TERADATA

What would you do if you knew?™

**Teradata ODBC Driver for Presto
Installation and Configuration Guide**

TERADATA
LABS

About This Guide

Purpose

The *Teradata ODBC Driver for Presto Installation and Configuration Guide* explains how to install and configure the Teradata Presto ODBC Driver with SQL Connector. 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:**
>
> This text box indicates a short note appended to a paragraph.

> **Important:**
>
> This text box 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 and PostgreSQL. Presto also integrates seamlessly with the Hive metastore 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: http://teradata/github.io/presto/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 macOS platforms.

The *Teradata ODBC Driver for Presto 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.

Windows Driver

Windows System Requirements

The Teradata Presto ODBC Driver supports open-source Presto Server versions 0.160 and earlier, and Teradata Presto Server versions 0.157-t and earlier.

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

- One of the following operating systems:
    - o Windows Vista, 7, 8, or 10
    - o Windows Server 2008 or later
- 75 MB of available disk space
- Visual C++ Redistributable for Visual Studio 2013 installed (with the same bitness as the driver that you are installing).

You can download the installation packages at https://www.microsoft.com/en-ca/download/details.aspx?id=40784.

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

Installing the Driver on Windows

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

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

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

To install the Teradata Presto ODBC Driver on Windows:

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**.

Creating a Data Source Name on Windows

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 36.

To create a Data Source Name on Windows:

1. Open the ODBC Administrator:

   - If you are using Windows 7 or earlier, click **Start ☐ > All Programs >** Teradata Presto ODBC Driver **1.1 > ODBC Administrator**.
   - Or, if you are using Windows 8 or later, on the Start screen, type **ODBC administrator**, and then click the **ODBC Administrator** search result.

   > **Note:**
   >
   > Make sure to select the ODBC Data Source Administrator that has the same bitness as the client application that you are using to connect to 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.

> **Note:**
>
> It is recommended that you create a System DSN instead of a User DSN. Some applications load the data using a different user account, and might not be able to detect User DSNs that are created under another user account.

4. Click **Add**.

5. In the Create New Data Source dialog box, select **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. If the database that you are connecting to requires authentication, then use the options in the Authentication area to configure authentication as needed. For more information, see Configuring Authentication on Windows on page 12.

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. In the **Schema** field, type the name of the the schema for the driver to use.

13. Optionally, in the **Time Zone ID** field, type the name of the time zone for the driver to use, in tz database format. For a list of time zones in tz database format, see https://en.wikipedia.org/wiki/List_of_tz_database_time_zones.

    If a time zone is not specified, the driver uses the system time zone.

14. To configure client-server verification over SSL, click **SSL Options**. For more information, see Configuring SSL Verification on Windows on page 13.

15. To configure advanced driver options, click **Advanced Options**. For more information, see Configuring Advanced Options on Windows on page 12.

16. To configure a connection to a datasource through a proxy server, click **Proxy Options**. For more information, see Configuring a Proxy Server on Windows on page 14.

17. To configure logging behavior for the driver, click **Logging Options**. For more information, see Configuring Logging Options on Windows on page 15.

18. 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.

19. To save your settings and close the Teradata Presto ODBC Driver DSN Setup dialog box, click **OK**.

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

Configuring Advanced Options on Windows

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

To configure advanced options on Windows:

1. Open the ODBC Data Source Administrator where you created the DSN, then select the DSN, then click **Configure**, and then click **Advanced Options**.

2. To specify the version of the Presto server that the driver is connecting to, in the **Server Version** field, type the server version number.

3. To automatically test the connection, select **Connection Test**.

4. To automatically populate the metadata for parameters, select **Auto Populate Parameter Metadata**.

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

Configuring Authentication on Windows

Some Presto data stores require authentication. You can configure the Teradata Presto ODBC Driver to provide your credentials and authenticate the connection to the database using one of the following methods:

- Using Kerberos Authentication on page 12
- Using LDAP Authentication on page 13

> **Note:**
>
> If Kerberos or LDAP authentication is enabled, then SSL is automatically enabled.

