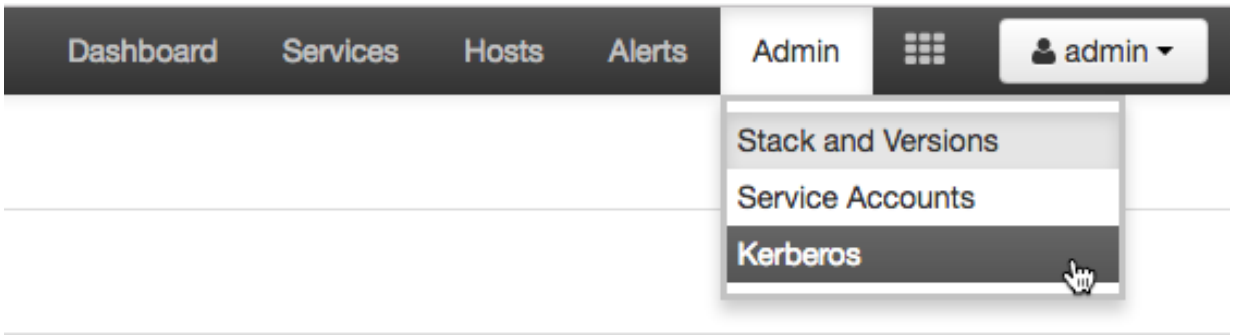
   ```
   unzip -o -j -q jce_policy-8.zip -d /usr/jdk64/jdk1.8.0_60/jre/lib/
   security/
   ```

4. Restart Ambari Server.

Enabling Kerberos on Ambari

Once you have completed the prerequisites, you are ready to enable Kerberos for Ambari.

1. From the Ambari UI, click **Admin**, and select **Kerberos**.



2. Click **Enable Kerberos** to launch the **Enable Kerberos Wizard**.
3. From the **Get Started** screen, select the type of KDC you want to use.
4. Provide information about the KDC and admin account.

   a. In the KDC section, enter the following information:

   - In the KDC Host field, the IP address or FQDN for the KDC host. Optionally a port number may be included.
   - In the Realm name field, the default realm to use when creating service principals.
   - (Optional) In the Domains field, provide a list of patterns to use to map hosts in the cluster to the appropriate realm. For example, if your hosts have a common domain in their FQDN such as host1.hortonworks.local and host2.hortonworks.local, you would set this to:

     .hortonworks.local,hortonworks.local

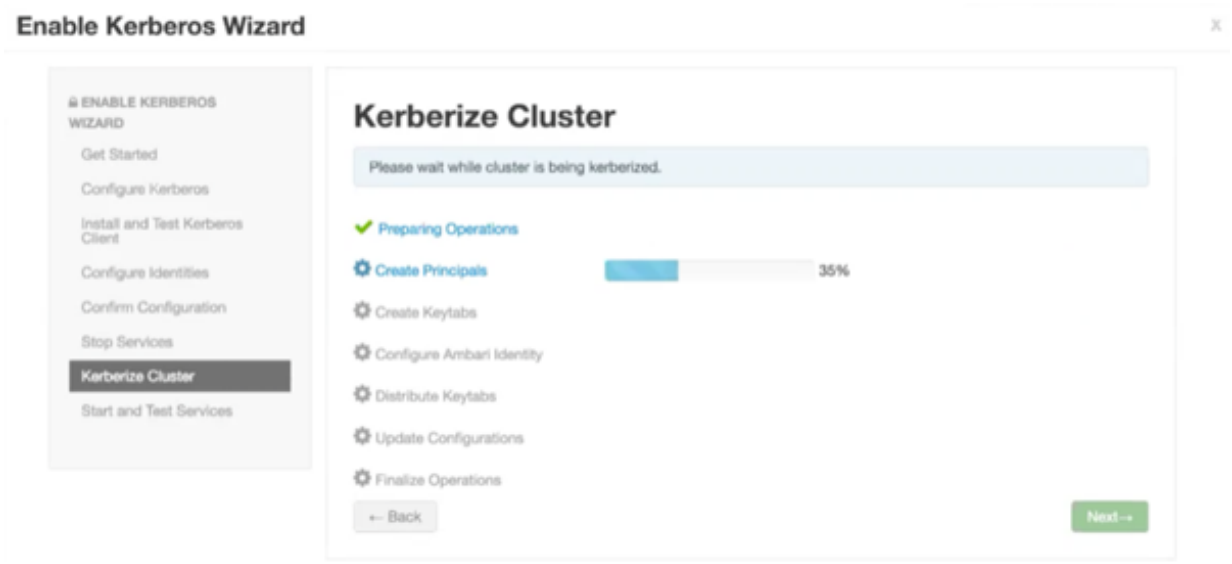
   b. In the Kadmin section, enter the following information:

   - In the Kadmin Host field, the IP address or FQDN for the KDC administrative host. Optionally a port number may be included.
   - The Admin principal and password that will be used to create principals and keytabs.
   - (Optional) If you have configured Ambari for encrypted passwords, the Save Admin Credentials option will be enabled. With this option, you can have Ambari store the KDC Admin credentials to use when making cluster changes.

