TERADATA®

What would you do if you knew?™

Teradata ODBC Driver for Presto

Installation and Configuration Guide

Release 1.1.0
B035-6060-056K
May 2016

TERADATA LABS

About This Guide

Purpose

The *Teradata Presto ODBC Driver Installation and Configuration Guide* explains how to install and configure the Teradata Presto ODBC Driver. The guide also provides details related to features of the driver.

Audience

The guide is intended for end users of the Teradata Presto ODBC Driver, as well as administrators and developers integrating the driver.

Knowledge Prerequisites

To use the Teradata Presto ODBC Driver, the following knowledge is helpful:

- Familiarity with the platform on which you are using the Teradata Presto ODBC Driver
- Ability to use the data source to which the Teradata Presto ODBC Driver is connecting
- An understanding of the role of ODBC technologies and driver managers in connecting to a data source
- Experience creating and configuring ODBC connections
- Exposure to SQL

Document Conventions

Italics are used when referring to book and document titles.

Bold is used in procedures for graphical user interface elements that a user clicks and text that a user types.

`Monospace font` indicates commands, source code, or contents of text files.

NOTE: A text box with a pencil icon indicates a short note appended to a paragraph.

IMPORTANT: A text box with an exclamation mark indicates an important comment related to the preceding paragraph.

Table of Contents

About the Teradata Presto ODBC Driver

About Presto

Presto is a low latency distributed query engine capable of querying large datasets from multiple data sources using SQL. Presto is designed for short, interactive queries useful for data exploration.

The data sources that Presto supports include MySQL, PostgreSQL, and HBase. Presto also integrates with the Hive metastore seamlessly to complement existing Hive environments with low latency queries. Unlike traditional RDBMS or SQL-on-Hadoop solutions that require centralized schema definitions, Presto can query self-describing data as well as complex or multi-structured data that is commonly seen in big data systems. Moreover, Presto does not require a fully structured schema and can support semi-structured or nested data types such as JSON.

Presto processes the data in record batches and discovers the schema during the processing of each record batch. Thus, Presto has the capability to support changing schemas over the lifetime of a query. Presto reconfigures its operators and handles these situations to ensure that data is not lost.

NOTE: For information about connecting Presto to data sources, see the Presto documentation: https://prestodb.io/docs/current/.

About the Driver

The Teradata Presto ODBC Driver lets organizations connect their BI tools to Presto. Presto provides an ANSI SQL query layer and also exposes the metadata information through an ANSI SQL standard metadata database called INFORMATION_SCHEMA. The Teradata Presto ODBC Driver leverages INFORMATION_SCHEMA to expose Presto's metadata to BI tools as needed.

The driver complies with the ODBC 3.80 data standard, including important functionality such as Unicode and 32- and 64-bit support for high-performance computing environments on all platforms.

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

The Teradata Presto ODBC Driver is available for Microsoft® Windows®, Linux, and Mac OS X platforms.

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

> **NOTE:** For basic configuration instructions that allow you to quickly set up the Windows driver so that you can evaluate and use it, see *Teradata Presto ODBC Driver Quickstart Guide for Windows*. The guide also explains how to use the driver in various applications.

Windows Driver

Installing the Driver on Windows

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

- `TeradataPrestoODBC32.msi` for 32-bit applications
- `TeradataPrestoODBC64.msi` for 64-bit applications

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

You install the Teradata Presto ODBC Driver on client machines that access data stored in a Hadoop cluster with Presto installed and running. Each machine that you install the driver on must meet the following minimum system requirements:

- One of the following operating systems:
  - Windows 7 SP1, 8, or 8.1
  - Windows Server 2008 R2 SP1, 2012, or 2012 R2
- 75 MB of available disk space

> **IMPORTANT:** To install the driver, you must have Administrator privileges on the machine.

