Open for Innovation
KNIME

Secured Cluster Connection Guide for KNIME Server

KNIME AG, Zurich, Switzerland
Version 4.2 (last updated on 2020-05-04)

Table of Contents

Overview

KNIME Server executes workflows, that may try to access Kerberos-secured services such as Apache Hive™, Apache Impala™ and Apache Hadoop® HDFS™.

This guide describes how to configure KNIME Server so that it can **authenticate** itself against Kerberos and then **impersonate** its own users towards Kerberos-secured cluster services.

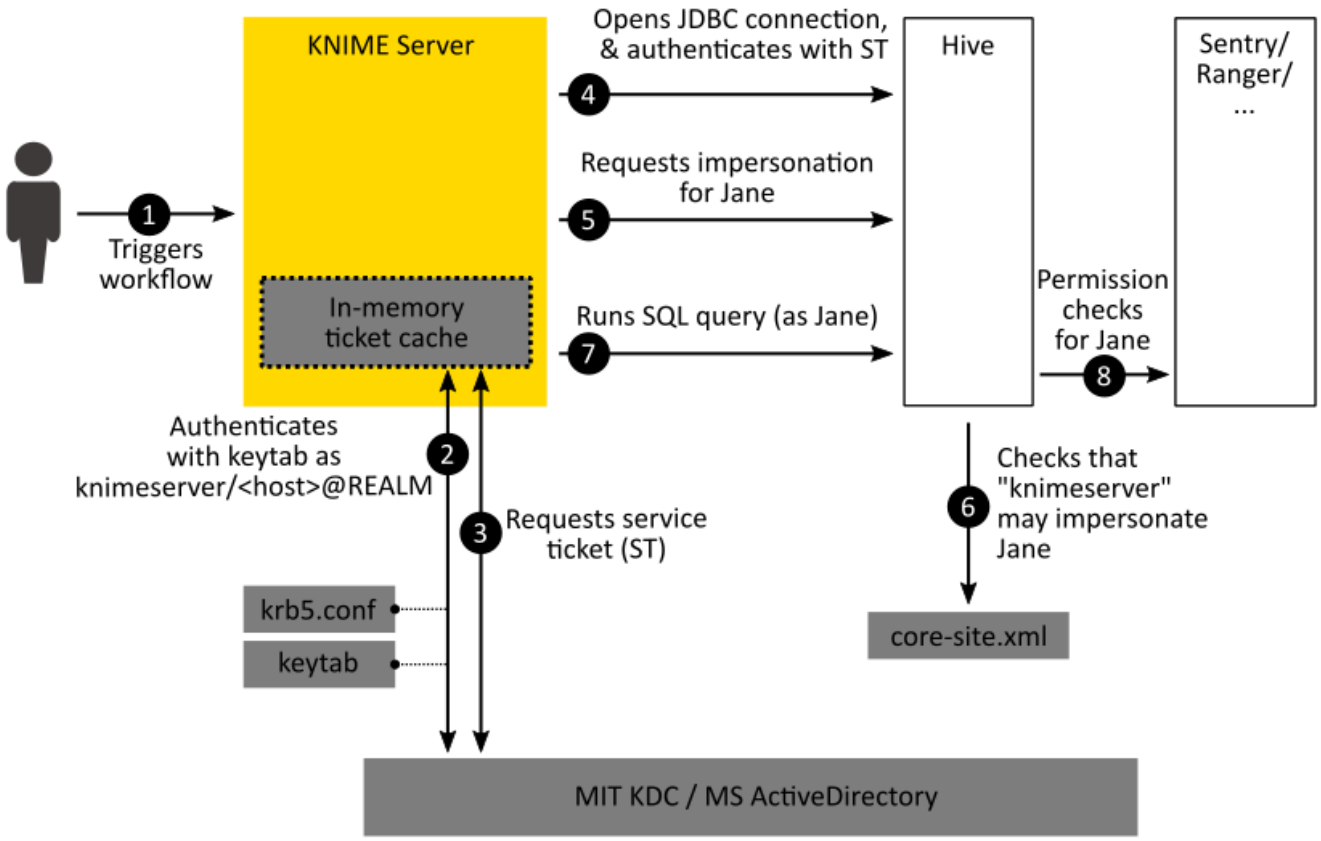
What is user impersonation?

