cloudera

# Cloudera ODBC Driver for Impala
# Version 2.5.36

**Important Notice**



**Cloudera, Inc.**
**1001 Page Mill Road, Building 2**
**Palo Alto, CA 94304-1008**
**info@cloudera.com**
**US: 1-888-789-1488**
**Intl: 1-650-843-0595**
**www.cloudera.com**

**Release Information**

Version: 2.5.36

Date: November 3, 2016

# Table of Contents

# About the Cloudera ODBC Driver for Impala

The Cloudera ODBC Driver for Impala is used for direct SQL and Impala SQL access to Apache Hadoop / Impala distributions, enabling Business Intelligence (BI), analytics, and reporting on Hadoop / Impala-based data. The driver efficiently transforms an application's SQL query into the equivalent form in Impala SQL, which is a subset of SQL-92. If an application is Impala-aware, then the driver is configurable to pass the query through to the database for processing. The driver interrogates Impala to obtain schema information to present to a SQL-based application. Queries, including joins, are translated from SQL to Impala SQL. For more information about the differences between Impala SQL and SQL, see "Features" on page 47.

The Cloudera ODBC Driver for Impala complies with the ODBC 3.80 data standard and adds important functionality such as Unicode and 32- and 64-bit support for high-performance computing environments.

ODBC is one of the most established and widely supported APIs for connecting to and working with databases. At the heart of the technology is the ODBC driver, which connects an application to the database. For more information about ODBC, see the *Data Access Standards Glossary*: http://www.simba.com/resources/data-access-standards-library. For complete information about the ODBC specification, see the *ODBC API Reference*: http://msdn.microsoft.com/en-us/library/windows/desktop/ms714562(v=vs.85).aspx.

The *Installation and Configuration Guide* is suitable for users who are looking to access data residing within Impala from their desktop environment. Application developers might also find the information helpful. Refer to your application for details on connecting via ODBC.

# Windows Driver

## Windows System Requirements

The Cloudera ODBC Driver for Impala supports Impala versions 1.0.1 through 2.5, and CDH versions 5.6 and 5.7.

Install the driver on client machines where the application is installed. Each machine that you install the driver on must meet the following minimum system requirements:

- One of the following operating systems:
  - Windows Vista, 7, 8, or 10
  - Windows Server 2008 or later
- 100 MB of available disk space
- Visual C++ Redistributable for Visual Studio 2013 installed (both 32- and 64-bit). You can download the installation packages at https://www.microsoft.com/en-ca/download/details.aspx?id=40784.

To install the driver, you must have Administrator privileges on the machine.

## Installing the Driver on Windows

On 64-bit Windows operating systems, you can execute both 32- and 64-bit applications. However, 64-bit applications must use 64-bit drivers, and 32-bit applications must use 32-bit drivers. Make sure that you use the version of the driver that matches the bitness of the client application:

- `ClouderaImpalaODBC32.msi` for 32-bit applications
- `ClouderaImpalaODBC64.msi` for 64-bit applications

You can install both versions of the driver on the same machine.

**To install the Cloudera ODBC Driver for Impala on Windows:**

1. Depending on the bitness of your client application, double-click to run **ClouderaImpalaODBC32.msi** or **ClouderaImpalaODBC64.msi**.
2. Click **Next**.
3. Select the check box to accept the terms of the License Agreement if you agree, and then click **Next**.
4. To change the installation location, click **Change**, then browse to the desired folder, and then click **OK**. To accept the installation location, click **Next**.
5. Click **Install**.
6. When the installation completes, click **Finish**.

## Creating a Data Source Name on Windows

Typically, after installing the Cloudera ODBC Driver for Impala, you need to create a Data Source Name (DSN).

Alternatively, for information about DSN-less connections, see "Using a Connection String" on page 44.

**To create a Data Source Name on Windows:**

1. Open the ODBC Administrator:
   - If you are using Windows 7 or earlier, click **Start** > **All Programs** > **Cloudera ODBC Driver for Impala 2.5** > **ODBC Administrator**.
   - Or, if you are using Windows 8 or later, on the Start screen, type **ODBC administrator**, and then click the **ODBC Administrator** search result.

   **Note:**

   Make sure to select the ODBC Data Source Administrator that has the same bitness as the client application that you are using to connect to Impala.
2. In the ODBC Data Source Administrator, click the **Drivers** tab, and then scroll down as needed to confirm that the Cloudera ODBC Driver for Impala appears in the alphabetical list of ODBC drivers that are installed on your system.
3. Choose one:
   - To create a DSN that only the user currently logged into Windows can use, click the **User DSN** tab.
   - Or, to create a DSN that all users who log into Windows can use, click the **System DSN** tab.

   **Note:**

   It is recommended that you create a System DSN instead of a User DSN. Some applications, such as Sisense, load the data using a different user account, and might not be able to detect User DSNs that are created under another user account.
4. Click **Add**.
5. In the Create New Data Source dialog box, select **Cloudera ODBC Driver for Impala** and then click **Finish**. The Cloudera ODBC Driver for Impala DSN Setup dialog box opens.
6. In the **Data Source Name** field, type a name for your DSN.
7. Optionally, in the **Description** field, type relevant details about the DSN.
8. In the **Host** field, type the IP address or host name of the network load balancer (NLB) or one of the Impala nodes if you are deployed without an NLB.
9. In the **Port** field, type the number of the TCP port that the Impala server uses to listen for client connections.

   **Note:**

   The default port number used by Impala is 21050.

10. In the **Database** field, type the name of the database schema to use when a schema is not explicitly specified in a query.

    **Note:**

    You can still issue queries on other schemas by explicitly specifying the schema in the query. To inspect your databases and determine the appropriate schema to use, type the `show databases` command at the Impala command prompt.

11. In the Authentication area, configure authentication as needed. For more information, see "Configuring Authentication on Windows" on page 9.

    **Note:**

    The default configuration of Impala requires the Cloudera ODBC Driver for Impala to be configured to use the No Authentication mechanism.

12. Optionally, if the operations against Impala are to be done on behalf of a user that is different than the authenticated user for the connection, type the name of the user to be delegated in the **Delegation UID** field.
13. To configure client-server verification over SSL, click **SSL Options**. For more information, see "Configuring SSL Verification on Windows" on page 13.
14. To configure advanced driver options, click **Advanced Options**. For more information, see "Configuring Advanced Options on Windows" on page 13.
15. To configure server-side properties, click **Advanced Options** and then click **Server Side Properties**. For more information, see "Configuring Server-Side Properties on Windows" on page 15.
16. To configure logging behavior for the driver, click **Logging Options**. For more information, see "Configuring Logging Options on Windows" on page 15.
17. To test the connection, click **Test**. Review the results as needed, and then click **OK**.

    **Note:**

    If the connection fails, then confirm that the settings in the Cloudera ODBC Driver for Impala DSN Setup dialog box are correct. Contact your Impala server administrator as needed.

18. To save your settings and close the Cloudera ODBC Driver for Impala DSN Setup dialog box, click **OK**.
19. To close the ODBC Data Source Administrator, click **OK**.

## Configuring Authentication on Windows

You must determine the authentication type your server is using and configure your DSN accordingly. The Impala server supports the following authentication methods:



**Note:**

In addition to authentication, you can configure the driver to connect over the Secure Sockets Layer (SSL). For more information, see "Configuring SSL Verification on Windows" on page 13.

### Using No Authentication

For this authentication mechanism, you do not need to configure any additional settings.

**Note:**

The default configuration of Impala requires the Cloudera ODBC Driver for Impala to be configured to use the No Authentication mechanism.

**To configure a connection without authentication:**

1. To access authentication options, open the ODBC Data Source Administrator where you created the DSN, then select the DSN, and then click **Configure**.
2. In the **Mechanism** drop-down list, select **No Authentication**.
3. If the Impala server is configured to use SSL, then click **SSL Options** to configure SSL for the connection. For more information, see "Configuring SSL Verification on Windows" on page 13.
4. To save your settings and close the dialog box, click **OK**.

### Using Kerberos

Kerberos must be installed and configured before you can use this authentication mechanism. For more information, see "Configuring Kerberos Authentication for Windows" on page 17.

**To configure Kerberos authentication:**

1. To access authentication options, open the ODBC Data Source Administrator where you created the DSN, then select the DSN, and then click **Configure**.
2. In the **Mechanism** drop-down list, select **Kerberos**.
3. Choose one:

- To use the default realm defined in your Kerberos setup, leave the **Realm** field empty.
- Or, if your Kerberos setup does not define a default realm or if the realm of your Impala server host is not the default, then, in the **Realm** field, type the Kerberos realm of the Impala server.

4. In the **Host FQDN** field, type the fully qualified domain name of the Impala server host.

   **Note:**

   To use the Impala server host name as the fully qualified domain name for Kerberos authentication, in the **Host FQDN** field, type **_HOST**.

5. To allow the driver to pass your credentials directly to the server for use in authentication, select **Delegate Kerberos Credentials**.
6. In the **Service Name** field, type the service name of the Impala server.
7. If the Impala server is configured to use SSL, then click **SSL Options** to configure SSL for the connection. For more information, see "Configuring SSL Verification on Windows" on page 13.
8. Optionally, in the **Transport Buffer Size** field, type the number of bytes to reserve in memory for buffering unencrypted data from the network.

   **Note:**

   In most circumstances, the default value of 1000 bytes is optimal.

9. To save your settings and close the dialog box, click **OK**.

#### Using Advanced Kerberos

The Advanced Kerberos authentication mechanism allows concurrent connections within the same process to use different Kerberos user principals.

This authentication mechanism is supported only when the driver is configured to handle Kerberos authentication using MIT Kerberos:

- MIT Kerberos must be installed on your machine.
- The Use Only SSPI Plugin option must be disabled. For more information, see "Use Only SSPI Plugin" on page 62.

When you use Advanced Kerberos authentication, you do not need to run the `kinit` command to obtain a Kerberos ticket. Instead, you use a JSON file to map your Impala user name to a Kerberos user principal name and a keytab that contains the corresponding keys. The driver obtains Kerberos tickets based on the specified mapping. As a fallback, you can specify a keytab that the driver uses by default if the mapping file is not available or if no matching keytab can be found in the mapping file.

**Note:**

- For information about the schema of the mapping file and how the driver handles invalid mappings, see "UPN Keytab Mapping File" on page 60.
- For information about how the driver searches for a keytab file if the keytab mapping and default keytab file are invalid, see "Default Keytab File" on page 52.

**To configure Advanced Kerberos authentication:**

1. To access authentication options, open the ODBC Data Source Administrator where you created the DSN, then select the DSN, and then click **Configure**.
2. In the **Mechanism** drop-down list, select **Kerberos**.
3. Choose one:
   - To use the default realm defined in your Kerberos setup, leave the **Realm** field empty.
   - Or, if your Kerberos setup does not define a default realm or if the realm of your Impala server host is not the default, then, in the **Realm** field, type the Kerberos realm of the Impala server.
4. In the **Host FQDN** field, type the fully qualified domain name of the Impala server host.

   **Note:**

   To use the Impala server host name as the fully qualified domain name for Kerberos authentication, in the **Host FQDN** field, type **_HOST**.
5. In the **Service Name** field, type the service name of the Impala server.
6. Select the **Use Keytab** check box.

   **Note:**

   If the check box is not available, make sure that MIT Kerberos is installed on your machine.
7. In the **User Name** field, type an appropriate user name for accessing the Impala server.
8. Click **Keytab Options** and then do the following in the Keytab Options dialog box:
   a. In the **UPN Keytab Mapping File** field, specify the full path to a JSON file that maps your Impala user name to a Kerberos user principal name and a keytab file.
   b. In the **Default Keytab File** field, specify the full path to a keytab file that the driver can use if the mapping file is not available or if no matching keytab can be found in the mapping file.
   c. To save your settings and close the dialog box, click **OK**.
9. If the Impala server is configured to use SSL, then click **SSL Options** to configure SSL for the connection. For more information, see "Configuring SSL Verification on Windows" on page 13.