Using Kerberos Authentication

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

When you log in to Windows, the operating system automatically caches your credentials. When the driver is run, it loads your Kerberos credentials from the Windows Kerberos cache.

To configure the driver to use Kerberos authentication on Windows:

1. Open the ODBC Data Source Administrator where you created the DSN, select the DSN, and then click **Configure**.

2. From the **Authentication Type** drop-down list, select **Kerberos Authentication**.

3. To configure client-server verification over SSL, click **SSL Options**. For more information, see Configuring SSL Verification on Windows on page 13.

4. 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 server.

Using LDAP Authentication

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

To configure LDAP authentication on Windows:

1. Open the ODBC Data Source Administrator where you created the DSN, select the DSN, and then click **Configure**.

2. From the **Authentication Type** drop-down list, select **LDAP Authentication**.

3. In the **User** field, type an appropriate user name for accessing the data store.

4. In the **Password** field, type the password corresponding to the user name that you specified above.

5. Optionally, to save the password in the Windows registry, select **Save Password (Encrypted)**.

6. To configure client-server verification over SSL, click **SSL Options**. For more information, see Configuring SSL Verification on Windows on page 13.

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

You can now use the driver to authenticate through LDAP and connect to your Presto server.

Configuring SSL Verification on Windows

If you are connecting to a Presto server that has Secure Sockets Layer (SSL) enabled, you can configure the driver to connect to an SSL-enabled socket. When using SSL to connect to a server, the driver can be configured to verify the identity of the server.

> **Note:**
>
> If Kerberos or LDAP authentication is enabled, then SSL is automatically enabled.

To configure SSL verification on Windows:

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

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

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

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

5. To specify the CA certificates that you want to use to verify the server, do one of the following:

    - To configure the driver to load certificates from a specific PEM file, specify the full path to the file in the **Trusted Certificates** field.

    - Or, to use the certificates in the PEM file that is installed with the driver, leave the **Trusted Certificates** field empty.

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

Configuring a Proxy Server on Windows

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

To configure a proxy server on Windows:

1. To access proxy server options, open the ODBC Data Source Administrator where you created the DSN, then select the DSN, then click **Configure**, and then click **Proxy Options**.

2. Select the **Use Proxy Server** check box.

3. In the **Proxy Host** field, type the host name or IP address of the proxy server.

4. In the **Proxy Port** field, type the number of the TCP port that the proxy server uses to listen for client connections.

5. In the **Proxy Username** field, type your user name for accessing the proxy server.

6. In the **Proxy Password** field, type the password corresponding to the user name.

7. Optionally, to save the proxy server password in the Windows registry, select **Save Password (Encrypted)**.

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

Exporting a Data Source Name on Windows

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 on Windows:

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 on Windows

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

To import a Data Source Name on Windows:

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 source.

Configuring Logging Options on Windows

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.

To enable driver logging on Windows:

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

2. From the **Log Level** drop-down list, select the logging level corresponding to the amount of information that you want to include in log files:

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

3. In the **Log Path** field, specify the full path to the folder where you want to save log files.

4. 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 on Windows:

1. Open the ODBC Data Source Administrator where you created the DSN, then select the DSN, then click **Configure**, and then click **Logging Options**.

2. From the **Log Level** drop-down list, select **LOG_OFF**.

3. Click **OK**.

4. Restart your ODBC application to make sure that the new settings take effect.

Verifying the Driver Version Number on Windows

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 driver version number on Windows:

1. Open the ODBC Administrator:

   - If you are using Windows 7 or earlier, click **Start □ > All Programs >** Teradata Presto ODBC Driver **1.1 > ODBC Administrator**.

   - Or, if you are using Windows 8 or later, on the Start screen, type **ODBC administrator**, and then click the **ODBC Administrator** search result.

