Hortonworks Architecting the Future of Big Data

# Hive ODBC Driver

## User Guide

Revised: February 19, 2013

© 2012-2013 Hortonworks Inc. All Rights Reserved.

Parts of this Program and Documentation include proprietary software and content that is copyrighted and licensed by Simba Technologies Incorporated. This proprietary software and content may include one or more feature, functionality or methodology within the ODBC, JDBC, ADO.NET, OLE DB, ODBO, XMLA, SQL and/or MDX component(s).
For information about Simba's products and services, visit: www.simba.com.

## Table of Contents

Introduction ....4
Contact Us ....4
Windows Driver ....5
System Requirements ....5
Installation ....5
Configuration ....5
Create a Data Source Name (DSN) ....5
Configure authentication ....12
Configuring your Driver DSN ....12
Linux Driver ....15
System Requirements ....15
Installation ....15
Driver Directories ....16
Configuration ....16
ODBC Configuration Files ....16
Sample ODBC Configuration Files ....16
ODBCINI and ODBCSYSINI Environment Configuration ....17
ODBC Data Source Configuration File Overview ....17
Create a Data Source ....18
Configuration Options ....19
ODBC Drivers Configuration File Overview ....20
Define a Driver ....21
Configure the Hortonworks Hive ODBC Driver ....21
Configure the Library Path ....22
Configure Kerberos Authentication ....22
Mac OS X Driver ....23
System Requirements ....23
Installation ....23
Driver Directories ....23
Configuration ....23
ODBC Configuration Files ....23
Sample ODBC Configuration Files ....24
ODBCINI and ODBCSYSINI Environment Configuration ....24
ODBC Data Source Configuration File Overview ....25

Create a Data Source ..... 25
Configuration Options ..... 26
ODBC Drivers Configuration File Overview ..... 28
Define a Driver ..... 28
Configure the Hortonworks Hive ODBC Driver ..... 29
Configure the Library Path ..... 29
Configure Kerberos Authentication ..... 29
Features ..... 30
SQL Query versus HiveQL Query ..... 30
SQL Connector ..... 30
Data Types ..... 30
Catalog and Schema Support ..... 30
Hive System Table ..... 31
Appendix A: Configuring Kerberos Authentication for Windows ..... 32
Download and install MIT Kerberos for Windows 4.0.1 ..... 32
Set up the Kerberos configuration file in the default location ..... 32
Set up the Kerberos configuration file in another location ..... 32
Set up the Kerberos credential cache file ..... 33
Obtain a ticket for a Kerberos principal using password ..... 33
Obtain a ticket for a Kerberos principal using a keytab file ..... 33
Obtain a ticket for a Kerberos principal using the default keytab file ..... 34
Appendix B: Driver Authentication Configuration for Windows ..... 35
Configure driver authentication ..... 35
Configuring your Driver ..... 37

## Introduction

Welcome to the Hortonworks Hive ODBC Driver with SQL Connector. 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.

The Hortonworks Hive ODBC Driver with SQL Connector is used for direct SQL and HiveQL access to Apache Hadoop / Hive distributions. It enables Business Intelligence (BI), analytics and reporting on Hadoop / Hive-based data. The Hortonworks Hive ODBC Driver efficiently transforms an application's SQL query into the equivalent form in HiveQL. The Hive Query Language is a subset of SQL-92. If an application is Hive-aware, the Hortonworks Hive ODBC Driver is configurable to pass the query through. The Hortonworks Hive ODBC Driver with SQL Connector 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, refer to the Features section of this document.

The Hortonworks Hive ODBC Driver with SQL Connector is available for both Microsoft Windows, Linux and Mac OS X. It 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 on all platforms. Any version of the ODBC driver will connect to a Hive server irrespective of the server's host OS.

This 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 here helpful. Please 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

- Windows® XP with SP3, Windows® Vista, Windows® 7 Professional or Windows® 2008 R2. Both 32-bit and 64-bit editions are supported.
- 25 MB of available disk space.

Installing the driver requires administrator privileges.

The Hortonworks Hive ODBC Driver with SQL Connector requires a Hadoop cluster with the Hive service installed and running. The Hortonworks Hive ODBC Driver with SQL Connector is suitable for use with all versions of Apache Hive.

## Installation

There are two versions of the driver for Windows:

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

The version of the driver that you select should match the bitness of the application. For example, if the application is 64-bit then you should install the 64-bit driver. It is allowable to install both versions of the driver.

The following document explains how to use ODBC on 64-bit editions of Windows: http://www.simba.com/docs/HOW-TO-32-bit-vs-64-bit-ODBC-Data-Source-Administrator.pdf.

