Hortonworks

Architecting the Future of Big Data

Hortonworks Hive ODBC Driver with SQL Connector

User Guide

Revised: August 7, 2015

Table of Contents

Introduction

The Hortonworks Hive ODBC Driver with SQL Connector is used for direct SQL and HiveQL access to Apache Hadoop / Hive distributions, enabling Business Intelligence (BI), analytics, and reporting on Hadoop / Hive-based data. The driver efficiently transforms an application's SQL query into the equivalent form in HiveQL, which is a subset of SQL-92. If an application is Hive-aware, then the driver is configurable to pass the query through to the database for processing. The driver interrogates Hive to obtain schema information to present to a SQL-based application. Queries, including joins, are translated from SQL to HiveQL. For more information about the differences between HiveQL and SQL, see "Features" on page 41.

The Hortonworks Hive ODBC Driver with SQL Connector complies with the ODBC 3.52 data standard and adds important functionality such as Unicode and 32- and 64-bit support for high-performance computing environments.

ODBC is one 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 http://www.simba.com/resources/data-access-standards-library. For complete information about the ODBC specification, see the *ODBC API Reference* at http://msdn.microsoft.com/en-us/library/windows/desktop/ms714562(v=vs.85).aspx.

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

Contact Us

If you have difficulty using the Hortonworks Hive ODBC Driver with SQL Connector, please contact our support staff. We welcome your questions, comments, and feature requests.

Please have a detailed summary of the client and server environment (OS version, patch level, Hadoop distribution version, Hive version, configuration, etc.) ready, before you call or write us. Supplying this information accelerates support.

By telephone:

USA: (855) 8-HORTON

International: (408) 916-4121

On the Internet:

Visit us at www.hortonworks.com

Windows Driver

System Requirements

You install the Hortonworks Hive ODBC Driver with SQL Connector on client computers accessing data in a Hadoop cluster with the Hive service installed and running. Each computer where you install the driver must meet the following minimum system requirements:

- One of the following operating systems (32- and 64-bit editions are supported):
  - Windows® XP with SP3
  - Windows® Vista
  - Windows® 7 Professional and Enterprise
  - Windows® 8 Pro and Enterprise
  - Windows® Server 2008 R2
- 25 MB of available disk space

Important: To install the driver, you must have Administrator privileges on the computer.

The driver supports Hive 0.11, 0.12, 0.13, 0.14, 1.0, 1.1, and 1.2.

Installing the Driver

On 64-bit Windows operating systems, you can execute 32- and 64-bit applications transparently. You must use the version of the driver matching the bitness of the client application accessing data in Hadoop / Hive:

- **HortonworksHiveODBC32.msi** for 32-bit applications
- **HortonworksHiveODBC64.msi** for 64-bit applications

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

Note: For an explanation of how to use ODBC on 64-bit editions of Windows, see http://www.simba.com/wp-content/uploads/2010/10/HOW-TO-32-bit-vs-64-bit-ODBC-Data-Source-Administrator.pdf

To install the Hortonworks Hive ODBC Driver with SQL Connector:

1. Depending on the bitness of your client application, double-click to run **HortonworksHiveODBC32.msi** or **HortonworksHiveODBC64.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**

Verifying the Version Number

If you need to verify the version of the Hortonworks Hive ODBC Driver with SQL Connector that is installed on your Windows machine, you can find the version number in the ODBC Data Source Administrator.

To verify the version number:

1. If you are using Windows 7 or earlier, click the **Start** button 🟦, then click **All Programs**, then click the **Hortonworks Hive ODBC Driver 2.0** program group corresponding to the bitness of the client application accessing data in Hadoop / Hive, and then click **ODBC Administrator**

   OR

   If you are using Windows 8 or later, on the Start screen, type **ODBC administrator**, and then click the **ODBC Administrator** search result corresponding to the bitness of the client application accessing data in Hadoop / Hive.

2. In the ODBC Data Source Administrator, click the **Drivers** tab and then find the Hortonworks Hive ODBC Driver in the list of ODBC drivers that are installed on your system. The version number is displayed in the Version column.

Creating a Data Source Name

Typically, after installing the Hortonworks Hive ODBC Driver with SQL Connector, you need to create a Data Source Name (DSN).

Alternatively, for information about DSN-less connections, see "Configuring a DSN-less Connection" on page 10.

To create a Data Source Name:

1. If you are using Windows 7 or earlier, click the **Start** button 🟦, then click **All Programs**, then click the **Hortonworks Hive ODBC Driver 2.0** program group corresponding to the bitness of the client application accessing data in Hadoop / Hive, and then click **ODBC Administrator**

   OR

   If you are using Windows 8 or later, on the Start screen, type **ODBC administrator**, and then click the **ODBC Administrator** search result corresponding to the bitness of the client application accessing data in Hadoop / Hive.

2. In the ODBC Data Source Administrator, click the **Drivers** tab, and then scroll down as needed to confirm that the Hortonworks Hive ODBC Driver appears in the alphabetical list of ODBC drivers that are installed on your system.

3. To create a DSN that only the user currently logged into Windows can use, click the **User DSN** tab.

<div align="center">OR</div>

To create a DSN that all users who log into Windows can use, click the **System DSN** tab.

4. Click **Add**

5. In the Create New Data Source dialog box, select **Hortonworks Hive ODBC Driver** and then click **Finish**

6. Use the options in the Hortonworks Hive ODBC Driver DSN Setup dialog box to configure your DSN:

    a) In the Data Source Name field, type a name for your DSN.

    b) Optionally, in the Description field, type relevant details about the DSN.

    c) In the Hive Server Type list, select **Hive Server 1** or **Hive Server 2**

       Note: If you are connecting through Apache ZooKeeper, then **Hive Server 1** is not supported.

    d) To connect to Hive without using the Apache ZooKeeper service, in the Service Discovery Mode list, select **No Service Discovery**

<div align="center">OR</div>

    To enable the driver to discover Hive Server 2 services via the ZooKeeper service, in the Service Discovery Mode list, select **ZooKeeper**

    e) In the Host(s) field, if you selected **No Service Discovery** in step d, then type the IP address or host name of the Hive server.

<div align="center">OR</div>

    If you selected **ZooKeeper** in step d, then type a comma-separated list of ZooKeeper servers. Use the following format, where *zk_host* is the IP address or host name of the ZooKeeper server and *zk_port* is the number of the port that the ZooKeeper server uses:

    `zk_host1:zk_port1,zk_host2:zk_port2`

    f) In the Port field, if you selected **No Service Discovery** in step d, then type the number of the TCP port that the Hive server uses to listen for client connections. Otherwise, do not type a value in the field.

    g) 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 Hive command prompt.

h) In the ZooKeeper Namespace field, if you selected **ZooKeeper** in step d, then type the namespace on ZooKeeper under which Hive Server 2 znodes are added. Otherwise, do not type a value in the field.

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

Note: Hive Server 1 does not support authentication. Most default configurations of Hive Server 2 require User Name authentication. To verify the authentication mechanism that you need to use for your connection, check the configuration of your Hadoop / Hive distribution. For more information, see "Authentication Options" on page 48.

j) Optionally, if the operations against Hive 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.

Note: This option is applicable only when connecting to a Hive Server 2 instance that supports this feature.

k) In the Thrift Transport list, select the transport protocol to use in the Thrift layer.

l) To configure HTTP options such as custom headers, click **HTTP Options**. For more information, see "Configuring HTTP Options" on page 18.

Note: The HTTP options are available only when the Thrift Transport option is set to HTTP.

m) To configure client-server verification over SSL, click **SSL Options**. For more information, see "Configuring SSL Verification" on page 19.

n) To configure advanced driver options, click **Advanced Options**. For more information, see "Configuring Advanced Options" on page 16.

o) To configure server-side properties, click **Advanced Options** and then click **Server Side Properties**. For more information, see "Configuring Server-Side Properties" on page 18.

p) To configure the Temporary Table feature, click **Advanced Options** and then click **Temporary Table Configuration**. For more information, see "Configuring the Temporary Table Feature" on page 20 and "Temporary Table" on page 44.

Important: When connecting to Hive 0.14 or later, the Temporary Tables feature is always enabled and you do not need to configure it in the driver.

q) To configure logging behavior for the driver, click **Logging Options**. For more information, see "Configuring Logging Options" on page 21.

7. 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 Hortonworks Hive ODBC Driver DSN Setup dialog box are correct. Contact your Hive server administrator as needed.

8. To save your settings and close the Hortonworks Hive ODBC Driver DSN Setup dialog box, click **OK**

9. To close the ODBC Data Source Administrator, click **OK**

Configuring a DSN-less Connection

Some client applications provide support for connecting to a data source using a driver without a Data Source Name (DSN). To configure a DSN-less connection, you can use a connection string or the Hortonworks Hive ODBC Driver Configuration tool that is installed with the Hortonworks Hive ODBC Driver with SQL Connector. The following section explains how to use the driver configuration tool. For information about using connection strings, see "DSN-less Connections" on page 47.

To configure a DSN-less connection using the driver configuration tool:

1. If you are using Windows 7 or earlier, click the **Start** button 🌐, then click **All Programs**, and then click the **Hortonworks Hive ODBC Driver 2.0** program group corresponding to the bitness of the client application accessing data in Hadoop / Hive.

    OR

    If you are using Windows 8 or later, click the arrow button at the bottom of the Start screen, and then find the **Hortonworks Hive ODBC Driver 2.0** program group corresponding to the bitness of the client application accessing data in Hadoop / Hive.

2. Click **Driver Configuration**, and then click **OK** if prompted for administrator permission to make modifications to the computer.

    Note: You must have administrator access to the computer in order to run this application because it makes changes to the registry.

3. In the Hive Server Type list, select **Hive Server 1** or **Hive Server 2**

    Note: If you are connecting through Apache ZooKeeper, then **Hive Server 1** is not supported.

4. To connect to Hive without using the Apache ZooKeeper service, in the Service Discovery Mode list, select **No Service Discovery**

    OR

    To enable the driver to discover Hive Server 2 services via the ZooKeeper service, in the Service Discovery Mode list, select **ZooKeeper**

5. In the ZooKeeper Namespace field, if you selected **ZooKeeper** in step 4, then type the namespace on ZooKeeper under which Hive Server 2 znodes are added. Otherwise, do not type a value in the field.

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

   Note: Hive Server 1 does not support authentication. Most default configurations of Hive Server 2 require User Name authentication. To verify the authentication mechanism that you need to use for your connection, check the configuration of your Hadoop / Hive distribution. For more information, see "Authentication Options" on page 48.

7. Optionally, if the operations against Hive 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.

   Note: This option is applicable only when connecting to a Hive Server 2 instance that supports this feature.

8. In the Thrift Transport list, select the transport protocol to use in the Thrift layer.

9. To configure HTTP options such as custom headers, click **HTTP Options**. For more information, see "Configuring HTTP Options" on page 18.

   Note: The HTTP options are available only when the Thrift Transport option is set to HTTP.

10. To configure client-server verification over SSL, click **SSL Options**. For more information, see "Configuring SSL Verification" on page 19.

11. To configure advanced options, click **Advanced Options**. For more information, see "Configuring Advanced Options" on page 16.

12. To configure server-side properties, click **Advanced Options** and then click **Server Side Properties**. For more information, see "Configuring Server-Side Properties" on page 18.

13. To configure the Temporary Table feature, click **Advanced Options** and then click **Temporary Table Configuration**. For more information, see "Temporary Table" on page 44 and "Configuring the Temporary Table Feature" on page 20.

    Important: When connecting to Hive 0.14 or later, the Temporary Tables feature is always enabled and you do not need to configure it in the driver.

14. To save your settings and close the Hortonworks Hive ODBC Driver Configuration tool, click **OK**

Configuring Authentication

Some Hive servers are configured to require authentication for access. To connect to a Hive server, you must configure the Hortonworks Hive ODBC Driver with SQL Connector 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 Hive server requires, see Appendix B "Authentication Options" on page 48.

ODBC applications that connect to Hive Server 2 using a DSN can pass in authentication credentials by defining them in the DSN. To configure authentication for a connection that uses a DSN, use the ODBC Data Source Administrator.

