

cloudera®

Cloudera ODBC Driver for Impala

Important Notice

© 2010-2019 Cloudera, Inc. All rights reserved.

Cloudera, the Cloudera logo, and any other product or service names or slogans contained in this document, except as otherwise disclaimed, are trademarks of Cloudera and its suppliers or licensors, and may not be copied, imitated or used, in whole or in part, without the prior written permission of Cloudera or the applicable trademark holder.

Hadoop and the Hadoop elephant logo are trademarks of the Apache Software Foundation. All other trademarks, registered trademarks, product names and company names or logos mentioned in this document are the property of their respective owners. Reference to any products, services, processes or other information, by trade name, trademark, manufacturer, supplier or otherwise does not constitute or imply endorsement, sponsorship or recommendation thereof by us.

Complying with all applicable copyright laws is the responsibility of the user. Without limiting the rights under copyright, no part of this document may be reproduced, stored in or introduced into a retrieval system, or transmitted in any form or by any means (electronic, mechanical, photocopying, recording, or otherwise), or for any purpose, without the express written permission of Cloudera.

Cloudera may have patents, patent applications, trademarks, copyrights, or other intellectual property rights covering subject matter in this document. Except as expressly provided in any written license agreement from Cloudera, the furnishing of this document does not give you any license to these patents, trademarks copyrights, or other intellectual property.

The information in this document is subject to change without notice. Cloudera shall not be liable for any damages resulting from technical errors or omissions which may be present in this document, or from use of this document.

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: 2.6.8

Date: September 27, 2019

Table of Contents

ABOUT THE CLUDERA ODBC DRIVER FOR IMPALA	5
WINDOWS DRIVER	6
WINDOWS SYSTEM REQUIREMENTS	6
INSTALLING THE DRIVER ON WINDOWS	6
CREATING A DATA SOURCE NAME ON WINDOWS	7
CONFIGURING AUTHENTICATION ON WINDOWS	9
CONFIGURING A PROXY CONNECTION ON WINDOWS	14
CONFIGURING HTTP OPTIONS ON WINDOWS	14
CONFIGURING SSL VERIFICATION ON WINDOWS	15
CONFIGURING ADVANCED OPTIONS ON WINDOWS	16
CONFIGURING SERVER-SIDE PROPERTIES ON WINDOWS	18
CONFIGURING LOGGING OPTIONS ON WINDOWS	18
SETTING DRIVER-WIDE CONFIGURATION OPTIONS ON WINDOWS	20
CONFIGURING KERBEROS AUTHENTICATION FOR WINDOWS	21
VERIFYING THE DRIVER VERSION NUMBER ON WINDOWS	25
MACOS DRIVER	26
MACOS SYSTEM REQUIREMENTS	26
INSTALLING THE DRIVER ON MACOS	26
VERIFYING THE DRIVER VERSION NUMBER ON MACOS	26
LINUX DRIVER	28
LINUX SYSTEM REQUIREMENTS	28
INSTALLING THE DRIVER USING THE RPM FILE	28
INSTALLING THE DRIVER ON DEBIAN	29
VERIFYING THE DRIVER VERSION NUMBER ON LINUX	30
AIX DRIVER	32
AIX SYSTEM REQUIREMENTS	32
INSTALLING THE DRIVER ON AIX	32
VERIFYING THE DRIVER VERSION NUMBER ON AIX	33
CONFIGURING THE ODBC DRIVER MANAGER ON NON-WINDOWS MACHINES	34
SPECIFYING ODBC DRIVER MANAGERS ON NON-WINDOWS MACHINES	34
SPECIFYING THE LOCATIONS OF THE DRIVER CONFIGURATION FILES	34
CONFIGURING ODBC CONNECTIONS ON A NON-WINDOWS MACHINE	36
CREATING A DATA SOURCE NAME ON A NON-WINDOWS MACHINE	36
CONFIGURING A DSN-LESS CONNECTION ON A NON-WINDOWS MACHINE	38
CONFIGURING AUTHENTICATION ON A NON-WINDOWS MACHINE	40

CONFIGURING SSL VERIFICATION ON A NON-WINDOWS MACHINE	44
CONFIGURING SERVER-SIDE PROPERTIES ON A NON-WINDOWS MACHINE	45
CONFIGURING LOGGING OPTIONS	46
SETTING DRIVER-WIDE CONFIGURATION OPTIONS ON A NON-WINDOWS MACHINE	47
TESTING THE CONNECTION	48
AUTHENTICATION OPTIONS	50
USING A CONNECTION STRING	51
DSN CONNECTION STRING EXAMPLE	51
DSN-LESS CONNECTION STRING EXAMPLES	51
FEATURES	54
DATA TYPES	54
CATALOG AND SCHEMA SUPPORT	55
SQL TRANSLATION	56
SERVER-SIDE PROPERTIES	56
ACTIVE DIRECTORY	56
WRITE-BACK	56
SECURITY AND AUTHENTICATION	57
DRIVER CONFIGURATION OPTIONS	58
CONFIGURATION OPTIONS APPEARING IN THE USER INTERFACE	58
CONFIGURATION OPTIONS HAVING ONLY KEY NAMES	78
ODBC API CONFORMANCE LEVEL	85
CONTACT US	87

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 54.

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 *Data Access Standards* on the Simba Technologies website: <https://www.simba.com/resources/data-access-standards-glossary>. For complete information about the ODBC specification, see the *ODBC API Reference* from the Microsoft documentation: <https://docs.microsoft.com/en-us/sql/odbc/reference/syntax/odbc-api-reference>.

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 is recommended for Impala versions 2.7 through 3.3, and CDH versions 5.9 through 5.16 and 6.0 through 6.2.

Install the driver on client machines where the application is installed. Before installing the driver, make sure that you have the following:

- Administrator rights on your machine.
- A machine that meets the following system requirements:
 - One of the following operating systems:
 - Windows 10, 8.1, or 7 SP1
 - Windows Server 2016, 2012, or 2008 R2 SP1
 - 100 MB of available disk space
 - Visual C++ Redistributable for Visual Studio 2013 installed (with the same bitness as the driver that you are installing).
You can download the installation packages at <https://www.microsoft.com/en-ca/download/details.aspx?id=40784>.

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 a driver whose bitness matches the bitness of the client application:

- Cloudera Impala 2.6 32-bit.msi for 32-bit applications
- Cloudera Impala 2.6 64-bit.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 **Cloudera Impala 2.6 32-bit.msi** or **Cloudera Impala 2.6 64-bit.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). A DSN is a data structure that stores connection information so that it can be used by the driver to connect to Impala.