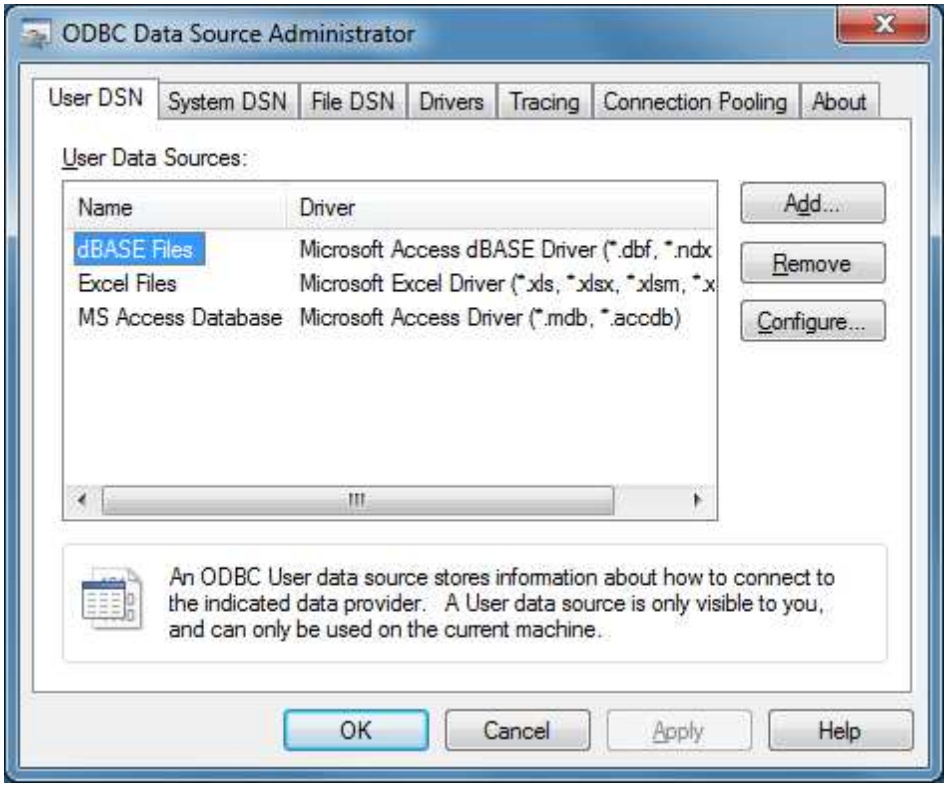
## Configuration

### Create a Data Source Name (DSN)

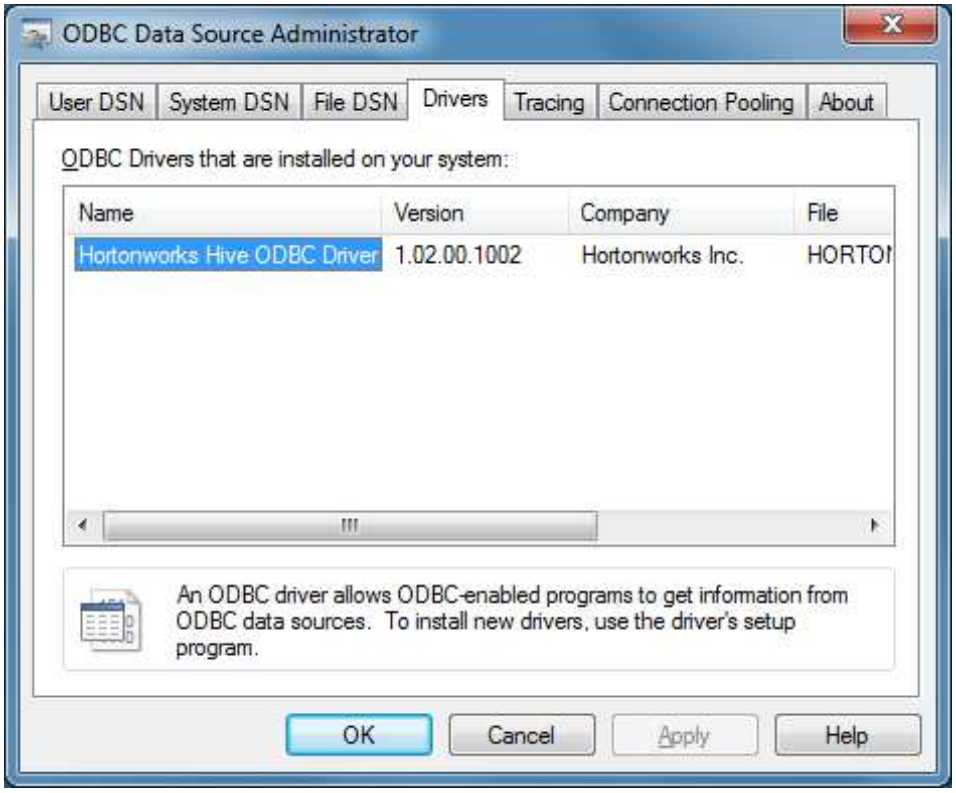
1. Click the **Start** button .
2. Click **All Programs.**
3. Click the **Hortonworks Hive ODBC Driver 1.2 (64-bit)** or the **Hortonworks Hive ODBC Driver 1.2 (32-bit)** program group. If you installed both versions of the driver, you will see two program groups.

   Because DSNs are bit-specific, select the version that matches the bitness of your application. For example, a DSN that is defined for the 32-bit driver will only be accessible from 32-bit applications.

4. Click **64-bit ODBC Administrator or 32-bit ODBC Administrator**. The ODBC Data Source Administrator window opens.



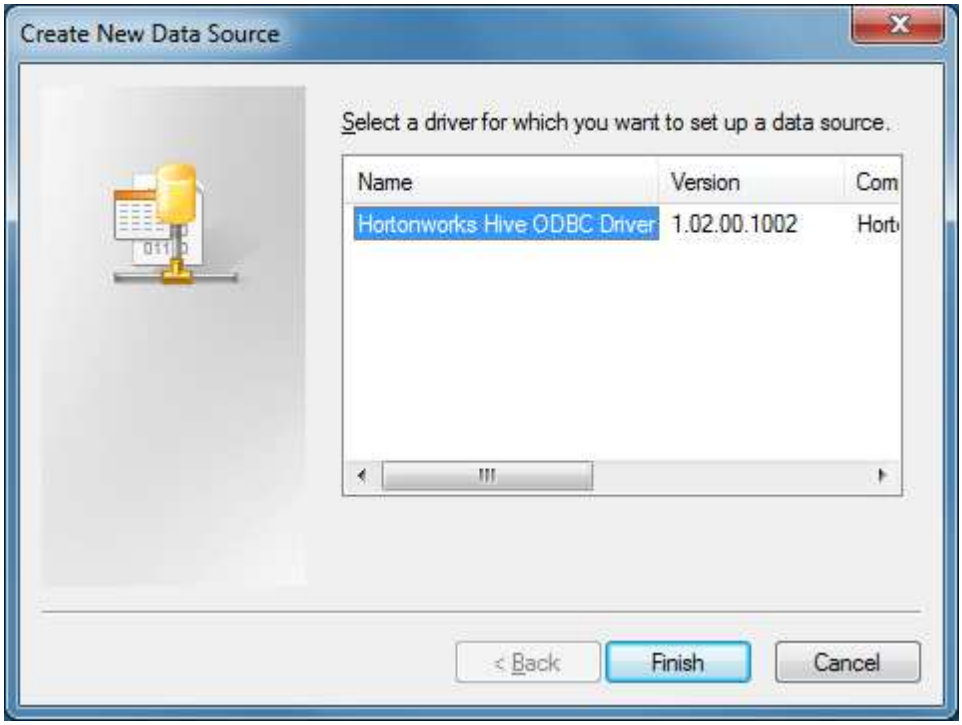
5. Click the **Drivers** tab and verify that the Hortonworks Hive ODBC Driver is displayed in the list of ODBC drivers that are installed on your system.