   > **Note:**
   >
   > Make sure to select the ODBC Data Source Administrator that has the same bitness as the client application that you are using to connect to 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.

macOS Driver

macOS System Requirements

The Teradata Presto ODBC Driver supports open-source Presto Server versions 0.160 and earlier, and Teradata Presto Server versions 0.157-t and earlier.

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

- macOS version 10.9, 10.10, or 10.11
- 150 MB of available disk space
- iODBC 3.52.7 or later

Installing the Driver on macOS

The Teradata Presto ODBC Driver is available for macOS as a `.dmg` file named `TeradataPrestoODBC.dmg`. The driver supports both 32- and 64-bit client applications.

To install the Teradata Presto ODBC Driver on macOS:

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**.

> **Note:**
>
> By default, the driver files are installed in the `/Library/teradata/prestoodbc` directory.

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

Next, configure the environment variables on your machine to make sure that the ODBC driver manager can work with the driver. For more information, see Configuring the ODBC Driver Manager on Non-Windows Machines on page 23.

Verifying the Driver Version Number on macOS

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

To verify the driver version number on macOS:

1. 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.

Linux Driver

Linux System Requirements

The Teradata Presto ODBC Driver supports open-source Presto Server versions 0.160 and earlier, and Teradata Presto Server versions 0.157-t and earlier.

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

- One of the following distributions:
  - o Red Hat® Enterprise Linux® (RHEL) 5, 6, or 7
  - o CentOS 5, 6, or 7
  - o SUSE Linux Enterprise Server (SLES) 11 or 12
  - o Debian 7 or 8
  - o Ubuntu 12.04, 14.04, or 16.04
- 90 MB of available disk space
- One of the following ODBC driver managers installed:
  - o iODBC 3.52.7 or later
  - o unixODBC 2.3.0 or later

The `krb5-libs` library that matches the bitness of the driver must be installed.

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

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

Installing the Driver Using the RPM File

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

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

- `TeradataPrestoODBC-64bit-`*[Version]*-*[Release]*`.rpm` for the 64-bit driver

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

The placeholders in the file names are defined as follows:

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

To install the Teradata Presto ODBC Driver using the RPM File:

1. Log in as the root user, and then navigate to the folder containing the RPM package for the driver.
2. Depending on the Linux distribution that you are using, run one of the following commands from the command line, where *[RPMFileName]* is the file name of the RPM package:

- If you are using Red Hat Enterprise Linux or CentOS, run the following command:

  ```
  yum --nogpgcheck localinstall [RPMFileName]
  ```

- Or, if you are using SUSE Linux Enterprise Server, run the following command:

  ```
  zypper install [RPMFileName]
  ```

The Teradata Presto ODBC Driver files are installed in the `/opt/teradata/prestoodbc` directory.

> **Note:**
>
> If the package manager in your Linux distribution cannot resolve the `krb5-libs` dependency automatically when installing the driver, then download and manually install the package.

Next, configure the environment variables on your machine to make sure that the ODBC driver manager can work with the driver. For more information, see Configuring the ODBC Driver Manager on Non-Windows Machines on page 23.

Verifying the Driver Version Number on Linux

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 driver version number on Linux:

1. 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.

Configuring the ODBC Driver Manager on Non-Windows Machines

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

- Make sure that your machine uses the correct ODBC driver manager by setting the library path environment variable. For more information, see Specifying ODBC Driver Managers on Non-Windows Machines on page 23.

- If the driver configuration files are not stored in the default locations, then make sure that the ODBC driver manager locates and uses those files by setting environment variables. For more information, see Specifying the Locations of the Driver Configuration Files on page 24.

After configuring the ODBC driver manager, you can configure a connection and access your data store through the driver. For more information, see Configuring ODBC Connections on a Non-Windows Machine on page 26.

Specifying ODBC Driver Managers on Non-Windows Machines

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

macOS

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

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

For information about setting an environment variable permanently, refer to the macOS shell documentation.

