

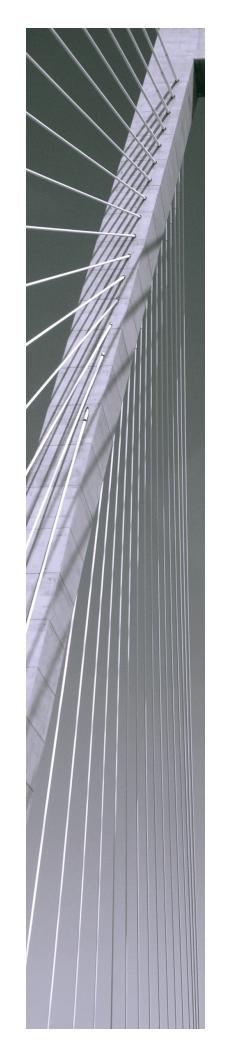
Simba Athena ODBC Driver with SQL Connector

Installation and Configuration Guide

Simba Technologies Inc.

Version 1.1.0 (Preview)

December 18, 2019



Copyright © 2019 Magnitude Software, Inc. All rights reserved.

No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording, or otherwise, without prior written permission from Magnitude.

The information in this document is subject to change without notice. Magnitude strives to keep this information accurate but does not warrant that this document is error-free.

Any Magnitude product described herein is licensed exclusively subject to the conditions set forth in your Magnitude license agreement.

Simba, the Simba logo, SimbaEngine, and Simba Technologies are registered trademarks of Simba Technologies Inc. in Canada, the United States and/or other countries. All other trademarks and/or servicemarks are the property of their respective owners.

All other company and product names mentioned herein are used for identification purposes only and may be trademarks or registered trademarks of their respective owners.

Information about the third-party products is contained in a third-party-licenses.txt file that is packaged with the software.

Contact Us

Simba Technologies Inc. 938 West 8th Avenue Vancouver, BC Canada V5Z 1E5

Tel: +1 (604) 633-0008

Fax: +1 (604) 633-0004

www.simba.com

About This Guide

Purpose

The Simba Athena ODBC Driver with SQL Connector Installation and Configuration Guide explains how to install and configure the Simba Athena 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 Simba Athena ODBC Driver, as well as administrators and developers integrating the driver.

Knowledge Prerequisites

To use the Simba Athena ODBC Driver, the following knowledge is helpful:

- Familiarity with the platform on which you are using the Simba Athena ODBC Driver
- Ability to use the data source to which the Simba Athena 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.



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 Simba Athena ODBC Driver	7
About Amazon Athena	7
About the Driver	7
Windows Driver	9
Windows System Requirements	9
Installing the Driver on Windows	9
Creating a Data Source Name on Windows	10
Configuring Authentication on Windows	12
Configuring Advanced Options on Windows	17
Configuring Proxy Connections on Windows	18
Exporting a Data Source Name on Windows	19
Importing a Data Source Name on Windows	19
Configuring Logging Options on Windows	20
Verifying the Driver Version Number on Windows	22
macOS Driver	23
macOS System Requirements	
Installing the Driver on macOS	
Verifying the Driver Version Number on macOS	
Linux Driver	25
Linux System Requirements	
Installing the Driver Using the RPM File	
Verifying the Driver Version Number on Linux	
Configuring the ODBC Driver Manager on Non-Windows Machines	20
Specifying ODBC Driver Managers on Non-Windows Machines	
Specifying the Locations of the Driver Configuration Files	
Specifying the Locations of the Driver Configuration Files	
Configuring ODBC Connections on a Non-Windows Machine	31
Creating a Data Source Name on a Non-Windows Machine	31
Configuring a DSN-less Connection on a Non-Windows Machine	34
Configuring Authentication on Non-Windows Machines	36
Configuring Proxy Connections on Non-Windows Machines	40
Configuring Query Result Encryption on a Non-Windows Machine	41
Configuring Logging Options on a Non-Windows Machine	42
Testing the Connection on a Non-Windows Machine	44