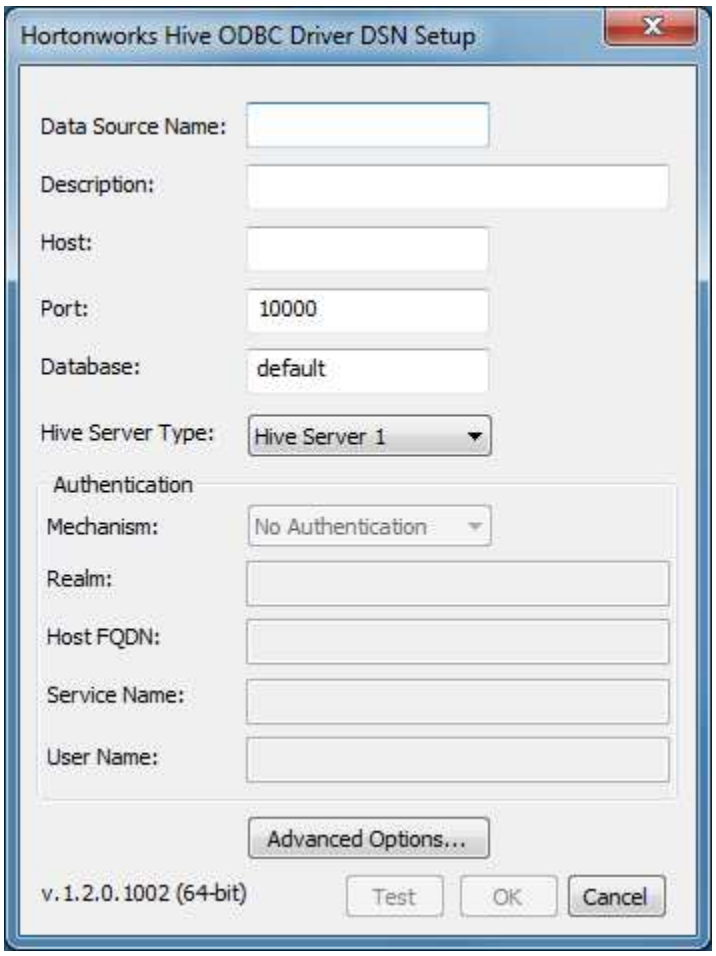

6. Click the **System DSN** tab to create a system DSN or click the **User DSN** tab to create a user DSN.

   A system DSN can be seen by all users that login to a workstation. A user DSN is specific to a user on the workstation. It can only be seen by the user who creates it.

7. Click **Add**.
   The Create New Data Source window opens.



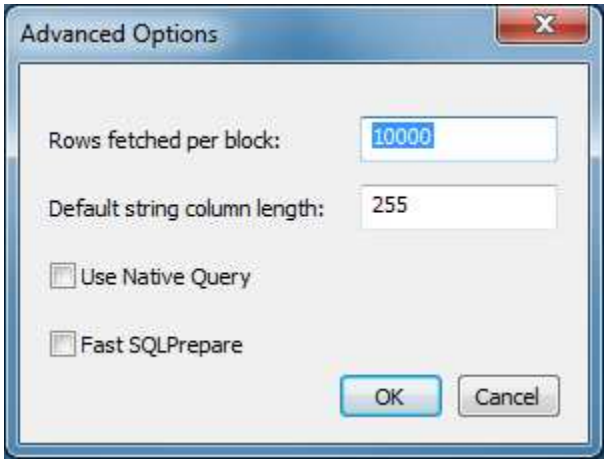
8. Select **Hortonworks Hive ODBC Driver** and then click **Finish**. The Hortonworks Hive ODBC Driver DSN Setup window opens.




9. In the Data Source Name text box, type a name for your DSN.
10. Optionally, In the Description text box, enter a description.
11. In the Host text box, type the IP address or hostname of the Hive server.
12. In the Port text box, type the listening port for the service.
13. In the Database text box, type the name of the database schema to use when a schema is not explicitly specified in a query. Queries on other schemas can still be issued by explicitly specifying the schema in the query. To determine the appropriate database schema to use, type the `show databases` command at the Hive command prompt to inspect your databases.
14. For the Hive Server Type, select either Hive Server 1 or Hive Server 2.

Optionally, if you selected Hive Server 2 as the Hive server type, you can configure authentication. For detailed instructions, refer to the section, "

15. Configure authentication".
16. Optionally, click **Advanced Options**.
    The Advanced Options window opens.




17. In the Rows fetched per block text box, type the number of rows to be fetched per block.
    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.
18. In the Default string column length text box, type the default string column length to use.
    Hive does not provide the length for String columns in its column metadata. This option allows you to tune the length of String columns.
19. Select the Use Native Query checkbox to disable the SQL Connector feature.
    The SQL Connector feature has been added to the driver to apply transformations to the queries emitted by an application to convert them into an equivalent form in HiveQL. If the application is Hive aware and already emits HiveQL then turning off the SQL Connector feature avoids the extra overhead of query transformation.
20. Select the Fast SQLPrepare checkbox to 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 this option.
21. Click **OK**.
22. Click **Test** to test the connection and then click **OK**.

### Configure authentication

**Note**: Authentication is only available for server of type Hive Server 2. Authentication is not available for server of type Hive Server 1.

If you are using an application that makes direct connections to Hive instead of using standard ODBC Data Sources, refer to "Appendix B: Driver Authentication Configuration for Windows".

Unlike Hive Server 1, Hive Server 2 supports multiple authentication mechanisms. You must determine the authentication type your server is using. The authentication methods available are as follows:

- No Authentication
- Kerberos
- User Name

For **No Authentication**, no additional details are required.

For **User Name** authentication, select **User Name** in the **Mechanism** field and then type a user name in the User Name field.

For **Kerberos** authentication, Kerberos must be configured before using the driver with Kerberos authentication. Refer to "Appendix A: Configuring Kerberos Authentication for Windows".

**Note:** If you installed HDP 1.2 using Ambari, by default the authentication method is **User Name**.

To discover how your Hive Server 2 is configured, examine your `hive-site.xml` file. Examine the following properties to determine which authentication mechanism your server is set to use:

- `hive.server2.authentication`
- `hive.server2.enable.doAs`

| hive.server2.authentication | hive.server2.enable.doAs | Driver Authentication Mechanism |
|---|---|---|
| NOSASL | False | No Authentication |
| KERBEROS | True or False | Kerberos |
| NONE | True or False | User Name |

Refer to chapter 13 (Setting Up Security for Manual Installs) of the HDP documentation (link) for a complete explanation of the authentication mechanisms.

## Configuring your Driver DSN

When `hive.server2.authentication` is set to `KERBEROS`, you must configure your DSN to also use **Kerberos**.

When `hive.server2.authentication` is set to `NONE`, you must configure your DSN to use **User Name**. Note that validation of the credentials that you include depends on `hive.server2.enable.doAs`.

- If `hive.server2.enable.doAs` is set to `true`, the User Name in the DSN or driver configuration must be an **existing OS user** on the host running Hive Server 2.
- If `hive.server2.enable.doAs` is set to `false`, the User Name in the DSN or driver configuration is ignored.

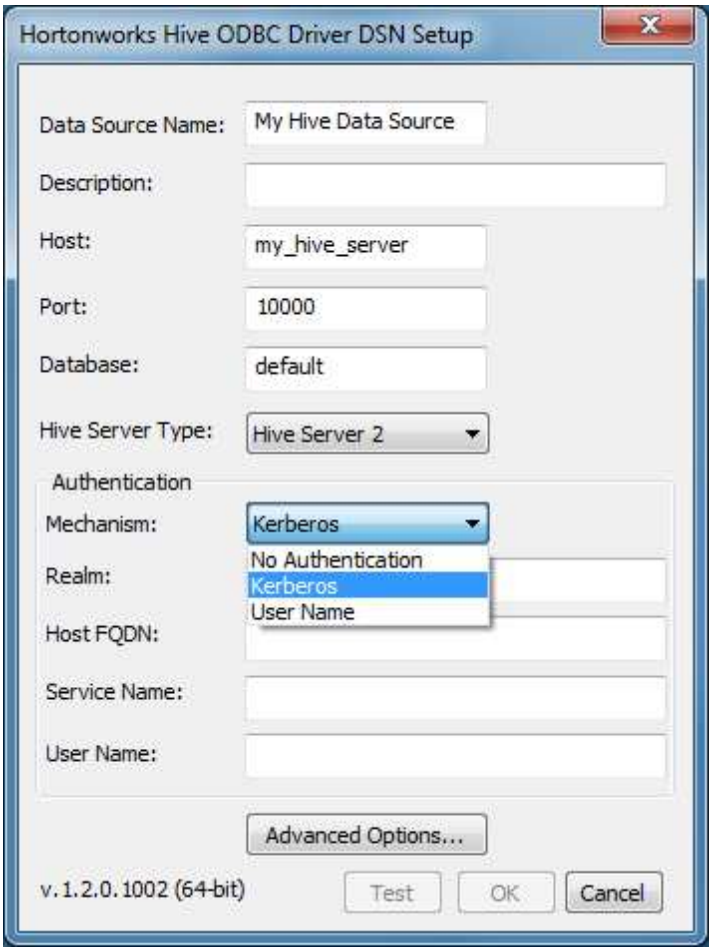
**Note**: If the User Name in the DSN or driver configuration is not supplied then the driver will default to using "anonymous" as the user name.