Linux

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

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

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

Specifying the Locations of the Driver Configuration Files

By default, ODBC driver managers are configured to use hidden versions of the `odbc.ini` and `odbcinst.ini` configuration files (named `.odbc.ini` and `.odbcinst.ini`) located in the home directory, as well as the `teradata.prestoodbc.ini` file in the `lib` subfolder of the driver installation directory. If you store these configuration files elsewhere, then you must set the environment variables described below so that the driver manager can locate the files.

If you are using iODBC, do the following:

- Set ODBCINI to the full path and file name of the `odbc.ini` file.

- Set ODBCINSTINI to the full path and file name of the `odbcinst.ini` file.

- Set SIMBAPRESTOINI to the full path and file name of the `teradata.prestoodbc.ini` file.

If you are using unixODBC, do the following:

- Set ODBCINI to the full path and file name of the `odbc.ini` file.

- Set ODBCSYSINI to the full path of the directory that contains the `odbcinst.ini` file.

- Set SIMBAPRESTOINI to the full path and file name of the `teradata.prestoodbc.ini` file.

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

For iODBC:
```
export ODBCINI=/usr/local/odbc/odbc.ini
export ODBCINSTINI=/usr/local/odbc/odbcinst.ini
export SIMBAPRESTOINI=/etc/teradata.prestoodbc.ini
```

For unixODBC:
```
export ODBCINI=/usr/local/odbc/odbc.ini
export ODBCSYSINI=/usr/local/odbc
export SIMBAPRESTOINI=/etc/teradata.prestoodbc.ini
```

To locate the `teradata.prestoodbc.ini` file, the driver uses the following search order:

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

2. The driver searches the directory that contains the driver library files for a file named `teradata.prestoodbc.ini`.

3. The driver searches the current working directory of the application for a file named `teradata.prestoodbc.ini`.

4. The driver searches the home directory for a hidden file named `.teradata.prestoodbc.ini` (prefixed with a period).

5. The driver searches the `/etc` directory for a file named `teradata.prestoodbc.ini`.

Configuring ODBC Connections on a Non-Windows Machine

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

- Creating a Data Source Name on a Non-Windows Machine on page 26
- Configuring a DSN-less Connection on a Non-Windows Machine on page 28
- Configuring Authentication on a Non-Windows Machine on page 30
- Configuring SSL Verification on a Non-Windows Machine on page 32
- Configuring Logging Options on a Non-Windows Machine on page 32
- Testing the Connection on a Non-Windows Machine on page 34

Creating a Data Source Name on a Non-Windows Machine

When connecting to your data store using a DSN, you only need to configure the `odbc.ini` file. Set the properties in the `odbc.ini` file to create a DSN that specifies the connection information for your data store. For information about configuring a DSN-less connection instead, see Configuring a DSN-less Connection on a Non-Windows Machine on page 28.

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

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

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

   > **Note:**
   >
   > If you are using a hidden copy of the `odbc.ini` file, you can remove the period (.) from the file name to make the file visible while you are editing it.

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

   For example, on a macOS machine:

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

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

```
[ODBC Data Sources]
Sample DSN=Teradata Presto ODBC Driver 32-bit
```

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

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

      For example, on a macOS machine:

      ```
      Driver=/Library/teradata/prestoodbc/lib/libprestoodbc_sbu.dylib
      ```

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

      ```
      Driver=/opt/teradata/prestoodbc/lib/32/libprestoodbc_sb32.so
      ```

   b. Set the `Host` property to the IP address or host name of the server, and then set the `Port` property to the number of the TCP port that the server uses to listen for client connections.

      For example:

      ```
      Host=192.168.222.160
      Port=8080
      ```

   c. Set the `Catalog` property to the name of the catalog to which you are connecting.

      For example:

      ```
      Catalog=hive
      ```

   d. If authentication is required to access the server, then specify the authentication mechanism and your credentials. For more information, see Configuring Authentication on a Non-Windows Machine on page 30.

   e. Optionally, set additional key-value pairs as needed to specify other optional connection settings. For detailed information about all the configuration options supported by the Teradata Presto ODBC Driver, see Driver Configuration Options on page 41.