10. Optionally, in the **Transport Buffer Size** field, type the number of bytes to reserve in memory for buffering unencrypted data from the network.

    **Note:**

    In most circumstances, the default value of 1000 bytes is optimal.

11. To save your settings and close the dialog box, click **OK**.

#### Using SASL User Name

This authentication mechanism requires a user name but not a password. The user name labels the session, facilitating database tracking.

**To configure SASL User Name authentication:**

1. To access authentication options, open the ODBC Data Source Administrator where you created the DSN, then select the DSN, and then click **Configure**.
2. In the **Mechanism** drop-down list, select **SASL User Name**.
3. In the **User Name** field, type an appropriate user name for accessing the Impala server.
4. Optionally, in the **Transport Buffer Size** field, type the number of bytes to reserve in memory for buffering unencrypted data from the network.

   **Note:**

   In most circumstances, the default value of 1000 bytes is optimal.

5. To save your settings and close the dialog box, click **OK**.

#### Using User Name And Password

This authentication mechanism requires a user name and a password.

**Note:**

This authentication mechanism should not be used with an Impala configuration that does not have LDAP enabled.

**To configure User Name And Password authentication:**

1. To access authentication options, open the ODBC Data Source Administrator where you created the DSN, then select the DSN, and then click **Configure**.
2. In the **Mechanism** drop-down list, select **User Name And Password**.
3. In the **User Name** field, type an appropriate user name for accessing the Impala server.
4. In the **Password** field, type the password corresponding to the user name you typed above.
5. To save the password, select the **Save Password (Encrypted)** check box.

**Important:**

The password is obscured, that is, not saved in plain text. However, it is still possible for the encrypted password to be copied and used.

6. If the Impala server is configured to use SSL, then click **SSL Options** to configure SSL for the connection. For more information, see "Configuring SSL Verification on Windows" on page 13.
7. Optionally, in the **Transport Buffer Size** field, type the number of bytes to reserve in memory for buffering unencrypted data from the network.

**Note:**

In most circumstances, the default value of 1000 bytes is optimal.

8. Optionally, to use SASL to handle authentication, select the **Use Simple Authentication and Security Layer (SASL)** check box.
9. To save your settings and close the dialog box, click **OK**.

## Configuring SSL Verification on Windows

If you are connecting to an Impala server that has Secure Sockets Layer (SSL) enabled, you can configure verification between the client and the Impala server over SSL.

**To configure SSL verification on Windows:**

1. To access SSL options, open the ODBC Data Source Administrator where you created the DSN, then select the DSN, then click **Configure**, and then click **SSL Options**.
2. Select the **Enable SSL** check box.
3. To allow self-signed certificates from the server, select the **Allow Self-signed Server Certificate** check box.
4. To allow the common name of a CA-issued SSL certificate to not match the host name of the Impala server, select the **Allow Common Name Host Name Mismatch** check box.
5. Choose one:
   - To configure the driver to load SSL certificates from a specific PEM file when verifying the server, specify the full path to the file in the **Trusted Certificates** field.
   - Or, to use the trusted CA certificates PEM file that is installed with the driver, leave the **Trusted Certificates** field empty.
6. To save your settings and close the SSL Options dialog box, click **OK**.

## Configuring Advanced Options on Windows

You can configure advanced options to modify the behavior of the driver.

**To configure advanced options on Windows:**

1. To access advanced options, open the ODBC Data Source Administrator where you created the DSN, then select the DSN, then click **Configure**, and then click **Advanced Options**.
2. To disable translation from ODBC SQL to Impala SQL, select the **Use Native Query** check box.

   **Note:**

   By default, the driver applies transformations to the queries emitted by an application to convert the queries into an equivalent form in Impala SQL. If the application is Impala-aware and already emits Impala SQL, then turning off the translation avoids the additional overhead of query transformation.

3. To enable the driver to successfully run queries that contain transaction statements, select the **Enable Simulated Transactions** check box.

   **Note:**

   The transaction statements are not executed, because ODBC does not support them. Enabling this option allows the driver to run the query without returning error messages.

4. To enable the driver to return SQL_WVARCHAR instead of SQL_VARCHAR for STRING and VARCHAR columns, and SQL_WCHAR instead of SQL_CHAR for CHAR columns, select the **Use SQL Unicode Types** check box.
5. To have the system automatically attempt to reconnect to the server if communications are lost, set **Enable Auto Reconnect**.
6. To handle Kerberos authentication using the SSPI plugin instead of MIT Kerberos by default, select one or both of the check boxes under the **Use Only SSPI Plugin** option:
   - To configure the current DSN to use the SSPI plugin by default, select **Enable For This DSN**.
   - To configure all DSN-less connections to use the SSPI plugin by default, select **Enable For DSN-less Connections**.
   - To configure all connections that use the Cloudera ODBC Driver for Impala to use the SSPI plugin by default, select both check boxes.
7. In the **Rows Fetched Per Block** field, type the number of rows to be fetched per block.
8. In the **Socket Timeout** field, type the number of seconds after which Impala closes the connection with the client application if the connection is idle.

   **Note:**

   Setting the Socket Timeout value to 0 disables the timeout feature.

9. In the **String Column Length** field, type the maximum data length for STRING columns.
10. To save your settings and close the Advanced Options dialog box, click **OK**.

## Configuring Server-Side Properties on Windows

When connecting to a server that is running Impala 2.0 or later, you can use the driver to apply configuration properties to the server.

**Important:**

This feature is not supported for earlier versions of Impala, where the SET statement can only be executed from within the Impala shell.

**To configure server-side properties on Windows:**

1. To configure server-side properties, open the ODBC Data Source Administrator where you created the DSN, then select the DSN, then click **Configure**, then click **Advanced Options**, and then click **Server Side Properties**.
2. To create a server-side property, click **Add**, then type appropriate values in the **Key** and **Value** fields, and then click **OK**. For example, to set the value of the MEM_LIMIT query option to 1 GB, type **MEM_LIMIT** in the **Key** field and then type **1000000000** in the **Value** field.
3. To edit a server-side property, select the property from the list, then click **Edit**, then update the **Key** and **Value** fields as needed, and then click **OK**.
4. To delete a server-side property, select the property from the list, and then click **Remove**. In the confirmation dialog box, click **Yes**.
5. To configure the driver to convert server-side property key names to all lower-case characters, select the **Convert Key Name To Lower Case** check box.
6. To save your settings and close the Server Side Properties dialog box, click **OK**.

## Configuring Logging Options on Windows

To help troubleshoot issues, you can enable logging. In addition to functionality provided in the Cloudera ODBC Driver for Impala, the ODBC Data Source Administrator provides tracing functionality.

**Important:**

Only enable logging or tracing long enough to capture an issue. Logging or tracing decreases performance and can consume a large quantity of disk space.

**To enable driver logging on Windows:**

1. To access logging options, open the ODBC Data Source Administrator where you created the DSN, then select the DSN, then click **Configure**, and then click **Logging Options**.
2. From the **Log Level** drop-down list, select the logging level corresponding to the amount of information that you want to include in log files:

| Logging Level | Description |
|---|---|
| OFF | Disables all logging. |
| FATAL | Logs severe error events that lead the driver to abort. |
| ERROR | Logs error events that might allow the driver to continue running. |
| WARNING | Logs events that might result in an error if action is not taken. |
| INFO | Logs general information that describes the progress of the driver. |
| DEBUG | Logs detailed information that is useful for debugging the driver. |
| TRACE | Logs all driver activity. |

3. In the **Log Path** field, specify the full path to the folder where you want to save log files.
4. If requested by Technical Support, type the name of the component for which to log messages in the **Log Namespace** field. Otherwise, do not type a value in the field.
5. In the **Max Number Files** field, type the maximum number of log files to keep.

   **Note:**

   After the maximum number of log files is reached, each time an additional file is created, the driver deletes the oldest log file.

6. In the **Max File Size** field, type the maximum size of each log file in megabytes (MB).

   **Note:**

   After the maximum file size is reached, the driver creates a new file and continues logging.

7. Click **OK**.
8. Restart your ODBC application to make sure that the new settings take effect.

The Cloudera ODBC Driver for Impala produces a log file named `ImpalaODBC_driver.log` at the location that you specify in the Log Path field.

**To disable driver logging on Windows:**

1. Open the ODBC Data Source Administrator where you created the DSN, then select the DSN, then click **Configure**, and then click **Logging Options**.
2. From the **Log Level** drop-down list, select **LOG_OFF**.

3. Click **OK**.
4. Restart your ODBC application to make sure that the new settings take effect.

**To start tracing using the ODBC Data Source Administrator:**

1. In the ODBC Data Source Administrator, click the **Tracing** tab.
2. In the **Log File Path** area, click **Browse**. In the Select ODBC Log File dialog box, browse to the location where you want to save the log file, then type a descriptive file name in the **File Name** field, and then click **Save**.
3. On the Tracing tab, click **Start Tracing Now**.

**To stop ODBC Data Source Administrator tracing:**

- On the Tracing tab in the ODBC Data Source Administrator, click **Stop Tracing Now**.

For more information about tracing using the ODBC Data Source Administrator, see "How to Generate an ODBC Trace with ODBC Data Source Administrator" on the Microsoft Support website: http://support.microsoft.com/kb/274551.

## Configuring Kerberos Authentication for Windows

### Active Directory

The Cloudera ODBC Driver for Impala supports Active Directory Kerberos on Windows. There are two prerequisites for using Active Directory Kerberos on Windows:

- MIT Kerberos is not installed on the client Windows machine.
- The MIT Kerberos Hadoop realm has been configured to trust the Active Directory realm, according to Cloudera's documentation, so that users in the Active Directory realm can access services in the MIT Kerberos Hadoop realm.

### MIT Kerberos

#### Downloading and Installing MIT Kerberos for Windows 4.0.1

For information about Kerberos and download links for the installer, see the MIT Kerberos website: http://web.mit.edu/kerberos/.

**To download and install MIT Kerberos for Windows 4.0.1:**

1. Download the appropriate Kerberos installer:
    - For a 64-bit machine, use the following download link from the MIT Kerberos website: http://web.mit.edu/kerberos/dist/kfw/4.0/kfw-4.0.1-amd64.msi.
    - For a 32-bit machine, use the following download link from the MIT Kerberos website: http://web.mit.edu/kerberos/dist/kfw/4.0/kfw-4.0.1-i386.msi.

> **Note:**
>
> The 64-bit installer includes both 32-bit and 64-bit libraries. The 32-bit installer includes 32-bit libraries only.

2. To run the installer, double-click the `.msi` file that you downloaded above.
3. Follow the instructions in the installer to complete the installation process.
4. When the installation completes, click **Finish**.

#### Setting Up the Kerberos Configuration File

Settings for Kerberos are specified through a configuration file. You can set up the configuration file as an `.ini` file in the default location, which is the `C:\ProgramData\MIT\Kerberos5` directory, or as a `.conf` file in a custom location.

Normally, the `C:\ProgramData\MIT\Kerberos5` directory is hidden. For information about viewing and using this hidden directory, refer to Microsoft Windows documentation.

> **Note:**
>
> For more information on configuring Kerberos, refer to the MIT Kerberos documentation.

**To set up the Kerberos configuration file in the default location:**

1. Obtain a `krb5.conf` configuration file. You can obtain this file from your Kerberos administrator, or from the `/etc/krb5.conf` folder on the machine that is hosting the Impala server.
2. Rename the configuration file from `krb5.conf` to `krb5.ini`.
3. Copy the `krb5.ini` file to the `C:\ProgramData\MIT\Kerberos5` directory and overwrite the empty sample file.