To install the Teradata Presto ODBC Driver:

1. Depending on the bitness of your client application, double-click to run **TeradataPrestoODBC32.msi** or **TeradataPrestoODBC64.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**.
7. If you received a license file via e-mail, then copy the license file into the `\lib` subfolder in the installation folder you selected above.

> **NOTE:** To avoid security issues, you might need to save the license file on your local machine prior to copying the file into the `\lib` subfolder.

Creating a Data Source Name

Typically, after installing the Teradata Presto ODBC Driver, you need to create a Data Source Name (DSN).

Alternatively, for information about DSN-less connections, see Using a Connection String on page 29.

To create a Data Source Name:

1. Open the ODBC Administrator:
   - If you are using Windows 7 or earlier, click the **Start** button, then click **All Programs**, then click the **Teradata Presto ODBC Driver 1.1** program group corresponding to the bitness of the client application accessing data in Presto, 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 Presto.
2. In the ODBC Data Source Administrator, click the **Drivers** tab, and then scroll down as needed to confirm that the Teradata Presto ODBC Driver appears in the alphabetical list of ODBC drivers that are installed on your system.
3. Choose one:
   - To create a DSN that only the user currently logged into Windows can use, click the **User DSN** tab.
   - Or, to create a DSN that all users who log into Windows can use, click the **System DSN** tab.
4. Click **Add**.
5. In the Create New Data Source dialog box, select **Teradata Presto ODBC Driver** and then click **Finish**. The Teradata Presto ODBC Driver DSN Setup dialog box opens.
6. In the **Data Source Name** field, type a name for your DSN.
7. Optionally, in the **Description** field, type relevant details about the DSN.
8. In the **User** field, type an appropriate user name to access the data source.
9. In the **Host** field, type the IP address or host name of the Presto server.
10. In the **Port** field, type the number of the TCP port that the Presto server uses to listen for client connections.

    NOTE: The default port number used by Presto is 8080.
11. In the **Catalog** field, type the name of the synthetic catalog under which all of the schemas/databases are organized.
12. Optionally, in the **Time Zone ID** field, type the name of the time zone for the driver to use.

13. To configure the driver to use Kerberos authentication, from the **Authentication Type** drop-down list, select **Kerberos**. For more instructions on configuring Kerberos authentication, see Configuring Kerberos Authentication on page 10.
14. To configure logging behavior for the driver, click **Logging Options**. For more information, see Configuring Logging Options on page 11.
15. 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 Teradata Presto ODBC Driver DSN Setup dialog box are correct. Contact your Presto server administrator as needed.
16. To save your settings and close the Teradata Presto ODBC Driver DSN Setup dialog box, click **OK**.
17. To close the ODBC Data Source Administrator, click **OK**.

Configuring Kerberos Authentication

You can configure the driver to use the Kerberos protocol to authenticate the connection.

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

When you configure your Kerberos server, in the `/etc/presto/config.properties` file, set the following properties:

- `http.server.authentication.krb5.service-name=HTTP`
- `http.server.authentication.krb5.keytab=HTTP.keytab`

When you use Kerberos authentication, the driver loads the credentials from the Kerberos credential cache. Therefore, the Kerberos ticket must be generated before you run the driver. To generate a Kerberos ticket, run the `kinit` Kerberos command with the appropriate principal.

To configure the driver to use Kerberos authentication:

1. Run the `kinit` command, using the following syntax, where *`[Principal]`* is the Kerberos user principal to use for authentication:
   ```
   kinit -k [Principal]
   ```
2. To access authentication options, open the ODBC Data Source Administrator where you created the DSN, select the DSN, and then click **Configure**.
3. From the **Mechanism** drop-down list, select **Kerberos**.
4. In the **SSL Certificate** field, type the path to the file containing the SSL certificate

used by Kerberos.

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

You can now use the driver to authenticate through Kerberos and connect to your Presto database.