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

> **Note:**
>
> If you are storing this file in its default location in the home directory, then prefix the file name with a period (`.`) so that the file becomes hidden. If you are storing this file in another location, then save it as a non-hidden file (without the prefix), and make sure that the ODBCINI environment variable specifies the location. For more information, see Specifying the Locations of the Driver Configuration Files on page 24.

For example, the following is an `odbc.ini` configuration file for macOS containing a DSN that connects to Google Presto using a user account:

```
[ODBC Data Sources]
Sample DSN=Teradata Presto ODBC Driver
[Sample DSN]
Driver=/Library/teradata/prestoodbc/lib/libprestoodbc_sbu.dylib
Host=192.168.222.160
Port=8080
Catalog=hive
```

As another example, the following is an `odbc.ini` configuration file for a 32-bit driver on a Linux machine, containing a DSN that connects to Google Presto using a user account:

```
[ODBC Data Sources]
Sample DSN=Teradata Presto ODBC Driver 32-bit
[Sample DSN]
Driver=/opt/teradata/prestoodbc/lib/32/libprestoodbc_sb32.so
Host=192.168.222.160
Port=8080
Catalog=hive
```

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

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

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

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

To define a driver on a non-Windows machine:

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

> **Note:**
>
> If you are using a hidden copy of the `odbcinst.ini` file, you can remove the period (.) from the start of the file name to make the file visible while you are editing it.

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

```
[ODBC Drivers]
Teradata Presto ODBC Driver=Installed
[Teradata Presto ODBC Driver]
Description=Teradata Presto ODBC Driver
Driver=/Library/teradata/prestoodbc/lib/libprestoodbc_sbu.dylib
```

For example:

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

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

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

      For example, on a macOS machine:

      ```
      Driver=/Library/teradata/prestoodbc/lib/libprestoodbc_sbu.dylib
      ```

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

      ```
      Driver=/opt/teradata/prestoodbc/lib/32/libprestoodbc_sb32.so
      ```

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

      For example:

      ```
      Description=Teradata Presto ODBC Driver
      ```

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

> **Note:**
>
> If you are storing this file in its default location in the home directory, then prefix the file name with a period (.) so that the file becomes hidden. If you are storing this file in another location, then save it as a non-hidden file (without the prefix), and make sure that the ODBCINSTINI or ODBCSYSINI environment variable specifies the location. For more information, see Specifying the Locations of the Driver Configuration Files on page 24.

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

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

You can now connect to your data store by providing your application with a connection string where the `Driver` property is set to the driver name specified in the `odbcinst.ini` file, and all the other necessary connection properties are also set. For more information, see "DSN-less Connection String Examples" in Using a Connection String on page 36.

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

- Configuring Authentication on a Non-Windows Machine on page 30
- Configuring SSL Verification on a Non-Windows Machine on page 32

For detailed information about all the connection properties that the driver supports, see Driver Configuration Options on page 41.

Configuring Authentication on a Non-Windows Machine

Some Presto data stores require authentication. You can configure the Teradata Presto ODBC Driver to provide your credentials and authenticate the connection to the database using one of the following methods:

- Using Kerberos Authentication on page 30
- Using LDAP Authentication on page 31

Note:
If Kerberos or LDAP authentication is enabled, then SSL is automatically enabled.

Using Kerberos Authentication

You can configure the driver to use the Kerberos protocol to authenticate the connection. You can set the connection properties in a connection string or in a DSN (in the `odbc.ini` file). Settings in the connection string take precedence over settings in the DSN.

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, a 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 on a non-Windows machine:

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 your `odbc.ini` configuration file or connection string, set the `AuthenticationType` property to `Kerberos Authentication`.