Installation and Configuration Guide

Jsing a Connection String	
DSN Connection String Example	46
DSN-less Connection String Examples	46
Example: Using Workgroups	51
Features	52
Catalog and Schema Support	52
File Formats	52
Data Types	52
Result Set Streaming Support	56
Query Execution Polling	56
Security and Authentication	57
Driver Configuration Options	58
Configuration Options Appearing in the User Interface	58
Configuration Options Having Only Key Names	77
Third-Party Trademarks	80

About the Simba Athena ODBC Driver

About Amazon Athena

Amazon Athena is a serverless interactive query service capable of querying data from Amazon Simple Storage Service (S3) using SQL. It is designed for short, interactive queries that are useful for data exploration. Athena enables you to run ad-hoc queries and quickly analyze data that is stored in S3 without ETL processes. Query results are stored in an S3 bucket and made available for analysis in BI tools.

The data formats that Athena supports include CSV, JSON, Parquet, Avro, and ORC. Unlike traditional RDBMS or SQL-on-Hadoop solutions that require centralized schema definitions, Athena can query self-describing data as well as complex or multi-structured data that is commonly seen in big data systems. Moreover, Athena does not require a fully structured schema and can support semi-structured or nested data types such as JSON.

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

Note:

- Access from Athena to your S3 data store is configured through Amazon Web Services (AWS). For information about enabling Athena to access S3 data stores, see the Amazon Athena documentation: http://docs.aws.amazon.com/athena/latest/ug/what-is.html.
- When using Athena, you are charged for each query that you run. The
 amount that you are charged is based on the amount of data scanned by the
 query. For more information, see *Amazon Athena Pricing*: https://aws.amazon.com/athena/pricing/.

About the Driver

The Simba Athena ODBC Driver enables organizations to connect their BI tools to the Amazon Athena query service, enabling Business Intelligence, analytics, and reporting on the data that Athena returns from Amazon S3 databases. The driver retrieves catalog metadata from the AthenaAPI.

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 of the most established and widely supported APIs for connecting to and working with databases. At the heart of the technology is the ODBC driver, which connects an application to the database. For more information about ODBC, see *Data Access Standards* on the Simba Technologies

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

The Simba Athena ODBC Driver is available for Microsoft® Windows®, Linux, and macOS platforms.

The Simba Athena ODBC Driver with SQL Connector Installation and Configuration Guide is suitable for users who are looking to access data returned by the Athena query service from their desktop environment. Application developers may also find the information helpful. Refer to your application for details on connecting via ODBC.



For information about how to use the driver in various BI tools, see the *Simba ODBC Drivers Quick Start Guide for Windows*: http://cdn.simba.com/docs/ODBC_QuickstartGuide/content/quick_start/intro.htm.

Windows Driver

Windows System Requirements

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

- · Administrator rights on your machine.
- A machine that meets the following system requirements:
 - One of the following operating systems:
 - Windows 10, 8.1, or 7 SP1
 - Windows Server 2016, 2012, or 2008 R2 SP1
 - 150 MB of available disk space

Before the driver can be used, the Visual C++ Redistributable for Visual Studio 2013 with the same bitness as the driver must also be installed. If you obtained the driver from the Simba website, then your installation of the driver automatically includes this dependency. Otherwise, you must install the redistributable manually. You can download the installation packages for the redistributable at https://www.microsoft.com/en-ca/download/details.aspx?id=40784.

Installing the Driver on Windows

If you did not obtain this driver from the Simba website, you might need to follow a different installation procedure. For more information, see the *Simba OEM ODBC Drivers Installation Guide*.

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

- Simba Athena 1.0 32-bit.msi for 32-bit applications
- Simba Athena 1.0 64-bit.msi for 64-bit applications

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

To install the Simba Athena ODBC Driver on Windows:

- 1. Depending on the bitness of your client application, double-click to run **Simba**Athena 1.0 32-bit.msi or Simba Athena 1.0 64-bit.msi.
- 2. Click Next.