Exporting a Data Source Name

After you configure a DSN, you can export it to be used on other machines. When you export a DSN, all of its configuration settings are saved in a `.sdc` file. You can then distribute the `.sdc` file to other users so that they can import your DSN configuration and use it on their machines.

To export a Data Source Name:

1. Open the ODBC Data Source Administrator where you created the DSN, select the DSN, click **Configure**, and then click **Logging Options**.
2. Click **Export Configuration**, specify a name and location for the exported DSN, and then click **Save**.

Your DSN is saved as a `.sdc` file in the location that you specified.

Importing a Data Source Name

You can import a DSN configuration from a `.sdc` file and then use those settings to connect to your data store.

To import a Data Source Name:

1. Open the ODBC Data Source Administrator where you created the DSN, select the DSN, click **Configure**, and then click **Logging Options**.
2. Click **Import Configuration**, browse to select the `.sdc` file that you want to import the DSN configuration from, and then click **Open**.
3. Click **OK** to close the Logging Options dialog box.

The Teradata Presto ODBC Driver DSN Setup dialog box loads the configuration settings from the selected `.sdc` file. You can now save this DSN and use it to connect to your data store.

Configuring Logging Options

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

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

The driver allows you to set the amount of detail included in log files. The following table lists the logging levels provided by the Teradata Presto ODBC Driver, in order from least verbose to most verbose.

Logging Level	Description
OFF	Disables all logging.
FATAL	Logs severe error events that lead the driver to abort.
ERROR	Logs error events that might allow the driver to continue running.
WARNING	Logs 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 all driver activity.

To enable driver 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. From the **Log Level** drop-down list, select the desired level of information to include in log files.
3. In the **Log Path** field, specify the full path to the folder where you want to save log files.
4. In the **Max Number Files** field, type the maximum number of log files to keep.

   NOTE: After the maximum number of log files is reached, each time an additional file is created, the driver deletes the oldest log file.
5. In the **Max File Size** field, type the maximum size of each log file in megabytes (MB).

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

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

The Teradata Presto ODBC Driver produces a log file named `driver.log` at the location that you specify in the Log Path field.

To disable driver logging:

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

To start tracing using the ODBC Data Source Administrator:

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

To stop ODBC Data Source Administrator tracing:

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

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

Verifying the Version Number

If you need to verify the version of the Teradata Presto ODBC Driver 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. Open the ODBC Administrator:
   - If you are using Windows 7 or earlier, click **Start**, then click **All Programs**, then click the **Teradata Presto ODBC Driver 1.1** program group corresponding to the bitness of the client application accessing data in Presto, 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 Presto.

2. Click the **Drivers** tab and then find the Teradata Presto ODBC Driver in the list of ODBC drivers that are installed on your system. The version number is displayed in the **Version** column.

Linux Driver

Linux System Requirements

You install the Teradata Presto ODBC Driver on client machines that access data stored in a Hadoop cluster with Presto installed and running. Each machine that you install the driver on must meet the following minimum system requirements:

- One of the following distributions:
  - Red Hat® Enterprise Linux® (RHEL) 5, 6, or 7
  - CentOS 5, 6, or 7
  - SUSE Linux Enterprise Server (SLES) 11 or 12
- 90 MB of available disk space
- One of the following ODBC driver managers installed:
  - iODBC 3.52.7 or later
  - unixODBC 2.3.0 or later

Installing the Driver

There are two versions of the driver for Linux:

- `TeradataPrestoODBC-32bit-`*`[Version]-[Release].[LinuxDistro]`*`.i686.rpm` for the 32-bit driver
- `TeradataPrestoODBC-`*`[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.

The bitness of the driver that you select should match the bitness of the client application accessing your 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:** Make sure that you install the driver using the RPM corresponding to your Linux distribution.

The Teradata Presto ODBC Driver driver files are installed in the following directories:

- `/opt/teradata/prestoodbc/ErrorMessages` contains error message files required by the driver.