**To set up the Kerberos configuration file in a custom location:**

1. Obtain a `krb5.conf` configuration file. You can obtain this file from your Kerberos administrator, or from the `/etc/krb5.conf` folder on the machine that is hosting the Impala server.
2. Place the `krb5.conf` file in an accessible directory and make note of the full path name.
3. Open the System window:
   - If you are using Windows 7 or earlier, click **Start**, then right-click **Computer**, and then click **Properties**.
   - Or, if you are using Windows 8 or later, right-click **This PC** on the Start screen, and then click **Properties**.
4. Click **Advanced System Settings**.
5. In the System Properties dialog box, click the **Advanced** tab and then click **Environment Variables**.
6. In the Environment Variables dialog box, under the System Variables list, click **New**.

7. In the New System Variable dialog box, in the **Variable Name** field, type **KRB5_CONFIG**.
8. In the **Variable Value** field, type the full path to the `krb5.conf` file.
9. Click **OK** to save the new variable.
10. Make sure that the variable is listed in the System Variables list.
11. Click **OK** to close the Environment Variables dialog box, and then click **OK** to close the System Properties dialog box.

#### Setting Up the Kerberos Credential Cache File

Kerberos uses a credential cache to store and manage credentials.

**To set up the Kerberos credential cache file:**

1. Create a directory where you want to save the Kerberos credential cache file. For example, create a directory named `C:\temp`.
2. Open the System window:
   - If you are using Windows 7 or earlier, click **Start**, then right-click **Computer**, and then click **Properties**.
   - Or, if you are using Windows 8 or later, right-click **This PC** on the Start screen, and then click **Properties**.
3. Click **Advanced System Settings**.
4. In the System Properties dialog box, click the **Advanced** tab and then click **Environment Variables**.
5. In the Environment Variables dialog box, under the System Variables list, click **New**.
6. In the New System Variable dialog box, in the **Variable Name** field, type **KRB5CCNAME**.
7. In the **Variable Value** field, type the path to the folder you created above, and then append the file name `krb5cache`. For example, if you created the folder `C:\temp`, then type `C:\temp\krb5cache`.

   **Note:**

   `krb5cache` is a file (not a directory) that is managed by the Kerberos software, and it should not be created by the user. If you receive a permission error when you first use Kerberos, make sure that the `krb5cache` file does not already exist as a file or a directory.

8. Click **OK** to save the new variable.
9. Make sure that the variable appears in the System Variables list.
10. Click **OK** to close the Environment Variables dialog box, and then click **OK** to close the System Properties dialog box.
11. To make sure that Kerberos uses the new settings, restart your machine.

#### Obtaining a Ticket for a Kerberos Principal

A principal refers to a user or service that can authenticate to Kerberos. To authenticate to Kerberos, a principal must obtain a ticket by using a password or a keytab file. You can specify a keytab file to use, or use the default keytab file of your Kerberos configuration.

**To obtain a ticket for a Kerberos principal using a password:**

1. Open MIT Kerberos Ticket Manager.
2. In MIT Kerberos Ticket Manager, click **Get Ticket**.
3. In the Get Ticket dialog box, type your principal name and password, and then click **OK**.

   If the authentication succeeds, then your ticket information appears in MIT Kerberos Ticket Manager.

**To obtain a ticket for a Kerberos principal using a keytab file:**

1. Open a command prompt:
   - If you are using Windows 7 or earlier, click **Start**, then click **All Programs**, then click **Accessories**, and then click **Command Prompt**.
   - If you are using Windows 8 or later, click the arrow button at the bottom of the Start screen, then find the Windows System program group, and then click **Command Prompt**.
2. In the Command Prompt, type a command using the following syntax:

   ```
   kinit -k -t [KeytabPath] [Principal]
   ```

   *[KeytabPath]* is the full path to the keytab file. For example: `C:\mykeytabs\myUser.keytab`.

   *[Principal]* is the Kerberos user principal to use for authentication. For example: `myUser@EXAMPLE.COM`.
3. If the cache location KRB5CCNAME is not set or used, then use the `-c` option of the `kinit` command to specify the location of the credential cache. In the command, the `-c` argument must appear last. For example:

   ```
   kinit -k -t C:\mykeytabs\myUser.keytab myUser@EXAMPLE.COM
   -c C:\ProgramData\MIT\krbcache
   ```

   `Krbcache` is the Kerberos cache file, not a directory.

**To obtain a ticket for a Kerberos principal using the default keytab file:**

> **Note:**
>
> For information about configuring a default keytab file for your Kerberos configuration, refer to the MIT Kerberos documentation.

1. Open a command prompt:
   - If you are using Windows 7 or earlier, click **Start**, then click **All Programs**, then click **Accessories**, and then click **Command Prompt**.
   - If you are using Windows 8 or later, click the arrow button at the bottom of the Start screen, then find the Windows System program group, and then click **Command Prompt**.
2. In the Command Prompt, type a command using the following syntax:

   ```
   kinit -k [principal]
   ```

   *[principal]* is the Kerberos user principal to use for authentication. For example: `MyUser@EXAMPLE.COM`.
3. If the cache location KRB5CCNAME is not set or used, then use the `-c` option of the `kinit` command to specify the location of the credential cache. In the command, the `-c` argument must appear last. For example:

   ```
   kinit -k -t C:\mykeytabs\myUser.keytab myUser@EXAMPLE.COM
   -c C:\ProgramData\MIT\krbcache
   ```

   `Krbcache` is the Kerberos cache file, not a directory.

## Verifying the Driver Version Number on Windows

If you need to verify the version of the Cloudera ODBC Driver for Impala that is installed on your Windows machine, you can find the version number in the ODBC Data Source Administrator.

**To verify the driver version number on Windows:**

1. Open the ODBC Administrator:
   - If you are using Windows 7 or earlier, click **Start** **> All Programs > Cloudera ODBC Driver for Impala 2.5 > ODBC Administrator**.
   - Or, if you are using Windows 8 or later, on the Start screen, type **ODBC administrator**, and then click the **ODBC Administrator** search result.

   **Note:**

   Make sure to select the ODBC Data Source Administrator that has the same bitness as the client application that you are using to connect to Impala.
2. Click the **Drivers** tab and then find the Cloudera ODBC Driver for Impala in the list of ODBC drivers that are installed on your system. The version number is displayed in the **Version** column.

# Mac OS X Driver

## Mac OS X System Requirements

The Cloudera ODBC Driver for Impala supports Impala versions 1.0.1 through 2.5, and CDH versions 5.6 and 5.7.

Install the driver on client machines where the application is installed. Each machine that you install the driver on must meet the following minimum system requirements:

- Mac OS X version 10.9, 10.10, or 10.11
- 100 MB of available disk space
- iODBC 3.52.7 or later

## Installing the Driver on Mac OS X

The Cloudera ODBC Driver for Impala is available for Mac OS X as a `.dmg` file named `ClouderaImpalaODBC.dmg`. The driver supports both 32- and 64-bit client applications.

**To install the Cloudera ODBC Driver for Impala on Mac OS X:**

1. Double-click **ClouderaImpalaODBC.dmg** to mount the disk image.
2. Double-click **ClouderaImpalaODBC.pkg** to run the installer.
3. In the installer, click **Continue**.
4. On the Software License Agreement screen, click **Continue**, and when the prompt appears, click **Agree** if you agree to the terms of the License Agreement.
5. Optionally, to change the installation location, click **Change Install Location**, then select the desired location, and then click **Continue**.

   **Note:**

   By default, the driver files are installed in the `/opt/cloudera/impalaodbc` directory.

6. To accept the installation location and begin the installation, click **Install**.
7. When the installation completes, click **Close**.

Next, configure the environment variables on your machine to make sure that the ODBC driver manager can work with the driver. For more information, see "Configuring the ODBC Driver Manager on Non-Windows Machines" on page 29.

## Verifying the Driver Version Number on Mac OS X

If you need to verify the version of the Cloudera ODBC Driver for Impala that is installed on your Mac OS X machine, you can query the version number through the Terminal.

**To verify the driver version number on Mac OS X:**

- At the Terminal, run the following command:

```
pkgutil --info com.cloudera.impalaodbc
```

The command returns information about the Cloudera ODBC Driver for Impala that is installed on your machine, including the version number.

# Linux Driver

For most Linux distributions, you can install the driver using the RPM file. If you are installing the driver on a Debian machine, you must use the Debian package.

## Linux System Requirements

The Cloudera ODBC Driver for Impala supports Impala versions 1.0.1 through 2.5, and CDH versions 5.6 and 5.7.

Install the driver on client machines where the application is installed. Each machine that you install the driver on must meet the following minimum system requirements:

- One of the following distributions:
  - Red Hat® Enterprise Linux® (RHEL) 5, 6, or 7
  - CentOS 5, 6, or 7
  - SUSE Linux Enterprise Server (SLES) 11 or 12
  - Debian 6 or 7
  - Ubuntu 12.04 or 14.04
- 50 MB of available disk space
- One of the following ODBC driver managers installed:
  - iODBC 3.52.7 or later
  - unixODBC 2.3.02.3.0 or later
- All of the following `libsasl` libraries installed:
  - `cyrus-sasl-2.1.22-7` or later
  - `cyrus-sasl-gssapi-2.1.22-7` or later
  - `cyrus-sasl-plain-2.1.22-7` or later

> **Note:**
>
> If the package manager in your Linux distribution cannot resolve the dependencies automatically when installing the driver, then download and manually install the packages.

To install the driver, you must have root access on the machine.

## Installing the Driver Using the RPM File

On 64-bit editions of Linux, you can execute both 32- and 64-bit applications. However, 64-bit applications must use 64-bit drivers, and 32-bit applications must use 32-bit drivers. Make sure that you use the version of the driver that matches the bitness of the client application:

- `ClouderaImpalaODBC-32bit-[Version]-[Release].i686.rpm` for the 32-bit driver
- `ClouderaImpalaODBC-[Version]-[Release].x86_64.rpm` for the 64-bit driver

*[Version]* is the version number of the driver, and *[Release]* is the release number for this version of the driver.

You can install both versions of the driver on the same machine.

**To install the Cloudera ODBC Driver for Impala using the RPM File:**

1. Log in as the root user, and then navigate to the folder containing the RPM package for the driver.
2. Depending on the Linux distribution that you are using, run one of the following commands from the command line, where *[RPMFileName]* is the file name of the RPM package:
   - If you are using Red Hat Enterprise Linux or CentOS, run the following command:
     ```
     yum --nogpgcheck localinstall [RPMFileName]
     ```
   - Or, if you are using SUSE Linux Enterprise Server, run the following command:
     ```
     zypper install [RPMFileName]
     ```

   The Cloudera ODBC Driver for Impala files are installed in the `/opt/cloudera/impalaodbc` directory.

   **Note:**

   If the package manager in your Linux distribution cannot resolve the `libsasl` dependencies automatically when installing the driver, then download and manually install the packages.

Next, configure the environment variables on your machine to make sure that the ODBC driver manager can work with the driver. For more information, see "Configuring the ODBC Driver Manager on Non-Windows Machines" on page 29.

## Installing the Driver on Debian

To install the driver on a Debian machine, use the Debian package instead of the RPM file or tarball package.

On 64-bit editions of Debian, you can execute both 32- and 64-bit applications. However, 64-bit applications must use 64-bit drivers, and 32-bit applications must use 32-bit drivers. Make sure that you use the version of the driver that matches the bitness of the client application:

- `ClouderaImpalaODBC-32bit-[Version]-[Release]_i386.deb` for the 32-bit driver
- `ClouderaImpalaODBC-[Version]-[Release]_amd64.deb` for the 64-bit driver

*[Version]* is the version number of the driver, and *[Release]* is the release number for this version of the driver.

You can install both versions of the driver on the same machine.

**To install the Cloudera ODBC Driver for Impala on Debian:**

1. Log in as the root user, and then navigate to the folder containing the Debian package for the driver.
2. Double-click **ClouderaImpalaODBC-32bit-*Version-Release*_i386.deb** or **ClouderaImpalaODBC-*Version-Release*_amd64.deb**.
3. Follow the instructions in the installer to complete the installation process.