Alternatively, you can specify connection settings in a connection string or as driver-wide settings. Settings in the connection string take precedence over settings in the DSN, and settings in the DSN take precedence over driver-wide settings.

The following instructions describe how to create a DSN. For information about specifying settings in a connection string, see "Using a Connection String" on page 51. For information about driver-wide settings, see "Setting Driver-Wide Configuration Options on Windows" on page 20.

To create a Data Source Name on Windows:

1. From the Start menu, go to **ODBC Data Sources**.

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 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. In the **Transport Mode** drop-down list, select the Thrift transport protocol to use in the Thrift layer.

Note:

For information about how to determine which Thrift transport protocols your Impala server supports, see "Authentication Options" on page 50.

14. To configure a connection through a proxy server, click **Proxy Options**. For more information, see "Configuring a Proxy Connection on Windows" on page 14.
15. If the Transport Mode option is set to HTTP, then to configure HTTP options such as custom headers, click **HTTP Options**. For more information, see "Configuring HTTP Options on Windows" on page 14.
16. To configure client-server verification over SSL, click **SSL Options**. For more information, see "Configuring SSL Verification on Windows" on page 15.
17. To configure advanced driver options, click **Advanced Options**. For more information, see "Configuring Advanced Options on Windows" on page 16.
18. 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 18.
19. To configure logging behavior for the driver, click **Logging Options**. For more information, see "Configuring Logging Options on Windows" on page 18.

20. 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.

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

Configuring Authentication on Windows

Some Impala servers are configured to require authentication for access. To connect to an Impala server, you must configure the Cloudera ODBC Driver for Impala to use the authentication mechanism that matches the access requirements of the server and provides the necessary credentials.

For information about how to determine the type of authentication your Impala server requires, see "Authentication Options" on page 50.

You can specify authentication settings in a DSN, in a connection string, or as driver-wide settings. Settings in the connection string take precedence over settings in the DSN, and settings in the DSN take precedence over driver-wide settings.

The following authentication methods are available:

- "Using No Authentication" on page 9
- "Using Kerberos" on page 10
- "Using Advanced Kerberos" on page 11
- "Using SASL User Name" on page 13
- "Using User Name And Password" on page 13

If cookie-based authentication is enabled in your Impala database, you can specify a list of authentication cookies in the `HTTPAuthCookies` connection property. In this case, the driver authenticates the connection once based on the provided authentication credentials. It then uses the cookie generated by the server for each subsequent request in the same connection. For more information, see "HTTPAuthCookies" on page 80.

Note:

On Windows, the `HTTPAuthCookies` property must be set in a connection string.

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. From 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 15.
4. To save your settings and close the dialog box, click **OK**.

Using Kerberos

If the Use Only SSPI advanced option is disabled, then Kerberos must be installed and configured before you can use this authentication mechanism. For information about configuring Kerberos on your machine, see "Configuring Kerberos Authentication for Windows" on page 21. For information about setting the Use Only SSPI advanced option, see "Configuring Advanced Options on Windows" on page 16.

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. From 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. Optionally, if you are using MIT Kerberos and a Kerberos realm is specified in the **Realm** field, then choose one:
 - To have the Kerberos layer canonicalize the server's service principal name, leave the **Canonicalize Principal FQDN** check box selected.

- Or, to prevent the Kerberos layer from canonicalizing the server's service principal name, clear the **Canonicalize Principal FQDN** check box.
7. To allow the driver to pass your credentials directly to the server for use in authentication, select **Delegate Kerberos Credentials**.
 8. 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 15.
 9. 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.

10. 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 option must be disabled. For more information, see "Use Only SSPI" on page 75.

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 73.
- 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 62.

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. Optionally, if you are using MIT Kerberos and a Kerberos realm is specified in the **Realm** field, then choose one:
 - To have the Kerberos layer canonicalize the server's service principal name, leave the **Canonicalize Principal FQDN** check box selected.
 - Or, to prevent the Kerberos layer from canonicalizing the server's service principal name, clear the **Canonicalize Principal FQDN** check box.
7. Select the **Use Keytab** check box.

Note:

If the check box is not available, make sure that MIT Kerberos is installed on your machine.

8. In the **User Name** field, type an appropriate user name for accessing the Impala server.
9. 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**.
10. 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 15.
11. 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.

12. 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. From 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. From 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 15.

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.

Note:

If the Transport Mode property is specified, it takes precedence over this property.

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

Configuring a Proxy Connection on Windows

If you are connecting to the data source through a proxy server, you must provide connection information for the proxy server.

To configure a proxy server connection on Windows:

1. To access proxy server options, open the ODBC Data Source Administrator where you created the DSN, then select the DSN, then click **Configure**, and then click **Proxy Options**.
2. Select the **Use Proxy** check box.
3. In the **Proxy Host** field, type the host name or IP address of the proxy server.
4. In the **Proxy Port** field, type the number of the TCP port that the proxy server uses to listen for client connections.
5. In the **Proxy Username** field, type your user name for accessing the proxy server.
6. In the **Proxy Password** field, type the password corresponding to the user name.
7. To encrypt your credentials, select one of the following:
 - If the credentials are used only by the current Windows user, select **Current User Only**.
 - Or, if the credentials are used by all users on the current Windows machine, select **All Users Of This Machine**.
8. To save your settings and close the HTTP Proxy Options dialog box, click **OK**.

Configuring HTTP Options on Windows

You can configure options such as custom headers when using the HTTP transport protocol in the Thrift layer. For information about how to determine if your Impala server supports the HTTP transport protocol, see "Authentication Options" on page 50.

The following instructions describe how to configure HTTP options in a DSN. You can specify the connection settings described below in a DSN, in a connection string, or as driver-wide settings.

Settings in the connection string take precedence over settings in the DSN, and settings in the DSN take precedence over driver-wide settings.

To configure HTTP options on Windows:

1. Open the ODBC Data Source Administrator where you created the DSN, then select the DSN, then click **Configure**, and then make sure that the Transport Mode option is set to **HTTP**.
2. To access HTTP options, click **HTTP Options**.

Note:

The HTTP options are available only when the Transport Mode option is set to HTTP.

3. In the **HTTP Path** field, type the partial URL corresponding to the Impala server.
4. To create a custom HTTP header, click **Add**, then type appropriate values in the **Key** and **Value** fields, and then click **OK**.
5. To edit a custom HTTP header, select the header from the list, then click **Edit**, then update the **Key** and **Value** fields as needed, and then click **OK**.
6. To delete a custom HTTP header, select the header from the list, and then click **Remove**. In the confirmation dialog box, click **Yes**.
7. To save your settings and close the HTTP Properties 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 the driver to connect to an SSL-enabled socket. When using SSL to connect to a server, the driver can be configured to verify the identity of the server.