Normally, applications that are not Hive Server 2 aware and that connect using a DSN-less connection do not have a facility for passing authentication credentials to the Hortonworks Hive ODBC Driver with SQL Connector for a connection. However, the Hortonworks Hive ODBC Driver Configuration tool enables you to configure authentication without using a DSN.

Important: Credentials defined in a DSN take precedence over credentials configured using the driver configuration tool. Credentials configured using the driver configuration tool apply for all connections that are made using a DSN-less connection unless the client application is Hive Server 2 aware and requests credentials from the user.

Using No Authentication

When connecting to a Hive server of type Hive Server 1, you must use No Authentication. When you use No Authentication, SASL is not supported.

To configure a connection without authentication:

1. To access authentication options for a DSN, open the ODBC Data Source Administrator where you created the DSN, then select the DSN, and then click **Configure**

<div align="center">OR</div>

To access authentication options for a DSN-less connection, open the Hortonworks Hive ODBC Driver Configuration tool.

2. In the Mechanism list, select **No Authentication**

3. In the Thrift Transport list, select the transport protocol to use in the Thrift layer.

   Important: When using this authentication mechanism, SASL is not supported.

4. If the Hive server is configured to use SSL, then click **SSL Options** to configure SSL for the connection. For more information, see "Configuring SSL Verification" on page 19.

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

Example connection string for Hive Server 1:

```
Driver=Hortonworks Hive ODBC Driver;Host=hs1_host;Port=hs1_
port
;HiveServerType=1;AuthMech=0;ThriftTransport=Binary;Schema=Hive_
database
```

Example connection string for Hive Server 2:

```
Driver=Hortonworks Hive ODBC Driver;Host=hs2_host;Port=hs2_
port
;HiveServerType=2;AuthMech=0;ThriftTransport=Binary;Schema=Hive_
database
```

Using Kerberos

Kerberos must be installed and configured before you can use this authentication mechanism. For more information, see Appendix C "Configuring Kerberos Authentication for Windows" on page 50.

This authentication mechanism is available only for Hive Server 2 on non-HDInsight distributions. When you use Kerberos authentication, the Binary transport protocol is not supported.

To configure Kerberos authentication:

1. To access authentication options for a DSN, open the ODBC Data Source Administrator where you created the DSN, then select the DSN, and then click **Configure**

   OR

   To access authentication options for a DSN-less connection, open the Hortonworks Hive ODBC Driver Configuration tool.

2. In the Mechanism list, select **Kerberos**

3. If your Kerberos setup does not define a default realm or if the realm of your Hive Server 2 host is not the default, then type the Kerberos realm of the Hive Server 2 host in the Realm field.

   OR

   To use the default realm defined in your Kerberos setup, leave the Realm field empty.

4. In the Host FQDN field, type the fully qualified domain name of the Hive Server 2 host.
   Note: To use the Hive server host name as the fully qualified domain name for Kerberos authentication, type **_HOST** in the Host FQDN field.

5. In the Service Name field, type the service name of the Hive server.

6. In the Thrift Transport list, select the transport protocol to use in the Thrift layer.
   Important: When using this authentication mechanism, the Binary transport protocol is not supported.

7. If the Hive server is configured to use SSL, then click **SSL Options** to configure SSL for the connection. For more information, see "Configuring SSL Verification" on page 19.

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

Example connection string:

```
Driver=Hortonworks Hive ODBC Driver;Host=hs2_host;Port=hs2_port;
HiveServerType=2;AuthMech=1;ThriftTransport=SASL;Schema=Hive_
database;KrbRealm=Kerberos_Realm;KrbHostFQDN=hs2_fully_qualified_
domain_name;KrbServiceName=hs2_service_name
```

Using User Name

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

This authentication mechanism is available only for Hive Server 2 on non-HDInsight distributions. Most default configurations of Hive Server 2 require User Name authentication. When you use User Name authentication, SSL is not supported and you must use SASL as the Thrift transport protocol.

To configure User Name authentication:

1. To access authentication options for a DSN, open the ODBC Data Source Administrator where you created the DSN, then select the DSN, and then click **Configure**

<div align="center">OR</div>

   To access authentication options for a DSN-less connection, open the Hortonworks Hive ODBC Driver Configuration tool.

2. In the Mechanism list, select **User Name**

3. In the User Name field, type an appropriate user name for accessing the Hive server.

4. In the Thrift Transport list, select **SASL**

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

Example connection string:

```
Driver=Hortonworks Hive ODBC Driver;Host=hs2_host;Port=hs2_
port;HiveServerType=2;AuthMech=2;ThriftTransport=SASL;Schema=Hive_
database;UID=user_name
```

Using User Name and Password

This authentication mechanism requires a user name and a password.

This authentication mechanism is available only for Hive Server 2 on non-HDInsight distributions.

To configure User Name and Password authentication:

1. To access authentication options for a DSN, open the ODBC Data Source Administrator where you created the DSN, then select the DSN, and then click **Configure**

OR

To access authentication options for a DSN-less connection, open the Hortonworks Hive ODBC Driver Configuration tool.

2. In the Mechanism list, select **User Name and Password**

3. In the User Name field, type an appropriate user name for accessing the Hive server.

4. In the Password field, type the password corresponding to the user name you typed in step 3.

5. In the Thrift Transport list, select the transport protocol to use in the Thrift layer.

6. If the Hive server is configured to use SSL, then click **SSL Options** to configure SSL for the connection. For more information, see "Configuring SSL Verification" on page 19.

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

Example connection string:

```
Driver=Hortonworks Hive ODBC Driver;Host=hs2_host;Port=hs2_
port;HiveServerType=2;AuthMech=3;ThriftTransport=SASL;Schema=Hive_
database;UID=user_name;PWD=password
```

Using Windows Azure HDInsight Emulator

This authentication mechanism is available only for Hive Server 2 instances running on Windows Azure HDInsight Emulator. When you use this authentication mechanism, SSL is not supported and you must use HTTP as the Thrift transport protocol.

To configure a connection to a Hive server on Windows Azure HDInsight Emulator:

1. To access authentication options for a DSN, open the ODBC Data Source Administrator where you created the DSN, then select the DSN, and then click **Configure**

OR

To access authentication options for a DSN-less connection, open the Hortonworks Hive ODBC Driver Configuration tool.

2. In the Mechanism list, select **Windows Azure HDInsight Emulator**

3. In the User Name field, type an appropriate user name for accessing the Hive server.

4. In the Password field, type the password corresponding to the user name you typed in step 3.

5. In the Thrift Transport list, select **HTTP**

6. Click **HTTP Options**, and in the HTTP Path field, type the partial URL corresponding to the Hive server. Click **OK** to save your HTTP settings and close the dialog box.

   Note: If necessary, you can create custom HTTP headers. For more information, see "Configuring HTTP Options" on page 18.

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

Example connection string:

```
Driver=Hortonworks Hive ODBC Driver;Host=HDInsight_Emulator_
host;Port=HDInsight_Emulator_
port;HiveServerType=2;AuthMech=5;Schema=Hive_database;UID=user_
name;PWD=password;ThriftTransport=HTTP;HTTPPath=hs2_HTTP_path
```

Using Windows Azure HDInsight Service

This authentication mechanism is available only for Hive Server 2 on HDInsight distributions. When you use this authentication mechanism, you must enable SSL and use HTTP as the Thrift transport protocol.

To configure a connection to a Hive server on Windows Azure HDInsight Service:

1. To access authentication options for a DSN, open the ODBC Data Source Administrator where you created the DSN, then select the DSN, and then click **Configure**

<div align="center">OR</div>

   To access authentication options for a DSN-less connection, open the Hortonworks Hive ODBC Driver Configuration tool.

2. In the Mechanism list, select **Windows Azure HDInsight Service**

3. In the User Name field, type an appropriate user name for accessing the Hive server.

4. In the Password field, type the password corresponding to the user name you typed in step 3.

5. In the Thrift Transport list, select **HTTP**

6. Click **HTTP Options**, and in the HTTP Path field, type the partial URL corresponding to the Hive server. Click **OK** to save your HTTP settings and close the dialog box.

   Note: If necessary, you can create custom HTTP headers. For more information, see "Configuring HTTP Options" on page 18.

7. Click **SSL Options**, select the **Enable SSL** check box, and configure other SSL settings as needed. For more information, see "Configuring SSL Verification" on page 19.

8. Click **OK** to save your SSL configuration and close the dialog box, and then click **OK** to save your authentication settings and close the dialog box.

Example connection string:

```
Driver=Hortonworks Hive ODBC Driver;Host=Azure_HDInsight_Service_
host;Port=443;HiveServerType=2;AuthMech=6;SSL=1;Schema=Hive_
database;UID=user_
name;PWD=password;ThriftTransport=HTTP;HTTPPath=hs2_HTTP_path
```

Configuring Advanced Options

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

To configure advanced options:

1. To access advanced options for a DSN, open the ODBC Data Source Administrator where you created the DSN, then select the DSN, then click **Configure**, and then click **Advanced Options**

   OR

   To access advanced options for a DSN-less connection, open the Hortonworks Hive ODBC Driver Configuration tool, and then click **Advanced Options**

2. To disable the SQL Connector feature, select the **Use Native Query** check box.

3. To defer query execution to SQLExecute, select the **Fast SQLPrepare** check box.

4. To allow driver-wide configurations to take precedence over connection and DSN settings, select the **Driver Config Take Precedence** check box.

5. To use the asynchronous version of the API call against Hive for executing a query, select the **Use Async Exec** check box.

   Note: This option is applicable only when connecting to a Hive cluster running Hive 0.12.0 or later.

6. To retrieve the names of tables in a database by using the SHOW TABLES query, select the **Get Tables With Query** check box.

   Note: This option is applicable only when connecting to Hive Server 2.

7. 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 **Unicode SQL character types** check box.

8. To enable the driver to return the hive_system table for catalog function calls such as SQLTables and SQLColumns, select the **Show System Table** check box.

9. To handle Kerberos authentication using the SSPI plugin instead of MIT Kerberos by default, select the **Use Only SSPI** check box.

10. In the Rows fetched per block field, type the number of rows to be fetched per block.

11. In the Default string column length field, type the maximum data length for STRING columns.

12. In the Binary column length field, type the maximum data length for BINARY columns.

13. In the Decimal column scale field, type the maximum number of digits to the right of the decimal point for numeric data types.

14. In the Async Exec Poll Interval (ms) field, type the time in milliseconds between each poll for the query execution status.

    Note: This option is applicable only to HDInsight clusters.

15. In the Socket Timeout field, type the number of seconds that an operation can remain idle before it is closed.

    Note: This option is applicable only when asynchronous query execution is being used against Hive Server 2 instances.

16. To save your settings and close the Advanced Options dialog box, click **OK**

Configuring Server-Side Properties

You can use the driver to apply configuration properties to the Hive server.

To configure server-side properties:

1. To configure server-side properties for a DSN, open the ODBC Data Source Administrator where you created the DSN, then select the DSN and click **Configure**, then click **Advanced Options**, and then click **Server Side Properties**

<div align="center">OR</div>

   To configure server-side properties for a DSN-less connection, open the Hortonworks Hive ODBC Driver Configuration tool, 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**

   Note: For a list of all Hadoop and Hive server-side properties that your implementation supports, type **set -v** at the Hive CLI command line or Beeline. You can also execute the set -v query after connecting using the driver.

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 apply each server-side property by executing a query when opening a session to the Hive server, select the **Apply Server Side Properties with Queries** check box.

<div align="center">OR</div>

   To configure the driver to use a more efficient method for applying server-side properties that does not involve additional network round-tripping, clear the **Apply Server Side Properties with Queries** check box.

   Note: The more efficient method is not available for Hive Server 1, and it might not be compatible with some Hive Server 2 builds. If the server-side properties do not take effect when the check box is clear, then select the check box.

6. To force the driver to convert server-side property key names to all lower case characters, select the **Convert Key Name to Lower Case** check box.

7. To save your settings and close the Server Side Properties dialog box, click **OK**

Configuring HTTP Options

You can configure options such as custom headers when using the HTTP transport protocol in the Thrift layer.