When `hive.server2.authentication` is set to `NOSASL`, you must configure your DSN to use **No Authentication**.

**Note**: It is an error to set `hive.server2.authentication` to `NOSASL` and `hive.server2.enable.doAs` to `true`. This configuration will not prevent the service from starting up but results in an unusable service.

For Kerberos authentication, once Kerberos has been installed and configured, set the following options in the Authentication group:

1. In the **Mechanism** field, select **Kerberos**.



2. If there is no default realm configured for your Kerberos setup, then type the value for the Kerberos realm of the Hive Server 2 host. Otherwise leave it blank. The **Realm** is only needed if your Kerberos setup does define a default realm or if the realm of your Hive Server 2 is not the default.
3. In the **Host FQDN** field, type the value for the fully qualified domain name of the Hive Server 2 host.
4. In the **Service Name** field, type the value for the service name of the Hive Server 2.

# Linux Driver

## System Requirements

- Red Hat® Enterprise Linux® (RHEL) 5.0, CentOS 5.0 or SUSE Linux Enterprise Server (SLES) 11. Both 32 and 64-bit editions are supported.
- 45 MB of available disk space.
- An installed ODBC Driver Manager, for example:
    - iODBC 3.52.7 or above
    - unixODBC 2.3.0 or above

The Hortonworks Hive ODBC Driver with SQL Connector requires a Hadoop cluster with the Hive service installed and running.

The Hortonworks Hive ODBC Driver with SQL Connector is suitable for use with all versions of Hive.

## Installation

There are two versions of the driver for Linux:

- **hive-odbc-native-32bit-*<version>*-*<release>*.i686.rpm** for 32-bit
- **hive-odbc-native-*<version>*-*<release>*.x86_64.rpm** for 64-bit

Please refer to your Linux distribution's documentation for instructions on how to install RPM packages.

The version of the driver that you select should match the bitness of the application. For example, if the 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. It is allowable to install both versions of the driver.

### Driver Directories

The Hortonworks Hive ODBC Driver files are installed in the following directories:

- /usr/lib/hive/lib/native/hiveodbc/ErrorMessages – Error messages files directory
- /usr/lib/hive/lib/native/hiveodbc/Setup – Sample configuration files directory
- /usr/lib/hive/lib/native/Linux-i386-32 – 32-bit shared libraries directory
- /usr/lib/hive/lib/native/Linux-amd64-64 – 64-bit shared libraries directory

## Configuration

### ODBC Configuration Files

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

- **.odbc.ini** – The file used to define ODBC data sources (required)
- **.odbcinst.ini** – The file used to define ODBC drivers (optional)
- **.hortonworks.hiveodbc.ini** – The file used to configure the Hortonworks Hive ODBC Driver (required)

### Sample ODBC Configuration Files

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

- **odbc.ini**
- **odbcinst.ini**
- **hortonworks.hiveodbc.ini**

The names of the sample configuration files do not begin with a period (.) so that they will appear in normal directory listings. A filename beginning with a period (.) is hidden. For **odbc.ini** and **odbcinst.ini**, if the default location is used, the filenames must begin with a period (.). For **hortonworks.hiveodbc.ini**, the filename must begin with a period (.) and must reside in the user's home directory.

If the configuration files do not already exist in the user's home directory, the sample configuration files can be copied to that directory and renamed. If the configuration files already exist in the user's home directory, the sample configuration files should be used as a guide for modifying the existing configuration files.

### ODBCINI and ODBCSYSINI Environment Configuration

By default, the configuration files reside in the user's home directory. However, two environment variables, **ODBCINI** and **ODBCSYSINI**, can be used to specify an alternative location of the **.odbc.ini** and **.odbcinst.ini** configuration files. For example, in the Bash shell, the location could be specified as follows:

```
export ODBCINI=/usr/local/odbc/myodbc.ini

export ODBCSYSINI=/usr/local/odbc/myodbcinst.ini
```

Refer to your Linux shell documentation for the exact syntax for setting environment variables.

### ODBC Data Source Configuration File Overview

ODBC Data Sources are defined in the **.odbc.ini** configuration file. The file is divided into several sections:

- [ODBC]
  The [ODBC] section is used to control global ODBC configuration such as ODBC tracing.
- [ODBC Data Sources]
  The [ODBC Data Sources] section is used to specify the available data sources.
- Data Source definitions ([<data source name>])
  The Data Source definitions are used to define the actual data source configurations.

For example, an **.odbc.ini** configuration file might look something like this:

```
[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
```

### Create a Data Source

To create a data source:

1. Open the **.odbc.ini** configuration file in a text editor.
2. Add a new entry to the [ODBC Data Sources] section. Type the data source name (DSN) and the driver name. It might look something like this:

   ```
   Sample Hortonworks Hive DSN 32=Hortonworks Hive ODBC Driver 32-bit
   ```

3. Add a new section with a name that matches the data source name (DSN). This section will contain the configuration options. They are specified as key-value pairs. For example, it might look something like this:

   ```
   [Sample Hortonworks Hive DSN 32]

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

   HOST=myhiveserver

   PORT=10000
   ```