- 3. Select the check box to accept the terms of the License Agreement if you agree, and then click **Next**.
- 4. To change the installation location, click **Change**, then browse to the desired folder, and then click **OK**. To accept the installation location, click **Next**.
- 5. Click Install.
- 6. When the installation completes, click **Finish**.
- 7. If you received a license file through email, then copy the license file into the \lib subfolder of the installation folder you selected above. You must have Administrator privileges when changing the contents of this folder.

Creating a Data Source Name on Windows

Typically, after installing the Simba Athena 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 46.

To create a Data Source Name on Windows:

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



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 Athena.

- 2. In the ODBC Data Source Administrator, click the **Drivers** tab, and then scroll down as needed to confirm that the Simba Athena 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.



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.

- In the Create New Data Source dialog box, select Simba Athena ODBC Driver and then click Finish. The Simba Athena 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 **AWS Region** field, type the AWS region of the Athena instance that you want to connect to.

Note:

For a list of valid regions, see the "Athena" section in the *AWS Regions and Endpoints* documentation:

http://docs.aws.amazon.com/general/latest/gr/rande.html#athena.

- 9. Optionally, in the **Catalog** field, type the name of the Athena catalog you want the driver to query. If Ithis filed is left empty, queries will be applied to AwsDataCatalog.
- 10. In the **Schema** field, type the name of the database schema to use when a schema is not explicitly specified in a query. You can still issue queries on other schemas by explicitly specifying the schema in the query.
- 11. Optionally, in the **Workgroup** field, type the name of the workgroup to use when signing in to Athena.
- 12. In the **S3 Output Location** field, type the path of the Amazon S3 location where you want to store query results, prefixed by s3://.

For example, to store results in a folder named "test-folder-1" inside an S3 bucket named "query-results-bucket", you would type **s3://query-results-bucket/test-folder-1** in this field.

- 13. To configure encryption for your query results, do the following:
 - a. From the **Encryption Options** drop-down list, select the encryption protocol that you want to use:

Option Name	Description
NOT_SET	The driver does not encrypt the data.
SSE_S3	The driver uses server-side encryption with an Amazon S3-managed key.
SSE_KMS	The driver uses server-side encryption with an AWS KMS-managed key.

Option Name	Description
CSE_KMS	The driver uses client-side encryption with an AWS KMS-managed key.

For detailed information about these encryption options, see "Configuring Encryption Options" in the *Amazon Athena User Guide*:

http://docs.aws.amazon.com/athena/latest/ug/encryption.html.

- b. If you selected SSE_KMS or CSE_KMS in the previous step, then in the **KMS Key** field, type the KMS customer key to use for encrypting data.
- 14. To configure authentication, click **Authentication Options**. For more information, see Configuring Authentication on Windows on page 12.
- 15. To configure advanced options, click **Advanced Options**. For more information, see Configuring Advanced Options on Windows on page 17.
- 16. To configure proxy connections, click **Proxy Options**. For more information, see Configuring Proxy Connections on Windows on page 18.
- 17. To configure logging behavior for the driver, click **Logging Options**. For more information, see Configuring Logging Options on Windows on page 20.
- 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 Simba Athena ODBC Driver DSN Setup dialog box are correct. Contact your AWS account administrator as needed.

- 19. To save your settings and close the Simba Athena ODBC Driver DSN Setup dialog box, click **OK**.
- 20. To close the ODBC Data Source Administrator, click **OK**.

Configuring Authentication on Windows

To access data from Athena, you must authenticate the connection. You can configure the Simba Athena ODBC Driver to provide your credentials and authenticate the connection using one of the following methods:

- Using the Default Credentials Provider Chain on Windows on page 13
- Using IAM Credentials on Windows on page 13
- Using an IAM Profile on Windows on page 14
- Using an Instance Profile on Windows on page 14

- Using the Active Directory Federation Services (AD FS) Credentials Provider on Windows on page 15
- Using the Okta Service on page 16

Using the Default Credentials Provider Chain on Windows

You can configure the driver to authenticate the connection using credentials that are stored in one of the locations in the default credentials provider chain. The driver looks for a valid access key and secret key pair by checking the following locations, in the following order:

- The AWS credentials file stored in the %USERPROFILE%.awscredentials directory.
- 2. The AWS_ACCESS_KEY_ID and AWS_SECRET_ACCESS_KEY system environment variables.
- 3. The instance profile from the Amazon EC2 Instance Metadata Service.