To configure HTTP options:

1. If you are configuring HTTP for a DSN, open the ODBC Data Source Administrator where you created the DSN, then select the DSN, then click **Configure**, and then ensure that the Thrift Transport option is set to **HTTP**

OR

If you are configuring HTTP for a DSN-less connection, open the Hortonworks Hive ODBC Driver Configuration tool and then ensure that the Thrift Transport option is set to **HTTP**

2. To access HTTP options, click **HTTP Options**

Note: The HTTP options are available only when the Thrift Transport option is set to HTTP.

3. In the HTTP Path field, type the partial URL corresponding to the Hive 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 Options dialog box, click **OK**

Configuring SSL Verification

You can configure verification between the client and the Hive server over SSL.

To configure SSL verification:

1. To access SSL options for a DSN, open the ODBC Data Source Administrator where you created the DSN, then select the DSN, then click **Configure**, and then click **SSL Options**

OR

To access advanced options for a DSN-less connection, open the Hortonworks Hive ODBC Driver Configuration tool, and then click **SSL Options**

2. Select the **Enable SSL** check box.

3. To allow self-signed certificates from the server, select the **Allow Self-signed Server Certificate** check box.

4. To allow the common name of a CA-issued SSL certificate to not match the host name of the Hive server, select the **Allow Common Name Host Name Mismatch** check box.

5. To configure the driver to load SSL certificates from a specific PEM file when verifying the server, specify the full path to the file in the Trusted Certificates field.

OR

To use the trusted CA certificates PEM file that is installed with the driver, leave the Trusted Certificates field empty.

6. If you want to configure two-way SSL verification, select the **Two Way SSL** check box and then do the following:

   a) In the Client Certificate File field, specify the full path of the PEM file containing the client's certificate.

   b) In the Client Private Key File field, specify the full path of the file containing the client's private key.

   c) If the private key file is protected with a password, type the password in the Client Private Key Password field. To save the password, select the **Save Password (Encrypted)** check box.

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

7. To save your settings and close the SSL Options dialog box, click **OK**

Configuring the Temporary Table Feature

You can configure the driver to create temporary tables. For more information about this feature, including details about the statement syntax used for temporary tables, see "Temporary Table" on page 44.

Important: When connecting to Hive 0.14 or later, the Temporary Tables feature is always enabled and you do not need to configure it in the driver.

To configure the Temporary Table feature:

1. To configure the temporary table feature for a DSN, open the ODBC Data Source Administrator where you created the DSN, then select the DSN and click **Configure**, then click **Advanced Options**, and then click **Temporary Table Configuration**

   OR

   To configure the temporary table feature for a DSN-less connection, open the Hortonworks Hive ODBC Driver Configuration tool, then click **Advanced Options**, and then click **Temporary Table Configuration**

2. To enable the Temporary Table feature, select the **Enable Temporary Table** check box.

3. In the Web HDFS Host field, type the host name or IP address of the machine hosting both the namenode of your Hadoop cluster and the WebHDFS service. If this field is left blank, then the host name of the Hive server will be used.

4. In the Web HDFS Port field, type the WebHDFS port for the namenode.

5. In the HDFS User field, type the name of the HDFS user that the driver will use to create the necessary files for supporting the Temporary Table feature.

6. In the Data file HDFS dir field, type the HDFS directory that the driver will use to store the necessary files for supporting the Temporary Table feature.

   Note: Due to a problem in Hive (see https://issues.apache.org/jira/browse/HIVE-4554), HDFS paths with space characters do not work with versions of Hive prior to 0.12.0.

7. In the Temp Table TTL field, type the number of minutes that a temporary table is guaranteed to exist in Hive after it is created.

8. To save your settings and close the Temporary Table Configuration dialog box, click **OK**

Configuring Logging Options

To help troubleshoot issues, you can enable logging. In addition to functionality provided in the Hortonworks Hive ODBC Driver with SQL Connector, the ODBC Data Source Administrator provides tracing functionality.

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

The driver allows you to set the amount of detail included in log files. Table 1 lists the logging levels provided by the Hortonworks Hive ODBC Driver with SQL Connector, in order from least verbose to most verbose.

Table 1. Hortonworks Hive ODBC Driver with SQL Connector Logging Levels

Logging Level	Description
OFF	Disables all logging.
FATAL	Logs very severe error events that will lead the driver to abort.
ERROR	Logs error events that might still allow the driver to continue running.
WARNING	Logs potentially harmful situations.
INFO	Logs general information that describes the progress of the driver.
DEBUG	Logs detailed information that is useful for debugging the driver.
TRACE	Logs more detailed information than the DEBUG level.

To enable the logging functionality available in the Hortonworks Hive ODBC Driver with SQL Connector:

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. In the Log Level list, select the desired level of information to include in log files.

3. In the Log Path field, type 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. Click **OK**

6. Restart your ODBC application to ensure that the new settings take effect.

The Hortonworks Hive ODBC Driver with SQL Connector produces a log file named HiveODBC_driver.log at the location you specify using the Log Path field.

To disable Hortonworks Hive ODBC Driver with SQL Connector logging:

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. In the Log Level list, select **LOG_OFF**

3. Click **OK**

To start tracing using the ODBC Data Source Administrator:

1. In the ODBC Data Source Administrator, click the **Tracing** tab.

2. In the Log File Path area, click **Browse**. In the Select ODBC Log File dialog box, browse to the location where you want to save the log file, then type a descriptive file name in the File name field, and then click **Save**

3. On the Tracing tab, click **Start Tracing Now**

To stop ODBC Data Source Administrator tracing:

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

For more information about tracing using the ODBC Data Source Administrator, see the article *How to Generate an ODBC Trace with ODBC Data Source Administrator* at http://support.microsoft.com/kb/274551

Linux Driver

System Requirements

You install the Hortonworks Hive ODBC Driver with SQL Connector on client computers accessing data in a Hadoop cluster with the Hive service installed and running. Each computer where you install the driver must meet the following minimum system requirements:

- One of the following distributions (32- and 64-bit editions are supported):
  - Red Hat® Enterprise Linux® (RHEL) 5.0 or 6.0
  - CentOS 5.0 or 6.0
  - SUSE Linux Enterprise Server (SLES) 11
- 45 MB of available disk space
- One of the following ODBC driver managers installed:
  - iODBC 3.52.7 or later
  - unixODBC 2.2.12 or later

The driver supports Hive 0.11, 0.12, 0.13, 0.14, 1.0, 1.1, and 1.2.

Installing the Driver

There are two versions of the driver for Linux:

- **hive-odbc-native-32bit-*Version-Release.LinuxDistro*.i686.rpm** for the 32-bit driver
- **hive-odbc-native-*Version-Release.LinuxDistro*.x86_64.rpm** for the 64-bit driver

Version is the version number of the driver, and *Release* is the release number for this version of the driver. *LinuxDistro* is either el5 or el6. For SUSE, the *LinuxDistro* placeholder is empty.

The bitness of the driver that you select should match the bitness of the client application accessing your Hadoop / Hive-based data. For example, if the client application is 64-bit, then you should install the 64-bit driver. Note that 64-bit editions of Linux support both 32- and 64-bit applications. Verify the bitness of your intended application and install the appropriate version of the driver.

Important: Ensure that you install the driver using the RPM corresponding to your Linux distribution.

The Hortonworks Hive ODBC Driver with SQL Connector driver files are installed in the following directories:

- **/usr/lib/hive/lib/native/hiveodbc** contains release notes, the *Hortonworks Hive ODBC Driver with SQL Connector User Guide* in PDF format, and a Readme.txt file that provides plain text installation and configuration instructions.
- **/usr/lib/hive/lib/native/hiveodbc/ErrorMessages** contains error message files required by the driver.
- **/usr/lib/hive/lib/native/hiveodbc/Setup** contains sample configuration files named odbc.ini and odbcinst.ini
- **/usr/lib/hive/lib/native/Linux-i386-32** contains the 32-bit driver and the hortonworks.hiveodbc.ini configuration file.
- **/usr/lib/hive/lib/native/Linux-amd64-64** contains the 64-bit driver and the hortonworks.hiveodbc.ini configuration file.

To install the Hortonworks Hive ODBC Driver with SQL Connector:

1. In Red Hat Enterprise Linux or CentOS, log in as the root user, then navigate to the folder containing the driver RPM packages to install, and then type the following at the command line, where *RPMFileName* is the file name of the RPM package containing the version of the driver that you want to install:

```
yum --nogpgcheck localinstall RPMFileName
```

OR

In SUSE Linux Enterprise Server, log in as the root user, then navigate to the folder containing the driver RPM packages to install, and then type the following at the command line, where *RPMFileName* is the file name of the RPM package containing the version of the driver that you want to install:

```
zypper install RPMFileName
```

The Hortonworks Hive ODBC Driver with SQL Connector depends on the following resources:

- cyrus-sasl-2.1.22-7 or above
- cyrus-sasl-gssapi-2.1.22-7 or above
- cyrus-sasl-plain-2.1.22-7 or above

If the package manager in your Linux distribution cannot resolve the 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.

Verifying the Version Number

If you need to verify the version of the Hortonworks Hive ODBC Driver with SQL Connector that is installed on your Linux machine, you can query the version number through the command-line interface.

To verify the version number:

➢ At the command prompt, run the following command:

```
yum list | grep hive-odbc-native
```

<div align="center">OR</div>

Run the following command:

```
rpm -qa | grep hive-odbc-native
```

The command returns information about the Hortonworks Hive ODBC Driver with SQL Connector that is installed on your machine, including the version number.

Setting the LD_LIBRARY_PATH Environment Variable

The LD_LIBRARY_PATH environment variable must include the paths to the installed ODBC driver manager libraries.

For example, if ODBC driver manager libraries are installed in /usr/local/lib, then set LD_LIBRARY_PATH as follows:

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

For information about how to set environment variables permanently, refer to your Linux shell documentation.

For information about creating ODBC connections using the Hortonworks Hive ODBC Driver with SQL Connector, see "Configuring ODBC Connections for Non-Windows Platforms" on page 29.

Mac OS X Driver

System Requirements

You install the Hortonworks Hive ODBC Driver with SQL Connector on client computers accessing data in a Hadoop cluster with the Hive service installed and running. Each computer where you install the driver must meet the following minimum system requirements:

- Mac OS X version 10.6.8 or later
- 100 MB of available disk space
- iODBC 3.52.7 or later

The driver supports Hive 0.11, 0.12, 0.13, 0.14, 1.0, 1.1, and 1.2. The driver supports both 32- and 64-bit client applications.

Installing the Driver

The Hortonworks Hive ODBC Driver with SQL Connector driver files are installed in the following directories:

- **/usr/lib/hive/lib/native/hiveodbc** contains release notes and the *Hortonworks Hive ODBC Driver with SQL Connector User Guide* in PDF format.
- **/usr/lib/hive/lib/native/hiveodbc/ErrorMessages** contains error message files required by the driver.
- **/usr/lib/hive/lib/native/hiveodbc/Setup** contains sample configuration files named odbc.ini and odbcinst.ini
- **/usr/lib/hive/lib/native/universal** contains the driver and the hortonworks.hiveodbc.ini configuration file.

To install the Hortonworks Hive ODBC Driver with SQL Connector:

1. Double-click **hive-odbc-native.dmg** to mount the disk image.
2. Double-click **hive-odbc-native.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**
6. To accept the installation location and begin the installation, click **Install**
7. When the installation completes, click **Close**

Verifying the Version Number

If you need to verify the version of the Hortonworks Hive ODBC Driver with SQL Connector that is installed on your Mac OS X machine, you can query the version number through the Terminal.

To verify the version number:

➢ At the Terminal, run the following command:

```
pkgutil --info hortonworks.hiveodbc
```

The command returns information about the Hortonworks Hive ODBC Driver with SQL Connector that is installed on your machine, including the version number.

Setting the DYLD_LIBRARY_PATH Environment Variable

The DYLD_LIBRARY_PATH environment variable must include the paths to the installed ODBC driver manager libraries.

For example, if ODBC driver manager libraries are installed in /usr/local/lib, then set DYLD_ LIBRARY_PATH as follows:

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

For information about how to set environment variables permanently, refer to your Mac OS X shell documentation.