3. Configure the SSL settings as described in Configuring SSL Verification on a Non-Windows Machine on page 32.

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

Using LDAP Authentication

You can configure the driver to use the LDAP protocol to authenticate the connection. You can set the connection properties in a connection string or in a DSN (in the `odbc.ini` file). Settings in the connection string take precedence over settings in the DSN.

To configure LDAP authentication on a non-Windows machine:

1. Set the `AuthenticationType` property to `LDAP Authentication`.

2. Set the `UID` property to an appropriate user name for accessing the data store.

3. Set the `PWD` property to the password corresponding to the user name that you specified above.

4. Configure the SSL settings as described in Configuring SSL Verification on a Non-Windows Machine on page 32.

You can now use the driver to authenticate through LDAP and connect to your Presto server.

Configuring SSL Verification on a Non-Windows Machine

If you are connecting to a Presto server that has Secure Sockets Layer (SSL) enabled, you can configure the driver to connect to an SSL-enabled socket. When connecting to a server over SSL, the driver can be configured to verify the identity of the server.

> **Note:**
>
> If either Kerberos or LDAP authentication are enabled, the driver automatically uses SSL to communicate with the Presto server.

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

To configure SSL verification on a non-Windows machine:

1. To enable SSL connections, set the `SSL` attribute to `1`.

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

3. To allow the common name of a CA-issued SSL certificate to not match the host name of the Presto server, set the `AllowHostNameCNMismatch` attribute to `1`.

4. Choose one:

   - To configure the driver to load SSL certificates from a specific `.pem` file when verifying the server, set the `TrustedCerts` attribute to the full path of the `.pem` file.

   - Or, to use the trusted CA certificates `.pem` file that is installed with the driver, do not specify a value for the `TrustedCerts` attribute.

Configuring Logging Options on a Non-Windows Machine

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

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

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

To enable logging on a non-Windows machine:

1. Open the `teradata.prestoodbc.ini` configuration file in a text editor.

2. To specify the level of information to include in log files, set the `LogLevel` property to one of the following numbers:

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

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

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

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

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

6. Save the `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 on a non-Windows machine:

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.

4. Restart your ODBC application to make sure that the new settings take effect.

Testing the Connection on a Non-Windows Machine

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

Using the iODBC Driver Manager

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

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

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

To test your connection using the iODBC driver manager:

1. Run **iodbctest** or **iodbctestw**.

2. Optionally, if you do not remember the DSN, then type a question mark (`?`) to see a list of available DSNs.

3. Type the connection string for connecting to your data store, and then press ENTER. For more information, see Using a Connection String on page 36.

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:

1. 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 41.

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.

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

```
Driver=Teradata Presto ODBC Driver;Catalog=[CatalogName];
Host=[Server];Port=[PortNumber];
```

For example:

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

Features

For more information on the features of the Teradata Presto ODBC Driver, see the following:

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. If Auto Populate Parameter Metadata is selected or the `AutoIPD` option is set to `1`, the driver automatically populates the metadata for parameters.

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	REAL
	Note: Only supported in Presto 0.152t and later.
BIGINT	ROW
BOOLEAN	SMALLINT
CHAR(x)	TIME
DATE	TIME WITH TIME ZONE
DECIMAL	TIMESTAMP
DOUBLE	TIMESTAMP WITH TIME ZONE
FLOAT **Note:** Deprecated in Presto 0.152t and later.	TINYINT
INTEGER	VARBINARY
INTERVAL DAY TO SECOND	VARCHAR (fixed length)
INTERVAL YEAR TO MONTH	VARCHAR (variable length)
JSON	VARCHAR(X)
MAP	

Security and Authentication

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

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

The driver provides a mechanism that enables you to authenticate your connection using the Kerberos protocol or the LDAP protocol. For detailed configuration instructions, see Configuring Authentication on Windows on page 12 or Configuring Authentication on a Non-Windows Machine on page 30.

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

It is recommended that you enable SSL whenever you connect to a server that is configured to support it. SSL encryption protects data and credentials when they are transferred over the network, and provides stronger security than authentication alone. For detailed configuration instructions, see Configuring SSL Verification on Windows on page 13 or Configuring SSL Verification on a Non-Windows Machine on page 32.