For detailed information about configuring default credentials, see "Providing AWS Credentials" in the *AWS SDK for C++ Developer Guide*: http://docs.aws.amazon.com/sdk-for-cpp/v1/developer-guide/credentials.html.

To configure authentication using the default credentials provider chain on Windows:

- 1. To access authentication options, open the ODBC Data Source Administrator where you created the DSN, then select the DSN, then click **Configure**, and then click **Authentication Options**.
- 2. From the Authentication Type drop-down list, select Default Credentials.
- 3. To save your settings and close the Authentication Options dialog box, click **OK**.

Using IAM Credentials on Windows

You can configure the driver to authenticate the connection using an access key and a secret key that is specified directly in the connection information.

If you are using temporary credentials, which are only valid for a limited amount of time, then you must also provide a session token. For more information, see "Temporary Security Credentials" in the AWS Identity and Access Management User Guide: http://docs.aws.amazon.com/IAM/latest/UserGuide/id_credentials_temp.html.

To configure authentication using IAM credentials on Windows:

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

- 2. From the Authentication Type drop-down list, select IAM Credentials.
- 3. In the **User** field, type the access key provided by your AWS account.
- 4. In the **Password** field, type the secret key provided by your AWS account.
- 5. To encrypt your credentials, click **Password Options** and then select one of the following:
 - If the credentials are used only by the current Windows user, select Current User Only.
 - Or, if the credentials are used by all users on the current Windows machine, select All Users Of This Machine.
- 6. If you are using temporary credentials, in the **Session Token** field, type the session token generated by the AWS Security Token Service.
- 7. To save your settings and close the Authentication Options dialog box, click **OK**.

Using an IAM Profile on Windows

You can configure the driver to authenticate the connection using credentials that are associated with an IAM profile in a credentials file.

By default, the driver uses the credentials associated with a profile named <code>default</code> in the credentials file found in the <code>%USERPROFILE%.awscredentials</code> directory. To use a different profile, specify the profile name in your connection settings. To use a different credentials file, set the AWS_SHARED_CREDENTIALS_FILE system environment variable to the full path of your credentials file.

For information about the format of a credentials file, see the "AWS Credentials File Format" section from the "Working with AWS Credentials" page in the *AWS SDK for Java Developer Guide*: http://docs.aws.amazon.com/sdk-for-java/v1/developer-guide/credentials.html.

To configure authentication using an IAM profile on Windows:

- 1. To access authentication options, open the ODBC Data Source Administrator where you created the DSN, then select the DSN, then click **Configure**, and then click **Authentication Options**.
- 2. From the **Authentication Type** drop-down list, select **IAM Profile**.
- 3. In the AWS Profile field, type the name of the profile to use.
- 4. To save your settings and close the Authentication Options dialog box, click **OK**.

Using an Instance Profile on Windows

You can configure the driver to authenticate the connection using credentials that have been loaded from the Amazon EC2 Instance Metadata Service into an instance profile.

Instance profiles contain authorization information such as roles, permissions, and credentials, and are automatically created by Amazon EC2 for each IAM role that is defined for an EC2 instance. For more information, see "IAM Roles for Amazon EC2" in the *Amazon Elastic Compute Cloud User Guide for Windows Instances*: http://docs.aws.amazon.com/AWSEC2/latest/WindowsGuide/iam-roles-for-amazon-ec2.html.

To configure authentication using an instance profile on Windows:

- 1. To access authentication options, open the ODBC Data Source Administrator where you created the DSN, then select the DSN, then click **Configure**, and then click **Authentication Options**.
- 2. From the Authentication Type drop-down list, select Instance Profile.
- 3. To save your settings and close the Authentication Options dialog box, click **OK**.