   The Cloudera ODBC Driver for Impala files are installed in the `/opt/cloudera/impalaodbc` directory.

   **Note:**

   If the package manager in your Ubuntu distribution cannot resolve the `libsasl` dependencies automatically when installing the driver, then download and manually install the packages required by the version of the driver that you want to install.

Next, configure the environment variables on your machine to make sure that the ODBC driver manager can work with the driver. For more information, see "Configuring the ODBC Driver Manager on Non-Windows Machines" on page 29.

## Verifying the Driver Version Number on Linux

If you need to verify the version of the Cloudera ODBC Driver for Impala that is installed on your Linux machine, you can query the version number through the command-line interface if the driver was installed using an RPM file or Debian package.

**To verify the driver version number on Linux:**

- Depending on your package manager, at the command prompt, run one of the following commands:
  - `yum list | grep ClouderaImpalaODBC`
  - `rpm -qa | grep ClouderaImpalaODBC`
  - `dpkg -l | grep ClouderaImpalaODBC`

The command returns information about the Cloudera ODBC Driver for Impala that is installed on your machine, including the version number.

# AIX Driver

## AIX System Requirements

The Cloudera ODBC Driver for Impala supports Impala versions 1.0.1 through 2.5, and CDH versions 5.6 and 5.7.

Install the driver on client machines where the application is installed. Each machine that you install the driver on must meet the following minimum system requirements:

- IBM AIX 5.3, 6.1, or 7.1
- 150 MB of available disk space
- One of the following ODBC driver managers installed:
  - iODBC 3.52.7 or later
  - unixODBC 2.3.0 or later

To install the driver, you must have root access on the machine.

## Installing the Driver on AIX

On 64-bit editions of AIX, you can execute both 32- and 64-bit applications. However, 64-bit applications must use 64-bit drivers, and 32-bit applications must use 32-bit drivers. Make sure that you use the version of the driver that matches the bitness of the client application:

- `ClouderaImpalaODBC-32bit-`*`[Version]`*`-`*`[Release]`*`.ppc.rpm` for the 32-bit driver
- `ClouderaImpalaODBC-`*`[Version]`*`-`*`[Release]`*`.ppc.rpm` for the 64-bit driver

*[Version]* is the version number of the driver, and *[Release]* is the release number for this version of the driver.

You can install both versions of the driver on the same machine.

**To install the Cloudera ODBC Driver for Impala on AIX:**

1. Log in as the root user, and then navigate to the folder containing the RPM package for the driver.
2. Run the following command from the command line, where *[RPMFileName]* is the file name of the RPM package:

   ```
   rpm --install [RPMFileName]
   ```

   The Cloudera ODBC Driver for Impala files are installed in the `/opt/cloudera/impalaodbc` directory.

Next, configure the environment variables on your machine to make sure that the ODBC driver manager can work with the driver. For more information, see "Configuring the ODBC Driver Manager on Non-Windows Machines" on page 29.

## Verifying the Driver Version Number on AIX

If you need to verify the version of the Cloudera ODBC Driver for Impala that is installed on your AIX machine, you can query the version number through the command-line interface.

**To verify the driver version number on AIX:**

- At the command prompt, run the following command:

```
rpm -qa | grep ClouderaImpalaODBC
```

The command returns information about the Cloudera ODBC Driver for Impala that is installed on your machine, including the version number.

# Configuring the ODBC Driver Manager on Non-Windows Machines

To make sure that the ODBC driver manager on your machine is configured to work with the Cloudera ODBC Driver for Impala, do the following:

- Make sure that your machine uses the correct ODBC driver manager by setting the library path environment variable. For more information, see "Specifying ODBC Driver Managers on Non-Windows Machines" on page 29.
- If the driver configuration files are not stored in the default locations, then make sure that the ODBC driver manager locates and uses those files by setting environment variables. For more information, see "Specifying the Locations of the Driver Configuration Files" on page 29.

After configuring the ODBC driver manager, you can configure a connection and access your data store through the driver. For more information, see "Configuring ODBC Connections" on page 31.

## Specifying ODBC Driver Managers on Non-Windows Machines

You need to make sure that your machine uses the correct ODBC driver manager to load the driver. To do this, set the library path environment variable.

**Mac OS X**

If you are using a Mac OS X machine, then set the DYLD_LIBRARY_PATH environment variable to include the paths to the ODBC driver manager libraries. For example, if the libraries are installed in `/usr/local/lib`, then run the following command to set DYLD_LIBRARY_PATH for the current user session:

```
export DYLD_LIBRARY_PATH=$DYLD_LIBRARY_PATH:/usr/local/lib
```

For information about setting an environment variable permanently, refer to the Mac OS X shell documentation.

**Linux or AIX**

If you are using a Linux or AIX machine, then set the LD_LIBRARY_PATH environment variable to include the paths to the ODBC driver manager libraries. For example, if the libraries are installed in `/usr/local/lib`, then run the following command to set LD_LIBRARY_PATH for the current user session:

```
export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/usr/local/lib
```

For information about setting an environment variable permanently, refer to the Linux or AIX shell documentation.

## Specifying the Locations of the Driver Configuration Files

By default, ODBC driver managers are configured to use hidden versions of the `odbc.ini` and `odbcinst.ini` configuration files (named `.odbc.ini` and `.odbcinst.ini`) located in the home directory, as well as the `cloudera.impalaodbc.ini` file in the `lib` subfolder of the

driver installation directory. If you want to store these configuration files in different locations, then you must set environment variables to indicate the locations of these files:

- Set ODBCINI to point to your `odbc.ini` file.
- Set ODBCSYSINI to point to the directory containing the `odbcinst.ini` file.
- Set CLOUDERAIMPALAODBCINI to point to your `cloudera.impalaodbc.ini` file.

For example, if your `odbc.ini` and `cloudera.impalaodbc.ini` files are located in `/etc` and your `odbcinst.ini` file is located in `/usr/local/odbc`, then set the environment variables as follows:

```
export ODBCINI=/etc/odbc.ini
export ODBCSYSINI=/usr/local/odbc
export CLOUDERAIMPALAODBCINI=/etc/cloudera.impalaodbc.ini
```

To locate the `cloudera.impalaodbc.ini` file, the driver uses the following search order:

1. If the CLOUDERAIMPALAODBCINI environment variable is defined, then the driver searches for the file specified by the environment variable.
2. The driver searches the directory that contains the driver library files for a file named `cloudera.impalaodbc.ini`.
3. The driver searches the current working directory of the application for a file named `cloudera.impalaodbc.ini`.
4. The driver searches the home directory for a hidden file named `.cloudera.impalaodbc.ini` (prefixed with a period).
5. The driver searches the /etc directory for a file named `cloudera.impalaodbc.ini`.

# Configuring ODBC Connections

The following sections describe how to configure ODBC connections when using the Cloudera ODBC Driver for Impala on non-Windows platforms:



## Creating a Data Source Name on a Non-Windows Machine

When connecting to your data store using a DSN, you only need to configure the `odbc.ini` file. Set the properties in the `odbc.ini` file to create a DSN that specifies the connection information for your data store. For information about configuring a DSN-less connection instead, see "Configuring a DSN-less Connection on a Non-Windows Machine" on page 33.

If your machine is already configured to use an existing `odbc.ini` file, then update that file by adding the settings described below. Otherwise, copy the `odbc.ini` file from the `Setup` subfolder in the driver installation directory to the home directory, and then update the file as described below.

**To create a Data Source Name on a non-Windows machine:**

1. In a text editor, open the `odbc.ini` configuration file.

   **Note:**

   If you are using a hidden copy of the `odbc.ini` file, then you need to remove the period (`.`) from the start of the file name before the file becomes editable.

2. In the `[ODBC Data Sources]` section, add a new entry by typing a name for the DSN, an equal sign (=), and then the name of the driver.

   For example, on a Mac OS X machine:

   ```
   [ODBC Data Sources]
   Sample DSN=Cloudera ODBC Driver for Impala
   ```

   As another example, for a 32-bit driver on a Linux/AIX/Debian machine:

   ```
   [ODBC Data Sources]
   Sample DSN=Cloudera ODBC Driver for Impala 32-bit
   ```

3. Create a section that has the same name as your DSN, and then specify configuration options as key-value pairs in the section:

a. Set the `Driver` property to the full path of the driver library file that matches the bitness of the application.

   For example, on a Mac OS X machine:

   ```
   Driver=/opt/cloudera/impalaodbc/lib/universal/libclouderaimpalaodbc.dylib
   ```

   As another example, for a 32-bit driver on a Linux/AIX/Debian machine:

   ```
   Driver=/opt/cloudera/impalaodbc/lib/32/libclouderaimpalaodbc32.so
   ```

b. Set the `Host` property to the IP address or host name of the server.

   For example:

   ```
   Host=192.168.222.160
   ```

c. Set the `Port` property to the number of the TCP port that the server uses to listen for client connections.

   For example:

   ```
   Port=21050
   ```

d. If authentication is required to access the server, then specify the authentication mechanism and your credentials. For more information, see "Configuring Authentication on a Non-Windows Machine" on page 35.

e. If you want to connect to the server through SSL, then enable SSL and specify the certificate information. For more information, see "Configuring SSL Verification on a Non-Windows Machine" on page 38.

f. If you want to configure server-side properties, then set them as key-value pairs using a special syntax. For more information, see "Configuring Server-Side Properties on a Non-Windows Machine" on page 39.

g. Optionally, set additional key-value pairs as needed to specify other optional connection settings. For detailed information about all the configuration options supported by the Cloudera ODBC Driver for Impala, see "Driver Configuration Options" on page 50.

4. Save the `odbc.ini` configuration file.

> **Note:**
>
> If you are storing this file in its default location in the home directory, then prefix the file name with a period (`.`) so that the file becomes hidden. If you are storing this file in another location, then save it as a non-hidden file (without the prefix), and make sure that the ODBCINI environment variable specifies the location. For more information, see "Specifying the Locations of the Driver Configuration Files" on page 29.

For example, the following is an `odbc.ini` configuration file for Mac OS X containing a DSN that connects to an Impala server that does not require authentication:

```
[ODBC Data Sources]
Sample DSN=Cloudera ODBC Driver for Impala
[Sample DSN]
Driver=/opt/cloudera/impalaodbc/lib/universal/libclouderaimpalaodbc.dylib
Host=192.168.222.160
Port=21050
```

As another example, the following is an `odbc.ini` configuration file for a 32-bit driver on a Linux/AIX/Debian machine, containing a DSN that connects to an Impala server that does not require authentication:

```
[ODBC Data Sources]
Sample DSN=Cloudera ODBC Driver for Impala 32-bit
[Sample DSN]
Driver=/opt/cloudera/impalaodbc/lib/32/libclouderaimpalaodbc32.so
Host=192.168.222.160
Port=21050
```

You can now use the DSN in an application to connect to the data store.

## Configuring a DSN-less Connection on a Non-Windows Machine

To connect to your data store through a DSN-less connection, you need to define the driver in the `odbcinst.ini` file and then provide a DSN-less connection string in your application.

If your machine is already configured to use an existing `odbcinst.ini` file, then update that file by adding the settings described below. Otherwise, copy the `odbcinst.ini` file from the `Setup` subfolder in the driver installation directory to the home directory, and then update the file as described below.

**To define a driver on a non-Windows machine:**

1. In a text editor, open the `odbcinst.ini` configuration file.

   > **Note:**
   >
   > If you are using a hidden copy of the `odbcinst.ini` file, then you need to remove the period (`.`) from the start of the file name before the file becomes editable.

2. In the `[ODBC Drivers]` section, add a new entry by typing a name for the driver, an equal sign (=), and then `Installed`.