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

### Configuration Options

The configuration options that can be used to control the behavior of the Hortonworks Hive ODBC Driver are described in the following table:

| Configuration Option | Default Value | Description |
|---|---|---|
| Driver | | The location of the Hortonworks Hive ODBC Driver shared object file. |
| HOST | | The IP address or hostname of the Hive server. |
| PORT | 10000 | The listening port for the service. |
| Schema | default | The name of the database schema to use when a schema is not explicitly specified in a query. Queries on other schemas can still be issued by explicitly specifying the schema in the query. To determine the appropriate database schema to use, type the show databases command at the Hive command prompt to inspect your databases. |
| DefaultStringColumnLength | 255 | The default string column length to use. Hive does not provide the length for String columns in its column metadata. This option allows you to tune the length of String columns. |
| UseNativeQuery | 0 | To enable the UseNativeQuery option, use a value of 1. This will disable the SQL Connector feature. The SQL Connector feature has been added to the driver to apply transformations to the queries emitted by an application to convert them into an equivalent form in HiveQL. If the application is Hive aware and already emits HiveQL then turning off the SQL Connector feature avoids the extra overhead of query transformation. |
| FastSQLPrepare | 0 | To enable the FastSQLPrepare option, use a value of 1. This will 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 this option. |
| RowsFetchedPerBlock | 10000 | The number of rows to be fetched per block. 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. |
| HiveServerType | 1 | The Hive Server Type. Set it to 1 for Hive Server and 2 for Hive Server 2. |
| HS2AuthMech | 0 | The authentication mechanism to use. Set it to 0 for no authentication, 1 for Kerberos or 2 for User Name. |
| HS2HostFQDN | | The fully qualified domain name of the Hive Server 2 host used. |

| | | |
|---|---|---|
| HS2KrbServiceName | | The Kerberos service principal name of the Hive Server 2. |
| HS2KrbRealm | | If there is no default realm configured or the realm of the Hive Server 2 host is different from the default realm for your Kerberos setup, define the realm of the Hive Server 2 host using this option. |
| UserName | | The user name of an existing user on the host running Hive Server 2. This option is used when using User Name authentication. |

### ODBC Drivers Configuration File Overview

ODBC Drivers are defined in the **.odbcinst.ini** configuration file. This configuration is optional because drivers can be specified directly in the **.odbc.ini** configuration file as discussed in the previous section.

The file is divided into these sections:

- [ODBC Drivers]
  The [ODBC Drivers] section is used to specify the available drivers.
- Driver definitions ([<driver name>])
  The Driver definitions are used to define the actual driver configurations.

For example, an **.odbcinst.ini** configuration file might look something like this:

```
[ODBC Drivers]
Hortonworks Hive ODBC Driver 32-bit=Installed
Hortonworks Hive ODBC Driver 64-bit=Installed

[Hortonworks Hive ODBC Driver 32-bit]
Driver=/usr/lib/hive/lib/native/Linux-i386-32/libhortonworkshiveodbc32.so
Description=Hortonworks Hive ODBC Driver (32-bit)

[Hortonworks Hive ODBC Driver 64-bit]
Driver=/usr/lib/hive/lib/native/Linux-amd64-64/libhortonworkshiveodbc64.so
Description=Hortonworks Hive ODBC Driver (64-bit)
```

### Define a Driver

To define a driver:

1. Open the **.odbcinst.ini** configuration file in a text editor.
2. Add a new entry to the [ODBC Drivers] section. Type driver name and the value "Installed". This driver name should be used for the "Driver" value in the data source definition instead of the driver shared library name.

   For example, it might look something like this:

   ```
   Hortonworks Hive ODBC Driver 32-bit=Installed
   ```

3. Add a new section with a name that matches the new driver name. This section will contain the configuration options. They are specified as key-value pairs. For example, it might look something like this:

   ```
   [Hortonworks Hive ODBC Driver 32-bit]

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

   Description=Hortonworks Hive ODBC Driver (32-bit)
   ```

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

### Configure the Hortonworks Hive ODBC Driver

To configure the Hortonworks Hive ODBC Driver to work with your ODBC Driver Manager:

1. Open the **.hortonworks.hiveodbc.ini** configuration file in a text editor.
2. Edit the DriverManagerEncoding setting.
   This setting is usually set to **UTF-16** or **UTF-32** depending on the ODBC Driver Manager being used. iODBC uses **UTF-32** and unixODBC uses **UTF-16**. Consult your ODBC Driver Manager documentation for the correct setting to use.
3. Edit the ODBCInstLib setting.
   This setting is set to the ODBCInst shared library for the ODBC Driver Manager being used. The configuration file defaults to iODBC's **libiodbcinst.so** shared library. You can specify the absolute or relative filename for the library. If you intend to use the relative filename for the library, the path to the library must be included in your **LD_LIBRARY_PATH** setting. Consult your ODBC Driver Manager documentation for the correct library to use.
4. Save the **.hortonworks.hiveodbc.ini** configuration file.

### Configure the Library Path

In the ODBC configuration files, the driver libraries can be specified using absolute or relative paths. If relative paths are desired, set **LD_LIBRARY_PATH** to include:

- /usr/lib/hive/lib/native/Linux-i386-32
- /usr/lib/hive/lib/native/Linux-amd64-64

Refer to your Linux shell documentation for the exact syntax for setting environment variables.

### Configure Kerberos Authentication

For more information about how to configure Kerberos authentication, refer to the documentation for your operating system.

**Note**: Authentication is not available for the server type of Hive Server 1.

# Mac OS X Driver

## System Requirements

- Mac OS X version 10.6.8 or later.
- 86 MB of available disk space.
- An installed ODBC Driver Manager, for example:
    - iODBC 3.52.7 or above

The Hortonworks Hive ODBC Driver with SQL Connector requires a Hadoop cluster with the Hive service installed and running.

The Hortonworks Hive ODBC Driver with SQL Connector is suitable for use with all versions of Hive and the driver works with both 32 and 64-bit applications.

## Installation

1. Double click the `hive-odbc-native.dmg` file.
   The Hortonworks Hive ODBC Driver volume is mounted.
2. Double click the `hive-odbc-native.pkg` file.
   The installer opens.
3. Follow the instructions in the installer and when it has finished installing, click Close.

## Driver Directories

The Hortonworks Hive ODBC Driver files are installed in the following directories:

- `/usr/lib/hive/lib/native/hiveodbc/ErrorMessages` – Error messages files directory
- `/usr/lib/hive/lib/native/hiveodbc/Setup` – Sample configuration files directory
- `/usr/lib/hive/lib/native/universal` – Binaries directory

## Configuration

### ODBC Configuration Files

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

- **.odbc.ini** – The file used to define ODBC data sources (required)
- **.odbcinst.ini** – The file used to define ODBC drivers (optional)
- **.hortonworks.hiveodbc.ini** – The file used to configure the Hortonworks Hive ODBC Driver (required)

### Sample ODBC Configuration Files

The driver installation contains the following sample configuration files in the `/usr/lib/hive/lib/native/hiveodbc/Setup` directory:

- **odbc.ini**
- **odbcinst.ini**
- **hortonworks.hiveodbc.ini**

The names of the sample configuration files do not begin with a period (.) so that they will appear in normal directory listings. A filename beginning with a period (.) is hidden. For **odbc.ini** and **odbcinst.ini**, if the default location is used, the filenames must begin with a period (.). For **hortonworks.hiveodbc.ini**, the filename must begin with a period (.) and must reside in the user's home directory.

If the configuration files do not already exist in the user's home directory, the sample configuration files can be copied to that directory and renamed. If the configuration files already exist in the user's home directory, the sample configuration files should be used as a guide for modifying the existing configuration files.