Using the Active Directory Federation Services (AD FS) Credentials Provider on Windows

You can configure the driver to authenticate the connection using credentials obtained from the AD FS credentials provider. To do this, you must specify information about the AD FS service, such as the host and port of the server where the service is hosted.

To configure authentication using AD FS on Windows:

- 1. To access authentication options, open the ODBC Data Source Administrator where you created the DSN, then select the DSN, then click **Configure**, and then click **Authentication Options**.
- From the Authentication Type drop-down list, select ADFS.
- 3. Optionally, to specify your credentials for accessing the AD FS server, do the following. If you do not specify any credentials, the driver attempts to authenticate to the AD FS server by using your Windows account credentials over the NTLM protocol.
 - a. In the **User** field, type the user name that you use to access the AD FS server. You can include the domain name using the format [DomainName] \ [UserName].
 - b. In the **Password** field, type the password corresponding to the user name that you provided in the previous step.
 - c. To encrypt your credentials, click **Password Options** and then select one of the following:
 - If the credentials are used only by the current Windows user, select **Current User Only**.
 - Or, if the credentials are used by all users on the current Windows machine, select **All Users Of This Machine**.

- 4. To specify AD FS service information, do the following:
 - a. In the IdP Host field, type the host name of the AD FS service.

! Important:

The host name cannot include any slashes (/).

b. Optionally, in the **IdP Port** field, type the number of the port that the AD FS service host uses to listen for requests.

Note:

The exact port number that you need to specify may differ depending on the AD FS server configuration. If you are not sure which port to specify, contact your system administrator.

- 5. Optionally, in the **Preferred Role** field, type the Amazon Resource Name (ARN) of the role that you want to assume when authenticated through AD FS.
- 6. If the AD FS service must be accessed through an HTTP proxy, select the **Use HTTP Proxy For IdP Host** check box. For information about configuring the
 proxy connection, see Configuring Proxy Connections on Windows on page 18.
- 7. Optionally, if you do not want the driver to verify the AD FS server certificate, select the **SSL Insecure** check box.
- 8. To save your settings and close the Authentication Options dialog box, click **OK**.

Using the Okta Service

You can configure the driver to authenticate the connection using credentials obtained from the Okta credentials provider. To do this, you must specify information about the Okta service, such as the host name of the Okta service and the Okta application ID.

To configure authentication using Okta on Windows:

- To access the authentication options, open the ODBC Data Source Administrator where you created the DSN, select the DSN, and then click Configure.
- 2. In the Authentication area, click the **Authentication Type** drop-down list, select **Okta**.
- 3. In the **User** field, type the user name associated with your Okta account.
- 4. In the **Password** field, type the password associated with your Okta user name.
- 5. In the **IdP Host** field, type the host name of the Okta service.
- 6. In the **Okta App ID** field, type the Okta-supplied ID associated with your Athena application.
- 7. Optionally, in the **Okta App Name** field, type the name of your Okta application.

- 8. Optionally, in the **Preferred Role** field, type the Amazon Resource Name (ARN) of the role that you want to assume when authenticated through Okta.
- 9. To save your settings and close the dialog box, 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:

- To access advanced options, open the ODBC Data Source Administrator where you created the DSN, then select the DSN, then click Configure, and then click Advanced Options.
- 2. In the **String Column Length** field, type the maximum data length for STRING columns.
- 3. In the **Binary Column Length** field, type the maximum data length for BINARY columns
- 4. In the **Max Complex Type Column Length** field, type the maximum data length for complex data types the driver casts to SQL VARCHAR.
- 5. In the **Max Catalog Name Length** field, type the maximum number of characters that catalog names contain.
- 6. In the **Max Schema Name Length** field, type the maximum number of characters that schema names contain.
- 7. In the **Max Table Name Length** field, type the maximum number of characters that table names contain.
- 8. In the **Max Column Name Length** field, type the maximum number of characters that column names contain.