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 macOS 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 macOS computer:

- Allow Common Name Host Name Mismatch on page 42
- Allow Self-Signed Server Certificate on page 42
- Authentication Type on page 43
- Auto Populate Parameter Metadata on page 43
- Catalog on page 44
- Connection Test on page 44
- Enable SSL on page 45
- Host on page 45
- Log Level on page 45
- Log Path on page 46
- Max File Size on page 47

- Max Number Files on page 47
- Password on page 48
- Port on page 48
- Proxy on page 49
- Proxy Password on page 49
- Proxy Port on page 49
- Proxy U on page 50
- Save Password (Encrypted) on page 50
- Schema on page 50
- Server Version on page 51
- Time Zone ID on page 51
- Trusted Certificates on page 52
- Use Proxy Server on page 52
- User on page 53

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 macOS computer, use the key names provided.

Allow Common Name Host Name Mismatch

Key Name	Default Value	Required
AllowHostNameCNMismatch	Clear (0)	No

Description

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

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

- Disabled (0): The CA-issued SSL certificate name must match the host name of the Prestoserver.

Note:

This setting is applicable only when SSL is enabled.

Allow Self-Signed Server Certificate

Key Name	Default Value	Required
AllowSelfSignedServerCert	Clear (0)	No

Description

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

- Enabled (1): The driver authenticates the Presto server even if the server is using a self-signed certificate.

- Disabled (0): The driver does not allow self-signed certificates from the server.

Note:

This setting is applicable only when SSL is enabled.

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 Authentication`: The driver uses Kerberos to authenticate the connection. For more information about Kerberos authentication on Windows, see the Windows Kerberos documentation: https://msdn.microsoft.com/en-us/library/windows/desktop/aa378747(v=vs.85).aspx. For more information about Kerberos authentication on macOS or Linux, see the MIT Kerberos Documentation: http://web.mit.edu/kerberos/krb5-latest/doc/.

- `LDAP Authentication`: The driver uses LDAP to authenticate the connection.

> **Note:**
>
> If either Kerberos Authentication or LDAP Authentication are specified, the driver automatically uses SSL to communicate with the Presto server.

Auto Populate Parameter Metadata

Key Name	Default Value	Required
AutoIPD	Enabled (1)	No

Description

This option specifies whether the driver automatically populates the parameter metadata for parameterized SQL statements.

The Presto server does not necessarily provide parameter metadata for every parameter in a parameterized SQL statement. When the server does not provide parameter metadata, the driver defines the parameter data type as SQL_VARCHAR.

Automatically populating parameter metadata may cause certain ODBC applications to not function correctly. In that case, this option should be disabled.

- Enabled (`1`): The driver automatically populates the metadata for parameters.

- Disabled (`0`): The driver does not automatically populate the metadata for parameters.

Catalog

Key Name	Default Value	Required
Catalog	None	Yes

Description

The current catalog context for all requests against the server.

Connection Test

Key Name	Default Value	Required
ConnectionTest	Enabled (1)	No

Description

This option specifies whether the driver should automatically attempt to test the connection by contacting the server while establishing the connection.

- Enabled (`1`): The driver automatically tests the connection while establishing the connection.

- Disabled (`0`): The driver does not automatically test the connection.

> **Note:**
>
> - Disabling this option may improve driver performance.
>
> - If this option is disabled, you should specify the version of the Presto server in the Server Version or `ServerVersion` configuration option (see Server Version on page 51).

Enable SSL

Key Name	Default Value	Required
SSL	Clear (0)	No

Description

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

- Enabled (1): The client communicates with the Presto server using SSL.

- Disabled (0): SSL is disabled.

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

Note:

If either Kerberos Authentication or LDAP Authentication are specified, the driver automatically uses SSL to communicate with the Presto server.