- `/opt/teradata/prestoodbc/Setup` contains sample configuration files named `odbc.ini` and `odbcinst.ini`.
- `/opt/teradata/prestoodbc/lib/32` contains the 32-bit shared libraries.
- `/opt/teradata/prestoodbc/lib/64` contains the 64-bit shared libraries.

To install the Teradata Presto ODBC Driver:

1. Choose one:
   - 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]
     ```
2. If you received a license file via e-mail, then copy the license file into the `/opt/teradata/prestoodbc/lib/32` or `/opt/teradata/prestoodbc/lib/64` folder, depending on the version of the driver that you installed.

   NOTE: To avoid security issues, you might need to save the license file on your local machine prior to copying the file into the folder.

The Teradata Presto ODBC Driver requires the `krb5-libs` resource. To use the 32-bit driver, you need to have the 32-bit version of `krb5-libs` installed. To use the 64-bit driver, you need the 64-bit version of `krb5-libs`.

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

Setting the LD_LIBRARY_PATH Environment Variable

The LD_LIBRARY_PATH environment variable must include the paths to the installed ODBC driver manager libraries, as well as the installed Teradata Presto ODBC Driver shared libraries.

> **IMPORTANT:** While you can have both 32- and 64-bit versions of the driver installed at the same time on the same machine, do not include the paths to both 32- and 64-bit shared libraries in LD_LIBRARY PATH at the same time. Only include the path to the shared libraries corresponding to the driver matching the bitness of the client application used.

For example, if you are using a 64-bit client application and ODBC driver manager libraries are installed in `/usr/local/lib`, then set LD_LIBRARY_PATH as follows:

```
export LD_LIBRARY_
PATH=/usr/local/lib:/opt/teradata/prestoodbc/lib/64
```

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

For information about creating ODBC connections using the Teradata Presto ODBC Driver, see Configuring ODBC Connections for Non-Windows Platforms on page 20.

Verifying the Version Number

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

To verify the version number:

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

The command returns information about the Teradata Presto ODBC Driver that is installed on your machine, including the version number.

Mac OS X Driver

Installing the Driver on Mac OS X

The Teradata Presto ODBC Driver supports both 32- and 64-bit client applications.

You install the Teradata Presto ODBC Driver on client machines that access data stored in a Hadoop cluster with Presto installed and running. Each machine that you install the driver on must meet the following minimum system requirements:

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

The Teradata Presto ODBC Driver driver files are installed in the following directories:

- `/opt/teradata/prestoodbc/ErrorMessages` contains error message files required by the driver.
- `/opt/teradata/prestoodbc/Setup` contains sample configuration files named `odbc.ini` and `odbcinst.ini`.
- `/opt/teradata/prestoodbc/lib` contains the driver binaries.

To install the Teradata Presto ODBC Driver:

1. Double-click **TeradataPrestoODBC.dmg** to mount the disk image.
2. Double-click **TeradataPrestoODBC.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**.
8. If you received a license file via e-mail, then copy the license file into the `/opt/teradata/prestoodbc/lib` folder.

> **NOTE:** To avoid security issues, you might need to save the license file on your local machine prior to copying the file into the folder.

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 Teradata Presto ODBC Driver, see Configuring ODBC Connections for Non-Windows Platforms on page 20.