For information about creating ODBC connections using the Hortonworks Hive ODBC Driver with SQL Connector, see "Configuring ODBC Connections for Non-Windows Platforms" on page 29.

Debian Driver

System Requirements

You install the Hortonworks Hive ODBC Driver with SQL Connector on client computers accessing data in a Hadoop cluster with the Hive service installed and running. Each computer where you install the driver must meet the following minimum system requirements:

- Debian 6 or 7 (Ubuntu 12.04 LTS and Ubuntu 14.04 LTS)
- 45 MB of available disk space
- One of the following ODBC driver managers installed:
  - iODBC 3.52.7 or later
  - unixODBC 2.2.12 or later

The driver supports Hive 0.11, 0.12, 0.13, 0.14, 1.0, 1.1, and 1.2. It supports both 32- and 64-bit client applications.

Installing the Driver

There are two versions of the driver for Debian:

- **hive-odbc-native-32bit-*Version-Release*_i386.deb** for the 32-bit driver
- **hive-odbc-native-*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.

The bitness of the driver that you select should match the bitness of the client application accessing your Hadoop / Hive-based data. For example, if the client application is 64-bit, then you should install the 64-bit driver. Note that 64-bit editions of Debian support both 32- and 64-bit applications. Verify the bitness of your intended application and install the appropriate version of the driver.

The Hortonworks Hive ODBC Driver with SQL Connector driver files are installed in the following directories:

- **/usr/lib/hive/lib/native/hiveodbc/ErrorMessages** contains error message files required by the driver.

- **/usr/lib/hive/lib/native/hiveodbc/Setup** contains sample configuration files named odbc.ini and odbcinst.ini

- **/usr/lib/hive/lib/native/Linux-i386-32** contains the 32-bit driver and the hortonworks.hiveodbc.ini configuration file.

- **/usr/lib/hive/lib/native/Linux-amd64-64** contains the 64-bit driver and the hortonworks.hiveodbc.ini configuration file.

To install the Hortonworks Hive ODBC Driver with SQL Connector:

1. In Ubuntu, log in as the root user, then navigate to the folder containing the driver Debian packages to install, and double-click **hive-odbc-native-32bit-*Version-Release*_i386.deb** or **hive-odbc-native-*Version-Release*_amd64.deb**

2. Follow the instructions in the installer to complete the installation process.

3. If you received a license file via e-mail, then copy the license file into the /opt/hortonworks/hiveodbc/lib/32 or /opt/hortonworks/hiveodbc/lib/64 folder, depending on the version of the driver that you installed.

   Note: To avoid security issues, you may need to save the license file on your local computer prior to copying the file into the folder.

The Hortonworks Hive ODBC Driver with SQL Connector depends on the following resources:

- cyrus-sasl-2.1.22-7 or above

- cyrus-sasl-gssapi-2.1.22-7 or above

- cyrus-sasl-plain-2.1.22-7 or above

If the package manager in your Ubuntu distribution cannot resolve the 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.

Verifying the Version Number

If you need to verify the version of the Hortonworks Hive ODBC Driver with SQL Connector that is installed on your Debian machine, you can query the version number through the command-line interface.

To verify the version number:

➢ At the command prompt, run the following command:

```
dpkg -l | grep hive-odbc-native
```

The command returns information about the Hortonworks Hive ODBC Driver with SQL Connector that is installed on your machine, including the version number.

Setting the LD_LIBRARY_PATH Environment Variable

The LD_LIBRARY_PATH environment variable must include the path to the installed ODBC driver manager libraries.

For example, if ODBC driver manager libraries are installed in /usr/local/lib, then set LD_LIBRARY_PATH as follows:

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

For information about how to set environment variables permanently, refer to your Ubuntu shell documentation.

For information about creating ODBC connections using the Hortonworks Hive ODBC Driver with SQL Connector, see "Configuring ODBC Connections for Non-Windows Platforms" on page 29.

Configuring ODBC Connections for Non-Windows Platforms

The following sections describe how to configure ODBC connections when using the Hortonworks Hive ODBC Driver with SQL Connector with non-Windows platforms:

Files

ODBC driver managers use configuration files to define and configure ODBC data sources and drivers. By default, the following configuration files residing in the user's home directory are used:

- **.odbc.ini** is used to define ODBC data sources, and it is required for DSNs.
- **.odbcinst.ini** is used to define ODBC drivers, and it is optional.

Also, by default the Hortonworks Hive ODBC Driver with SQL Connector is configured using the hortonworks.hiveodbc.ini file, which is located in one of the following directories depending on the version of the driver that you are using:

- **/usr/lib/hive/lib/native/Linux-i386-32** for the 32-bit driver on Linux/Debian
- **/usr/lib/hive/lib/native/Linux-amd64-64** for the 64-bit driver on Linux/Debian
- **/usr/lib/hive/lib/native/universal** for the driver on Mac OS X

The hortonworks.hiveodbc.ini file is required.

Note: The hortonworks.hiveodbc.ini file in the /lib subfolder provides default settings for most configuration options available in the Hortonworks Hive ODBC Driver with SQL Connector.

You can set driver configuration options in your odbc.ini and hortonworks.hiveodbc.ini files. Configuration options set in a hortonworks.hiveodbc.ini file apply to all connections, whereas configuration options set in an odbc.ini file are specific to a connection. Configuration options set in odbc.ini take precedence over configuration options set in hortonworks.hiveodbc.ini. For information about the configuration options available for controlling the behavior of DSNs that are using the Hortonworks Hive ODBC Driver with SQL Connector, see Appendix D "Driver Configuration Options" on page 55.

Sample Files

The driver installation contains the following sample configuration files in the Setup directory:

- odbc.ini
- odbcinst.ini

These sample configuration files provide preset values for settings related to the Hortonworks Hive ODBC Driver with SQL Connector.

The names of the sample configuration files do not begin with a period (.) so that they will appear in directory listings by default. A file name beginning with a period (.) is hidden. For odbc.ini and odbcinst.ini, if the default location is used, then the file names must begin with a period (.).

If the configuration files do not exist in the home directory, then you can copy the sample configuration files to the home directory, and then rename the files. If the configuration files already exist in the home directory, then use the sample configuration files as a guide to modify the existing configuration files.

Configuring the Environment

Optionally, you can use three environment variables—ODBCINI, ODBCSYSINI, and HORTONWORKSHIVEINI—to specify different locations for the odbc.ini, odbcinst.ini, and hortonworks.hiveodbc.ini configuration files by doing the following:

- Set ODBCINI to point to your odbc.ini file.
- Set ODBCSYSINI to point to the directory containing the odbcinst.ini file.
- Set HORTONWORKSHIVEINI to point to your hortonworks.hiveodbc.ini file.

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

```
export ODBCINI=/etc/odbc.ini
export ODBCSYSINI=/usr/local/odbc
export HORTONWORKSHIVEINI=/etc/hortonworks.hiveodbc.ini
```

The following search order is used to locate the hortonworks.hiveodbc.ini file:

1. If the HORTONWORKSHIVEINI environment variable is defined, then the driver searches for the file specified by the environment variable.

   Important: HORTONWORKSHIVEINI must specify the full path, including the file name.

2. The directory containing the driver's binary is searched for a file named hortonworks.hiveodbc.ini (not beginning with a period).

3. The current working directory of the application is searched for a file named hortonworks.hiveodbc.ini (not beginning with a period).

4. The directory ~/ (that is, $HOME) is searched for a hidden file named .hortonworks.hiveodbc.ini

5. The directory /etc is searched for a file named hortonworks.hiveodbc.ini (not beginning with a period).

Configuring the odbc.ini File

Note: If you are using a DSN-less connection, then you do not need to configure the odbc.ini file. For information about configuring a DSN-less connection, see "DSN-less Connections" on page 47.

ODBC Data Source Names (DSNs) are defined in the odbc.ini configuration file. The file is divided into several sections:

- **[ODBC]** is optional and used to control global ODBC configuration, such as ODBC tracing.
- **[ODBC Data Sources]** is required, listing DSNs and associating DSNs with a driver.
- A section having the same name as the data source specified in the [ODBC Data Sources] section is required to configure the data source.

The following is an example of an odbc.ini configuration file for Linux/Debian:

```
[ODBC]
InstallDir=/usr/local/odbc
[ODBC Data Sources]
Sample Hortonworks Hive DSN 32=Hortonworks Hive ODBC Driver 32-bit
[Sample Hortonworks Hive DSN 32]
Driver=/usr/lib/hive/lib/native/Linux-i386-
32/libhortonworkshiveodbc32.so
HOST=MyHiveServer
PORT=10000
```

MyHiveServer is the IP address or host name of the Hive server.

The following is an example of an odbc.ini configuration file for Mac OS X:

```
[ODBC]
InstallDir=/usr/local/odbc
[ODBC Data Sources]
Sample Hortonworks Hive DSN=Hortonworks Hive ODBC Driver
[Sample Hortonworks Hive DSN]
Driver=/usr/lib/hive/lib/native/universal/libhortonworkshiveodbc.dy
lib
HOST=MyHiveServer
PORT=10000
```

MyHiveServer is the IP address or host name of the Hive server.

To create a Data Source Name:

1. Open the .odbc.ini configuration file in a text editor.
2. In the [ODBC Data Sources] section, add a new entry by typing the Data Source Name (DSN), then an equal sign (=), and then the driver name.

3. In the .odbc.ini file, add a new section with a name that matches the DSN you specified in step 2, and then add configuration options to the section. Specify configuration options as key-value pairs.

   Note: Hive Server 1 does not support authentication. Most default configurations of Hive Server 2 require User Name authentication, which you configure by setting the AuthMech key to 2. To verify the authentication mechanism that you need to use for your connection, check the configuration of your Hadoop / Hive distribution. For more information, see "Authentication Options" on page 48.

4. Save the .odbc.ini configuration file.

For information about the configuration options available for controlling the behavior of DSNs that are using the Hortonworks Hive ODBC Driver with SQL Connector, see Appendix D "Driver Configuration Options" on page 55.

Configuring the odbcinst.ini File

ODBC drivers are defined in the odbcinst.ini configuration file. The configuration file is optional because drivers can be specified directly in the odbc.ini configuration file, as described in "Configuring the odbc.ini File" on page 31.

The odbcinst.ini file is divided into the following sections:

- **[ODBC Drivers]** lists the names of all the installed ODBC drivers.
- A section having the same name as the driver name specified in the [ODBC Drivers] section lists driver attributes and values.

The following is an example of an odbcinst.ini configuration file for Linux/Debian:

```
[ODBC Drivers]

Hortonworks Hive ODBC Driver 32-bit=Installed

Hortonworks Hive ODBC Driver 64-bit=Installed

[Hortonworks Hive ODBC Driver 32-bit]

Description=Hortonworks Hive ODBC Driver (32-bit)

Driver=/usr/lib/hive/lib/native/Linux-i386-
32/libhortonworkshiveodbc32.so

[Hortonworks Hive ODBC Driver 64-bit]

Description=Hortonworks Hive ODBC Driver (64-bit)

Driver=/usr/lib/hive/lib/native/Linux-amd64-
64/libhortonworkshiveodbc64.so
```

The following is an example of an odbcinst.ini configuration file for Mac OS X:

```
[ODBC Drivers]

Hortonworks Hive ODBC Driver=Installed

[Hortonworks Hive ODBC Driver]
```

```
Description=Hortonworks Hive ODBC Driver
Driver=/usr/lib/hive/lib/native/universal/libhortonworkshiveodbc.dy
lib
```

To define a driver:

1. Open the .odbcinst.ini configuration file in a text editor.

2. In the [ODBC Drivers] section, add a new entry by typing the driver name and then typing **=Installed**

   Note: Type a symbolic name that you want to use to refer to the driver in connection strings or DSNs.

3. In the .odbcinst.ini file, add a new section with a name that matches the driver name you typed in step 2, and then add configuration options to the section based on the sample odbcinst.ini file provided in the Setup directory. Specify configuration options as key-value pairs.

4. Save the .odbcinst.ini configuration file.

Configuring the hortonworks.hiveodbc.ini File

The hortonworks.hiveodbc.ini file contains configuration settings for the Hortonworks Hive ODBC Driver with SQL Connector. Settings that you define in the hortonworks.hiveodbc.ini file apply to all connections that use the driver.