5. From the **Install and Test Kerberos Client** page, proceed with the install. Click **Next** when complete.
6. From the **Configure Identities** page, you can customize the Kerberos identities as needed, and proceed to kerberize the cluster.

   Be sure to review the principal names, particularly the Ambari Principals on the General tab. These principal names, by default, append the name of the cluster to each of the Ambari principals. You can leave this as default or adjust these by removing the "-${cluster-name}" from principal name string.

   Click the **Advanced** tab to review the principals and keytabs for each service.
7. Confirm your configurations, and click next to proceed kerberizing your cluster.



Cluster Component Configuration Updates

After you have enabled Kerberos, some cluster components require additional configuration updates.

SAM Configuration Changes

Steps

1. From the Ambari UI, select **Admin | Kerberos**.
2. From **Ambari Principals**, set the SAM principal name:

```
streamline_principal_name : memo the principal without @${realm} – (1)
```

3. From **Ambari Infra | Configs | Advanced | Advanced infra-solr-security-json**, add the following to the bottom text box. Replace <<streamline>> (bolded) to streamline principal (1) before pasting.

```
{
  "authentication": {
    "class": "org.apache.solr.security.KerberosPlugin"
  },
  "authorization": {
    "class":
 "org.apache.ambari.infra.security.InfraRuleBasedAuthorizationPlugin",
    "user-role": {
      "{{infra_solr_kerberos_service_user}}@{{kerberos_realm}}": "admin",
      "{{logsearch_kerberos_service_user}}@{{kerberos_realm}}":
 ["{{infra_solr_role_logsearch}}", "{{infra_solr_role_ranger_admin}}",
 "{{infra_solr_role_dev}}"],
      "<<streamline>>@{{kerberos_realm}}":
 ["{{infra_solr_role_logsearch}}", "{{infra_solr_role_ranger_admin}}",
 "{{infra_solr_role_dev}}"],
      "{{logfeeder_kerberos_service_user}}@{{kerberos_realm}}":
 ["{{infra_solr_role_logfeeder}}", "{{infra_solr_role_dev}}"],
      "{{atlas_kerberos_service_user}}@{{kerberos_realm}}":
 ["{{infra_solr_role_atlas}}", "{{infra_solr_role_ranger_audit}}",
 "{{infra_solr_role_dev}}"],
{% if infra_solr_ranger_audit_service_users %}
{%   for ranger_audit_service_user in
 infra_solr_ranger_audit_service_users %}
```

```
      "{{ranger_audit_service_user}}@{{kerberos_realm}}":
 ["{{infra_solr_role_ranger_audit}}", "{{infra_solr_role_dev}}"],
{%   endfor %}
{% endif %}
      "{{ranger_admin_kerberos_service_user}}@{{kerberos_realm}}":
 ["{{infra_solr_role_ranger_admin}}", "{{infra_solr_role_ranger_audit}}",
 "{{infra_solr_role_dev}}"]
    },
    "permissions": [
      {
        "name" : "collection-admin-read",
        "role" :null
      },
      {
        "name" : "collection-admin-edit",
        "role" : ["admin", "{{infra_solr_role_logsearch}}",
 "{{infra_solr_role_logfeeder}}", "{{infra_solr_role_atlas}}",
 "{{infra_solr_role_ranger_admin}}"]
      },
      {
        "name":"read",
        "role": "{{infra_solr_role_dev}}"
      },
      {
        "collection": ["{{logsearch_service_logs_collection}}",
 "{{logsearch_audit_logs_collection}}", "history"],
        "role": ["admin", "{{infra_solr_role_logsearch}}",
 "{{infra_solr_role_logfeeder}}"],
        "name": "logsearch-manager",
        "path": "/*"
      },
      {
        "collection": ["vertex_index", "edge_index", "fulltext_index"],
        "role": ["admin", "{{infra_solr_role_atlas}}"],
        "name": "atlas-manager",
        "path": "/*"
      },
      {
        "collection": "{{ranger_solr_collection_name}}",
        "role": ["admin", "{{infra_solr_role_ranger_admin}}",
 "{{infra_solr_role_ranger_audit}}"],
        "name": "ranger-manager",
        "path": "/*"
      }
    ]
  }
}
```