   For example:

   ```
   [ODBC Drivers]
   Cloudera ODBC Driver for Impala=Installed
   ```

3. Create a section that has the same name as the driver (as specified in the previous step), and then specify the following configuration options as key-value pairs in the section:

a. Set the `Driver` property to the full path of the driver library file that matches the bitness of the application.

   For example, on a Mac OS X machine:

   ```
   Driver=/opt/cloudera/impalaodbc/lib/universal/libcloude
   raimpalaodbc.dylib
   ```

   As another example, for a 32-bit driver on a Linux/AIX/Debian machine:

   ```
   Driver=/opt/cloudera/impalaodbc/lib/32/libclouderaimpal
   aodbc32.so
   ```

b. Optionally, set the `Description` property to a description of the driver.

   For example:

   ```
   Description=Cloudera ODBC Driver for Impala
   ```

4. Save the `odbcinst.ini` configuration file.

> **Note:**
>
> If you are storing this file in its default location in the home directory, then prefix the file name with a period (`.`) so that the file becomes hidden. If you are storing this file in another location, then save it as a non-hidden file (without the prefix), and make sure that the ODBCSYSINI environment variable specifies the location. For more information, see "Specifying the Locations of the Driver Configuration Files" on page 29.

For example, the following is an `odbcinst.ini` configuration file for Mac OS X:

```
[ODBC Drivers]
Cloudera ODBC Driver for Impala=Installed
[Cloudera ODBC Driver for Impala]
Description=Cloudera ODBC Driver for Impala
Driver=/opt/cloudera/impalaodbc/lib/universal/libclouderaimpalao
dbc.dylib
```

As another example, the following is an `odbcinst.ini` configuration file for both the 32- and 64-bit drivers on Linux/AIX/Debian:

```
[ODBC Drivers]
Cloudera ODBC Driver for Impala 32-bit=Installed
Cloudera ODBC Driver for Impala 64-bit=Installed
[Cloudera ODBC Driver for Impala 32-bit]
Description=Cloudera ODBC Driver for Impala (32-bit)
Driver=/opt/cloudera/impalaodbc/lib/32/libclouderaimpalaodbc32.s
o
[Cloudera ODBC Driver for Impala 64-bit]
Description=Cloudera ODBC Driver for Impala (64-bit)
Driver=/opt/cloudera/impalaodbc/lib/64/libclouderaimpalaodbc64.s
o
```

You can now connect to your data store by providing your application with a connection string where the `Driver` property is set to the driver name specified in the `odbcinst.ini` file, and all the other necessary connection properties are also set. For more information, see "DSN-less Connection String Examples" in "Using a Connection String" on page 44.

For instructions about configuring specific connection features, see the following:

- "Configuring Authentication on a Non-Windows Machine" on page 35
- "Configuring SSL Verification on a Non-Windows Machine" on page 38
- "Configuring Server-Side Properties on a Non-Windows Machine" on page 39

For detailed information about all the connection properties that the driver supports, see "Driver Configuration Options" on page 50.

## Configuring Authentication on a Non-Windows Machine

The Impala server supports multiple authentication mechanisms. You must determine the authentication type your server is using and configure your DSN accordingly. The available authentication methods are as follows:

- "Using No Authentication" on page 35
- "Using Kerberos" on page 36
- "Using Advanced Kerberos" on page 36
- "Using SASL User Name" on page 37
- "Using User Name And Password" on page 38

You can set the connection properties for authentication in a connection string or in a DSN (in the `odbc.ini` file). Settings in the connection string take precedence over settings in the DSN.

**Note:**

In addition to authentication, you can configure the driver to connect over the Secure Sockets Layer (SSL). For more information, see "Configuring SSL Verification on a Non-Windows Machine" on page 38.

### Using No Authentication

For this authentication mechanism, you do not need to configure any additional settings.

**Note:**

The default configuration of Impala requires the Cloudera ODBC Driver for Impala to be configured to use the No Authentication mechanism.

**To configure a connection without authentication:**

- Set the `AuthMech` connection attribute to `0`.

### Using Kerberos

Kerberos must be installed and configured before you can use this authentication mechanism. For more information, refer to the MIT Kerberos Documentation: http://web.mit.edu/kerberos/krb5-latest/doc/.

**To configure Kerberos authentication:**

1. Set the `AuthMech` connection attribute to `1`.
2. Choose one:
   - To use the default realm defined in your Kerberos setup, do not set the `KrbRealm` attribute.
   - Or, if your Kerberos setup does not define a default realm or if the realm of your Impala server is not the default, then set the appropriate realm using the `KrbRealm` attribute.
3. Set the `KrbFQDN` attribute to the fully qualified domain name of the Impala server host.

   **Note:**

   To use the Impala server host name as the fully qualified domain name for Kerberos authentication, set `KrbFQDN` to `_HOST`.
4. Set the `KrbServiceName` attribute to the service name of the Impala server.
5. Optionally, set the `TSaslTransportBufSize` attribute to the number of bytes to reserve in memory for buffering unencrypted data from the network.

   **Note:**

   In most circumstances, the default value of 1000 bytes is optimal.

### Using Advanced Kerberos

This authentication mechanism allows concurrent connections within the same process to use different Kerberos user principals.

When you use Advanced Kerberos authentication, you do not need to run the `kinit` command to obtain a Kerberos ticket. Instead, you use a JSON file to map your Impala user name to a Kerberos user principal name and a keytab that contains the corresponding keys. The driver obtains Kerberos tickets based on the specified mapping. As a fallback, you can specify a keytab that the driver uses by default if the mapping file is not available or if no matching keytab can be found in the mapping file.

**Note:**

- For information about the schema of the mapping file and how the driver handles invalid mappings, see "UPN Keytab Mapping File" on page 60.
- For information about how the driver searches for a keytab file if the keytab mapping and default keytab file are invalid, see "Default Keytab File" on page 52.

**To configure Advanced Kerberos authentication:**

1. Set the `AuthMech` connection attribute to `1`.
2. Choose one:
   - To use the default realm defined in your Kerberos setup, do not set the `KrbRealm` attribute.
   - Or, if your Kerberos setup does not define a default realm or if the realm of your Impala server is not the default, then set the appropriate realm using the `KrbRealm` attribute.
3. Set the `KrbFQDN` attribute to the fully qualified domain name of the Impala server host.

   **Note:**

   To use the Impala server host name as the fully qualified domain name for Kerberos authentication, set `KrbFQDN` to `_HOST`.
4. Set the `KrbServiceName` attribute to the service name of the Impala server.
5. Set the `UseKeytab` attribute to `1`.
6. Set the `UID` attribute to an appropriate user name for accessing the Impala server.
7. Set the `UPNKeytabMappingFile` attribute to the full path to a JSON file that maps your Impala user name to a Kerberos user principal name and a keytab file.
8. Set the `DefaultKeytabFile` attribute to the full path to a keytab file that the driver can use if the mapping file is not available or if no matching keytab can be found in the mapping file.
9. If the Impala server is configured to use SSL, then configure SSL for the connection. For more information, see "Configuring SSL Verification on a Non-Windows Machine" on page 38.
10. Optionally, set the `TSaslTransportBufSize` attribute to the number of bytes to reserve in memory for buffering unencrypted data from the network.

    **Note:**

    In most circumstances, the default value of 1000 bytes is optimal.

#### Using SASL User Name

This authentication mechanism requires a user name but does not require a password. The user name labels the session, facilitating database tracking.

**To configure SASL User Name authentication:**

1. Set the `AuthMech` connection attribute to `2`.
2. Set the `UID` attribute to an appropriate user name for accessing the Impala server.
3. Optionally, set the `TSaslTransportBufSize` attribute to the number of bytes to reserve in memory for buffering unencrypted data from the network.

> **Note:**
>
> In most circumstances, the default value of 1000 bytes is optimal.

#### Using User Name And Password

This authentication mechanism requires a user name and a password.

> **Note:**
>
> This authentication mechanism should not be used with an Impala configuration that does not have LDAP enabled.

**To configure User Name And Password authentication:**

1. Set the `AuthMech` connection attribute to `3`.
2. Set the `UID` attribute to an appropriate user name for accessing the Impala server.
3. Set the `PWD` attribute to the password corresponding to the user name you provided above.
4. Optionally, set the `TSaslTransportBufSize` attribute to the number of bytes to reserve in memory for buffering unencrypted data from the network.

> **Note:**
>
> In most circumstances, the default value of 1000 bytes is optimal.

5. Optionally, to use SASL to handle authentication, set the `UseSASL` attribute to `1`.

## Configuring SSL Verification on a Non-Windows Machine

If you are connecting to an Impala server that has Secure Sockets Layer (SSL) enabled, you can configure the driver to connect to an SSL-enabled socket.

You can set the connection properties described below in a connection string or in a DSN (in the `odbc.ini` file). Settings in the connection string take precedence over settings in the DSN.

**To configure SSL verification on a non-Windows machine:**

1. To enable SSL connections, set the `SSL` attribute to `1`.
2. To allow self-signed certificates from the server, set the `AllowSelfSignedServerCert` attribute to `1`.

3. To allow the common name of a CA-issued SSL certificate to not match the host name of the Impala server, set the `CAIssuedCertNamesMismatch` attribute to `1`.
4. Choose one:
   - To configure the driver to load SSL certificates from a specific PEM file when verifying the server, set the `TrustedCerts` attribute to the full path of the PEM file.
   - Or, to use the trusted CA certificates PEM file that is installed with the driver, do not specify a value for the `TrustedCerts` attribute.

## Configuring Server-Side Properties on a Non-Windows Machine

When connecting to a server that is running Impala 2.0 or later, you can use the driver to apply configuration properties to the Impala server.

You can set the connection properties described below in a connection string or in a DSN (in the `odbc.ini` file). Settings in the connection string take precedence over settings in the DSN.

**Important:**

This feature is not supported for earlier versions of Impala, where the SET statement can only be executed from within the Impala shell.

**To configure server-side properties on a non-Windows machine:**

1. To set a server-side property, use the syntax `SSP_[SSPKey]=[SSPValue]`, where *[SSPKey]* is the name of the server-side property and *[SSPValue]* is the value to specify for that property. For example, to set the MEM_LIMIT query option to 1 GB and the REQUEST_POOL query option to myPool, type the following in the `odbc.ini` file:

   ```
   SSP_MEM_LIMIT=1000000000
   SSP_REQUEST_POOL=myPool
   ```

   Or, to set those properties in a connection string, type the following:

   ```
   SSP_MEM_LIMIT={1000000000};SSP_REQUEST_POOL={myPool}
   ```

   **Note:**

   When setting a server-side property in a connection string, it is recommended that you enclose the value in braces (`{ }`) to make sure that special characters can be properly escaped.

2. To disable the driver's default behavior of converting server-side property key names to all lower-case characters, set the `LCaseSspKeyName` property to `0`.

## Configuring Logging Options

To help troubleshoot issues, you can enable logging in the driver.

**Important:**

Only enable logging long enough to capture an issue. Logging decreases performance and can consume a large quantity of disk space.

Logging is configured through driver-wide settings in the `cloudera.impalaodbc.ini` file, which apply to all connections that use the driver.

**To enable logging:**

1. Open the `cloudera.impalaodbc.ini` configuration file in a text editor.
2. To specify the level of information to include in log files, set the `LogLevel` property to one of the following numbers:

| LogLevel Value | Description |
|---|---|
| 0 | Disables all logging. |
| 1 | Logs severe error events that lead the driver to abort. |
| 2 | Logs error events that might allow the driver to continue running. |
| 3 | Logs events that might result in an error if action is not taken. |
| 4 | Logs general information that describes the progress of the driver. |
| 5 | Logs detailed information that is useful for debugging the driver. |
| 6 | Logs all driver activity. |

3. Set the `LogPath` key to the full path to the folder where you want to save log files.
4. Set the `LogFileCount` key to the maximum number of log files to keep.