Verifying the Version Number

If you need to verify the version of the Teradata Presto ODBC Driver 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
com.teradata.prestoodbc
```

The command returns information about the Teradata Presto ODBC Driver that is installed on your machine, including the version number.

Configuring ODBC Connections for Non-Windows Platforms

The following sections describe how to configure ODBC connections when using the Teradata Presto ODBC Driver with non-Windows platforms:

Configuration Files

ODBC driver managers use configuration files to define and configure ODBC data sources and drivers. By default, the following configuration files 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.

These files are located in the user's home directory.

Also, by default the Teradata Presto ODBC Driver is configured using the `teradata.prestoodbc.ini` file. This file is located in one of the following directories depending on the version of the driver that you are using:

- `/opt/teradata/prestoodbc/lib/32` for the 32-bit driver on Linux.
- `/opt/teradata/prestoodbc/lib/64` for the 64-bit driver on Linux.
- `/opt/teradata/prestoodbc/lib` for the driver on Mac OS X.

The `teradata.prestoodbc.ini` file is required.

NOTE: The `teradata.prestoodbc.ini` file in the `/lib` subfolder provides default settings for most configuration options available in the Teradata Presto ODBC Driver.

Also, the installer for the Mac OS X version of the driver creates a sample User DSN in the following two files:

- `~/Library/ODBC/odbc.ini`
- `~/.odbc.ini`

You can set driver configuration options in your `odbc.ini` and `teradata.prestoodbc.ini` files. Configuration options set in a `teradata.prestoodbc.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 `teradata.prestoodbc.ini`. For information about the configuration options available for controlling the behavior of DSNs that are using the Teradata Presto ODBC Driver, see Driver Configuration Options on page 33.

Sample Configuration 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 Teradata Presto ODBC Driver.

The names of the sample configuration files do not begin with a period (`.`) so that they 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 TERADATAPRESTOINI, to specify different locations for the `odbc.ini`, `odbcinst.ini`, and `teradata.prestoodbc.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 TERADATAPRESTOINI to point to your `teradata.prestoodbc.ini` file.

For example, if your `odbc.ini` and `teradata.prestoodbc.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 TERADATAPRESTOINI=/etc/teradata.prestoodbc.ini
```

The following search order is used to locate the `teradata.prestoodbc.ini` file:

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

   NOTE: TERADATAPRESTOINI must specify the full path, including the file name.

2. The directory containing the driver's binary is searched for a file named `teradata.prestoodbc.ini` (not beginning with a period).
3. The current working directory of the application is searched for a file named `teradata.prestoodbc.ini` (not beginning with a period).
4. The directory `~/`, that is, $HOME, is searched for a hidden file named `teradata.prestoodbc.ini` (beginning with a period).
5. The directory `/etc` is searched for a file named `teradata.prestoodbc.ini` (not beginning with a period).

Defining DSNs in odbc.ini

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

- `[ODBC]` is optional. This section is used to control global ODBC configuration, such as ODBC tracing.
- `[ODBC Data Sources]` is required. This section lists the DSNs and associates them 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:

```
[ODBC Data Sources]
Teradata Presto DSN 32=Teradata Presto ODBC Driver 32-bit
[Teradata Presto DSN 32]
Driver=/opt/teradata/prestoodbc/lib/32/libprestoodbc_sb32.so
```

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

```
[ODBC Data Sources]
Teradata Presto DSN=Teradata Presto ODBC Driver
[Teradata Presto DSN]
```

```
Driver=/opt/teradata/prestoodbc/lib/libprestoodbc_
sbUniversal.dylib
```

To create a Data Source Name:

1. In a text editor, open the `odbc.ini` configuration file.
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. Add a new section to the file, with a section name that matches the DSN you specified above, and then add configuration options to the section. Specify the configuration options as key-value pairs.
4. Save the `odbc.ini` configuration file.

For information about the configuration options available for controlling the behavior of DSNs that are using the Teradata Presto ODBC Driver, see Driver Configuration Options on page 33.

Specifying ODBC Drivers in odbcinst.ini

ODBC drivers are defined in the `odbcinst.ini` configuration file. This configuration file is optional because drivers can be specified directly in the `odbc.ini` configuration file, as described in Defining DSNs in odbc.ini on page 22.

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

- `[ODBC Drivers]` lists the names of all the installed ODBC drivers.
- For each driver, a section having the same name as the driver name specified in the `[ODBC Drivers]` section lists the driver attributes and values.