### ODBCINI and ODBCSYSINI Environment Configuration

By default, the configuration files reside in the user's home directory. However, two environment variables, **ODBCINI** and **ODBCSYSINI**, can be used to specify an alternative location of the **.odbc.ini** and **.odbcinst.ini** configuration files. For example, in the Bash shell, the location could be specified as follows:

```
export ODBCINI=/usr/local/odbc/myodbc.ini

export ODBCSYSINI=/usr/local/odbc/myodbcinst.ini
```

### ODBC Data Source Configuration File Overview

ODBC Data Sources are defined in the **.odbc.ini** configuration file. The file is divided into several sections:

- [ODBC]
  The [ODBC] section is used to control global ODBC configuration such as ODBC tracing.
- [ODBC Data Sources]
  The [ODBC Data Sources] section is used to specify the available data sources.
- Data Source definitions ([<data source name>])
  The Data Source definitions are used to define the actual data source configurations.

For example, an **.odbc.ini** configuration file might look something like this:

```
[ODBC]

InstallDir=/usr/local/odbc


[ODBC Data Sources]

Hortonworks Hive DSN=Hortonworks Hive ODBC Driver


[Hortonworks Hive DSN]

Driver=/usr/lib/hive/lib/native/universal/libhortonworkshiveodbc.dylib

HOST=myhiveserver

PORT=10000
```

### Create a Data Source

To create a data source:

1. Open the **.odbc.ini** configuration file in a text editor.
2. Add a new entry to the [ODBC Data Sources] section. Type the data source name (DSN) and the driver name. It might look something like this:

   ```
   Hortonworks Hive DSN=Hortonworks Hive ODBC Driver
   ```

3. Add a new section with a name that matches the data source name (DSN). This section will contain the configuration options. They are specified as key-value pairs. For example, it might look something like this:

   ```
   [Hortonworks Hive DSN]
   Driver=/usr/lib/hive/lib/native/universal/libhortonworkshiveodbc.dylib
   HOST=myhiveserver
   PORT=10000
   ```

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

### Configuration Options

The configuration options that can be used to control the behavior of the Hortonworks Hive ODBC Driver are described in the following table:

| Configuration Option | Default Value | Description |
|---|---|---|
| Driver | | The location of the Hortonworks Hive ODBC Driver shared object file. |
| HOST | | The IP address or hostname of the Hive server. |
| PORT | 10000 | The listening port for the service. |
| Schema | default | The name of the database schema to use when a schema is not explicitly specified in a query. Queries on other schemas can still be issued by explicitly specifying the schema in the query. To determine the appropriate database schema to use, type the show databases command at the Hive command prompt to inspect your databases. |
| DefaultStringColumnLength | 255 | The default string column length to use. Hive does not provide the length for String columns in its column metadata. This option allows you to tune the length of String columns. |
| UseNativeQuery | 0 | To enable the UseNativeQuery option, use a value of 1. This will disable the SQL Connector feature. The SQL Connector feature has been added to the driver to apply transformations to the queries emitted by an application to convert them into an equivalent form in HiveQL. If the application is Hive aware and already emits HiveQL then turning off the SQL Connector feature avoids the extra overhead of query transformation. |
| FastSQLPrepare | 0 | To enable the FastSQLPrepare option, use a value of 1. This will 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 this option. |
| RowsFetchedPerBlock | 10000 | The number of rows to be fetched per block. 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. |
| HiveServerType | 1 | The Hive Server Type. Set it to 1 for Hive Server and 2 for Hive Server 2. |
| HS2AuthMech | 0 | The authentication mechanism to use. Set it to 0 for no authentication, 1 for Kerberos or 2 for User Name. |

| | | |
|---|---|---|
| HS2HostFQDN | | The fully qualified domain name of the Hive Server 2 host used. |
| HS2KrbServiceName | | The Kerberos service principal name of the Hive Server 2. |
| HS2KrbRealm | | If there is no default realm configured or the realm of the Hive Server 2 host is different from the default realm for your Kerberos setup, define the realm of the Hive Server 2 host using this option. |
| UserName | | The user name of an existing user on the host running Hive Server 2. This option is used when using User Name authentication. |

### ODBC Drivers Configuration File Overview

ODBC Drivers are defined in the **.odbcinst.ini** configuration file. This configuration is optional because drivers can be specified directly in the **.odbc.ini** configuration file as discussed in the previous section.

The file is divided into these sections:

- [ODBC Drivers]
  The [ODBC Drivers] section is used to specify the available drivers.
- Driver definitions ([<driver name>])
  The Driver definitions are used to define the actual driver configurations.

For example, an **.odbcinst.ini** configuration file might look something like this:

```
[ODBC Drivers]

Hortonworks Hive ODBC Driver=Installed


[Hortonworks Hive ODBC Driver]

Driver=/usr/lib/hive/lib/native/universal/libhortonworkshiveodbc.dylib

Description=Hortonworks Hive ODBC Driver
```

### Define a Driver

To define a driver:

1. Open the **.odbcinst.ini** configuration file in a text editor.
2. Add a new entry to the [ODBC Drivers] section. Type driver name and the value “Installed”. This driver name should be used for the “Driver” value in the data source definition instead of the driver shared library name.

   For example, it might look something like this:

   ```
   Hortonworks Hive ODBC Driver=Installed
   ```
3. Add a new section with a name that matches the new driver name. This section will contain the configuration options. They are specified as key-value pairs. For example, it might look something like this:

   ```
   [Hortonworks Hive ODBC Driver]

   Driver=/usr/lib/hive/lib/native/universal/libhortonworkshiveodbc.dylib

   Description=Hortonworks Hive ODBC Driver
   ```
4. Save the **.odbcinst.ini** configuration file.

### Configure the Hortonworks Hive ODBC Driver

To configure the Hortonworks Hive ODBC Driver to work with your ODBC Driver Manager:

1. Open the **.hortonworks.hiveodbc.ini** configuration file in a text editor.
2. Edit the DriverManagerEncoding setting.
   This setting is usually set to **UTF-16** or **UTF-32** depending on the ODBC Driver Manager being used. iODBC uses **UTF-32**. Consult your ODBC Driver Manager documentation for the correct setting to use.
3. Edit the ODBCInstLib setting.
   This setting is set to the ODBCInst shared library for the ODBC Driver Manager being used. The configuration file defaults to iODBC's **libiodbcinst.dylib** shared library. You can specify the absolute or relative filename for the library. If you intend to use the relative filename for the library, the path to the library must be included in your **DYLD_LIBRARY_PATH** setting. Consult your ODBC Driver Manager documentation for the correct library to use.
4. Save the **.hortonworks.hiveodbc.ini** configuration file.

### Configure the Library Path

In the ODBC configuration files, the driver libraries can be specified using absolute or relative paths. If relative paths are desired, set **DYLD_LIBRARY_PATH** to include `/usr/lib/hive/lib/native/universal`.

### Configure Kerberos Authentication

For more information about how to configure Kerberos authentication, refer to the documentation for your operating system.