The following instructions describe how to configure SSL in a DSN. You can specify the connection settings described below in a DSN, in a connection string, or as driver-wide settings. Settings in the connection string take precedence over settings in the DSN, and settings in the DSN take precedence over driver-wide settings.

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 authentication using self-signed certificates that have not been added to the list of trusted certificates, 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. To specify the CA certificates that you want to use to verify the server, do one of the following:
 - To verify the server using the trusted CA certificates from a specific `.pem` file, specify the full path to the file in the **Trusted Certificates** field and clear the **Use System Trust Store** check box.
 - Or, to use the trusted CA certificates `.pem` file that is installed with the driver, leave the **Trusted Certificates** field empty, and clear the **Use System Trust Store** check box.
 - Or, to use the Windows trust store, select the **Use System Trust Store** check box.

Important:

- If you are using the Windows trust store, make sure to import the trusted CA certificates into the trust store.
- If the trusted CA supports certificate revocation, select the **Check Certificate Revocation** check box.

6. From the **Minimum TLS Version** drop-down list, select the minimum version of TLS to use when connecting to your data store.
7. 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.

The following instructions describe how to configure advanced options in a DSN. You can specify the connection settings described below in a DSN, in a connection string, or as driver-wide settings. Settings in the connection string take precedence over settings in the DSN, and settings in the DSN take precedence over driver-wide settings.

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.

Important:

- When this option is enabled, the driver cannot execute parameterized queries.
- 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 **Ignore 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 driver automatically attempt to reconnect to the server if communications are lost, select **Enable Auto Reconnect**.
6. To have the driver restrict catalog queries to the current schema when no schema is specified, or when the schema is specified with the wildcard character %, select **Restrict Metadata with Current Schema**.
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 that the TCP socket waits for a response from the server before timing out the request and returning an error message.

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. In the **Async Exec Poll Interval** field, type the time in milliseconds between each poll for the query execution status.
11. 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** 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.
12. Optionally, if you want the driver to retry queries that fail, select the **Enable Query Retry** check box and then do the following:
 - a. In the **Max Retries** field, type the maximum number of times that the driver retries each query.
 - b. In the **Result Set Cache Size** field, type the maximum amount of memory that the result set cache can occupy. Values must be specified in: B (bytes), KB (kilobytes), MB (megabytes), or GB (gigabytes).
 - c. In the **Retry Interval** field, type the amount of time that the driver waits between query retry attempts. Values must be specified in S (seconds) or MS (milliseconds).
13. 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.

The following instructions describe how to configure server-side properties in a DSN. You can specify the connection settings described below in a DSN, in a connection string, or as driver-wide settings. Settings in the connection string take precedence over settings in the DSN, and settings in the DSN take precedence over driver-wide settings.

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.

The settings for logging apply to every connection that uses the Cloudera ODBC Driver for Impala, so make sure to disable the feature after you are done using it.

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 the following log files at the location you specify in the Log Path field:

- A `clouderaimpalaodbcdriver.log` file that logs driver activity that is not specific to a connection.
- A `clouderaimpalaodbcdriver_connection_[Number].log` file for each connection made to the database, where `[Number]` is a number that identifies each log file. This file logs driver activity that is specific to the connection.

If you enable the `UseLogPrefix` connection property, the driver prefixes the log file name with the user name associated with the connection and the process ID of the application through which the connection is made. For more information, see "UseLogPrefix" on page 83.

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.


Setting Driver-Wide Configuration Options on Windows

When you specify connection settings in a DSN or connection string, those settings apply only when you connect to Impala using that particular DSN or string. As an alternative, you can specify settings that apply to every connection that uses the Cloudera ODBC Driver for Impala by configuring them in the Windows Registry.

Note:

Settings in the connection string take precedence over settings in the DSN, and settings in the DSN take precedence over driver-wide settings.

To set driver-wide configuration options on Windows:

1. Choose one:
 - If you are using Windows 7 or earlier, click **Start** , then type **regedit** in the **Search** field, and then click **regedit.exe** in the search results.
 - Or, if you are using Windows 8 or later, on the Start screen, type **regedit**, and then click the **regedit** search result.
2. Navigate to the appropriate registry key for the bitness of your driver and your machine:
 - If you are using the 32-bit driver on a 64-bit machine, then browse to the following registry key:
HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Cloudera\Cloudera ODBC Driver for Impala\Driver
 - Otherwise, browse to the following registry key:

HKEY_LOCAL_MACHINE\SOFTWARE\Cloudera\Cloudera ODBC Driver for Impala\Driver

3. For each connection property that you want to configure, do the following:
 - a. Right-click the **Driver** subkey and then select **New > String Value**.
 - b. Type the key name of the connection property, and then press **Enter**.

For example, to specify the authentication mechanism to use, type `AuthMech`. To verify the supported key name for each driver configuration option, refer to the "Key Name" column in the description of the option in "Driver Configuration Options" on page 58.

- c. Right-click the value that you created in the previous steps and then click **Modify**.
 - d. In the Edit String dialog box, in the **Value Data** field, type the value that you want to set the connection property to and then click **OK**.

For example, to specify the Kerberos authentication mechanism, type `1`.

4. Close the Registry Editor.

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 Apache'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. From the Start menu, go to **ODBC Data Sources**.

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.

macOS Driver

macOS System Requirements

The Cloudera ODBC Driver for Impala is recommended for Impala versions 2.7 through 3.3, and CDH versions 5.9 through 5.16 and 6.0 through 6.2.

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

- macOS version 10.12, 10.13, or 10.14
- 100 MB of available disk space
- One of the following ODBC driver managers installed:
 - iODBC 3.52.9, 3.52.10, 3.52.11, or 3.52.12
 - unixODBC 2.3.1

Installing the Driver on macOS

The Cloudera ODBC Driver for Impala is available for macOS 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 macOS:

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 34.