For the options described in steps 4 to 8, you can specify 0 to indicate that there is no maximum length or that the length is unknown.

- 9. In the **Rows To Fetch Per Block** field, type the maximum number of rows to fetch per stream if using the result set streaming API.
- 10. In the **Max Execution Polling Interval** field, type the maximum time, in milliseconds, to wait between attempts when polling the server for the query execution result.
- 11. In the **Min Execution Polling Interval** field, type the minimum value of the polling interval in milliseconds.
- 12. In the **Execution Polling Interval Multiplier** field, type the multiplier by which the driver increases the amount of time between polls.

- To enable the driver to return SQL_WVARCHAR instead of SQL_VARCHAR for ARRAY, MAP, STRING, STRUCT, and VARCHAR columns, select the Use SQL Unicode Types check box.
- 14. To enable the use of the AWS result set streaming API, select the **Use Result Set Streaming** check box.
- 15. To save your settings and close the Advanced Options dialog box, click **OK**.

Configuring Proxy Connections on Windows

You can configure the driver to connect through a proxy server instead of connecting directly to the Athena service.

! Important:

If you are connecting to Athena through a proxy server, make sure that the proxy server does not block port 444. The result set streaming API uses port 444 on the Athena server for outbound communications. For more information, see Use Result Set Streaming on page 75.

To configure a proxy connection on Windows:

- To access proxy 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. To enable proxy connections, select the **Use Proxy** check box.
- 3. In the **Proxy Host** field, type the IP address or host name of your 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 your password for accessing the proxy server.
- 7. To encrypt your credentials, click **Password Options** and then select one of the following:
 - If the credentials are used only by the current Windows user, select Current User Only.
 - Or, if the credentials are used by all users on the current Windows machine, select All Users Of This Machine.
- 8. To save your settings and close the HTTP Proxy Options dialog box, click **OK**.

If the proxy server is configured to intercept SSL-encrypted connections, then in addition to specifying the proxy server information described above, you must also import the proxy server's root certificate into the Windows trust store.

To import the proxy server's root certificate to the Windows trust store:

1. Export the proxy server's root certificate file. You can do this using OpenSSL.

For example, the following command exports the root certificate, originally a .pem file, to a .crt file:

```
openssl x509 -outform der -in clientPublicKey.pem -out clientPublicKey.crt
```

For more information, see "OpenSSL Commands" in the OpenSSL documentation: https://www.openssl.org/docs/manmaster/man1/.

Import the certificate into the Window trust store. For detailed instructions, see
"Installing a Certificate in the Trusted Root Certification Authorities Store" in the
Microsoft Windows documentation: https://docs.microsoft.com/enus/dotnet/framework/wcf/feature-details/how-to-create-temporary-certificates-foruse-during-development#to-install-a-self-signed-certificate-in-the-trusted-rootcertification-authorities.

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

To enable driver logging on Windows:

- 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.

Logging Level	Description
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. You can type the path into the field, or click **Browse** and then browse to select the folder.
- 4. In the **Max Number Files** field, type 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.

5. In the **Max File Size** field, type 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.

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

The Simba Athena ODBC Driver produces the following log files at the location you specify in the Log Path field:

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

If you enable the UseLogPrefix connection property, the driver prefixes the log file name with the user name associated with the connection and the process ID of the application through which the connection is made. For more information, see UseLogPrefix on page 78.

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



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 Athena.

2. Click the **Drivers** tab and then find the Simba Athena 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

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

- macOS version 10.12, 10.13, or 10.14
- 150MB of available disk space
- One of the following ODBC driver managers installed:
 - iODBC 3.52.9 or later
 - unixODBC 2.2.14 or later

Installing the Driver on macOS

If you did not obtain this driver from the Simba website, you might need to follow a different installation procedure. For more information, see the *Simba OEM ODBC Drivers Installation Guide*.

The Simba Athena ODBC Driver is available for macOS as a .dmg file named Simba Athena 1.0.dmg. The driver supports both 32- and 64-bit client applications.