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

```
[ODBC Drivers]
Teradata Presto ODBC Driver 32-bit=Installed
Teradata Presto ODBC Driver 64-bit=Installed
[Teradata Presto ODBC Driver 32-bit]
Description=Teradata Presto ODBC Driver (32-bit)
Driver=/opt/teradata/prestoodbc/lib/32/libprestoodbc_sb32.so
[Teradata Presto ODBC Driver 64-bit]
Description=Teradata Presto ODBC Driver (64-bit)
Driver=/opt/teradata/prestoodbc/lib/64/libprestoodbc_sb64.so
```

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

```
[ODBC Drivers]
Teradata Presto ODBC Driver=Installed
[Teradata Presto ODBC Driver]
Description=Teradata Presto ODBC Driver
```

```
Driver=/opt/teradata/prestoodbc/lib/libteradataprestoodbc_
sbUniversal.dylib
```

To define a driver:

1. In a text editor, open the `odbcinst.ini` configuration file.
2. In the `[ODBC Drivers]` section, add a new entry by typing the driver name and then typing `=Installed`.

   NOTE: Give the driver a symbolic name that you want to use to refer to the driver in connection strings or DSNs.
3. Add a new section to the file with a name that matches the driver name you typed above, and then add configuration options to the section based on the sample `odbcinst.ini` file provided in the Setup directory. Specify the configuration options as key-value pairs.
4. Save the `odbcinst.ini` configuration file.

Configuring Driver Settings in teradata.prestoodbc.ini

The `teradata.prestoodbc.ini` file contains configuration settings for the Teradata Presto ODBC Driver. Settings that you define in this file apply to all connections that use the driver.

You do not need to modify the settings in the `teradata.prestoodbc.ini` file to use the driver and connect to your data source.

However, to help troubleshoot issues, you can configure the `teradata.prestoodbc.ini` file to enable logging in the driver. For information about configuring logging, see Configuring Logging Options on page 25.

Configuring Kerberos Authentication

You can configure the driver to use the Kerberos protocol to authenticate the connection.

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

When you configure your Kerberos server, in the `/etc/presto/config.properties` file, set the following properties:

- `http.server.authentication.krb5.service-name=HTTP`
- `http.server.authentication.krb5.keytab=HTTP.keytab`

When you use Kerberos authentication, the driver loads the credentials from the Kerberos credential cache. Therefore, the Kerberos ticket must be generated before you run the driver. To generate a Kerberos ticket, run the `kinit` Kerberos command with the appropriate principal.

To configure Kerberos authentication:

1. Run the `kinit` command, using the following syntax, where *`[Principal]`* is the Kerberos user principal to use for authentication:
   `kinit -k [Principal]`
2. In a text editor, open the `odbc.ini` configuration file.
3. Set the `AuthMechanism` property to `Kerberos`.
4. Set the `SSLCert` property to the service name of the Presto server.
5. Save the `odbc.ini` configuration file.

You can now use the driver to authenticate through Kerberos and connect to your Presto database.

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. The following table lists the logging levels provided by the Teradata Presto ODBC Driver, in order from least verbose to most verbose.

LogLevel Value	Description
0	Disables all logging.
1	Logs severe error events that lead the driver to abort.
2	Logs error events that might allow the driver to continue running.
3	Logs potentially harmful situations.
4	Logs general information that describes the progress of the driver.

LogLevel Value	Description
5	Logs detailed information that is useful for debugging the driver.
6	Logs all driver activity.

To enable logging:

1. Open the `teradata.prestoodbc.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. Set the `LogFileCount` key to the maximum number of log files to keep.

   NOTE: After the maximum number of log files is reached, each time an additional file is created, the driver deletes the oldest log file.
5. Set the `LogFileSize` key to the maximum size of each log file in megabytes (MB).

   NOTE: After the maximum file size is reached, the driver creates a new file and continues logging.
6. Save the `teradata.prestoodbc.ini` configuration file.
7. Restart your ODBC application to make sure that the new settings take effect.