   **Note:**

   After the maximum number of log files is reached, each time an additional file is created, the driver deletes the oldest log file.

5. Set the `LogFileSize` key to the maximum size of each log file in megabytes (MB).

   **Note:**

   After the maximum file size is reached, the driver creates a new file and continues logging.

6. Save the `cloudera.impalaodbc.ini` configuration file.
7. Restart your ODBC application to make sure that the new settings take effect.

The Cloudera ODBC Driver for Impala produces a log file named `ImpalaODBC_driver.log` at the location you specify using the `LogPath` key.

**To disable logging:**

1. Open the `cloudera.impalaodbc.ini` configuration file in a text editor.
2. Set the `LogLevel` key to `0`.
3. Save the `cloudera.impalaodbc.ini` configuration file.
4. Restart your ODBC application to make sure that the new settings take effect.

## Testing the Connection

To test the connection, you can use an ODBC-enabled client application. For a basic connection test, you can also use the test utilities that are packaged with your driver manager installation. For example, the iODBC driver manager includes simple utilities called iodbctest and iodbctestw. Similarly, the unixODBC driver manager includes simple utilities called isql and iusql.

### Using the iODBC Driver Manager

You can use the iodbctest and iodbctestw utilities to establish a test connection with your driver. Use iodbctest to test how your driver works with an ANSI application, or use iodbctestw to test how your driver works with a Unicode application.

**Note:**

There are 32-bit and 64-bit installations of the iODBC driver manager available. If you have only one or the other installed, then the appropriate version of iodbctest (or iodbctestw) is available. However, if you have both 32- and 64-bit versions installed, then you need to make sure that you are running the version from the correct installation directory.

For more information about using the iODBC driver manager, see http://www.iodbc.org.

**To test your connection using the iODBC driver manager:**

1. Run **iodbctest** or **iodbctestw**.
2. Optionally, if you do not remember the DSN, then type a question mark (`?`) to see a list of available DSNs.
3. Type the connection string for connecting to your data store, and then press ENTER. For more information, see "Using a Connection String" on page 44.

If the connection is successful, then the `SQL>` prompt appears.

### Using the unixODBC Driver Manager

You can use the isql and iusql utilities to establish a test connection with your driver and your DSN. isql and iusql can only be used to test connections that use a DSN. Use isql to test how your driver works with an ANSI application, or use iusql to test how your driver works with a Unicode application.

**Note:**

There are 32-bit and 64-bit installations of the unixODBC driver manager available. If you have only one or the other installed, then the appropriate version of isql (or iusql) is available. However, if you have both 32- and 64-bit versions installed, then you need to make sure that you are running the version from the correct installation directory.

For more information about using the unixODBC driver manager, see http://www.unixodbc.org.

**To test your connection using the unixODBC driver manager:**

- Run isql or iusql by using the corresponding syntax:
  - `isql` *`[DataSourceName]`*
  - `iusql` *`[DataSourceName]`*

  *`[DataSourceName]`* is the DSN that you are using for the connection.

If the connection is successful, then the `SQL>` prompt appears.

**Note:**

For information about the available options, run isql or iusql without providing a DSN.

# Authentication Options

Impala supports multiple authentication mechanisms. You must determine the authentication type that your server is using. The authentication methods available in the Cloudera ODBC Driver for Impala are as follows:

- No Authentication
- Kerberos
- SASL User Name
- User Name And Password

**Note:**

- The default configuration of Impala requires the Cloudera ODBC Driver for Impala to be configured to use the No Authentication mechanism.
- In addition to regular Kerberos authentication, the driver also supports an advanced configuration of Kerberos authentication that allows concurrent connections within the same process to use different Kerberos user principals.

In addition to authentication, you can configure the driver to connect over SSL or use SASL to handle authentication.

The Impala server uses SASL (Simple Authentication and Security Layer) to support some of the authentication methods. Kerberos is supported with the SASL GSSAPI mechanism. SASL User Name and User Name And Password (with SASL enabled) are supported with the SASL PLAIN mechanism.

| SASL mechanisms | Non-SASL mechanisms |
|---|---|
| • Kerberos<br>• SASL User Name<br>• User Name And Password (with SASL enabled) | • No Authentication<br>• User Name And Password (without SASL enabled) |

**Note:**

Thrift (the layer for handling remote process communication between the Cloudera ODBC Driver for Impala and the Impala server) has a limitation where it cannot detect a mix of non-SASL and SASL mechanisms being used between the driver and the server. If this happens, the driver will appear to hang during connection establishment.

# Using a Connection String

For some applications, you might need to use a connection string to connect to your data source. For detailed information about how to use a connection string in an ODBC application, refer to the documentation for the application that you are using.

The connection strings in the following sections are examples showing the minimum set of connection attributes that you must specify to successfully connect to the data source. Depending on the configuration of the data source and the type of connection you are working with, you might need to specify additional connection attributes. For detailed information about all the attributes that you can use in the connection string, see "Driver Configuration Options" on page 50.

## DSN Connection String Example

The following is an example of a connection string for a connection that uses a DSN:

```
DSN=[DataSourceName];
```

*[DataSourceName]* is the DSN that you are using for the connection.

You can set additional configuration options by appending key-value pairs to the connection string. Configuration options that are passed in using a connection string take precedence over configuration options that are set in the DSN.

## DSN-less Connection String Examples

Some applications provide support for connecting to a data source using a driver without a DSN. To connect to a data source without using a DSN, use a connection string instead.

The placeholders in the examples are defined as follows, in alphabetical order:

- *[DomainName]* is the fully qualified domain name of the Impala server host.
- *[MappingFile]* is the full path to a JSON file that maps your Impala user name to a Kerberos user principal name and a keytab file.
- *[PortNumber]* is the number of the TCP port that the Impala server uses to listen for client connections.
- *[Realm]* is the Kerberos realm of the Impala server host.
- *[Server]* is the IP address or host name of the Impala server to which you are connecting.
- *[ServiceName]* is the Kerberos service principal name of the Impala server.
- *[YourPassword]* is the password corresponding to your user name.
- *[YourUserName]* is the user name that you use to access the Impala server.

### Connecting to an Impala Server Without Authentication

The following is the format of a DSN-less connection string that connects to an Impala server that does not require authentication:

```
Driver=Cloudera ODBC Driver for Impala;Host=[Server];
```

```
Port=[PortNumber];
```

For example:

```
Driver=Cloudera ODBC Driver for Impala;Host=192.168.222.160;
Port=21050;
```

If you are connecting to the server through SSL, then set the SSL property to 1. For example:

```
Driver=Cloudera ODBC Driver for Impala;Host=192.168.222.160;
Port=21050;SSL=1;
```

**Connecting to an Impala Server that Requires Kerberos Authentication**

The following is the format of a DSN-less connection string that connects to an Impala server requiring Kerberos authentication:

```
Driver=Cloudera ODBC Driver for Impala;Host=[Server];
Port=[PortNumber];AuthMech=1;KrbRealm=[Realm];
KrbFQDN=[DomainName];KrbServiceName=[ServiceName];
```

For example:

```
Driver=Cloudera ODBC Driver for Impala;Host=192.168.222.160;
Port=21050;AuthMech=1;KrbRealm=CLOUDERA;
KrbFQDN=localhost.localdomain;KrbServiceName=impala;
```

If you are connecting to the server through SSL, then set the SSL property to 1. For example:

```
Driver=Cloudera ODBC Driver for Impala;Host=192.168.222.160;
Port=21050;AuthMech=1;KrbRealm=CLOUDERA;
KrbFQDN=localhost.localdomain;KrbServiceName=impala;SSL=1;
```

**Connecting to an Impala Server using Advanced Kerberos Authentication**

The following is the format of a DSN-less connection string that connects to an Impala server using Advanced Kerberos authentication:

```
Driver=Cloudera ODBC Driver for Impala;Host=[Server];
Port=[PortNumber];AuthMech=1;KrbRealm=[Realm];
KrbFQDN=[DomainName];KrbServiceName=[ServiceName];
UseKeytab=1;UID=[YourUserName];
UPNKeytabMappingFile=[MappingFile];
```

For example:

```
Driver=Cloudera ODBC Driver for Impala;Host=192.168.222.160;
Port=21050;AuthMech=1;KrbRealm=CLOUDERA;
KrbFQDN=localhost.localdomain;KrbServiceName=impala;
UseKeytab=1;UID=cloudera;
UPNKeytabMappingFile=C:\Temp\cloudera.keytab;
```

If you are connecting to the server through SSL, then set the SSL property to 1. For example:

```
Driver=Cloudera ODBC Driver for Impala;Host=192.168.222.160;
Port=21050;AuthMech=1;KrbRealm=CLOUDERA;
KrbFQDN=localhost.localdomain;KrbServiceName=impala;
UseKeytab=1;UID=cloudera;
UPNKeytabMappingFile=C:\Temp\cloudera.keytab;SSL=1;
```

**Connecting to an Impala Server that Requires User Name Authentication**

The following is the format of a DSN-less connection string that connects to an Impala server requiring User Name authentication. By default, the driver uses **anonymous** as the user name.

```
Driver=Cloudera ODBC Driver for Impala;Host=[Server];
Port=[PortNumber];AuthMech=2;
```

For example:

```
Driver=Cloudera ODBC Driver for Impala;Host=192.168.222.160;
Port=21050;AuthMech=2;
```

If you are connecting to the server through SSL, then set the SSL property to 1. For example:

```
Driver=Cloudera ODBC Driver for Impala;Host=192.168.222.160;
Port=21050;AuthMech=2;SSL=1;
```

**Connecting to a Impala Server that Requires User Name And Password Authentication**

The following is the format of a DSN-less connection string that connects to an Impala server requiring User Name And Password authentication:

```
Driver=Cloudera ODBC Driver for Impala;Host=[Server];
Port=[PortNumber];AuthMech=3;UID=[YourUserName];
PWD=[YourPassword];
```

For example:

```
Driver=Cloudera ODBC Driver for Impala;Host=192.168.222.160;
Port=21050;AuthMech=3;UID=cloudera;PWD=cloudera;
```

If you are connecting to the server through SSL, then set the SSL property to 1. For example:

```
Driver=Cloudera ODBC Driver for Impala;Host=192.168.222.160;
Port=21050;AuthMech=3;UID=cloudera;PWD=cloudera;SSL=1;
```

# Features

For more information on the features of the Cloudera ODBC Driver for Impala, see the following:



## Data Types

The Cloudera ODBC Driver for Impala supports many common data formats, converting between Impala data types and SQL data types.

The table below lists the supported data type mappings.

| Impala Type | SQL Type |
|---|---|
| ARRAY | SQL_VARCHAR |
| BIGINT | SQL_BIGINT |
| BINARY | SQL_VARBINARY |
| BOOLEAN | SQL_BOOLEAN |
| CHAR<br><br>**Note:**<br>Only available in CDH 5.2 or later. | SQL_CHAR<br><br>**Note:**<br>SQL_WCHAR is returned instead if the Use SQL Unicode Types configuration option (the `UseUnicodeSqlCharacterTypes` key) is enabled. |
| DATE | SQL_DATE |
| DECIMAL<br><br>**Note:**<br>Only available in CDH 5.2 or later. | SQL_DECIMAL |

| Impala Type | SQL Type |
|---|---|
| DOUBLE<br><br>**Note:**<br>REAL is an alias for DOUBLE. | SQL_DOUBLE |
| FLOAT | SQL_REAL |
| INT | SQL_INTEGER |
| MAP | SQL_VARCHAR |
| SMALLINT | SQL_SMALLINT |
| STRUCT | SQL_VARCHAR |
| TIMESTAMP | SQL_TIMESTAMP |
| TINYINT | SQL_TINYINT |
| VARCHAR<br><br>**Note:**<br>Only available in CDH 5.2 or later. | SQL_VARCHAR<br><br>**Note:**<br>SQL_WVARCHAR is returned instead if the Use SQL Unicode Types configuration option (the `UseUnicodeSqlCharacterTypes` key) is enabled. |

## Catalog and Schema Support

The Cloudera ODBC Driver for Impala supports both catalogs and schemas to make it easy for the driver to work with various ODBC applications. Since Impala only organizes tables into schemas/databases, the driver provides a synthetic catalog named IMPALA under which all of the schemas/databases are organized. The driver also maps the ODBC schema to the Impala schema/database.