**Note**: Authentication is not available for the server type of Hive Server 1.

# Features

## 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, for most applications, the syntax is different enough that most applications do not work with native HiveQL.

## SQL Connector

To bridge the difference between SQL and HiveQL, we have added the **SQL Connector** feature to translate standard SQL-92 queries into equivalent HiveQL queries. The **SQL Connector** performs syntactical translations and structural transformations. For example:

1. Quoted Identifiers

   HiveQL uses back-quote while SQL uses double quote when quoting identifiers. Even when a driver reports the back-quote as the quote character, some applications still generate double quoted identifiers.

2. Table Aliases

   HiveQL does not support the AS keyword between a table reference and its alias.

3. JOIN, INNER JOIN and CROSS JOIN

   SQL INNER JOIN and CROSS JOIN syntax is translated to HiveQL JOIN syntax.

4. TOP N/LIMIT

   SQL TOP N queries are transformed to HiveQL LIMIT queries.

## Data Types

The following data types are supported: TINYINT, SMALLINT, INT, BIGINT, FLOAT, DOUBLE, BOOLEAN, STRING and TIMESTAMP.

The aggregate types (ARRAY, MAP and STRUCT) are not yet supported.

## Catalog and Schema Support

The Hortonworks Hive ODBC Driver 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 schema/database, we have added 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 **HIVE_SYSTEM**. The table has two String type columns **ENVKEY** and **ENVVALUE**. Standard SQL can be executed against the Hive system table. For example, the following query:

```
SELECT * FROM HIVE_SYSTEM.HIVE_SYSTEM WHERE ENVKEY LIKE '%hive%'
```

will return all of the Hive system environment entries whose key has the word "hive" in it. A special query, "set –v", has to be executed to fetch this information and this is not supported by all Hive versions. For versions of Hive that do not support this type of query, the driver will return an empty result set.

# Appendix A: Configuring Kerberos Authentication for Windows

## Download and install MIT Kerberos for Windows 4.0.1

1. For 64-bit machines: http://web.mit.edu/kerberos/dist/kfw/4.0/kfw-4.0.1-amd64.msi. The installer includes both 32-bit and 64-bit libraries.
2. For 32-bit machines: http://web.mit.edu/kerberos/dist/kfw/4.0/kfw-4.0.1-i386.msi. The installer includes 32-bit libraries only.

## Set up the Kerberos configuration file in the default location

1. Obtain a **krb5.conf** configuration file from your Kerberos administrator. The configuration file should also be present at /**etc/krb5.conf** on the machine hosting the Hive Server 2.
2. The default location is **C:\ProgramData\MIT\Kerberos5** but this is normally a hidden directory. Consult your Windows documentation if you wish to view and use this hidden directory.
3. Rename the configuration file from **krb5.conf** to **krb5.ini**.
4. Copy **krb5.ini** to the default location and overwrite the empty sample file.

Consult the MIT Kerberos documentation for more information on configuration.

## Set up the Kerberos configuration file in another location

If you do not want to put the Kerberos configuration file in the default location then you can use another location. The steps required to do this are as follows:

1. Obtain a **krb5.conf** configuration file for your Kerberos setup.
2. Store **krb5.conf** in an accessible directory and make note of the full path name.
3. Click the Windows **Start** menu.
4. Right-click **Computer**.
5. Click **Properties**.
6. Click **Advanced system settings**.
7. Click **Environment Variables**.
8. Click **New** for **System variables**.
9. Enter variable name: **KRB5_CONFIG**.
10. Enter variable value: **<absolute pathname to krb5.conf file>**.
11. Click **OK** to save the new variable.
12. Ensure the variable is listed in the **System variables** list.
13. Click **OK** to close Environment Variables Window.
14. Click **OK** to close System Properties Window.

## Set up the Kerberos credential cache file

1. Create a writable directory. For example, `C:\temp`
3. Click the Windows **Start** menu.
4. Right-click **Computer**.
5. Click **Properties**.
6. Click **Advanced system settings**.
7. Click **Environment Variables**.
8. Click **New** for System variables.
9. Enter variable name: `KRB5CCNAME`.
10. Enter variable value: <*writable directory from step 1*>`\krb5cache`. For example, `C:\temp\krb5cache`

    **Note:** `krb5cache` is a regular file (not a directory) managed by the Kerberos software and should not be created by the user. If you receive a permission error when you first use Kerberos, check to make sure that the `krb5cache` file does not exist as a file or a directory.
11. Click **OK** to save the new variable.
12. Ensure the variable is listed in the System variables list.
13. Click **OK** to close Environment Variables Window.
14. Click **OK** to close System Properties Window.
15. Restart your computer to ensure that MIT Kerberos for Windows uses the new settings.

## Obtain a ticket for a Kerberos principal using password

**Note:** If your Kerberos environment uses keytab files please see the next section.

1. Click the **Start** button .
2. Click **All Programs**.
3. Click the **Kerberos for Windows (64-bit)** or the **Kerberos for Windows (32-bit)** program group.
4. Use **MIT Kerberos Ticket Manager** to obtain a ticket for the principal that will be connecting to Hive Server 2.

## Obtain a ticket for a Kerberos principal using a keytab file

1. Click the **Start** button .
2. Click **All Programs**.
3. Click **Accessories**.
4. Click **Command Prompt**.
5. Type: `kinit -k -t` <*keytab pathname*> <*principal*>

<*keytab pathname*> is the full pathname to the keytab file. For example, `C:\mykeytabs\hiveserver2.keytab`

<*principal*> is the Kerberos principal to use for authentication. For example, `hive/hiveserver2.example.com@EXAMPLE.COM`

## Obtain a ticket for a Kerberos principal using the default keytab file

A default keytab file can be set for your Kerberos configuration. Consult the MIT Kerberos documentation for instructions on configuring a default keytab file.

1. Click the **Start** button .
2. Click **All Programs**.
3. Click **Accessories**.
4. Click **Command Prompt**.
5. Type: `kinit -k <principal>`

   <*principal*> is the Kerberos principal to use for authentication. For example, `hive/hiveserver2.example.com@EXAMPLE.COM`

# Appendix B: Driver Authentication Configuration for Windows

Windows applications that connect with Hive Server 1 will connect in one of two ways:

1. ODBC Data Source
2. Direct driver connection

Applications that connect using ODBC Data Sources will work with Hive Server 2 by sending the appropriate authentication credentials defined in the Data Source.

Applications that connect using a direct driver connection that is Hive Server 1 aware but not Hive Server 2 aware will not have a facility for sending authentication credentials to Hive Server 2. However, the Hortonworks Hive ODBC driver can be configured with authentication credentials using the supplied configuration tool.

**Note:** The credentials configured using the configuration application will apply for all connections made using a direct driver connection unless the application is Hive Server 2 aware and requests credentials from the user.