To configure the Hortonworks Hive ODBC Driver with SQL Connector to work with your ODBC driver manager:

1. Open the hortonworks.hiveodbc.ini configuration file in a text editor.

2. Edit the DriverManagerEncoding setting. The value is usually **UTF-16** or **UTF-32**, depending on the ODBC driver manager you use. iODBC uses **UTF-32**, and unixODBC uses **UTF-16**. To determine the correct setting to use, refer to your ODBC driver manager documentation.

3. Edit the ODBCInstLib setting. The value is the name of the ODBCInst shared library for the ODBC driver manager you use. To determine the correct library to specify, refer to your ODBC driver manager documentation.

   The configuration file defaults to the shared library for iODBC. In Linux/Debian, the shared library name for iODBC is **libiodbcinst.so**. In Mac OS X, the shared library name for iODBC is **libiodbcinst.dylib**.

   Note: You can specify an absolute or relative filename for the library. If you intend to use the relative filename, then the path to the library must be included in the library path environment variable. In Linux/Debian, the library path environment variable is named LD_LIBRARY_PATH. In Mac OS X, the library path environment variable is named DYLD_LIBRARY_PATH.

4. Optionally, configure logging by editing the LogLevel and LogPath settings. For more information, see "Configuring Logging Options" on page 40.

5. Save the hortonworks.hiveodbc.ini configuration file.

Configuring Service Discovery Mode

You can configure the Hortonworks Hive ODBC Driver with SQL Connector to discover Hive Server 2 services via ZooKeeper.

To enable service discovery via ZooKeeper:

1. Open the odbc.ini configuration file in a text editor.

2. Set the ServiceDiscoveryMode connection attribute to 1

3. Set the ZKNamespace connection attribute to specify the namespace on ZooKeeper under which Hive Server 2 znodes are added.

4. Set the Host connection attribute to specify the ZooKeeper ensemble as a comma-separated list of ZooKeeper servers. For example, type the following, where *zk_host* is the IP address or host name of the ZooKeeper server and *zk_port* is the number of the port that the ZooKeeper server uses:

   zk_host1:*zk_port1*,*zk_host2*:*zk_port2*

Important: When ServiceDiscoveryMode is set to 1, connections to Hive Server 1 are not supported and the Port connection attribute is not applicable.

Depending on whether service discovery mode is enabled or disabled, you may need to provide different connection attributes or values in your connection string or DSN. For more information about connection attributes, see "Driver Configuration Options" on page 55.

Configuring Authentication

Some Hive servers are configured to require authentication for access. To connect to a Hive server, you must configure the Hortonworks Hive ODBC Driver with SQL Connector 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 Hive server requires, see Appendix B "Authentication Options" on page 48.

You can select the type of authentication to use for a connection by defining the AuthMech connection attribute in a connection string or in a DSN (in the odbc.ini file). Depending on the authentication mechanism you use, there may be additional connection attributes that you must define. For more information about the attributes involved in configuring authentication, see Appendix D "Driver Configuration Options" on page 55.

Using No Authentication

When connecting to a Hive server of type Hive Server 1, you must use No Authentication. When you use No Authentication, SASL is not supported.

To configure a connection without authentication:

1. Set the AuthMech connection attribute to 0

2. Set the ThriftTransport connection attribute to the transport protocol to use in the Thrift layer.

   Important: When using this authentication mechanism, SASL (**ThriftTransport=1**) is not supported.

3. If the Hive server is configured to use SSL, then configure SSL for the connection. For more information, see "Configuring SSL Verification" on page 39.

Example connection string for Hive Server 1:

```
Driver=Hortonworks Hive ODBC Driver;Host=hs1_host;Port=hs1_
port;HiveServerType=1;AuthMech=0;ThriftTransport=BinarySchema=Hive_
database
```

Example connection string for Hive Server 2:

```
Driver=Hortonworks Hive ODBC Driver;Host=hs2_host;Port=hs2_
port;HiveServerType=2;AuthMech=0;ThriftTransport=BinarySchema=Hive_
database
```

Using Kerberos

Kerberos must be installed and configured before you can use this authentication mechanism. For more information, refer to the MIT Kerberos documentation.

This authentication mechanism is available only for Hive Server 2 on non-HDInsight distributions. When you use Kerberos authentication, the Binary transport protocol is not supported.

To configure Kerberos authentication:

1. Set the AuthMech connection attribute to 1

2. If your Kerberos setup does not define a default realm or if the realm of your Hive server is not the default, then set the appropriate realm using the KrbRealm attribute.

   OR

   To use the default realm defined in your Kerberos setup, do not set the KrbRealm attribute.

3. Set the KrbHostFQDN attribute to the fully qualified domain name of the Hive Server 2 host.

   Note: To use the Hive server host name as the fully qualified domain name for Kerberos authentication, set KrbHostFQDN to **_HOST**

4. Set the KrbServiceName attribute to the service name of the Hive server.

5. Set the ThriftTransport connection attribute to the transport protocol to use in the Thrift layer.

Important: When using this authentication mechanism, Binary (**ThriftTransport=0**) is not supported.

6. If the Hive server is configured to use SSL, then configure SSL for the connection. For more information, see "Configuring SSL Verification" on page 39.

Example connection string:

```
Driver=Hortonworks Hive ODBC Driver;Host=hs2_host;Port=hs2_port;
HiveServerType=2;AuthMech=1;ThriftTransport=SASL;Schema=Hive_
database;KrbRealm=Kerberos_Realm;KrbHostFQDN=hs2_fully_qualified_
domain_name;KrbServiceName=hs2_service_name
```

Using User Name

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

This authentication mechanism is available only for Hive Server 2 on non-HDInsight distributions. Most default configurations of Hive Server 2 require User Name authentication. When you use User Name authentication, SSL is not supported and you must use SASL as the Thrift transport protocol.

To configure User Name authentication:
1. Set the AuthMech connection attribute to 2
2. Set the UID attribute to an appropriate user name for accessing the Hive server.
3. Set the ThriftTransport connection attribute to 1

Example connection string:

```
Driver=Hortonworks Hive ODBC Driver;Host=hs2_host;Port=hs2_
port;HiveServerType=2;AuthMech=2;ThriftTransport=SASL;Schema=Hive_
database;UID=user_name
```

Using User Name and Password

This authentication mechanism requires a user name and a password.

This authentication mechanism is available only for Hive Server 2 on non-HDInsight distributions.

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 Hive server.
3. Set the PWD attribute to the password corresponding to the user name you provided in step 2.

4. Set the ThriftTransport connection attribute to the transport protocol to use in the Thrift layer.

5. If the Hive server is configured to use SSL, then configure SSL for the connection. For more information, see "Configuring SSL Verification" on page 39.

Example connection string:

```
Driver=Hortonworks Hive ODBC Driver;Host=hs2_host;Port=hs2_
port;HiveServerType=2;AuthMech=3;ThriftTransport=SASL;Schema=Hive_
database;UID=user_name;PWD=password
```

Using Windows Azure HDInsight Emulator

This authentication mechanism is available only for Hive Server 2 instances running on Windows Azure HDInsight Emulator. When you use this authentication mechanism, SSL is not supported and you must use HTTP as the Thrift transport protocol.

To configure a connection to a Hive server on Windows Azure HDInsight Emulator:

1. Set the AuthMech connection attribute to 5

2. Set the HTTPPath attribute to the partial URL corresponding to the Hive server.

3. Set the UID attribute to an appropriate user name for accessing the Hive server.

4. Set the PWD attribute to the password corresponding to the user name you provided in step 3.

5. Set the ThriftTransport connection attribute to 2

   Note: If necessary, you can create custom HTTP headers. For more information, see "http.header." on page 74.

Example connection string:

```
Driver=Hortonworks Hive ODBC Driver;Host=HDInsight_Emulator_
host;Port=HDInsight_Emulator_
port;HiveServerType=2;AuthMech=5;Schema=Hive_database;UID=user_
name;PWD=password;ThriftTransport=HTTP;HTTPPath=hs2_HTTP_path
```

Using Windows Azure HDInsight Service

This authentication mechanism is available only for Hive Server 2 on HDInsight distributions. When you use this authentication mechanism, you must enable SSL and use HTTP as the Thrift transport protocol.

To configure a connection to a Hive server on Windows Azure HDInsight Service:

1. Set the AuthMech connection attribute to 6

2. Set the HTTPPath attribute to the partial URL corresponding to the Hive server.

3. Set the UID attribute to an appropriate user name for accessing the Hive server.

4. Set the PWD attribute to the password corresponding to the user name you typed in step 3.

5. Set the ThriftTransport connection attribute to 2

   Note: If necessary, you can create custom HTTP headers. For more information, see "http.header." on page 74.

6. Set the SSL connection attribute to 1, and configure other SSL settings as needed. For more information, see "Configuring SSL Verification" on page 39.

Example connection string:

```
Driver=Hortonworks Hive ODBC Driver;Host=Azure_HDInsight_Service_
host;Port=443;HiveServerType=2;AuthMech=6;SSL=1;Schema=Hive_
database;UID=user_
name;PWD=password;ThriftTransport=HTTP;HTTPPath=hs2_HTTP_path
```

Configuring SSL Verification

You can configure verification between the client and the Hive server over SSL.

To configure SSL verification:

1. Open the odbc.ini configuration file in a text editor.

2. To enable SSL, set the SSL attribute to 1

3. To allow self-signed certificates from the server, set the AllowSelfSignedServerCert attribute to 1.

4. To allow the common name of a CA-issued SSL certificate to not match the host name of the Hive server, set the CAIssuedCertNamesMismatch attribute to 1.

5. 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.

6. If you want to configure two-way SSL verification, set the TwoWaySSL attribute to 1 and then do the following:

   a) Set the ClientCert attribute to the full path of the PEM file containing the client's certificate.

   b) Set the ClientPrivateKey attribute to the full path of the file containing the client's private key.

   c) If the private key file is protected with a password, set the ClientPrivateKeyPassword attribute to the password.

7. Save the odbc.ini configuration file.

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.

Use the LogLevel key to set the amount of detail included in log files. Table 2 lists the logging levels provided by the Hortonworks Hive ODBC Driver with SQL Connector, in order from least verbose to most verbose.

Table 2. Hortonworks Hive ODBC Driver with SQL Connector Logging Levels

LogLevel Value	Description
0	Disables all logging.
1	Logs very severe error events that will lead the driver to abort.
2	Logs error events that might still allow the driver to continue running.
3	Logs potentially harmful situations.
4	Logs general information that describes the progress of the driver.
5	Logs detailed information that is useful for debugging the driver.
6	Logs more detailed information than LogLevel=5

To enable logging:

1. Open the hortonworks.hiveodbc.ini configuration file in a text editor.
2. Set the LogLevel key to the desired level of information to include in log files. For example:

   ```
   LogLevel=2
   ```
3. Set the LogPath key to the full path to the folder where you want to save log files. For example:

   ```
   LogPath=/localhome/employee/Documents
   ```
4. Save the hortonworks.hiveodbc.ini configuration file.
5. Restart your ODBC application to ensure that the new settings take effect.

The Hortonworks Hive ODBC Driver with SQL Connector produces a log file named HiveODBC_driver.log at the location you specify using the LogPath key.

To disable logging:

1. Open the hortonworks.hiveodbc.ini configuration file in a text editor.

2. Set the LogLevel key to 0

3. Save the hortonworks.hiveodbc.ini configuration file.

Features

More information is provided on the following features of the Hortonworks Hive ODBC Driver with SQL Connector:

- "SQL Query versus HiveQL Query" on page 41
- "SQL Connector" on page 41
- "Data Types" on page 42
- "Catalog and Schema Support" on page 43
- "hive_system Table" on page 43
- "Server-Side Properties" on page 43
- "Get Tables With Query" on page 45
- "Active Directory" on page 45
- "Write-back" on page 46
- "Dynamic Service Discovery using ZooKeeper" on page 46

SQL Query versus HiveQL Query

The native query language supported by Hive is HiveQL. For simple queries, HiveQL is a subset of SQL-92. However, the syntax is different enough that most applications do not work with native HiveQL.