## SQL Translation

The Cloudera ODBC Driver for Impala can parse queries locally before sending them to the Impala server. This feature allows the driver to calculate query metadata without executing the query, support query parameters, and support extra SQL features such as ODBC escape sequences and additional scalar functions that are not available in the Impala-shell tool.

**Note:**

The driver does not support translation for queries that reference a field contained in a nested column (an ARRAY, MAP, or STRUCT column). To retrieve data from a nested column, make sure that the query is written in valid Impala SQL syntax.

## Server-Side Properties

The Cloudera ODBC Driver for Impala allows you to set server-side properties via a DSN. Server-side properties specified in a DSN affect only the connection that is established using the DSN.

For more information about setting server-side properties when using the Windows driver, see "Configuring Server-Side Properties on Windows" on page 15. For information about setting server-side properties when using the driver on a non-Windows platform, see "Configuring Server-Side Properties on a Non-Windows Machine" on page 39.

## Active Directory

The Cloudera ODBC Driver for Impala supports Active Directory Kerberos on Windows. There are two prerequisites for using Active Directory Kerberos on Windows:

- MIT Kerberos is not installed on the client Windows machine.
- The MIT Kerberos Hadoop realm has been configured to trust the Active Directory realm, according to Cloudera's documentation, so that users in the Active Directory realm can access services in the MIT Kerberos Hadoop realm.

# Driver Configuration Options

Driver Configuration Options lists the configuration options available in the Cloudera ODBC Driver for Impala alphabetically by field or button label. Options having only key names, that is, not appearing in the user interface of the driver, are listed alphabetically by key name.

When creating or configuring a connection from a Windows machine, the fields and buttons are available in the following dialog boxes:

- Cloudera ODBC Driver for Impala DSN Setup
- Advanced Options
- Keytab Options
- Server Side Properties
- SSL Options

When using a connection string or configuring a connection from a Linux/Mac OS X/AIX/Debian machine, use the key names provided.

## Configuration Options Appearing in the User Interface

The following configuration options are accessible via the Windows user interface for the Cloudera ODBC Driver for Impala, or via the key name when using a connection string or configuring a connection from a Linux/Mac OS X/AIX/Debian machine:

- "Password" on page 57
- "Port" on page 57

### Allow Common Name Host Name Mismatch

| Key Name | Default Value | Required |
|---|---|---|
| AllowHostNameCNMismatch | Clear (0) | No |

#### Description

This option specifies whether a CA-issued SSL certificate name must match the host name of the Impala server.

**Note:**

The key for this option used to be CAIssuedCertNamesMismatch, and will still be recognized by the driver under that key. If both keys are defined in the Windows registry or .ini file, AllowHostNameCNMismatch will take precedence.

- Enabled (1): The driver allows a CA-issued SSL certificate name to not match the host name of the Impala server.
- Disabled (0): The CA-issued SSL certificate name must match the host name of the Impala server.

**Note:**

This setting is applicable only when SSL is enabled.

### Allow Self-Signed Server Certificate

| Key Name | Default Value | Required |
|---|---|---|
| AllowSelfSignedServerCert | Clear (0) | No |

#### Description

This option specifies whether the driver allows self-signed certificates from the server.

- Enabled (1): The driver authenticates the Impala server even if the server is using a self-signed certificate.
- Disabled (0): The driver does not allow self-signed certificates from the server.

**Note:**

This setting is applicable only when SSL is enabled.

**Convert Key Name to Lower Case**

| Key Name | Default Value | Required |
|---|---|---|
| LCaseSspKeyName | Selected (1) | No |

**Description**

This option specifies whether the driver converts server-side property key names to all lower-case characters.

- Enabled (1): The driver converts server-side property key names to all lower-case characters.
- Disabled (0): The driver does not modify the server-side property key names.

**Database**

| Key Name | Default Value | Required |
|---|---|---|
| Schema | default | No |

**Description**

The name of the database schema to use when a schema is not explicitly specified in a query. You can still issue queries on other schemas by explicitly specifying the schema in the query.

**Note:**

To inspect your databases and determine the appropriate schema to use, at the Impala command prompt, type `show databases`.

**Default Keytab File**

| Key Name | Default Value | Required |
|---|---|---|
| DefaultKeytabFile | None | No |

**Description**

The full path to the keytab file that the driver uses to obtain the ticket for Kerberos authentication.

**Note:**

- This option is applicable only when the authentication mechanism is set to Kerberos (AuthMech=1) and the Use Keytab option is enabled (UseKeytab=1).
- If the UPN Keytab Mapping File option (the UPNKeytabMappingFile key) is set to a JSON file with a valid mapping to a keytab, then that keytab takes precedence.

If you do not set this option but the Use Keytab option is enabled (`UseKeytab=1`), then the MIT Kerberos library will search for a keytab using the following search order:

- The file specified by the KRB5_KTNAME environment variable.
- The `default_keytab_name` setting in the `[libdefaults]` section of the Kerberos configuration file (`krb5.conf/krb5.ini`).
- The default keytab file specified in the MIT Kerberos library. Typically, the default file is `C:\Windows\krb5kt` for Windows platforms and `/etc/krb5.keytab` for non-Windows platforms.

**Delegation UID**

| Key Name | Default Value | Required |
|---|---|---|
| DelegationUID | None | No |

**Description**

If a value is specified for this setting, the driver delegates all operations against Impala to the specified user, rather than to the authenticated user for the connection.

**Enable Auto Reconnect**

| Key Name | Default Value | Required |
|---|---|---|
| AutoReconnect | 1 | Yes |

**Description**

Controls whether or not the driver will attempt to automatically reconnect to the server when a communication link error occurs.

- `0`: Set this to not attempt to reconnect.
- `1`: Set this to attempt to reconnect.

**Enable Simulated Transactions**

| Key Name | Default Value | Required |
|---|---|---|
| EnableSimulatedTransactions | Clear (0) | No |

**Description**

This option specifies whether the driver should simulate transactions, or return an error.

- Enabled (`1`): The driver simulates transactions, enabling queries that contain transaction statements to be run successfully. The transactions are not executed.
- Disabled (`0`): The driver returns an error if it attempts to run a query that contains transaction statements.

**Note:**

ODBC does not support transaction statements, so they cannot be executed.

**Enable SSL**

| Key Name | Default Value | Required |
|---|---|---|
| SSL | Clear (0) | No |

**Description**

This option specifies whether the client uses an SSL encrypted connection to communicate with the Impala.

- Enabled (1): The client communicates with the Impala using SSL.
- Disabled (0): SSL is disabled.

SSL is configured independently of authentication. When authentication and SSL are both enabled, the driver performs the specified authentication method over an SSL connection.

**Host**

| Key Name | Default Value | Required |
|---|---|---|
| Host | None | Yes |

**Description**

The IP address or host name of the Impala server.

**Host FQDN**

| Key Name | Default Value | Required |
|---|---|---|
| KrbFQDN | _HOST | No. |

**Description**

The fully qualified domain name of the Impala host.

When the value of Host FQDN is _HOST, the driver uses the Impala server host name as the fully qualified domain name for Kerberos authentication.

**Log Level**

| Key Name | Default Value | Required |
|---|---|---|
| LogLevel | OFF (0) | No |

#### Description

Use this property to enable or disable logging in the driver and to specify the amount of detail included in log files.

**Important:**

- Only enable logging long enough to capture an issue. Logging decreases performance and can consume a large quantity of disk space.
- This option is not supported in connection strings. To configure logging for the Windows driver, you must use the Logging Options dialog box. To configure logging for a non-Windows driver, you must use the `cloudera.impalaodbc.ini` file.

Set the property to one of the following values:

- OFF (`0`): Disable all logging.
- FATAL (`1`): Logs severe error events that lead the driver to abort.
- ERROR (`2`): Logs error events that might allow the driver to continue running.
- WARNING (`3`): Logs events that might result in an error if action is not taken.
- INFO (`4`): Logs general information that describes the progress of the driver.
- DEBUG (`5`): Logs detailed information that is useful for debugging the driver.
- TRACE (`6`): Logs all driver activity.

When logging is enabled, the driver produces a log file named `ImpalaODBC_driver.log` in the location specified in the Log Path (`LogPath`) property.

### Log Path

| Key Name | Default Value | Required |
|---|---|---|
| `LogPath` | None | Yes, if logging is enabled. |

#### Description

The full path to the folder where the driver saves log files when logging is enabled.

**Important:**

This option is not supported in connection strings. To configure logging for the Windows driver, you must use the Logging Options dialog box. To configure logging for a non-Windows driver, you must use the `cloudera.impalaodbc.ini` file.

### Max File Size

| Key Name | Default Value | Required |
|---|---|---|
| `LogFileSize` | `20` | No |

#### Description

The maximum size of each log file in megabytes (MB). After the maximum file size is reached, the driver creates a new file and continues logging.

**Important:**

This option is not supported in connection strings. To configure logging for the Windows driver, you must use the Logging Options dialog box. To configure logging for a non-Windows driver, you must use the `cloudera.impalaodbc.ini` file.

### Max Number Files

| Key Name | Default Value | Required |
|---|---|---|
| `LogFileCount` | `50` | No |

#### Description

The maximum number of log files to keep. After the maximum number of log files is reached, each time an additional file is created, the driver deletes the oldest log file.

**Important:**

This option is not supported in connection strings. To configure logging for the Windows driver, you must use the Logging Options dialog box. To configure logging for a non-Windows driver, you must use the `cloudera.impalaodbc.ini` file.

### Mechanism

| Key Name | Default Value | Required |
|---|---|---|
| `AuthMech` | No Authentication (`0`) | No |

#### Description

The authentication mechanism to use.

Select one of the following settings, or set the key to the corresponding number:

- No Authentication (`0`)
- Kerberos (`1`)
- SASL User Name (`2`)
- User Name And Password (`3`)

**Password**

| Key Name | Default Value | Required |
|---|---|---|
| PWD | None | Yes, if the authentication mechanism is User Name And Password (3). |

**Description**

The password corresponding to the user name that you provided in the User Name field (the UID key).

**Port**

| Key Name | Default Value | Required |
|---|---|---|
| Port | 21050 | Yes |

**Description**

The number of the TCP port that the Impala server uses to listen for client connections.

**Realm**

| Key Name | Default Value | Required |
|---|---|---|
| KrbRealm | NULL | No |

**Description**

The realm of the Impala host.

If your Kerberos configuration already defines the realm of the Impala host as the default realm, then you do not need to configure this option.

**Rows Fetched Per Block**

| Key Name | Default Value | Required |
|---|---|---|
| RowsFetchedPerBlock | 10000 | No |

**Description**

The maximum number of rows that a query returns at a time.

Valid values for this setting include any positive 32-bit integer. However, testing has shown that performance gains are marginal beyond the default value of 10000 rows.

### Save Password (Encrypted)

| Key Name | Default Value | Required |
|---|---|---|
| N/A | Selected | No |

#### Description

This option specifies whether the password is saved in the registry.

- Enabled: The password is saved in the registry.
- Disabled: The password is not saved in the registry.

This option is available only in the Windows driver. It appears in the Cloudera ODBC Driver for Impala DSN Setup dialog box.

**Important:**

The password is obscured (not saved in plain text). However, it is still possible for the encrypted password to be copied and used.

### Service Name

| Key Name | Default Value | Required |
|---|---|---|
| KrbServiceName | impala | No. |

#### Description

The Kerberos service principal name of the Impala server.