Host

Key Name	Default Value	Required
Host	None	Yes

Description

The IP address or host name of the Presto server.

Log Level

Key Name	Default Value	Required
LogLevel	OFF (0)	No

Description

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

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

Set the property to one of the following values:

- OFF (`0`): Disable all logging.

- FATAL (`1`): Logs severe error events that lead the driver to abort.

- ERROR (`2`): Logs error events that might allow the driver to continue running.

- WARNING (`3`): Logs events that might result in an error if action is not taken.

- INFO (`4`): Logs general information that describes the progress of the driver.

- DEBUG (`5`): Logs detailed information that is useful for debugging the driver.

- TRACE (`6`): Logs all driver activity.

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

Log Path

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

Description

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

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

Max File Size

Key Name	Default Value	Required
LogFileSize	20	No

Description

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

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

Max Number Files

Key Name	Default Value	Required
LogFileCount	50	No

Description

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

> **Important:**
>
> This option is not supported in connection strings. To configure logging for the Windows driver, you must use the Logging Options dialog box. To configure logging for a non-Windows driver, you must use the `teradata.prestoodbc.ini` file.

Password

Key Name	Default Value	Required
PWD	None	No

Description

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

> **Note:**
>
> This option is only available when the Authentication Type is set to LDAP Authentication.

Port

Key Name	Default Value	Required
Port	8080	Yes

Description

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

Proxy

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

Description

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

Proxy Password

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

Description

The password that you use to access the proxy server.

Proxy Port

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

Description

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

Proxy U

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

Description

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

Save Password (Encrypted)

Key Name	Default Value	Required
N/A	Clear	No

Description

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

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

This option is available only in the Windows driver. It appears in the Teradata Presto ODBC Driver DSN Setup dialog box and the Proxy Options dialog box.

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

Schema

Key Name	Default Value	Required
Schema	None	No

Description

The current schema context for all requests against the server.

> **Note:**
>
> This option is only used when the `Catalog` option is specified and the value is not an empty string.

Server Version

Key Name	Default Value	Required
ServerVersion	None	No

Description

This option specifies the version of the Presto server that the driver connects to, for example, `0.148-t`. This value is used when the driver cannot automatically detect the server version.

> **Note:**
>
> If Connection Test is cleared or `ConnectionTest` is set to `0`, this option should be set to the version of the Presto server that is being used.

Time Zone ID

Key Name	Default Value	Required
TimeZoneID	System time zone	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 https://en.wikipedia.org/wiki/List_of_tz_database_time_zones.

Trusted Certificates

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

Description

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

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

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

Use Proxy Server

Key Name	Default Value	Required
UseProxy	Clear (0)	No

Description

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

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

- Disabled (0): The driver connects directly to the Presto server.

User

Key Name	Default Value	Required
UID	PrestoODBC_Driver	No

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 macOS or Linux.

Driver

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

Description

On Windows, the name of the installed driver (`Teradata Presto ODBC Driver`).

On other platforms, the name of the installed driver as specified in `odbcinst.ini,` or the absolute path of the 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, macOS, 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) 1998-2016 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 msinttypes r29
Copyright (c) 2006-2013 Alexander Chemeris
All rights reserved.

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

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

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

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

THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS AND 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.

Open Source Software Licensed Under the JSON License:

json.org
Copyright (c) 2002 JSON.org
All Rights Reserved.

JSON_checker
Copyright (c) 2002 JSON.org
All Rights Reserved.

Terms of the JSON 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 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

All rights reserved.

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

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

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

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

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

This is the standard "new" BSD license:

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

zlib License

Copyright (C) 1995-2013 Jean-loup Gailly and Mark Adler

This software is provided 'as-is', without any express or implied warranty. In no event will the authors be held liable for any damages arising from the use of this software.

Permission is granted to anyone to use this software for any purpose, including commercial applications, and to alter it and redistribute it freely, subject to the following restrictions:

1. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an 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 Gai   Mark Adler

jloup@gzip.or   madler@alumni.cal