SQL Connector

To bridge the difference between SQL and HiveQL, the SQL Connector feature translates standard SQL-92 queries into equivalent HiveQL queries. The SQL Connector performs syntactical translations and structural transformations. For example:

- **Quoted Identifiers** — The double quotes (") that SQL uses to quote identifiers are translated into back quotes (`) to match HiveQL syntax. The SQL Connector needs to handle this translation because even when a driver reports the back quote as the quote character, some applications still generate double-quoted identifiers.

- **Table Aliases** — Support is provided for the AS keyword between a table reference and its alias, which HiveQL normally does not support.

- **JOIN**, **INNER JOIN**, and **CROSS JOIN** — SQL JOIN, INNER JOIN, and CROSS

JOIN syntax is translated to HiveQL JOIN syntax.

- **TOP N/LIMIT** — SQL TOP N queries are transformed to HiveQL LIMIT queries.

Data Types

The Hortonworks Hive ODBC Driver with SQL Connector supports many common data formats, converting between Hive data types and SQL data types.

Table 3 lists the supported data type mappings.

Table 3. Supported Data Types

Hive Type	SQL Type
TINYINT	SQL_TINYINT
SMALLINT	SQL_SMALLINT
INT	SQL_INTEGER
BIGINT	SQL_BIGINT
FLOAT	SQL_REAL
DOUBLE	SQL_DOUBLE
DECIMAL	SQL_DECIMAL
BOOLEAN	SQL_BIT
STRING	SQL_VARCHAR **Note:** SQL_WVARCHAR is returned instead if the Unicode SQL Character Types configuration option (the UseUnicodeSqlCharacterTypes key) is enabled.
TIMESTAMP	SQL_TYPE_TIMESTAMP
VARCHAR(n)	SQL_VARCHAR
DATE	SQL_TYPE_DATE
DECIMAL(p,s)	SQL_DECIMAL
CHAR(n)	SQL_CHAR

Hive Type	SQL Type
	Note: SQL_WCHAR is returned instead if the Unicode SQL Character Types configuration option (the UseUnicodeSqlCharacterTypes key) is enabled.
BINARY	SQL_VARBINARY

Note:
The aggregate types (ARRAY, MAP, and STRUCT) are not yet supported.
Columns of aggregate types are treated as STRING columns.

The interval types (YEAR TO MONTH and DAY TIME) are supported only in query expressions and predicates. Interval types are not supported as column data types in tables.

Catalog and Schema Support

The Hortonworks Hive ODBC Driver with SQL Connector supports both catalogs and schemas in order to make it easy for the driver to work with various ODBC applications. Since Hive only organizes tables into schemas/databases, the driver provides a synthetic catalog called "HIVE" under which all of the schemas/databases are organized. The driver also maps the ODBC schema to the Hive schema/database.

hive_system Table

A pseudo-table called hive_system can be used to query for Hive cluster system environment information. The pseudo-table is under the pseudo-schema called hive_ system. The table has two STRING type columns, envkey and envvalue. Standard SQL can be executed against the hive_system table. For example:

```
SELECT * FROM HIVE.hive_system.hive_system WHERE envkey LIKE
'%hive%'
```

The above query returns all of the Hive system environment entries whose key contains the word "hive." A special query, set -v, is executed to fetch system environment information. Some versions of Hive do not support this query. For versions of Hive that do not support querying system environment information, the driver returns an empty result set.

Server-Side Properties

The Hortonworks Hive ODBC Driver with SQL Connector 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.

You can also specify server-side properties for connections that do not use a DSN. To do this, use the Hortonworks Hive ODBC Driver Configuration tool that is installed with the

Windows version of the driver, or set the appropriate configuration options in your connection string or the hortonworks.hiveodbc.ini file. Properties specified in the driver configuration tool or the hortonworks.hiveodbc.ini file apply to all connections that use the Hortonworks Hive ODBC Driver with SQL Connector.

For more information about setting server-side properties when using the Windows driver, see "Configuring Server-Side Properties" on page 18. For information about setting server-side properties when using the driver on a non-Windows platform, see "Driver Configuration Options" on page 55.

Temporary Table

The Temporary Table feature adds support for creating temporary tables and inserting literal values into temporary tables. Temporary tables are only accessible by the ODBC connection that created them and they will be dropped upon disconnect.

CREATE TABLE Statement for Temporary Tables

The driver supports the following DDL syntax for creating temporary tables:

```
<create table statement> := CREATE TABLE <temporary table name>
<left paren><column definition list><right paren>
```

```
<column definition list> := <column definition>[, <column
definition>]*
```

```
<column definition> := <column name> <data type>
```

```
<temporary table name> := <double quote><number sign><table
name><double quote>
```

```
<left paren> := (
```

```
<right paren> := )
```

```
<double quote> := "
```

```
<number sign> := #
```

The following is an example of a SQL statement for creating a temporary table:

```
CREATE TABLE "#TEMPTABLE1" (C1 DATATYPE_1, C2 DATATYPE_2, …, Cn
DATATYPE_n)
```

The temporary table name in a SQL query must be surrounded by double quotes ("), and the name must begin with a number sign (#).

Note: You can only use data types that are supported by Hive.

INSERT Statement for Temporary Tables

The driver supports the following DDL syntax for inserting data into temporary tables:

```
<insert statement> := INSERT INTO <temporary table name> <left
paren><column name list><right paren> VALUES <left paren><literal
value list><right paren>

<column name list> := <column name>[, <column name>]*

<literal value list> := <literal value>[, <literal value>]*

<temporary table name> := <double quote><number sign><table
name><double quote>

<left paren> := (

<right paren> := )

<double quote> := "

<number sign> := #
```

The following is an example of a SQL statement for inserting data into temporary tables:

```
INSERT INTO "#TEMPTABLE1" values (VAL(C1), VAL(C2) … VAL(Cn) )
```

VAL(C1) is the literal value for the first column in the table, and VAL(Cn) is the literal value for the nth column in the table.

Note: The INSERT statement is only supported for temporary tables.

Get Tables With Query

The Get Tables With Query configuration option allows you to choose whether to use the SHOW TABLES query or the GetTables API call to retrieve table names from a database.

Hive Server 2 has a limit on the number of tables that can be in a database when handling the GetTables API call. When the number of tables in a database is above the limit, the API call will return a stack overflow error or a timeout error. The exact limit and the error that appears depends on the JVM settings.

As a workaround for this issue, enable the Get Tables with Query configuration option (or GetTablesWithQuery key) to use the query instead of the API call.

Active Directory

The Hortonworks Hive ODBC Driver with SQL Connector 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 so that users in the Active Directory realm can access services in the MIT Kerberos Hadoop realm. For more information, see *Setting up One-Way Trust with Active Directory* in the Hortonworks documentation at http://-

docs.hortonworks.com/HDPDocuments/HDP2/HDP-2.1.7/bk_installing_manually_
book/content/ch23s05.html

Write-back

The Hortonworks Hive ODBC Driver with SQL Connector supports translation for INSERT, UPDATE, and DELETE syntax when connecting to a Hive Server 2 instance that is running Hive 0.14 or later.

Dynamic Service Discovery using ZooKeeper

The Hortonworks Hive ODBC Driver with SQL Connector can be configured to discover Hive Server 2 services via the ZooKeeper service.

For information about configuring this feature in the Windows driver, see "Creating a Data Source Name" on page 7 or "Configuring a DSN-less Connection" on page 10. For information about configuring this feature when using the driver on a non-Windows platform, see "Configuring Service Discovery Mode" on page 35.

Appendix A Using a Connection String

For some applications, you may need to use a connection string to connect to your data source.

DSN Connections

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

```
DSN=DataSourceName;Key=Value
```

DataSourceName is the DSN that you are using for the connection. *Key* is any connection attribute that is not already specified as a configuration key in the DSN, and *Value* is the value for the attribute. Add key-value pairs to the connection string as needed, separating each pair with a semicolon (;). For information about the connection attributes that are available, see "Driver Configuration Options" on page 55.

DSN-less Connections

Some client applications provide support for connecting to a data source using a driver without a DSN. The following is an example of a connection string for a DSN-less connection:

```
Driver=DriverNameOrFile;HOST=MyHiveServer;PORT=PortNumber;
Schema=DefaultSchema;HiveServerType=ServerType
```

The placeholders in the connection string are defined as follows:

- *DriverNameOrFile* is either the symbolic name of the installed driver defined in the odbcinst.ini file or the absolute path of the shared object file for the driver. If you use the symbolic name, then you must ensure that the odbcinst.ini file is configured to point to the symbolic name to the shared object file. For more information, see "Configuring the odbcinst.ini File" on page 33.
- *MyHiveServer* is the IP address or host name of the Hive server.
- *PortNumber* is the number of the port that the Hive server uses.
- *DefaultSchema* is the database schema to use when a schema is not explicitly specified in a query.
- *ServerType* is either 1 (for Hive Server 1) or 2 (for Hive Server 2).

Add key-value pairs to the connection string as needed, separating each pair with a semicolon (;). For information about the connection attributes that are available, see "Driver Configuration Options" on page 55.

Appendix B Authentication Options

To connect to a Hive server, you must configure the Hortonworks Hive ODBC Driver with SQL Connector to use the authentication mechanism that matches the access requirements of the server and provides the necessary credentials. To determine the authentication settings that your Hive server requires, check the server type and then refer to the corresponding section below.

Hive Server 1

You must use No Authentication as the authentication mechanism. Hive Server 1 instances do not support authentication.

Hive Server 2 on an HDInsight Distribution

If you are connecting to HDInsight Emulator running on Windows Azure, then you must use the Windows Azure HDInsight Emulator mechanism.

If you are connecting to HDInsight Service running on Windows Azure, then you must use the Windows Azure HDInsight Service mechanism.

Hive Server 2 on a non-HDInsight Distribution

Note: Most default configurations of Hive Server 2 on non-HDInsight Distributions require User Name authentication.

Configuring authentication for a connection to a Hive Server 2 instance on a non-HDInsight Distribution involves setting the authentication mechanism, the Thrift transport protocol, and SSL support. To determine the settings that you need to use, check the following three properties in the hive-site.xml file in the Hive server that you are connecting to:

- hive.server2.authentication
- hive.server2.transport.mode
- hive.server2.use.SSL

Use Table 4 to determine the authentication mechanism that you need to configure, based on the hive.server2.authentication value in the hive-site.xml file:

Table 4. Authentication Mechanismto Use

hive.server2.authentication	Authentication Mechanism
NOSASL	No Authentication
KERBEROS	Kerberos

hive.server2.authentication	Authentication Mechanism
NONE	User Name
LDAP	User Name and Password

Use Table 5 to determine the Thrift transport protocol that you need to configure, based on the hive.server2.authentication and hive.server2.transport.mode values in the hive-site.xml file:

Table 5. Thrift Transport Protocol Setting to Use

hive.server2.authentication	hive.server2.transport.mode	Thrift Transport Protocol
NOSASL	binary	Binary
KERBEROS	binary or http	SASL or HTTP
NONE	binary or http	SASL or HTTP
LDAP	binary or http	SASL or HTTP

To determine whether SSL should be enabled or disabled for your connection, check the hive.server2.use.SSL value in the hive-site.xml file. If the value is true, then you must enable and configure SSL in your connection. If the value is false, then you must disable SSL in your connection.

For detailed instructions on how to configure authentication when using the Windows driver, see "Configuring Authentication" on page 11.

For detailed instructions on how to configure authentication when using a non-Windows driver, see "Configuring Authentication" on page 35.

Appendix C Configuring Kerberos Authentication for Windows

Active Directory

The Hortonworks Hive ODBC Driver with SQL Connector 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 so that users in the Active Directory realm can access services in the MIT Kerberos Hadoop realm. For more information, see *Setting up One-Way Trust with Active Directory* in the Hortonworks documentation at http://docs.hortonworks.com/HDPDocuments/HDP2/HDP-2.1.7/bk_installing_manually_book/content/ch23s05.html

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 at http://web.mit.edu/kerberos/

To download and install MIT Kerberos for Windows 4.0.1:

1. To download the Kerberos installer for 64-bit computers, 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

<div align="center">OR</div>

   To download the Kerberos installer for 32-bit computers, 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 in step 1.

