Enable SSO for Relational Connections

Contents:

This section describes how to enable relational connections to leverage your Hadoop Single Sign-On (SSO) infrastructure.When this feature is enabled and properly configured, users can create relational (JDBC) connections that use SSO that you have already configured.

Connections that were created before this feature is enabled continue to operate as expected without modification.

Limitations

- For this release, this feature applies to SQL Server connections only.
- Cross-realm is not supported. As a result, the SQL Server instance, service principal, and Trifacta principal must be in the same Kerberos realm.

Pre-requisites

- **Kerberos SSO:** You must set up SSO authentication to the Hadoop cluster using Kerberos. This feature uses the global Kerberos keytab. For more information, see *Set up for a Kerberos-enabled Hadoop cluster*.

Enable

You can apply this change through the *Admin Settings Page* (recommended) or

`trifacta-conf.json`

. For more information, see *Platform Configuration Methods*.

Parameter	Description
`webapp.connectivity.enableSsoKerberosDelegate`	Set this flag to `true` to enable Kerberos-based SSO connections to be created for supported connection types.

Configure

Configure JAAS file and path

You can apply this change through the *Admin Settings Page* (recommended) or

`trifacta-conf.json`

. For more information, see *Platform Configuration Methods*.

Parameter	Description
`webapp.connectivity.kerberosDelegateConfigPath`	Path on the Trifacta node to the location of the JAAS configuration file required by the DataDirect driver. **NOTE:** The default location is listed below. You may wish to move this file to a location outside of the Trifacta installation to ensure that the file is not overwritten during upgrades. More information on this file is provided below.

JAAS file

For connections that support Kerberos-delegated authentication, the underlying driver supports a JAAS file in which you can provide environment-specific configuration to the driver. As needed, you can modify this file.

Connection Type	Default path to JAAS file
SQL Server	`%(topOfTree)s/services/data-service/build/conf/kerberosdelegate.config`

Example JAAS file for SQL Server

Below is an example file, where you must apply the Kerberos global keytab and principal values that are to be used to authenticate to use the Kerberos-delegated connections of this type:

```
trifacta_jaas_config {
  com.sun.security.auth.module.Krb5LoginModule required
  useKeyTab=true
  storeKey=true
  doNotPrompt=true
  keyTab="</absolute/path/to/trifacta_jdbc_sso.keytab>"
  principal="<principal_name>";
};

JDBC_DRIVER_01 {
  com.sun.security.auth.module.Krb5LoginModule required debug=false
  useTicketCache=true;
};
```

where:

- `keytab` = the absolute path on the Trifacta node where the Kerberos global keytab is located.
- `principal` = Set to the service principal name of the user's service account in LDAP.

Specify Kerberos configuration file

On the Trifacta node, locate the following file:

```
<root>/etc/krb5.conf
```

If it doesn't exist, create it with the following content, some of which you must specify:

```
[libdefaults]
    default_realm = <my_default_realm>
    forwardable = true # Important that this is set!

[realms]
    <my_default_realm> = {
        kdc = <kdc_domain>
    }

[domain_realm]
    <my_domain> = <my_default_realm>
```

Setting	Description
`default_realm`	Set this value to your default Kerberos realm.
`forwardable`	This value must be set to `true`.
`kdc`	For each realm that you create, you must create an entry in `[realms]`. For the `kdc` entry, apply the KDC domain that the JDBC connection should use.
`my_domain`	For each domain to which the Kerberos delegation applies, you must create an entry in `[domain_realm]`. Entries should look like the following: `trifacta.com = TRIFACTA.COM`

Modify the location of the Kerberos configuration file

If you need to move the location of the file from the default one, please complete the following:

Steps:

1. If you haven't already done so, copy the file from its current location to its preferred location.
2. You can apply this change through the *Admin Settings Page* (recommended) or `trifacta-conf.json`. For more information, see *Platform Configuration Methods*.
3. Specify the path to the new location in the following parameter:

   ```
   "webapp.connectivity.krb5Path": "/etc/krb5.conf";
   ```

4. Save your changes.

Example Setup

The following example uses the default Kerberos realm to set an SSO connection to a SQL Server instance. This example is intended to demonstrate one way in which you can set up your SSO connections.

Steps:

1. Create the Trifacta service principal:
   a. Form: `HTTP/serviceprincipal@REAM`
   b. Enable this flag: `ok_to_auth_as_delegate`
   c. Example:

```
kadmin -q "addprinc -randkey +ok_to_auth_as_delegate HTTP/serviceprincipal"
kadmin -q "addprinc -randkey +ok_to_auth_as_delegate HTTP/serviceprincipal@REALM"
```

   d. For more information on delegation flags, see *https://web.mit.edu/kerberos/krb5-1.12/doc/admin/admin_commands/kadmin_local.html*
2. Generate a keytab for the Trifacta service principal.
3. Register the Trifacta service principal for Microsoft Sql Server instance:
   a. Enable this flag: `ok_as_delegated`
   b. Example:

```
kadmin -q "addprinc -randkey +ok_as_delegate MSSQLSvc/<FQDN>:<port>"
kadmin -q "addprinc -randkey +ok_as_delegate MSSQLSvc/<FQDN>:<port>@REALM"
kadmin -q "addprinc -randkey +ok_as_delegate MSSQLSvc/<FQDN>"
kadmin -q "addprinc -randkey +ok_as_delegate MSSQLSvc/<FQDN>@REALM"
```

   c. For more information on setting this flag, see *https://docs.microsoft.com/en-us/sql/database-engine/configure-windows/register-a-service-principal-name-for-kerberos-connections?view=sql-server-2017* .
4. Create a linked SQL Server account:
   a. Account must have the same name as the end-user principal.
   b. Account needs connect permissions at least.

NOTE: If you are using LDAP/AD SSO, you can register all of the above SPNs using AD mechanisms. You do not have to use the delegation flags. Delegation can be managed through the UI for the service account.

Use

When you create a new connection of a supported type, you can select the Kerberos Delegate credentials type. When selected, no username or credentials are applied as part of the connection object. Instead, authentication is determined via Kerberos authentication with the cluster.

- *Create SQL Server Connections*

Sharing

When sharing SSO connections, the credentials for the connection cannot be shared for security reasons. The Kerberos principal for the user with whom the connection is shared is applied. That user must have the appropriate permissions to access any required data through the connection. See *Overview of Sharing*.