4. Restart Ambari Infra.

Druid Configuration Changes

Update the Druid property druid.hadoop.security.spnego.excludedPaths to the following value:

```
["/status", "/druid/worker/v1", "/druid/indexer/v1"]
```

HDFS Configuration Changes

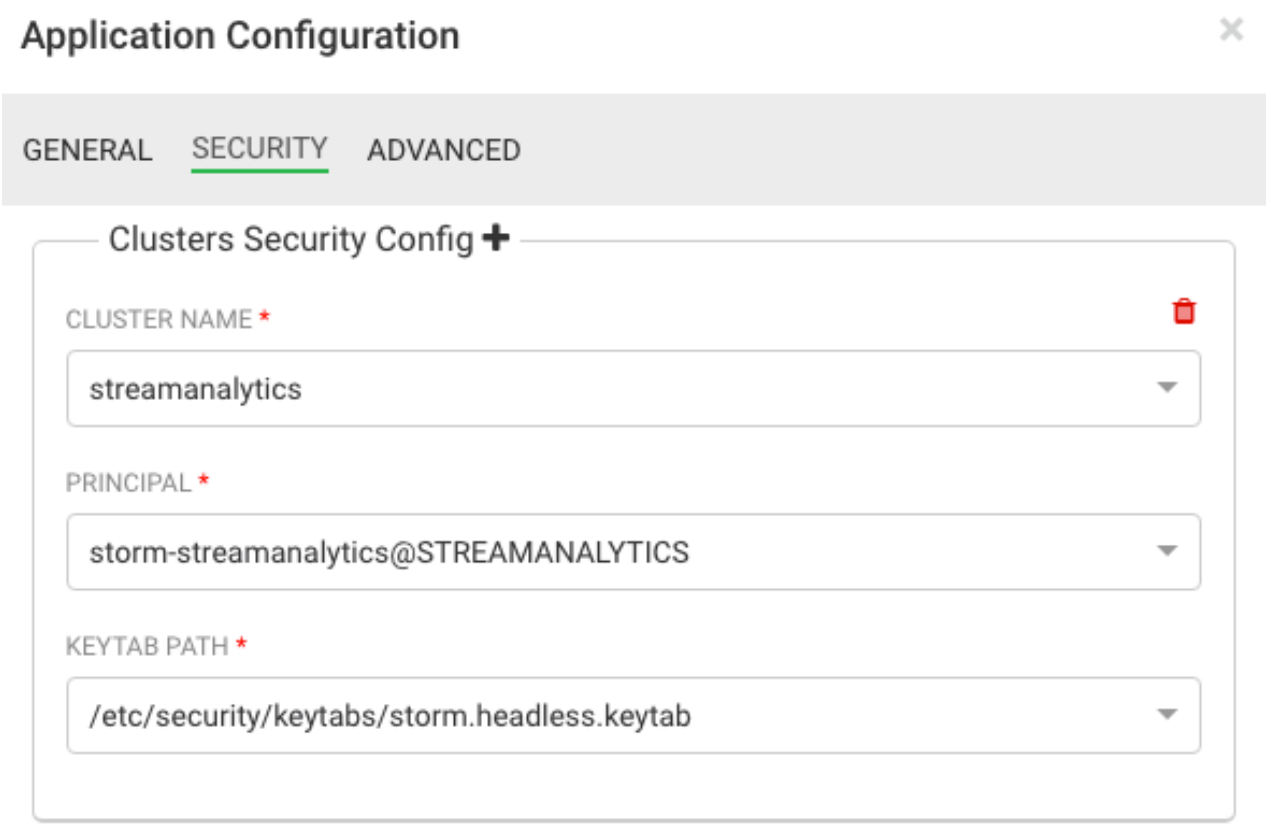
If you are going to use the HDFS processor in your application in secure mode, add the following properties in the HDFS service under custom core-site.xml.

Property Name	Value
hadoop.proxyuser.$principal_you_configured_in_sam_app_settings.groups	*

Property Name	Value
hadoop.proxyuser.$principal_you_configured_in_sam_app_settings.hosts	*

Example

In SAM, you have configured the following principal and keytab under Application Settings:



The configuration for the 2 HDF properties is:

```
hadoop.proxyuser.storm-streamanalytics.hosts=*
hadoop.proxyuser.storm-streamanalytics.groups=*
```

HBase Configuration

In the HBase service, under custom hbase-site.xml add the following properties

- hbase.thrift.support.proxyuser=true
- hbase.regionserver.thrift.http=true

In HDFS service, add the following under custom core-site.xm

- hadoop.proxyuser.streamline-streamanalytics.hosts=*
- hadoop.proxyuser.streamline-streamanalytics.groups=*