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 a .INI file in the default location—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 from your Kerberos administrator.

OR

Obtain the configuration file from the /etc/krb5.conf folder on the computer that is hosting the Hive Server 2 instance.

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 from your Kerberos administrator.

OR

Obtain the configuration file from the /etc/krb5.conf folder on the computer that is hosting the Hive Server 2 instance.

2. Place the krb5.conf file in an accessible directory and make note of the full path name.

3. If you are using Windows 7 or earlier, click the **Start** button 🌐, 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 absolute path to the krb5.conf file from step 2.

9. Click **OK** to save the new variable.

10. Ensure 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. If you are using Windows 7 or earlier, click the **Start** button 🌐, then right-click **Computer**, and then click **Properties**

<div align="center">OR</div>

   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 in step 1, and then append the file name krb5cache. For example, if you created the folder C:\temp in step 1, 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, ensure that the krb5cache file does not already exist as a file or a directory.

8. Click **OK** to save the new variable.

9. Ensure 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 ensure that Kerberos uses the new settings, restart your computer.

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. If you are using Windows 7 or earlier, click the **Start** button 🌐, then click **All Programs**, and then click the **Kerberos for Windows (64-bit)** or **Kerberos for Windows (32-bit)** program group.

OR

If you are using Windows 8 or later, click the arrow button at the bottom of the Start screen, and then find the **Kerberos for Windows (64-bit)** or **Kerberos for Windows (32-bit)** program group.

2. Click **MIT Kerberos Ticket Manager**

3. In the MIT Kerberos Ticket Manager, click **Get Ticket**

4. 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 the MIT Kerberos Ticket Manager.

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

1. If you are using Windows 7 or earlier, click the **Start** button 🌐, then click **All Programs**, then click **Accessories**, and then click **Command Prompt**

OR

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 keytab_path principal
```

keytab_path 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. Click the **Start** button 🌐, then click **All Programs**, then click **Accessories**, 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.

Appendix D Driver Configuration Options

Appendix D "Driver Configuration Options" on page 55 lists the configuration options available in the Hortonworks Hive ODBC Driver with SQL Connector alphabetically by field or button label. Options having only key names—not appearing in the user interface of the driver—are listed alphabetically by key name.

When creating or configuring a connection from a Windows computer, the fields and buttons are available in the Hortonworks Hive ODBC Driver Configuration tool and the following dialog boxes:

- Hortonworks Hive ODBC Driver DSN Setup
- HTTP Properties
- SSL Options
- Advanced Options
- Server Side Properties

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

Note: You can pass in configuration options in your connection string or set them in your odbc.ini and hortonworks.hiveodbc.ini files. Configuration options set in a hortonworks.hiveodbc.ini file apply to all connections, whereas configuration options passed in in the connection string or set in an odbc.ini file are specific to a connection. Configuration options passed in using the connection string take precedence over configuration options set in odbc.ini. Configuration options set in odbc.ini take precedence over configuration options set in hortonworks.hiveodbc.ini

Configuration Options Appearing in the User Interface

The following configuration options are accessible via the Windows user interface for the Hortonworks Hive ODBC Driver with SQL Connector, or via the key name when using a connection string or configuring a connection from a Linux/Mac OS X/Debian computer:

- "Allow Common Name Host Name Mismatch" on page 56
- "Allow Self-signed Server Cer-tificate" on page 57
- "Apply Properties with Queries" on page 57
- "Async Exec Poll Interval" on page 57
- "Binary Column Length" on page 58

- "Host FQDN" on page 63
- "HTTP Path" on page 64
- "Mechanism" on page 64
- "Password" on page 65
- "Port" on page 65
- "Realm" on page 65
- "Rows Fetched Per Block" on page 66

Allow Common Name Host Name Mismatch

Key Name	Default Value	Required
CAIssuedCertNamesMismatch	Clear (0)	No

Description

When this option is enabled (1), the driver allows a CA-issued SSL certificate name to not match the host name of the Hive server.

When this option is disabled (0), the CA-issued SSL certificate name must match the host name of the Hive 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

When this option is enabled (1), the driver authenticates the Hive server even if the server is using a self-signed certificate.

When this option is disabled (0), the driver does not allow self-signed certificates from the server.

Note: This setting is applicable only when SSL is enabled.

Apply Properties with Queries

Key Name	Default Value	Required
ApplySSPWithQueries	Selected (1)	No

Description

When this option is enabled (1), the driver applies each server-side property by executing a `set SSPKey=SSPValue` query when opening a session to the Hive server.

When this option is disabled (0), the driver uses a more efficient method for applying server-side properties that does not involve additional network round-tripping. However, some Hive Server 2 builds are not compatible with the more efficient method.

Note: When connecting to a Hive Server 1 instance, ApplySSPWithQueries is always enabled.

Async Exec Poll Interval

Key Name	Default Value	Required
AsyncExecPollInterval	100	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 Hive is asynchronous. It does not mean that ODBC asynchronous operations are supported.

Note: This option is applicable only to HDInsight clusters.

Binary Column Length

Key Name	Default Value	Required
BinaryColumnLength	32767	No

Description

The maximum data length for BINARY columns.

By default, the columns metadata for Hive does not specify a maximum data length for BINARY columns.

Client Certificate File

Key Name	Default Value	Required
ClientCert	None	Yes, if two-way SSL verification is enabled.

Description

The full path to the PEM file containing the client's SSL certificate.

Note: This setting is applicable only when two-say SSL is enabled.

Client Private Key File

Key Name	Default Value	Required
ClientPrivateKey	None	Yes, if two-way SSL verification is enabled.

Description

The full path to the PEM file containing the client's SSL private key.

If the private key file is protected with a password, then provide the password using the driver configuration option "Client Private Key Password" on page 59.

Note: This setting is applicable only when two-say SSL is enabled.

Client Private Key Password

Key Name	Default Value	Required
ClientPrivateKeyPassword	None	Yes, if two-way SSL verification is enabled and the client's private key file is protected with a password.

Description

The password of the private key file that is specified in the Client Private Key File field (the ClientPrivateKey key).

Convert Key Name to Lower Case

Key Name	Default Value	Required
LCaseSspKeyName	Selected (1)	No

Description

When this option is enabled (1), the driver converts server-side property key names to all lower case characters.

When this option is disabled (0), the driver does not modify the server-side property key names.

Data File HDFS Dir

Key Name	Default Value	Required
HDFSTempTableDir	/tmp/simba	No

Description

The HDFS directory that the driver will use to store the necessary files for supporting the Temporary Table feature.

Note: Due to a problem in Hive (see https://issues.apache.org/jira/browse/HIVE-4554), HDFS paths with space characters do not work with versions of Hive prior to 0.12.0.

This option is not applicable when connecting to Hive 0.14 or later.

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, type the **show databases** command at the Hive command prompt.

Decimal Column Scale

Key Name	Default Value	Required
DecimalColumnScale	10	No

Description

The maximum number of digits to the right of the decimal point for numeric data types.

Default String Column Length

Key Name	Default Value	Required
DefaultStringColumnLength	255	No

Description

The maximum data length for STRING columns.

By default, the columns metadata for Hive does not specify a maximum data length for STRING columns.

Delegation UID

Key Name	Default Value	Required
DelegationUID	None	No

Description

Use this option to delegate all operations against Hive to a user that is different than the authenticated user for the connection.

Note: This option is applicable only when connecting to a Hive Server 2 instance that supports this feature.

Driver Config Take Precedence

Key Name	Default Value	Required
DriverConfigTakePrecedence	Clear (0)	No

Description

When this option is enabled (1), driver-wide configurations take precedence over connection and DSN settings.

When this option is disabled (0), connection and DSN settings take precedence instead.

Enable SSL

Key Name	Default Value	Required
SSL	Clear (0)	No

Description

When this option is set to (1), the client verifies the Hive server using SSL.

When this option is set to (0), SSL is disabled.

Note: This option is applicable only when connecting to a Hive server that supports SSL.

Enable Temporary Table

Key Name	Default Value	Required
EnableTempTable	Clear (0)	No

Description

When this option is enabled (1), the driver supports the creation and use of temporary tables.

When this option is disabled (0), the driver does not support temporary tables.

Important: When connecting to Hive 0.14 or later, the Temporary Tables feature is always enabled and you do not need to configure it in the driver.

Fast SQLPrepare

Key Name	Default Value	Required
FastSQLPrepare	Clear (0)	No

Description

When this option is enabled (1), the driver defers query execution to SQLExecute.

When this option is disabled (0), the driver does not defer query execution to SQLExecute.

When using Native Query mode, the driver will execute the HiveQL query to retrieve the result set metadata for SQLPrepare. As a result, SQLPrepare might be slow. If the result set metadata is not required after calling SQLPrepare, then enable Fast SQLPrepare.

Get Tables With Query

Key Name	Default Value	Required
GetTablesWithQuery	Clear (0)	No

Description

When this option is enabled (1), the driver uses the SHOW TABLES query to retrieve the names of the tables in a database.

When this option is disabled (0), the driver uses the GetTables Thrift API call to retrieve the names of the tables in a database.

Note: This option is applicable only when connecting to a Hive Server 2 instance.

HDFS User

Key Name	Default Value	Required
HDFSUser	hdfs	No

Description

The name of the HDFS user that the driver will use to create the necessary files for supporting the Temporary Tables feature.

This option is not applicable when connecting to Hive 0.14 or later.

Hive Server Type

Key Name	Default Value	Required
HiveServerType	Hive Server 2 (2)	No

Description

Select **Hive Server 1** or set the key to 1 if you are connecting to a Hive Server 1 instance.

Select **Hive Server 2** or set the key to 2 if you are connecting to a Hive Server 2 instance.

Note: If Service Discovery Mode is enabled, then connections to Hive Server 1 are not supported.

Host(s)

Key Name	Default Value	Required
HOST	None	Yes

Description

If Service Discovery Mode is disabled, specify the IP address or host name of the Hive server.

If Service Discovery Mode is enabled, specify a comma-separated list of ZooKeeper servers in the following format, where *zk_host* is the IP address or host name of the ZooKeeper server and *zk_port* is the number of the port that the ZooKeeper server uses:

```
zk_host1:zk_port1,zk_host2:zk_port2
```

Host FQDN

Key Name	Default Value	Required
KrbHostFQDN	None	Yes, if the authentication mechanism is **Kerberos**.

Description

The fully qualified domain name of the Hive Server 2 host.

You can set the value of Host FQDN to **_HOST** in order to use the Hive server host name as the fully qualified domain name for Kerberos authentication. If Service Discovery Mode is disabled, then the driver uses the value specified in the Host connection attribute. If Service Discovery Mode is enabled, then the driver uses the Hive Server 2 host name returned by ZooKeeper.

HTTP Path

Key Name	Default Value	Required
HTTPPath	None	Yes, if you are using **HTTP** as the Thrift transport protocol.

Description

The partial URL corresponding to the Hive server.

Note: This option is applicable only when you are using HTTP as the Thrift transport protocol. For more information, see "Thrift Transport" on page 68.

Mechanism

Key Name	Default Value	Required
AuthMech	No Authentication (0) if you are connecting to Hive Server 1. User Name (2) if you are connecting to Hive Server 2.	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)
- User Name (2)

- User Name and Password (3)
- Windows Azure HDInsight Emulator (5)
- Windows Azure HDInsight Service (6)

Password

Key Name	Default Value	Required
PWD	None	Yes, if the authentication mechanism is **User Name and Password**, **Windows Azure HDInsight Emulator**, or **Windows Azure HDInsight Service**.

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	**non-HDInsight clusters**—10000 **Windows Azure HDInsight Emulator**—10001 **Windows Azure HDInsight Service**—443	Yes, if Service Discovery Mode is disabled.

Description

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

Realm

Key Name	Default Value	Required
KrbRealm	Depends on your Kerberos configuration.	No

Description

The realm of the Hive Server 2 host.

If your Kerberos configuration already defines the realm of the Hive Server 2 host as the default realm, then you do not need to configure this option.

Rows Fetched Per Block

Key Name	Default Value	Required
RowsFetchedPerBlock	10000	No

Description

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

Any positive 32-bit integer is a valid value, but 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	Clear (0)	No

Description

This option is available only in the Windows driver. It appears in the Hortonworks Hive ODBC Driver DSN Setup dialog box and the SSL Options dialog box.

When this option is enabled (1), the specified password is saved in the registry.

When this option is disabled (0), the specified password is not saved in the registry.

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

Service Discovery Mode

Key Name	Default Value	Required
ServiceDiscoveryMode	No Service Discovery (0)	No

Description

When this option is enabled (1), the driver discovers Hive Server 2 services via the ZooKeeper service.

When this option is disabled (0), the driver connects to Hive without using the ZooKeeper service.

Service Name

Key Name	Default Value	Required
KrbServiceName	None	Yes, if the authentication mechanism is **Kerberos**.

Description

The Kerberos service principal name of the Hive server.

Show System Table

Key Name	Default Value	Required
ShowSystemTable	Clear (0)	No

Description

When this option is enabled (1), the driver returns the hive_system table for catalog function calls such as `SQLTables` and `SQLColumns`

When this option is disabled (0), the driver does not return the hive_system table for catalog function calls.

Socket Timeout

Key Name	Default Value	Required
SocketTimeout	30	No

Description

The number of seconds that an operation can remain idle before it is closed.

Note: This option is applicable only when asynchronous query execution is being used against Hive Server 2 instances.

Temp Table TTL

Key Name	Default Value	Required
TempTableTTL	10	No

Description

The number of minutes a temporary table is guaranteed to exist in Hive after it is created.

This option is not applicable when connecting to Hive 0.14 or later.

Thrift Transport

Key Name	Default Value	Required
ThriftTransport	Binary (0) if you are connecting to Hive Server 1. SASL (1) if you are connecting to Hive Server 2.	No

Description

The transport protocol to use in the Thrift layer.

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

- Binary (0)
- SASL (1)
- HTTP (2)

Trusted Certificates

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

Description

The location of the PEM file containing trusted CA certificates for authenticating the Hive server when using SSL.

If this option is not set, then the driver will default to using the trusted CA certificates PEM file installed by the driver.

Note: This setting is applicable only when SSL is enabled.

Two Way SSL

Key Name	Default Value	Required
TwoWaySSL	Clear (0)	No

Description

When this option is enabled (1), the client and the Hive server verify each other using SSL. See also the driver configuration options "Client Certificate File" on page 58, "Client Private Key File" on page 58, and "Client Private Key Password" on page 59.

When this option is disabled (0), then the server does not verify the client. Depending on whether one-way SSL is enabled, the client may verify the server. For more information, see "Enable SSL" on page 61.

Note: This option is applicable only when connecting to a Hive server that supports SSL. You must enable SSL before Two Way SSL can be configured. For more information, see "Enable SSL" on page 61.

Unicode SQL Character Types

Key Name	Default Value	Required
UseUnicodeSqlCharacterTypes	Clear (0)	No

Description

When this option is enabled (1), the driver returns SQL_WVARCHAR for STRING and VARCHAR columns, and returns SQL_WCHAR for CHAR columns.

When this option is disabled (0), the driver returns SQL_VARCHAR for STRING and VARCHAR columns, and returns SQL_CHAR for CHAR columns.

Use Async Exec

Key Name	Default Value	Required
EnableAsyncExec	Clear (0)	No

Description

When this option is enabled (1), the driver uses an asynchronous version of the API call against Hive for executing a query.

When this option is disabled (0), the driver executes queries synchronously.

Due to a problem in Hive 0.12.0 (see https://issues.apache.org/jira/browse/HIVE-5230), Hive returns generic error messages for errors that occur during query execution. To see the actual error message relevant to the problem, turn off asynchronous query execution and execute the query again.

Note: This option only takes effect when connecting to a Hive cluster running Hive 0.12.0 or higher.

Use Native Query

Key Name	Default Value	Required
UseNativeQuery	Clear (0)	No

Description

When this option is enabled (1), the driver does not transform the queries emitted by an application, so the native query is used.

When this option is disabled (0), the driver transforms the queries emitted by an application and converts them into an equivalent from in HiveQL.

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

Use Only SSPI Plugin

Key Name	Default Value	Required
UseOnlySSPI	Clear (0)	No

Description

When this option is enabled (1), the driver handles Kerberos authentication by using the SSPI plugin instead of MIT Kerberos by default.

When this option is 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.

User Name

Key Name	Default Value	Required
UID	For User Name authentication only, the default value is anonymous	No, if the authentication mechanism is **User Name**. Yes, if the authentication mechanism is **User Name and Password**, **Windows Azure HDInsight Emulator**, or **Windows Azure HDInsight Service**.

Description

The user name that you use to access Hive Server 2.

Web HDFS Host

Key Name	Default Value	Required
WebHDFSHost	The Hive server host.	No

Description

The host name or IP address of the machine hosting both the namenode of your Hadoop cluster and the WebHDFS service.

This option is not applicable when connecting to Hive 0.14 or later.

Web HDFS Port

Key Name	Default Value	Required
WebHDFSPort	50070	No

Description

The WebHDFS port for the namenode.

This option is not applicable when connecting to Hive 0.14 or later.

ZooKeeper Namespace

Key Name	Default Value	Required
ZKNamespace	None	Yes, if Service Discovery Mode is enabled.

Description

The namespace on ZooKeeper under which Hive Server 2 znodes are added.

Configuration Options Having Only Key Names

The following configuration options do not appear in the Windows user interface for the Hortonworks Hive ODBC Driver with SQL Connector and are only accessible when using a connection string or configuring a connection from a Linux/Mac OS X/Debian computer:

- "ADUserNameCase" on page 73
- "Driver" on page 73

- "ForceSynchronousExec" on page 73
- "http.header." on page 74
- "SSP_" on page 74

ADUserNameCase

Default Value	Required
Unchanged	No

Description

Use this option to control whether the driver changes the user name part of an AD Kerberos UPN to all upper case or all lower case. The following values are possible:

- **Upper**—Change the user name to all upper case.
- **Lower**—Change the user name to all lower case.
- **Unchanged**—Do not modify the user name.

Note: This option is applicable only when using Active Directory Kerberos from a Windows client machine to authenticate.

Driver

Default Value	Required
The default value varies depending on the version of the driver that is installed. For example, the value for the Windows driver is different from the value of the Mac OS X driver.	Yes

Description

The name of the installed driver (Hortonworks Hive ODBC Driver) or the absolute path of the Hortonworks Hive ODBC Driver with SQL Connector shared object file.

ForceSynchronousExec

Default Value	Required
0	No

Description

When this option is enabled (1), the driver is forced to execute queries synchronously when connected to an HDInsight cluster.

When this option is disabled (0), the driver is able to execute queries asynchronously when connected to an HDInsight cluster.

Note: This option is applicable only to HDInsight clusters.

http.header.

Default Value	Required
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 "Thrift Transport" on page 68.

SSP_

Default Value	Required
None	No

Description

Set a server-side property by using the following syntax, where *SSPKey* is the name of the server-side property to set and *SSPValue* is the value to assign to the server-side property:

```
SSP_SSPKey=SSPValue
```

For example:

```
SSP_mapred.queue.names=myQueue
```

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

Note: The SSP_ prefix must be upper case.

Third Party Trademarks

ICU License - ICU 1.8.1 and later

COPYRIGHT AND PERMISSION NOTICE

Copyright (c) 1995-2014 International Business Machines Corporation and others

All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, provided that the above copyright notice(s) and this permission notice appear in all copies of the Software and that both the above copyright notice(s) and this permission notice appear in supporting documentation.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS NOTICE BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization of the copyright holder.

All trademarks and registered trademarks mentioned herein are the property of their respective owners.

OpenSSL

Copyright (c) 1998-2011 The OpenSSL Project. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. All advertising materials mentioning features or use of this software must display the following acknowledgment:

"This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit. (http://www.openssl.org/)"

4. The names "OpenSSL Toolkit" and "OpenSSL Project" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact openssl-core@openssl.org.

5. Products derived from this software may not be called "OpenSSL" nor may "OpenSSL" appear in their names without prior written permission of the OpenSSL Project.

6. Redistributions of any form whatsoever must retain the following acknowledgment:

"This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit (http://www.openssl.org/)"

THIS SOFTWARE IS PROVIDED BY THE OpenSSL PROJECT "AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE OpenSSL PROJECT OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Original SSLeay License

Copyright (C) 1995-1998 Eric Young (eay@cryptsoft.com)

All rights reserved.

This package is an SSL implementation written by Eric Young (eay@cryptsoft.com). The implementation was written so as to conform with Netscapes SSL.

This library is free for commercial and non-commercial use as long as the following conditions are aheared to. The following conditions apply to all code found in this distribution, be it the RC4, RSA, lhash, DES, etc., code; not just the SSL code. The SSL documentation included with this distribution is covered by the same copyright terms except that the holder is Tim Hudson (tjh@cryptsoft.com).

Copyright remains Eric Young's, and as such any Copyright notices in the code are not to be removed. If this package is used in a product, Eric Young should be given attribution as the

author of the parts of the library used. This can be in the form of a textual message at program startup or in documentation (online or textual) provided with the package.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. All advertising materials mentioning features or use of this software must display the following acknowledgement:

   "This product includes cryptographic software written by Eric Young (eay@cryptsoft.com)"

   The word 'cryptographic' can be left out if the rouines from the library being used are not cryptographic related :-).

4. If you include any Windows specific code (or a derivative thereof) from the apps directory (application code) you must include an acknowledgement:

   "This product includes software written by Tim Hudson (tjh@cryptsoft.com)"

THIS SOFTWARE IS PROVIDED BY ERIC YOUNG ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The licence and distribution terms for any publically available version or derivative of this code cannot be changed. i.e. this code cannot simply be copied and put under another distribution licence [including the GNU Public Licence.]

Expat

Copyright (c) 1998, 1999, 2000 Thai Open Source Software Center Ltd

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish,

distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NOINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Stringencoders License

Copyright 2005, 2006, 2007

Nick Galbreath -- nickg [at] modp [dot] com

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

> Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

> Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

> Neither the name of the modp.com nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This is the standard "new" BSD license:

http://www.opensource.org/licenses/bsd-license.php

dtoa License

The author of this software is David M. Gay.

Copyright (c) 1991, 2000, 2001 by Lucent Technologies.

Permission to use, copy, modify, and distribute this software for any purpose without fee is hereby granted, provided that this entire notice is included in all copies of any software which is or includes a copy or modification of this software and in all copies of the supporting documentation for such software.

THIS SOFTWARE IS BEING PROVIDED "AS IS", WITHOUT ANY EXPRESS OR IMPLIED WARRANTY. IN PARTICULAR, NEITHER THE AUTHOR NOR LUCENT MAKES ANY REPRESENTATION OR WARRANTY OF ANY KIND CONCERNING THE MERCHANTABILITY OF THIS SOFTWARE OR ITS FITNESS FOR ANY PARTICULAR PURPOSE.

Apache Hive
 Copyright 2008-2011 The Apache Software Foundation.

Apache Thrift
 Copyright 2006-2010 The Apache Software Foundation.

Apache ZooKeeper
 Copyright 2009-2012 The Apache Software Foundation.

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Cyrus SASL

Copyright (c) 1994-2012 Carnegie Mellon University. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. The name "Carnegie Mellon University" must not be used to endorse or promote products derived from this software without prior written permission. For permission or any other legal details, please contact

   Office of Technology Transfer
   Carnegie Mellon University
   5000 Forbes Avenue
   Pittsburgh, PA 15213-3890
   (412) 268-4387, fax: (412) 268-7395
   tech-transfer@andrew.cmu.edu

4. Redistributions of any form whatsoever must retain the following acknowledgment:

   "This product includes software developed by Computing Services at Carnegie Mellon University (http://www.cmu.edu/computing/)."

CARNEGIE MELLON UNIVERSITY DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS, IN NO EVENT SHALL CARNEGIE MELLON UNIVERSITY BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

libcurl

COPYRIGHT AND PERMISSION NOTICE

Copyright (c) 1996 - 2015, Daniel Stenberg, daniel@haxx.se.

All rights reserved.

Permission to use, copy, modify, and distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization of the copyright holder.