With user impersonation, it does not matter whether a user runs a workflow in KNIME Analytics Platform or on KNIME Server. In both cases, all operations on the cluster will be performed **as that particular user** and the **same permissions and authorization rules** apply. This has the following advantages:

- Workflows that access a secured cluster run without modifications on KNIME Server.
- Authorization to access cluster resources (Hive tables, HDFS files, …) is administered with the usual mechanisms, e.g. Apache Sentry™ or Apache Ranger™.

How does user impersonation work?

Let us assume that a user Jane runs a workflow on KNIME Server. The workflow is supposed to run a Hive query.



The following sequence of events now takes place:

1. She starts a workflow that connects to Hive. This workflow is now executed on KNIME Server, not Jane's machine.
2. When the *Hive Connector* node in the workflow is executed, KNIME Server first checks for a TGT (ticket granting ticket) in its own ticket cache. If there is no TGT, it reads the `krb5.conf` configuration file, connects to the KDC and authenticates itself. Instead of Jane's credentials, it uses the credentials configured on KNIME Server, i.e. a service principal such as `knimeserver/<host>@REALM` and a keytab file. The TGT will be stored in an in-memory ticket cache.
3. To make a JDBC connection to Hive, the Hive JDBC driver on KNIME Server still requires an ST (service ticket), which it now requests from the KDC. The ST is only valid for connections between KNIME Server and the Hive instance.
4. Now, the Hive JDBC driver opens a connection to Hive and authenticates itself with the ST as `knimeserver/<host>@REALM`.
5. Since the workflow was started by Jane, the JDBC driver tells Hive, that all operations shall be performed **as user Jane**.
6. Hive consults the Hadoop `core-site.xml` to verify that KNIME Server is indeed allowed to impersonate Jane. If not, it will return an error.
7. Now, the workflow submits an SQL query via the JDBC connection. The query is

executed on the cluster **as user Jane**.

8. Hive checks whether user Jane has the necessary permissions to run the query. It employs its usual permission checking mechanism, e.g. Apache Sentry™ or Apache Ranger™. The query will succeed or fail, depending on whether **Jane** has the necessary permissions.

Prerequisites

Setting up KNIME Server for Kerberos authentication and user impersonation has the following prerequisites.

- For Kerberos:
  - An existing Kerberos KDC such as MIT Kerberos or Microsoft ActiveDirectory
  - A service principal for KNIME Server. The recommended format is `knimeserver/<host>@<REALM>`, where
    - `<host>` is the fully-qualified domain name of the machine where KNIME Server runs,
    - `<REALM>` is the Kerberos realm.
  - A keytab file for the KNIME Server service principal.
  - A Kerberos client configuration file (`krb5.conf`). The recommended way to obtain this file, is to copy the `/etc/krb5.conf` from a node in the cluster. Alternatively, the file can be created manually (see Setting up `krb5.conf`).
- For the cluster:
  - A Kerberos-secured cluster.
  - An account with administrative privileges in the cluster management software, that can configure and restart cluster services. On Cloudera CDH this means a Cloudera Manager account.
- For KNIME Server:
  - An existing KNIME Server installation.
  - An account with administrative privileges on the machine where KNIME Server is installed. This accounts needs to be able to edit the KNIME Server configuration files and restart KNIME Server.

Supported cluster services

KNIME Server supports Kerberos authentication and user impersonation for connections to the following services:

- Apache Hive
- Apache Impala
- Apache Hadoop HDFS (including HttpFS)
- Apache Livy

Setting up Kerberos authentication

This section describes how to set up KNIME Server to authenticate itself against Kerberos.

Kerberos client configuration (krb5.conf)

The KNIME Server executor needs to read the `krb5.conf` file during Kerberos authentication. A valid `krb5.conf` file is needed. The KNIME Server executor will check several locations for the `krb5.conf` file. The section Possible locations for `krb5.conf` describes the process by which KNIME Server executor locates the `krb5.conf` file. In case the location of the file is unknown or the file is not available, please contact the local administrator.

As an alternative, the `krb5.conf` file can be created manually. Please consult the section Setting up `krb5.conf` of the Kerberos Admin Guide for more information and an example on how to create a simple `krbf5.conf` file.

For Hadoop, if the user is in the same Kerberos realm as the Hadoop cluster, then the `/etc/krb5.conf` file can be downloaded directly from a cluster node, e.g. using WinSCP or pscp from PuTTY.

Kerberos customization profiles

Kerberos configurations, such as the `krb5.conf` location, service principal, keytab values, and many others, are stored in a preferences file (`.epf` file), which can be distributed to all connected KNIME server executors via customatization profiles.