Verifying the Driver Version Number on macOS

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

To verify the driver version number on macOS:

- At the Terminal, run the following command:

```
pkgutil --info 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 is recommended for Impala versions 2.7 through 3.3, and CDH versions 5.9 through 5.16 and 6.0 through 6.2.

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

- One of the following distributions:
 - Red Hat® Enterprise Linux® (RHEL) 6 or 7
 - CentOS 6 or 7
 - SUSE Linux Enterprise Server (SLES) 11 or 12
 - Debian 6 or 7
 - Oracle Linux 7.5 or 7.6
 - Ubuntu 14.04
- 50 MB of available disk space
- One of the following ODBC driver managers installed:
 - iODBC 3.52.9, 3.52.10, 3.52.11, or 3.52.12
 - unixODBC 2.3.2, 2.3.3, or 2.3.4
- 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 a driver whose bitness 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

The placeholders in the file names are defined as follows:

- `[Version]` is the version number of the driver.
- `[Release]` is the release number for this version of the driver.

You can install both the 32-bit and 64-bit 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.
2. Navigate to the folder containing the RPM package for the driver.
3. 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 34.

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 34.

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. Alternatively, you can search the driver's binary file for version number information.

To verify the driver version number on Linux using the command-line interface:

- 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.

To verify the driver version number on Linux using the binary file:

1. Navigate to the `/lib` subfolder in your driver installation directory. By default, the path to this directory is: `/opt/cloudera/impalaodbc/lib`.
2. Open the driver's `.so` binary file in a text editor, and search for the text `$driver_version_sb$:`. The driver's version number is listed after this text.

AIX Driver

AIX System Requirements

The Cloudera ODBC Driver for Impala is recommended for Impala versions 2.7 through 3.3, and CDH versions 5.9 through 5.16 and 6.0 through 6.2.

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 3.52.9, 3.52.10, 3.52.11, or 3.52.12
 - unixODBC 2.3.2, 2.3.3, or 2.3.4

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 34.

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:

- Set the library path environment variable to make sure that your machine uses the correct ODBC driver manager. For more information, see "Specifying ODBC Driver Managers on Non-Windows Machines" on page 34.
- If the driver configuration files are not stored in the default locations expected by the ODBC driver manager, then set environment variables to make sure that the driver manager locates and uses those files. For more information, see "Specifying the Locations of the Driver Configuration Files" on page 34.

After configuring the ODBC driver manager, you can configure a connection and access your data store through the driver.

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.

macOS

If you are using a macOS 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 macOS 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 store these configuration files elsewhere, then you must set the environment variables described below so that the driver manager can locate the files.

If you are using iODBC, do the following:

- Set ODBCINI to the full path and file name of the `odbc.ini` file.
- Set ODBCINSTINI to the full path and file name of the `odbcinst.ini` file.
- Set CLOUDERAIMPALAINI to the full path and file name of the `cloudera.impalaodbc.ini` file.

If you are using unixODBC, do the following:

- Set ODBCINI to the full path and file name of the `odbc.ini` file.
- Set ODBCSYSINI to the full path of the directory that contains the `odbcinst.ini` file.
- Set CLOUDERAIMPALAINI to the full path and file name of the `cloudera.impalaodbc.ini` file.

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

For iODBC:

```
export ODBCINI=/usr/local/odbc/odbc.ini
export ODBCINSTINI=/usr/local/odbc/odbcinst.ini
export CLOUDERAIMPALAINI=/etc/cloudera.impalaodbc.ini
```

For unixODBC:

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

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

1. If the CLOUDERAIMPALAINI 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 on a Non-Windows Machine

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" on page 36
- "Configuring a DSN-less Connection on a Non-Windows Machine" on page 38
- "Configuring Authentication on a Non-Windows Machine" on page 40
- "Configuring SSL Verification on a Non-Windows Machine" on page 44
- "Configuring Server-Side Properties on a Non-Windows Machine" on page 45
- "Configuring Logging Options" on page 46
- "Setting Driver-Wide Configuration Options on a Non-Windows Machine" on page 47
- "Testing the Connection" on page 48

Creating a Data Source Name on a Non-Windows Machine

Typically, after installing the Cloudera ODBC Driver for Impala, you need to create a Data Source Name (DSN). A DSN is a data structure that stores connection information so that it can be used by the driver to connect to Impala.

You can specify connection settings in a DSN (in the `odbc.ini` file), in a connection string, or as driver-wide settings (in the `cloudera.impalaodbc.ini` file). Settings in the connection string take precedence over settings in the DSN, and settings in the DSN take precedence over driver-wide settings.

The following instructions describe how to create a DSN by specifying connection settings in the `odbc.ini` file. 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.

For information about specifying settings in a connection string, see "Configuring a DSN-less Connection on a Non-Windows Machine" on page 38 and "Using a Connection String" on page 51. For information about driver-wide settings, see "Setting Driver-Wide Configuration Options on a Non-Windows Machine" on page 47.

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, you can remove the period (.) from the start of the file name to make the file visible while you are editing it.

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 macOS 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 macOS 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 40.
 - 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 44.
 - 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 45.
 - 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 58.

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 34.

For example, the following is an `odbc.ini` configuration file for macOS 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, you can remove the period (.) from the start of the file name to make the file visible while you are editing it.

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 macOS 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. 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 `ODBCINSTINI` or `ODBCSYSINI` environment variable specifies the location. For more information, see "Specifying the Locations of the Driver Configuration Files" on page 34.

For example, the following is an `odbcinst.ini` configuration file for macOS:

```
[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/libclouderaimpalaodbc.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.so
[Cloudera ODBC Driver for Impala 64-bit]
Description=Cloudera ODBC Driver for Impala (64-bit)
Driver=/opt/cloudera/impalaodbc/lib/64/libclouderaimpalaodbc64.so
```

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 51.

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

- "Configuring Authentication on a Non-Windows Machine" on page 40
- "Configuring SSL Verification on a Non-Windows Machine" on page 44
- "Configuring Server-Side Properties on a Non-Windows Machine" on page 45

For detailed information about all the connection properties that the driver supports, see "Driver Configuration Options" on page 58.

Configuring Authentication on a Non-Windows Machine

Some Impala servers are configured to require authentication for access. To connect to an Impala server, you must configure the Cloudera ODBC Driver for Impala to use the authentication mechanism that matches the access requirements of the server and provides the necessary credentials.

For information about how to determine the type of authentication your Impala server requires, see "Authentication Options" on page 50.

You can set the connection properties for authentication in a connection string, in a DSN (in the `odbc.ini` file), or as a driver-wide setting (in the `cloudera.impalaodbc.ini` file). Settings in the connection string take precedence over settings in the DSN, and settings in the DSN take precedence over driver-wide settings.

The following authentication methods are available:

- "Using No Authentication" on page 41
- "Using Kerberos" on page 41
- "Using Advanced Kerberos" on page 42
- "Using SASL User Name" on page 43
- "Using User Name And Password" on page 44

If cookie-based authentication is enabled in your Impala database, you can specify a list of authentication cookies in the `HTTPAuthCookies` connection property. In this case, the driver authenticates the connection once based on the provided authentication credentials. It then uses the cookie generated by the server for each subsequent request in the same connection. For more information, see "HTTPAuthCookies" on page 80.

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. Optionally, if you are using MIT Kerberos and a Kerberos realm is specified using the `KrbRealm` connection attribute, then choose one:
 - To have the Kerberos layer canonicalize the server's service principal name, leave the `ServicePrincipalCanonicalization` attribute set to 1.

- Or, to prevent the Kerberos layer from canonicalizing the server's service principal name, set the `ServicePrincipalCanonicalization` attribute to 0.
4. 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`.

5. Set the `KrbServiceName` attribute to the service name of the Impala server.
6. 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 73.
- 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 62.

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. Optionally, if you are using MIT Kerberos and a Kerberos realm is specified using the `KrbRealm` connection attribute, then choose one:
 - To have the Kerberos layer canonicalize the server's service principal name, leave the `ServicePrincipalCanonicalization` attribute set to 1.
 - Or, to prevent the Kerberos layer from canonicalizing the server's service principal name, set the `ServicePrincipalCanonicalization` attribute to 0.
4. 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`.

5. Set the `KrbServiceName` attribute to the service name of the Impala server.
6. Set the `UseKeytab` attribute to 1.
7. Set the `UID` attribute to an appropriate user name for accessing the Impala server.
8. 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.
9. 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.
10. 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 44.
11. 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.

Note:

If the `Transport Mode` property is specified, it takes precedence over this property.

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. When using SSL to connect to a server, the driver can be configured to verify the identity of the server.

You can set the connection properties described below in a connection string, in a DSN (in the `odbc.ini` file), or as a driver-wide setting (in the `cloudera.impalaodbc.ini` file). Settings in the connection string take precedence over settings in the DSN, and settings in the DSN take precedence over driver-wide settings.

To configure SSL verification on a non-Windows machine:

1. To enable SSL connections, set the `SSL` attribute to 1.
2. To allow authentication using self-signed certificates that have not been added to the list of trusted certificates, 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 `AllowHostNameCNMismatch` 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.
5. To specify the minimum version of TLS to use, set the `Min_TLS` property to the minimum version of TLS. Supported options include 1.0 for TLS 1.0, 1.1 for TLS 1.1, and 1.2 for TLS 1.2.

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, in a DSN (in the `odbc.ini` file), or as a driver-wide setting (in the `cloudera.impalaodbc.ini` file). Settings in the connection string take precedence over settings in the DSN, and settings in the DSN take precedence over driver-wide settings.

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.

The settings for logging apply to every connection that uses the Cloudera ODBC Driver for Impala, so make sure to disable the feature after you are done using it.

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 bytes.

Note:

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

6. Optionally, to prefix the log file name with the user name and process ID associated with the connection, set the `UseLogPrefix` property to 1.
7. Save the `cloudera.impalaodbc.ini` configuration file.
8. Restart your ODBC application to make sure that the new settings take effect.

The Cloudera ODBC Driver for Impala produces the following log files at the location you specify using the `LogPath` key:

- A `clouderaimpalaodbcdriver.log` file that logs driver activity that is not specific to a connection.
- A `clouderaimpalaodbcdriver_connection_[Number].log` file for each connection made to the database, where `[Number]` is a number that identifies each log file. This file logs driver activity that is specific to the connection.

If you set the `UseLogPrefix` property to 1, then each file name is prefixed with `[UserName]_[ProcessID]_`, where `[UserName]` is the user name associated with the connection and `[ProcessID]` is the process ID of the application through which the connection is made. For more information, see "UseLogPrefix" on page 83.

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.

Setting Driver-Wide Configuration Options on a Non-Windows Machine

When you specify connection settings in a DSN or connection string, those settings apply only when you connect to Impala using that particular DSN or string. As an alternative, you can specify settings that apply to every connection that uses the Cloudera ODBC Driver for Impala by configuring them in the `cloudera.impalaodbc.ini` file.

Note:

Settings in the connection string take precedence over settings in the DSN, and settings in the DSN take precedence over driver-wide settings.

To set driver-wide configuration options on a non-Windows machine:

1. In a text editor, open the `cloudera.impalaodbc.ini` configuration file.
2. In the `[Driver]` section, specify configuration options as key-value pairs. Start a new line for each key-value pair.

For example, to enable SASL User Name authentication using "cloudera" as the user name, type the following:

```
AuthMech=2
UID=cloudera
```

For detailed information about all the configuration options supported by the driver, see "Driver Configuration Options" on page 58.

3. Save the `cloudera.impalaodbc.ini` configuration file.

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 51.

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
<ul style="list-style-type: none"> • Kerberos • SASL User Name • User Name And Password (with SASL enabled) 	<ul style="list-style-type: none"> • No Authentication • 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 58.

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 an Impala Server with LDAP Authentication or other User Name and Password Authentication Enabled

The following is the format of a DSN-less connection string that connects to an Impala server with LDAP authentication, or another form of username/password authentication, enabled:

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

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 LDAP-enabled 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" on page 54
- "Catalog and Schema Support" on page 55
- "SQL Translation" on page 56
- "Server-Side Properties" on page 56
- "Active Directory" on page 56
- "Write-back" on page 56
- "Security and Authentication" on page 57

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
BOOLEAN	SQL_BOOLEAN
CHAR	SQL_CHAR
<p>Note: Only available in CDH 5.2 or later.</p>	<p>Note: SQL_WCHAR is returned instead if the Use SQL Unicode Types configuration option (the <code>UseUnicodeSqlCharacterTypes</code> key) is enabled.</p>
DATE	SQL_DATE
<p>Note: DATE data types are only supported in Impala 3.3 or later.</p>	
DECIMAL	SQL_DECIMAL

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

Catalog and Schema Support

The Cludera 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 18. 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 45.

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 Apache's documentation, so that users in the Active Directory realm can access services in the MIT Kerberos Hadoop realm.

Write-back

The Cloudera ODBC Driver for Impala supports translation for the following syntax:

- INSERT
- CREATE
- DROP

The driver also supports translation for UPDATE and DELETE syntax, but only when querying Kudu tables while connected to an Impala server that is running Impala 2.7 or later.

If the statement contains non-standard SQL-92 syntax, then the driver is unable to translate the statement to SQL and instead falls back to using Impala SQL.

Security and Authentication

To protect data from unauthorized access, some Impala data stores require connections to be authenticated with user credentials or the SSL protocol. The Cloudera ODBC Driver for Impala provides full support for these authentication protocols.

Note:

In this documentation, "SSL" refers to both TLS (Transport Layer Security) and SSL (Secure Sockets Layer). The driver supports TLS 1.0, 1.1, and 1.2. The SSL version used for the connection is the highest version that is supported by both the driver and the server.

The driver provides mechanisms that enable you to authenticate your connection using the Kerberos protocol, your Impala user name only, or your Impala user name and password. You must use the authentication mechanism that matches the security requirements of the Impala server. For information about determining the appropriate authentication mechanism to use based on the Impala server configuration, see "Authentication Options" on page 50. For detailed driver configuration instructions, see "Configuring Authentication on Windows" on page 9 or "Configuring Authentication on a Non-Windows Machine" on page 40.

Additionally, the driver supports SSL connections with or without one-way authentication. If the server has an SSL-enabled socket, then you can configure the driver to connect to it.

It is recommended that you enable SSL whenever you connect to a server that is configured to support it. SSL encryption protects data and credentials when they are transferred over the network, and provides stronger security than authentication alone. For detailed configuration instructions, see "Configuring SSL Verification on Windows" on page 15 or "Configuring SSL Verification on a Non-Windows Machine" on page 44.

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, configuring driver-wide settings, or configuring a connection from a Linux/macOS/AIX machine, use the key names provided.

Note:

Settings in the connection string take precedence over settings in the DSN, and settings in the DSN take precedence over driver-wide settings.

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, configuring driver-wide settings, or configuring a connection from a Linux/macOS/AIX machine:

- "Allow Common Name Host Name Mismatch" on page 59
- "Allow Self-Signed Server Certificate" on page 60
- "Async Exec Poll Interval" on page 60
- "Canonicalize Principal FQDN" on page 60
- "Check Certificate Revocation" on page 61
- "Convert Key Name to Lower Case" on page 61
- "Database" on page 61
- "Default Keytab File" on page 62
- "Delegation UID" on page 62
- "Enable Auto Reconnect" on page 63
- "Proxy Host" on page 68
- "Proxy Password" on page 69
- "Proxy Port" on page 68
- "Proxy Username" on page 69
- "Realm" on page 69
- "Restrict Metadata with Current Schema" on page 69
- "Result Set Cache Size" on page 70
- "Retry Interval" on page 70
- "Rows Fetched Per Block" on page 71
- "Save Password (Encrypted)" on page 71
- "Service Name" on page 71
- "Socket Timeout" on page 71

- "Enable Query Retry" on page 63
- "Enable SSL" on page 63
- "Host" on page 64
- "Host FQDN" on page 64
- "HTTP Path" on page 64
- "Ignore Transactions" on page 64
- "Log Level" on page 65
- "Log Path" on page 66
- "Max File Size" on page 66
- "Max Number Files" on page 66
- "Maximum Retries" on page 67
- "Mechanism" on page 67
- "Minimum TLS" on page 67
- "Password" on page 68
- "Port" on page 68
- "String Column Length" on page 72
- "Transport Buffer Size" on page 72
- "Transport Mode" on page 72
- "Trusted Certificates" on page 73
- "UPN Keytab Mapping File" on page 73
- "Use Keytab" on page 74
- "Use Native Query" on page 75
- "Use Only SSPI" on page 75
- "Use Proxy" on page 76
- "Use Simple Authentication and Security Layer (SASL)" on page 76
- "Use SQL Unicode Types" on page 77
- "Use System Trust Store" on page 77
- "User Name" on page 78

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 is still recognized by the driver under that key. If both keys are defined, `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 a connection to an Impala server that uses a self-signed certificate.

- 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.

Async Exec Poll Interval

Key Name	Default Value	Required
AsyncExecPollInterval	10	No

Description

The time in milliseconds between each poll for the query execution status.

"Asynchronous execution" refers to the fact that the RPC call used to execute a query against Impala is asynchronous. It does not mean that ODBC asynchronous operations are supported.

Canonicalize Principal FQDN

Key Name	Default Value	Required
ServicePrincipalCanonicalization	Selected (1)	No

Description

This option specifies whether the Kerberos layer canonicalizes the host FQDN in the server's service principal name.

- Enabled (1): The Kerberos layer canonicalizes the host FQDN in the server's service principal name.
- Disabled (0): The Kerberos layer does not canonicalize the host FQDN in the server's service principal name.

Note:

- This option only affects MIT Kerberos, and is ignored when using Active Directory Kerberos.
- This option can only be disabled if the Kerberos Realm or `KrbRealm` key is specified.

Check Certificate Revocation

Key Name	Default Value	Required
CheckCertRevocation	Selected (1)	No

Description

This option specifies whether the driver checks to see if a certificate has been revoked while retrieving a certificate chain from the Windows Trust Store.

This option is only applicable if you are using a CA certificate from the Windows Trust Store (see "Use System Trust Store" on page 77).

- Enabled (1): The driver checks for certificate revocation while retrieving a certificate chain from the Windows Trust Store.
- Disabled (0): The driver does not check for certificate revocation while retrieving a certificate chain from the Windows Trust Store.

Note:

This option is only available on Windows.

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	Selected (1)	Yes

Description

This option specifies whether the driver attempts to automatically reconnect to the server when a communication link error occurs.

- Enabled (1): The driver attempts to reconnect.
- Disabled (0): The driver does not attempt to reconnect.

Enable Query Retry

Key Name	Default Value	Required
EnableQueryRetry	Clear (0)	No

Description

This option specifies whether the driver automatically retries queries that are sent to the server but fail.

- Enabled (1): The driver retries failed queries.
- Disabled (0): The driver only submits each query once.

Query retry settings are configured through the following options:

- "Maximum Retries" on page 67
- "Result Set Cache Size" on page 70
- "Retry Interval" on page 70
- "GlobalResultSetCache" on page 80

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.

Driver Configuration Options

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

HTTP Path

Key Name	Default Value	Required
HTTPPath	None	No

Description

The partial URL corresponding to the Impala server.

The driver forms the HTTP address to connect to by appending the HTTP Path value to the host and port specified in the DSN or connection string. For example, to connect to the HTTP address `http://localhost:21050/gateway/sandbox/impala/version`, you would set HTTP Path to `/gateway/sandbox/impala/version`.

Host

Key Name	Default Value	Required
Host (or Server)	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.

Ignore Transactions

Key Name	Default Value	Required
IgnoreTransactions	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.

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.
- The settings for logging apply to every connection that uses the Cludera ODBC Driver for Impala, so make sure to disable the feature after you are done using it.
- 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 the following log files at the location you specify in the Log Path (`LogPath`) property:

- A `clouderaimpalaodbcdriver.log` file that logs driver activity that is not specific to a connection.

Driver Configuration Options

- A `clouderaimpalaodbcdriver_connection_[Number].log` file for each connection made to the database, where `[Number]` is a number that identifies each log file. This file logs driver activity that is specific to the connection.

If you enable the `UseLogPrefix` connection property, the driver prefixes the log file name with the user name associated with the connection and the process ID of the application through which the connection is made. For more information, see "UseLogPrefix" on page 83.

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	20971520	No

Description

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

If this property is set using the Windows UI, the entered value is converted from megabytes (MB) to bytes before being set.

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.

Maximum Retries

Key Name	Default Value	Required
MaxNumQueryRetries	5	No

Description

The maximum number of times that the driver retries a failed query, when query retry is enabled.

Also see "Enable Query Retry" on page 63.

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)

Minimum TLS

Key Name	Default Value	Required
Min_TLS	TLS 1.2 (1.2)	No

Description

The minimum version of TLS/SSL that the driver allows the data store to use for encrypting connections. For example, if TLS 1.1 is specified, TLS 1.0 cannot be used to encrypt connections.

- TLS 1.0 (1 . 0): The connection must use at least TLS 1.0.
- TLS 1.1 (1 . 1): The connection must use at least TLS 1.1.
- TLS 1.2 (1 . 2): The connection must use at least TLS 1.2.

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.

Proxy Host

Key Name	Default Value	Required
ProxyHost	None	Yes, if connecting through a proxy server.

Description

The host name or IP address of a proxy server that you want to connect through.

Proxy Port

Key Name	Default Value	Required
ProxyPort	None	Yes, if connecting through a proxy server.

Description

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

Proxy Password

Key Name	Default Value	Required
ProxyPWD	None	Yes, if connecting to a proxy server that requires authentication.

Description

The password that you use to access the proxy server.

Proxy Username

Key Name	Default Value	Required
ProxyUID	None	Yes, if connecting to a proxy server that requires authentication.

Description

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

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.

Restrict Metadata with Current Schema

Key Name	Default Value	Required
CurrentSchema RestrictedMetadata	Clear (0)	No

Description

This option specifies whether the driver should restrict catalog function results to tables and views in the current schema if a catalog function call is made without specifying a schema, or if the schema is specified as the wildcard character %.

Note:

Restricting results to the tables and views in the current schema may improve the performance of catalog calls that do not specify a schema.

- Enabled (1): The driver restricts catalog function results to the current schema if a schema is not specified.
- Disabled (0): The driver does not restrict catalog function results to the current schema if a schema is not specified.

Result Set Cache Size

Key Name	Default Value	Required
ResultSetCacheSize	20MB	No

Description

The maximum amount of memory that the result set cache for the current connection can occupy. Values must be specified in: B (bytes), KB (kilobytes), MB (megabytes), or GB (gigabytes).

When query retries are enabled, the driver temporarily caches result sets. For more information about this feature, see "Enable Query Retry" on page 63.

For information about setting a maximum total cache size for all concurrent connections, see "GlobalResultSetCache" on page 80.

Retry Interval

Key Name	Default Value	Required
QueryRetryInterval	5S	No

Description

The amount of time that the driver waits between query retry attempts, when query retry is enabled. Values can be specified in S (seconds) or MS (milliseconds).

Also see "Enable Query Retry" on page 63.

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	30	No

Description

The number of seconds that the TCP socket waits for a response from the server before timing out the request and returning an error message.

When this option is set to 0, the TCP socket does not time out any requests.

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.

Transport Mode

Key Name	Default Value	Required
TransportMode	Binary (0) if using No Authentication, otherwise SASL (1).	No

Description

The transport protocol to use in the Thrift layer.

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

- Binary (0 or binary)
- SASL (1 or sasl)
- HTTP (2 or http)

Note:

- If this property is specified, it takes precedence over the Use SASL property (`UseSASL`).
- If this property is not specified and the Use SASL property is specified, the Use SASL property takes precedence over the default value of this property.

For more information, see "Use Simple Authentication and Security Layer (SASL)" on page 76.

Trusted Certificates

Key Name	Default Value	Required
<code>TrustedCerts</code>	<p>The <code>cacerts.pem</code> file in the <code>\lib</code> subfolder within the driver's installation directory.</p> <p>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 macOS driver.</p>	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.

Important:

If you are connecting from a Windows machine and the Use System Trust Store option is enabled, the driver uses the certificates from the Windows trust store instead of your specified `.pem` file. For more information, see "Use System Trust Store" on page 77.

UPN Keytab Mapping File

Key Name	Default Value	Required
<code>UPNKeytabMappingFile</code>	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 *[Keytab]* is the full path to the keytab file:

```
{
  "[UserName]": {
    "principal" : "[KerberosUPN]",
    "keytab" : "[Keytab]"
  },
  ... }
```

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

```
{
  "cloudera": {
    "principal" : "cloudera@CLLOUDERA",
    "keytab" : "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 Impala SQL queries, or converts the queries emitted by an application into an equivalent form in Impala SQL. If the application is Impala-aware and already emits Impala SQL, then enable this option to avoid the extra overhead of query transformation.

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

Important:

When this option is enabled, the driver cannot execute parameterized queries.

Use Only SSPI

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 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 Proxy

Key Name	Default Value	Required
UseProxy	Clear (0)	No

Description

This option specifies whether the driver uses a proxy server to connect to the data store.

- **Enabled (1):** The driver connects to a proxy server based on the information provided in the Proxy Host, Proxy Port, Proxy Username, and Proxy Password fields or the ProxyHost, ProxyPort, ProxyUID, and ProxyPWD keys.
- **Disabled (0):** The driver connects directly to the Impala server.

Use Simple Authentication and Security Layer (SASL)

Key Name	Default Value	Required
UseSASL	0 if using No Authentication. 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.

Note:

- If the Transport Mode property (`TransportMode`) is specified, it takes precedence over this property.
- If the Transport Mode property is not specified and this property is specified, this property takes precedence over the default value of Transport Mode.

For more information, see "Transport Mode" on page 72.

Use SQL Unicode Types

Key Name	Default Value	Required
<code>UseSQLUnicodeTypes</code>	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
<code>UseSystemTrustStore</code>	Clear (0)	No

Description

This option specifies whether to use a CA certificate from the system trust store, or from a specified `.pem` file.

- Enabled (1): The driver verifies the connection using a certificate in the system trust store.
- Disabled (0): The driver verifies the connection using a specified `.pem` file. For information about specifying a `.pem` file, see "Trusted Certificates" on page 73.

Note:

This option is only available on Windows.