   The Teradata Presto ODBC Driver produces a log file named `driver.log` at the location you specify using the `LogPath` key.

To disable logging:

1. Open the `teradata.prestoodbc.ini` configuration file in a text editor.
2. Set the `LogLevel` key to `0`.
3. Save the `teradata.prestoodbc.ini` configuration file.

Testing the Connection

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

Using the iODBC Driver Manager

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

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

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

To test your connection using the iODBC driver manager:

1. Run iodbctest or iodbctestw.
2. Optionally, if you do not remember the DSN, then type a question mark (?) to see a list of available DSNs.
3. Type an ODBC connection string using the following format, specifying additional connection attributes as needed:

   `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 (`;`).

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

Using the unixODBC Driver Manager

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

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

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

To test your connection using the unixODBC driver manager:

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

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

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

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

Using a Connection String

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

The connection strings in the following sections are examples showing the minimum set of connection attributes that you must specify to successfully connect to the data source. Depending on the configuration of the data source and the type of connection you are working with, you might need to specify additional connection attributes. For detailed information about all the attributes that you can use in the connection string, see Driver Configuration Options on page 33.

DSN Connection String Example

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

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

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

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

DSN-less Connection String Examples

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

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

- *CatalogName* is the name of the catalog to which you are connecting.
- *PortNumber* is the number of the port that the Presto server uses to listen for client connections.
- *Server* is the IP address or host name of the Presto server to which you are connecting.
- *UserName* is the user name that you use to access the Presto server.

The following is the format of a DSN-less connection string:

```
Driver=Teradata Presto ODBC
Driver;Catalog=
CatalogName;Host=Server;Port=PortNumber;User=UserName
```

For example:

```
Driver=Teradata Presto ODBC Driver;Catalog=hive;
Host=192.168.222.160;Port=8080;User=presto
```

Features

More information is provided on the following features of the Teradata Presto ODBC Driver:

Catalog and Schema Support

The Teradata Presto ODBC Driver supports both catalogs and schemas to make it easy for the driver to work with various ODBC applications.

The Teradata Presto ODBC Driver supports querying against Hive, MySQL, and PostgreSQL schemas.

Parameters

A parameterized query contains placeholders that are used for parameters. The values of those parameters are supplied at execution time.

The Teradata Presto ODBC Driver fully supports parameterized queries.

Data Types

The Teradata Presto ODBC Driver supports many common SQL data types.

The table below lists the supported data types.

Supported SQL types	
ARRAY	MAP
BIGINT	ROW
BOOLEAN	TIME
DATE	TIME WITH TIME ZONE
DECIMAL	TIMESTAMP
DOUBLE	TIMESTAMP WITH TIME ZONE

Supported SQL types	
INTERVAL DAY TO SECOND	VARBINARY
INTERVAL YEAR TO MONTH	VARCHAR (fixed length)
JSON	VARCHAR (variable length)

Driver Configuration Options

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

When creating or configuring a connection from a Windows computer, the fields and buttons are available in the Teradata Presto ODBC Driver DSN Setup dialog box. When using a connection string or configuring a connection from a Linux or Mac OS X computer, use the key names provided.

Configuration Options Appearing in the User Interface

The following configuration options are accessible via the Windows user interface for the Teradata Presto ODBC Driver, or via the key name when using a connection string or configuring a connection from a Linux or Mac OS X computer:

When creating or configuring a connection from a Windows computer, the fields and buttons are available in the Teradata Presto ODBC Driver DSN Setup dialog box. When using a connection string or configuring a connection from a Linux or Mac OS X computer, use the key names provided.

Authentication Type

Key Name	Default Value	Required
`AuthenticationType`	`No Authentication`	No

Description

This option specifies the type of authentication that the driver uses.