Please consult the Kerberos configuration table for a list of all supported Kerberos configuration options and how to write them inside a preferences file. For an in-depth guide

to create a customization profile to distribute Kerberos preferences to all KNIME Server executors, please check the Kerberos Admin Guide.

For troubleshooting Kerberos, please check the Troubleshooting section of the Kerberos Admin Guide.

Setting up proprietary JDBC drivers (optional)

Each KNIME Server Executor is a headless instance of KNIME Analytics Platform. If the *KNIME Big Data Connectors* extension is installed, KNIME Server Executor includes a fully-functional embedded JDBC driver for Hive and Impala. If the use of this driver is preferred, then this section can be skipped.

The **embedded** Apache Hive JDBC Driver for Impala **does not support impersonation**. For impersonation, when connecting to Impala, please setup a proprietary driver.

The currently embedded JDBC driver is the open-source Apache Hive™ JDBC driver version `1.1.0-cdh5.13.0` (Release Notes). The driver has been verified to be compatible with CDH 5.3 and later.

If the set up of a proprietary Cloudera JDBC driver for Hive/Impala **(recommended)** is chosen, please consult the following sections for a step-by-step JDBC driver registration guide depending on the specific Hadoop vendor:

- Register Hive Cloudera JDBC driver on KNIME Server
- Register Impala Cloudera JDBC driver on KNIME Server

The JDBC driver registration process described above also creates a customization profile to distribute the driver to all connected KNIME Server executors. If a profile folder is already existing during the Setting up Kerberos authentication step, then the creation of a new one is not necessary in this step.

Setting up user impersonation

This section describes how to set up both ends of user impersonation, which requires configuration on two sides: KNIME Server **and** the cluster.

User impersonation on KNIME Server

By default, KNIME Server tries to impersonate its users on Kerberos-secured connections towards the following cluster services:

- HDFS (including httpFS)
- Apache Livy
- Apache Hive

Impersonation for HDFS and Apache Livy is done automatically and does not require any further setup. Connections to Apache Hive require further setup steps depending on the used JDBC driver.

The **embedded** Apache Hive JDBC Driver (for Impala) **does not support impersonation**. For impersonation when connecting to Impala please setup the proprietary driver.

If the instructions about JDBC driver registration contained in KNIME Server guide for Hive or Impala in the previous section, the user impersonation activation is already included. Please go to the next section.

For **embedded** Apache Hive JDBC driver, please follow the instruction in the User impersonation on Hive section.

Check the driver documentation for the appropriate impersonation parameter if any third party JDBC driver is in use.

User impersonation on Apache Hadoop™ and Apache Hive™

Apache Hadoop™ and Apache Hive™ consult the `core-site.xml` file to determine whether KNIME Server is allowed to impersonate users.

Changing the `core-site.xml` file must be done via Ambari (on HDP) or Cloudera Manager (on CDH). A restart of the affected Hadoop services is required.

Please add the following settings to the Hadoop `core-site.xml` on the cluster:

```
<property>
    <name>hadoop.proxyuser.knimeserver.hosts</name>①
    <value>*</value>
</property>
<property>
    <name>hadoop.proxyuser.knimeserver.groups</name>①
    <value>*</value>
</property>
```

① If a service principal was created for KNIME Server other than `knimeserver/<host>@<REALM>`, then the property name needs to be adjusted accordingly.

User impersonation on Apache Impala™

Apache Impala™ requires a configuration setting to determine whether KNIME Server is allowed to impersonate users.

It is recommended to also enable Apache Sentry™ authorization in Apache Impala™. Otherwise Impala perform all read and write operations with the privileges of the `impala` user.

The required steps are similar to Configuring Impala Delegation for Hue. In Cloudera Manager, navigate to **Impala** > **Configuration** > **Impala Daemon Command Line Argument Advanced Configuration Snippet (Safety Valve)** and add the following line:

```
-authorized_proxy_user_config='hue=*;knimeserver=*'
```

Then click **Save** and restart all Impala daemons.

Please note:

- This will make `hue` and `knimeserver` the only services that can impersonate users in Impala. If other services should be allowed to do the same, they need to be included here as well.
- If a service principal for KNIME Server other than `knimeserver/<host>@<REALM>` was created, then adjust the above setting accordingly.

KNIME AG
Talacker 50
8001 Zurich, Switzerland
www.knime.com
info@knime.com