User Name

Key Name	Default Value	Required
UID	For User Name (2) authentication only, the default value is <code>anonymous</code>	Yes, if the authentication mechanism is User Name And Password (3). 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, configure driver-wide settings, or configure a connection from a Linux/macOS/AIX machine:

- "DelegationUserIDCase" on page 79
- "DisableOptimizedEncodingConverter" on page 79
- "Driver" on page 79
- "GlobalResultSetCache" on page 80
- "HTTPAuthCookies" on page 80
- "http.header." on page 81
- "MaxCatalogNameLen" on page 82
- "MaxColumnNameLen" on page 82
- "MaxSchemaNameLen" on page 82
- "MaxTableNameLen" on page 83
- "SSP_" on page 81

The `UseLogPrefix` property must be configured as a Windows Registry key value, or as a driver-wide property in the `cloudera.impalaodbc.ini` file for macOS or Linux.

- "UseLogPrefix" on page 83

DelegationUserIDCase

Key Name	Default Value	Required
DelegationUserIDCase	Unchanged	No

Description

This option specifies whether the driver changes the Delegation UID (or `DelegationUID`) value to all upper-case or all lower-case. The following values are supported:

- `Upper`: Change the delegated user name to all upper-case.
- `Lower`: Change the delegated user name to all lower-case.
- `Unchanged`: Do not modify the delegated user name.

For more information about delegating a user name, see "Delegation UID" on page 62.

DisableOptimizedEncodingConverter

Key Name	Default Value	Required
DisableOptimizedEncodingConverter	false	No

Description

This driver-wide option enables the driver to read and replace invalid UTF-8 data.

Note:

Enabling this option may result in decreased performance.

- `true`: The driver replaces invalid UTF-8 characters with the replacement character (🔍), and returns the rest of the data successfully.
- `false`: The driver throws an exception when it encounters invalid UTF-8 data.

Important:

This option applies to every connection that uses the Cloudera ODBC Driver for Impala. For more information, see "Setting Driver-Wide Configuration Options on Windows" on page 20 or "Setting Driver-Wide Configuration Options on a Non-Windows Machine" on page 47.

Driver

Key Name	Default Value	Required
Driver	Cloudera ODBC Driver for Impala	Yes

Key Name	Default Value	Required
	when installed on Windows, or the absolute path of the driver shared object file when installed on a non-Windows machine.	

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.

GlobalResultSetCache

Key Name	Default Value	Required
<code>GlobalResultSetCache</code>	<code>200MB</code>	<code>No</code>

Description

The maximum total amount of memory that can be occupied by result set caches across concurrent connections. Values must be specified in: B (bytes), KB (kilobytes), MB (megabytes), or GB (gigabytes).

For example, when this property is set to the default value of `200MB`, if you have 3 concurrent Impala connections, the total amount of memory consumed by the 3 result set caches for those connections cannot exceed `200MB`.

These result set caches are used only when query retries are enabled. For more information about this feature, see "Enable Query Retry" on page 63.

For information about setting a maximum cache size for the current connection only, see "Result Set Cache Size" on page 70.

HTTPAuthCookies

Key Name	Default Value	Required
<code>HTTPAuthCookies</code>	<code>None</code>	<code>No</code>

Description

A comma-separated list of authentication cookies that are supported by the driver.

If cookie-based authentication is enabled in your server, the driver authenticates the connection once based on the provided authentication credentials. It then uses the cookie generated by the

server for each subsequent request in the same connection.

http.header.

Key Name	Default Value	Required
http.header.	None	No

Description

Set a custom HTTP header by using the following syntax, where *[HeaderKey]* is the name of the header to set and *[HeaderValue]* is the value to assign to the header:

```
http.header.[HeaderKey]=[HeaderValue]
```

For example:

```
http.header.AUTHENTICATED_USER=john
```

After the driver applies the header, the `http.header.` prefix is removed from the DSN entry, leaving an entry of *[HeaderKey]=[HeaderValue]*

The example above would create the following custom HTTP header:

```
AUTHENTICATED_USER: john
```

Note:

- The `http.header.` prefix is case-sensitive.
- This option is applicable only when you are using HTTP as the Thrift transport protocol. For more information, see "Transport Mode" on page 72.

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.

MaxCatalogNameLen

Key Name	Default Value	Required
MaxCatalogNameLen	0	No

Description

The maximum number of characters that can be returned for catalog names.

This option can be set to any integer from 0 to 65535, inclusive. To indicate that there is no maximum length or that the length is unknown, set this option to 0.

MaxColumnNameLen

Key Name	Default Value	Required
MaxColumnNameLen	0	No

Description

The maximum number of characters that can be returned for column names.

This option can be set to any integer from 0 to 65535, inclusive. To indicate that there is no maximum length or that the length is unknown, set this option to 0.

MaxSchemaNameLen

Key Name	Default Value	Required
MaxSchemaNameLen	256	No

Description

The maximum number of characters that can be returned for schema names.

This option can be set to any integer from 0 to 65535, inclusive. To indicate that there is no maximum length or that the length is unknown, set this option to 0.

MaxTableNameLen

Key Name	Default Value	Required
MaxTableNameLen	0	No

Description

The maximum number of characters that can be returned for table names.

This option can be set to any integer from 0 to 65535, inclusive. To indicate that there is no maximum length or that the length is unknown, set this option to 0.

UseLogPrefix

Key Name	Default Value	Required
UseLogPrefix	0	No

Description

This option specifies whether the driver includes a prefix in the names of log files so that the files can be distinguished by user and application.

Set the property to one of the following values:

- 1: The driver prefixes log file names with the user name and process ID associated with the connection that is being logged.

For example, if you are connecting as a user named "jdoe" and using the driver in an application with process ID 7836, the generated log files would be named `jdoe_7836_clouderaimpalaodbcdriver.log` and `jdoe_7836_clouderaimpalaodbcdriver_connection_[Number].log`, where *[Number]* is a number that identifies each connection-specific log file.

- 0: The driver does not include the prefix in log file names.

To configure this option for the Windows driver, you create a value for it in one of the following registry keys:

- For a 32-bit driver installed on a 64-bit machine: **HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Cloudera\Cloudera ODBC Driver for Impala\Driver**
- Otherwise: **HKEY_LOCAL_MACHINE\SOFTWARE\Cloudera\Cloudera ODBC Driver for Impala\Driver**

Driver Configuration Options

Use `UseLogPrefix` as the value name, and either `0` or `1` as the value data.

To configure this option for a non-Windows driver, you must use the `cloudera.impalaodbc.ini` file.

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
Core	SQLDriverconnect		Core	SQLSetDescField
Core	SQLEndTran		Core	SQLSetDescRec
Core	SQLExecDirect		Core	SQLSetEnvAttr
Core	SQLExecute		Core	SQLSetStmtAttr
Core	SQLFetch		Core	SQLSpecialColumns
Core	SQLFetchScroll		Core	SQLStatistics

Conformance Level	INTERFACES		Conformance Level	INTERFACES
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
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.