Select from the following:

- `No Authentication`: The driver does not authenticate the connection.
- `Kerberos`: The driver uses Kerberos to authenticate the connection. For more information about Kerberos authentication, see the MIT Kerberos Documentation: http://web.mit.edu/kerberos/krb5-latest/doc/.

Catalog

Key Name	Default Value	Required
`Catalog`	`Hive`	Yes

Description

The name of the synthetic catalog under which all of the schemas/databases are organized.

Host

Key Name	Default Value	Required
`HOST`	`localhost`	Yes

Description

The IP address or host name of the Presto server.

Port

Key Name	Default Value	Required
`Port`	`8080`	Yes

Description

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

SSL Certificate

Key Name	Default Value	Required
`SSLCert`	None	Yes, if the authentication type is set to Kerberos

Description

This option specifies the path to the SSL certificate that is used by the Kerberos server.

Time Zone ID

Key Name	Default Value	Required
`TimeZoneID`	None	No

Description

This option specifies the local time zone that the driver uses.

Valid values for this option are specified in the IANA Time Zone Database. For a complete list of time zones, see http://www.iana.org/time-zones.

User

Key Name	Default Value	Required
`UID`	None	Yes

Description

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

Configuration Options Having Only Key Names

The `Driver` configuration option does not appear in the Windows user interface for the Teradata Presto ODBC Driver. It is accessible only when you use a connection string or configure a connection on Mac OS X or Linux.

Driver

Key Name	Default Value	Required
`Driver`	`Teradata Presto ODBC Driver`	Yes

Description

The name of the installed driver (`Teradata Presto ODBC Driver`) or the absolute path of the Teradata Presto ODBC Driver shared object file.

Third-Party Trademarks

Debian is a trademark or registered trademark of Software in the Public Interest, Inc. or its subsidiaries in Canada, United States and/or other countries.

Linux is the registered trademark of Linus Torvalds in Canada, United States and/or other countries.

Mac, Mac OS, and OS X are trademarks or registered trademarks of Apple, Inc. or its subsidiaries in Canada, United States and/or other countries.

Microsoft, MSDN, Windows, Windows Server, Windows Vista, and the Windows start button are trademarks or registered trademarks of Microsoft Corporation or its subsidiaries in Canada, United States and/or other countries.

Red Hat, Red Hat Enterprise Linux, and CentOS are trademarks or registered trademarks of Red Hat, Inc. or its subsidiaries in Canada, United States and/or other countries.

SUSE is a trademark or registered trademark of SUSE LLC or its subsidiaries in Canada, United States and/or other countries.

Ubuntu is a trademark or registered trademark of Canonical Ltd. or its subsidiaries in Canada, United States and/or other countries.

All other trademarks are trademarks of their respective owners.

Third-Party Licenses

The licenses for the third-party libraries that are included in this product are listed below.

cURL License

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.

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.

Expat License

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.

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 License

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.

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

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.]

RapidJSON License

Tencent is pleased to support the open source community by making RapidJSON available.

Copyright (C) 2015 THL A29 Limited, a Tencent company, and Milo Yip. All rights reserved.

If you have downloaded a copy of the RapidJSON binary from Tencent, please note that the RapidJSON binary is licensed under the MIT License.

If you have downloaded a copy of the RapidJSON source code from Tencent, please note that RapidJSON source code is licensed under the MIT License, except for the third-party components listed below which are subject to different license terms. Your integration of RapidJSON into your own projects may require compliance with the MIT License, as well as the other licenses applicable to the third-party components included within RapidJSON.

A copy of the MIT License is included in this file.

Other dependencies and licenses:

Open Source Software Licensed Under the BSD License:

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 shall be used for Good, not Evil.

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. 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.

Terms of the MIT License:

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 NONINFRINGEMENT. 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

acknowledgment in the product documentation would be appreciated but is not required.
2. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.
3. This notice may not be removed or altered from any source distribution.

Jean-loup Gailly jloup@gzip.org

Mark Adler madler@alumni.caltech.edu