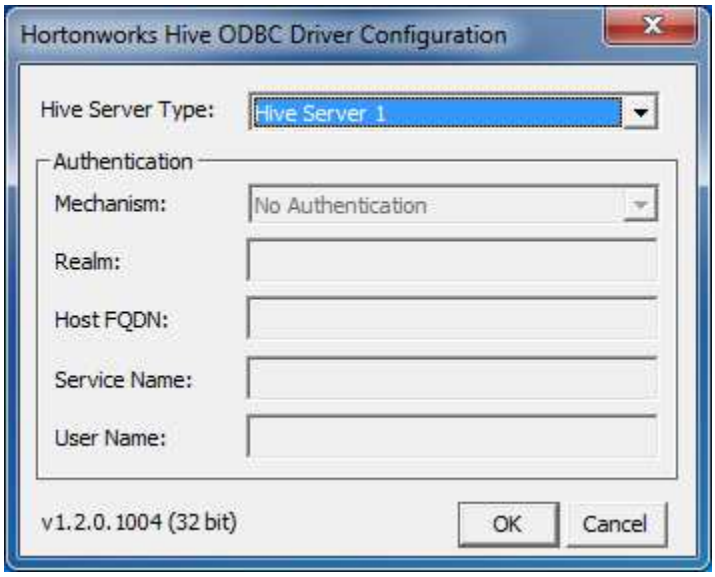
## Configure driver authentication

1. Click the **Start** button .
2. Click **All Programs.**
3. Click the **Hortonworks Hive ODBC Driver 1.2 (64-bit)** or the **Hortonworks Hive ODBC Driver 1.2 (32-bit)** program group. If you installed both versions of the driver, you will see two program groups.

   Because drivers are bit-specific, select the version that matches the bitness of your application. For example, a 32-bit driver will only be accessible from 32-bit applications.

4. Click **Driver Configuration** and click **OK** when prompted for administrator permission make modification 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.




**Note**: Authentication is only available for server of type Hive Server 2. Authentication is not available for server of type Hive Server 1.

Unlike Hive Server 1, Hive Server 2 supports multiple authentication mechanisms. You must determine the authentication type your server is using. The authentication methods available are as follows:

- No Authentication
- Kerberos
- User Name

For **No Authentication**, no additional details are required.

For **User Name** authentication, select **User Name** in the **Mechanism** field and then type a user name in the User Name field.

For **Kerberos** authentication, Kerberos must be configured before using the driver with Kerberos authentication. Refer to “Appendix A: Configuring Kerberos Authentication for Windows”.

**Note:** If you installed HDP 1.2 using Ambari, by default the authentication method is **User Name**.

To discover how your Hive Server 2 is configured, examine your `hive-site.xml` file. Examine the following properties to determine which authentication mechanism your server is set to use:

- `hive.server2.authentication`
- `hive.server2.enable.doAs`

| hive.server2.authentication | hive.server2.enable.doAs | Driver Authentication Mechanism |
|---|---|---|
| NOSASL | False | No Authentication |
| KERBEROS | True or False | Kerberos |
| NONE | True or False | User Name |

Refer to chapter 13 (Setting Up Security for Manual Installs) of the HDP documentation (link) for a complete explanation of the authentication mechanisms.

## Configuring your Driver

When `hive.server2.authentication` is set to `KERBEROS`, you must configure your DSN to also use **Kerberos**.

When `hive.server2.authentication` is set to `NONE`, you must configure your DSN to use **User Name**. Note that validation of the credentials that you include depends on `hive.server2.enable.doAs`.

- If `hive.server2.enable.doAs` is set to `true`, the User Name in the DSN or driver configuration must be an **existing OS user** on the host running Hive Server 2.
- If `hive.server2.enable.doAs` is set to `false`, the User Name in the DSN or driver configuration is ignored.

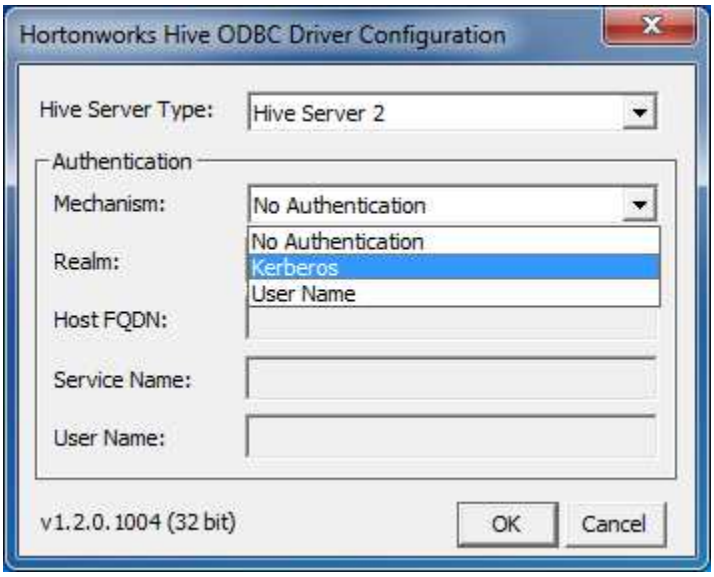
**Note**: If the User Name in the DSN or driver configuration is not supplied then the driver will default to using "anonymous" as the user name.

When `hive.server2.authentication` is set to `NOSASL`, you must configure your DSN to use **No Authentication**.

**Note**: It is an error to set `hive.server2.authentication` to `NOSASL` and `hive.server2.enable.doAs` to `true`. This configuration will not prevent the service from starting up but results in an unusable service.

For Kerberos authentication, once Kerberos has been installed and configured, set the following options in the Authentication group:

1. In the **Mechanism** field, select **Kerberos**.




2. If there is no default realm configured for your Kerberos setup, then type the value for the Kerberos realm of the Hive Server 2 host. Otherwise leave it blank. The **Realm** is only needed if your Kerberos setup does define a default realm or if the realm of your Hive Server 2 is not the default.
3. In the **Host FQDN** field, type the value for the fully qualified domain name of the Hive Server 2 host.
4. In the **Service Name** field, type the value for the service name of the Hive Server 2.

## Third Party Trademarks

**Cyrus SASL**

Copyright (c) 1998-2003 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.

**ICU License - ICU 1.8.1 and later**

COPYRIGHT AND PERMISSION NOTICE

Copyright (c) 1995-2010 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) 1998-2008 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.

**Apache Hive**

Copyright 2008-2011 The Apache Software Foundation.

**Apache Thrift**

Copyright 2006-2010 The Apache Software Foundation.

**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."

**About Hortonworks**
Hortonworks is a leading commercial vendor of Apache Hadoop, the preeminent open source platform for storing, managing and analyzing big data. Hortonworks Data Platform provides an open and stable foundation for enterprises and a growing ecosystem to build and deploy big data solutions. Hortonworks is the trusted source for information on Hadoop, and together with the Apache community, Hortonworks is making Hadoop easier to install, manage and use. Hortonworks provides technical support, training & certification programs for enterprises, systems integrators & technology vendors.

Hortonworks
3460 West Bayshore Rd.
Palo Alto, CA 94303 USA

**US**: 1.855.846.7866
**International:** 1.408.916.4121
**www.hortonworks.com**