### Socket Timeout

| Key Name | Default Value | Required |
|---|---|---|
| SocketTimeout | 0 | No |

#### Description

The number of seconds after which Impala closes the connection with the client application if the connection is idle.

When this option is set to 0, the connection does not time out.

### String Column Length

| Key Name | Default Value | Required |
|---|---|---|
| StringColumnLength | 32767 | No |

**Description**

The maximum number of characters that can be contained in STRING columns.

**Transport Buffer Size**

| Key Name | Default Value | Required |
|---|---|---|
| TSaslTransportBufSize | 1000 | No |

**Description**

The number of bytes to reserve in memory for buffering unencrypted data from the network.

**Note:**

In most circumstances, the default value of 1000 bytes is optimal.

**Trusted Certificates**

| Key Name | Default Value | Required |
|---|---|---|
| TrustedCerts | The cacerts.pem file in the \lib subfolder within the driver's installation directory. The exact file path varies depending on the version of the driver that is installed. For example, the path for the Windows driver is different from the path for the Mac OS X driver. | No |

**Description**

The full path of the .pem file containing trusted CA certificates for verifying the server when using SSL.

If this option is not set, then the driver defaults to using the trusted CA certificates .pem file installed by the driver.

**Note:**

This setting is applicable only when SSL is enabled.

### UPN Keytab Mapping File

| Key Name | Default Value | Required |
|---|---|---|
| UPNKeytabMappingFile | None | No |

#### Description

The full path to a JSON file that maps your Impala user name to a Kerberos user principal name and a keytab file.

**Note:**

This option is applicable only when the authentication mechanism is set to Kerberos (AuthMech=1) and the Use Keytab option is enabled (UseKeytab=1).

The mapping in the JSON file must be written using the following schema, where *[UserName]* is the Impala user name, *[KerberosUPN]* is the Kerberos user principal name, and *[KeytabFile]* is the full path to the keytab file:

```
{
     "[UserName]": {
          "principal" : "[KerberosUPN]",
          "keytabfile": "[KeytabFile]"
     },
... }
```

For example, the following file maps the Impala user name **cloudera** to the **cloudera@CLOUDERA** Kerberos user principal name and the C:\Temp\cloudera.keytab file:

```
{
     "cloudera": {
          "principal" : "cloudera@CLOUDERA",
          "keytabfile": "C:\Temp\cloudera.keytab"
     },
... }
```

If parts of the mapping are invalid or not defined, then the following occurs:

- If the mapping file fails to specify a Kerberos user principal name, then the driver uses the Impala user name as the Kerberos user principal name.
- If the mapping file fails to specify a keytab file, then the driver uses the keytab file that is specified in the Default Keytab File setting.
- If the entire mapping file is invalid or not defined, then the driver does both of the actions described above.

### Use Keytab

| Key Name | Default Value | Required |
|---|---|---|
| UseKeytab | Clear (0) | No |

#### Description

This option specifies whether the driver obtains the ticket for Kerberos authentication by using a keytab.

- Enabled (1): The driver uses a keytab to obtain a ticket before authenticating the connection using Kerberos.
- Disabled (0): The driver does not attempt to obtain the Kerberos ticket, and assumes that a valid ticket is already available in the credentials cache.

**Note:**

This option is applicable only when the authentication mechanism is set to Kerberos (AuthMech=1).

If you enable this option but do not set the Default Keytab File option (the DefaultKeytabFile key), then the MIT Kerberos library will search for a keytab file using the following search order:

1. The file specified by the KRB5_KTNAME environment variable.
2. The default_keytab_name setting in the [libdefaults] section of the Kerberos configuration file (krb5.conf/krb5.ini).
3. The default keytab file specified in the MIT Kerberos library. Typically, the default file is C:\Windows\krb5kt for Windows platforms.

### Use Native Query

| Key Name | Default Value | Required |
|---|---|---|
| UseNativeQuery | Clear (0) | No |

#### Description

This option specifies whether the driver uses native queries, or converts them into an equivalent form in .

- Enabled (1): The driver does not transform the queries emitted by an application, and executes queries directly.
- Disabled (0): The driver transforms the queries emitted by an application and converts them into an equivalent form in .

**Note:**

If the application is Impala-aware and already emits , then enable this option to avoid the extra overhead of query transformation.

### Use Only SSPI Plugin

| Key Name | Default Value | Required |
|---|---|---|
| UseOnlySSPI | Clear (0) | No |

#### Description

This option specifies how the driver handles Kerberos authentication: either with the SSPI plugin or with MIT Kerberos.

- Enable For This DSN (1 in the DSN entry in the registry): The driver handles Kerberos authentication in the DSN connection by using the SSPI plugin instead of MIT Kerberos by default.
- Enable For DSN-less Connections (1 in the driver configuration section of the registry): The driver handles Kerberos authentication in DSN-less connections by using the SSPI plugin instead of MIT Kerberos by default.

  If you want all connections that use the Cloudera ODBC Driver for Impala to use the SSPI plugin by default, then enable Use Only SSPI Plugin for both DSN and DSN-less connections.
- Disabled (0): The driver uses MIT Kerberos to handle Kerberos authentication, and only uses the SSPI plugin if the gssapi library is not available.

**Important:**

This option is available only in the Windows driver.

### Use Simple Authentication and Security Layer (SASL)

| Key Name | Default Value | Required |
|---|---|---|
| UseSASL | 0 if using No Authentication.<br><br>1 if using User Name And Password or Kerberos or SASL User Name authentication. | No |

#### Description

This option specifies whether the driver uses SASL to handle authentication.

- Enabled (1): The driver uses SASL to handle authentication.
- Disabled (0): The driver does not use SASL.

This option is configurable only when you are using the User Name And Password authentication mechanism. If the driver is configured to use the other authentication mechanisms, then it uses the default setting for the Use Simple Authentication and Security Layer (SASL) option.

### Use SQL Unicode Types

| Key Name | Default Value | Required |
|---|---|---|
| UseSQLUnicodeTypes | Clear (0) | No |

#### Description

This option specifies the SQL types to be returned for string data types.

- Enabled (1): The driver returns SQL_WVARCHAR for STRING and VARCHAR columns, and returns SQL_WCHAR for CHAR columns.
- Disabled (0): The driver returns SQL_VARCHAR for STRING and VARCHAR columns, and returns SQL_CHAR for CHAR columns.

### Use System Trust Store

| Key Name | Default Value | Required |
|---|---|---|
| UseSystemTrustStore | 0 | Yes |

#### Description

Controls whether to use the CA certificates in the Windows system trust store to verify the server's certificate for SSL connection.

- 0: Set this to not use the Windows system trust store.
- 1: Set this to use the Windows system trust store.

### User Name

| Key Name | Default Value | Required |
|---|---|---|
| UID | For User Name (2) authentication only, the default value is anonymous | Yes, if the authentication mechanism is User Name And Password (3).<br>No, if the authentication mechanism is SASL User Name (2). |

#### Description

The user name that you use to access the Impala server.

## Configuration Options Having Only Key Names

The following configuration options do not appear in the Windows user interface for the Cloudera ODBC Driver for Impala. They are accessible only when you use a connection string or configure a connection from a Linux/Mac OS X/AIX/Debian machine:


- "Driver" on page 64
- "SSP_" on page 64


**Driver**

| Key Name | Default Value | Required |
|---|---|---|
| Driver | Cloudera ODBC Driver for Impala when installed on Windows, or the absolute path of the driver shared object file when installed on a non-Windows machine. | Yes |

**Description**

On Windows, the name of the installed driver(`Cloudera ODBC Driver for Impala`).

On other platforms, the name of the installed driver as specified in `odbcinst.ini`, or the absolute path of the driver shared object file.

**SSP_**

| Key Name | Default Value | Required |
|---|---|---|
| SSP_ | None | No |

**Description**

Set a server-side property by using the following syntax, where *[SSPKey]* is the name of the server-side property and *[SSPValue]* is the value for that property:

```
SSP_[SSPKey]=[SSPValue]
```

For example:

```
SSP_MEM_LIMIT=1000000000
SSP_REQUEST_POOL=myPool
```

Or, to set those properties in a connection string, type the following:

```
SSP_MEM_LIMIT={1000000000};SSP_REQUEST_POOL={myPool}
```

After the driver applies the server-side property, the `SSP_` prefix is removed from the DSN entry, leaving an entry of *[SSPKey]=[SSPValue]*.

**Important:**

This property is supported only for connections to Impala 2.0 or later. In earlier versions of Impala, the SET statement can only be executed from within the Impala shell.

**Note:**

- The `SSP_` prefix must be upper case.
- When setting a server-side property in a connection string, it is recommended that you enclose the value in braces (`{ }`) to make sure that special characters can be properly escaped.

# ODBC API Conformance Level

The following table lists the ODBC interfaces that the Cloudera ODBC Driver for Impala implements and the ODBC compliance level of each interface.

ODBC compliance levels are Core, Level 1, and Level 2. These compliance levels are defined in the ODBC Specification published with the Interface SDK from Microsoft.

Interfaces include both the Unicode and non-Unicode versions. For more information, see "Unicode Function Arguments" in the *ODBC Programmer's Reference*: http://msdn.microsoft.com/en-us/library/ms716246%28VS.85%29.aspx.

| Conformance Level | INTERFACES | | Conformance Level | INTERFACES |
|---|---|---|---|---|
| Core | SQLAllocHandle | | Core | SQLGetStmtAttr |
| Core | SQLBindCol | | Core | SQLGetTypeInfo |
| Core | SQLBindParameter | | Core | SQLNativeSql |
| Core | SQLCancel | | Core | SQLNumParams |
| Core | SQLCloseCursor | | Core | SQLNumResultCols |
| Core | SQLColAttribute | | Core | SQLParamData |
| Core | SQLColumns | | Core | SQLPrepare |
| Core | SQLConnect | | Core | SQLPutData |
| Core | SQLCopyDesc | | Core | SQLRowCount |
| Core | SQLDescribeCol | | Core | SQLSetConnectAttr |
| Core | SQLDisconnect | | Core | SQLSetCursorName |

| Conformance Level | INTERFACES | | Conformance Level | INTERFACES |
|---|---|---|---|---|
| Core | SQLDriverconnect | | Core | SQLSetDescField |
| Core | SQLEndTran | | Core | SQLSetDescRec |
| Core | SQLExecDirect | | Core | SQLSetEnvAttr |
| Core | SQLExecute | | Core | SQLSetStmtAttr |
| Core | SQLFetch | | Core | SQLSpecialColumns |
| Core | SQLFetchScroll | | Core | SQLStatistics |
| Core | SQLFreeHandle | | Core | SQLTables |
| Core | SQLFreeStmt | | Core | SQLBrowseConnect |
| Core | SQLGetConnectAttr | | Core | SQLPrimaryKeys |
| Core | SQLGetCursorName | | Core | SQLGetInfo |
| Core | SQLGetData | | Level 1 | SQLProcedureColumns |
| Core | SQLGetDescField | | Level 1 | SQLProcedures |
| Core | SQLGetDescRec | | Level 2 | SQLColumnPrivileges |
| Core | SQLGetDiagField | | Level 2 | SQLDescribeParam |
| Core | SQLGetDiagRec | | Level 2 | SQLForeignKeys |

| Conformance Level | INTERFACES | | Conformance Level | INTERFACES |
|---|---|---|---|---|
| Core | SQLGetEnvAttr | | Level 2 | SQLTablePrivileges |
| Core | SQLGetFunctions | | | |

# Contact Us

If you are having difficulties using the driver, our Community Forum may have your solution. In addition to providing user to user support, our forums are a great place to share your questions, comments, and feature requests with us.

If you are a Subscription customer you may also use the Cloudera Support Portal to search the Knowledge Base or file a Case.

**Important:**

To help us assist you, prior to contacting Cloudera Support please prepare a detailed summary of the client and server environment including operating system version, patch level, and configuration