To install the Simba Athena ODBC Driver on macOS:

- 1. Double-click **Simba Athena 1.0.dmg** to mount the disk image.
- 2. Double-click Simba Athena 1.0.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**.



By default, the driver files are installed in the /Library/simba/athenaodbc directory.

- 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 through email, then copy the license file into the /lib subfolder in the driver installation directory. You must have root privileges when changing the contents of this folder.

For example, if you installed the driver to the default location, you would copy the license file into the /Library/simba/athenaodbc/lib folder.

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 28.

Verifying the Driver Version Number on macOS

If you need to verify the version of the Simba Athena 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:

At the Terminal, run the following command:

```
pkgutil --info com.simba.athenaodbc
```

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

Linux Driver

Linux System Requirements

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

- One of the following distributions:
 - Red Hat® Enterprise Linux® (RHEL) 6 or 7
 - CentOS 6 or 7
 - SUSE Linux Enterprise Server (SLES) 11 or 12
 - Debian 8 or 9
 - Ubuntu 14.04, 16.04, or 18.04
- 150 MB of available disk space
- One of the following ODBC driver managers installed:
 - iODBC 3.52.9 or later
 - unixODBC 2.2.14 or later

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

If you are using the RPM file to install the driver on Debian or Ubuntu, you must also have the alien utility installed. The alien utility is available on SourceForge: https://sourceforge.net/projects/alien-pkg-convert/.

Installing the Driver Using the RPM File

If you did not obtain this driver from the Simba website, you might need to follow a different installation procedure. For more information, see the *Simba OEM ODBC Drivers Installation Guide*.

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

- simbaathena-[Version]-[Release].i686.rpm for the 32-bit driver
- simbaathena-[Version]-[Release].x86_64.rpm for the 64-bit driver

The placeholders in the file names are defined as follows:

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

You can install both the 32-bit and 64-bit versions of the driver on the same machine.

To install the Simba Athena ODBC Driver using the RPM File:

- 1. Log in as the root user.
- 2. If you are installing the driver on a Debian or Ubuntu machine, download and install the alien utility:
 - a. Download the package from SourceForge: https://sourceforge.net/projects/alien-pkg-convert/.
 - b. From the command line, run the following command:

```
sudo apt-get install alien
```

- 3. Navigate to the folder containing the RPM package for the driver.
- 4. 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]
```

• Or, if you are using Debian or Ubuntu, run the following command:

```
alien -i [RPMFileName]
```

The Simba Athena ODBC Driver files are installed in the

/opt/simba/athenaodbc directory.

5. If you received a license file through email, then copy the license file into the /opt/simba/athenaodbc/lib/32 or /opt/simba/athenaodbc/lib/64 folder, depending on the version of the driver that you installed.

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 28.

Verifying the Driver Version Number on Linux

If you need to verify the version of the Simba Athena 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. Alternatively, you can search the driver's binary file for version number information.

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

Depending on your package manager, at the command prompt, run one of the following commands:

```
    yum list | grep SimbaAthenaODBC
    rpm -qa | grep SimbaAthenaODBC
    dpkg -l | grep simbaathenaodbc
```

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

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

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

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 Simba Athena ODBC Driver, do the following:

- Set the library path environment variable to make sure that your machine uses the correct ODBC driver manager. For more information, see Specifying ODBC Driver Managers on Non-Windows Machines on page 28.
- If the driver configuration files are not stored in the default locations expected by the ODBC driver manager, then set environment variables to make sure that the driver manager locates and uses those files. For more information, see Specifying the Locations of the Driver Configuration Files on page 29.

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

Specifying ODBC Driver Managers on Non-Windows Machines

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

macOS

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

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

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

Linux

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 simba.athenaodbc.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 SIMBA_ATHENA_ODBC_INI to the full path and file name of the simba.athenaodbc.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 SIMBA_ATHENA_ODBC_INI to the full path and file name of the simba.athenaodbc.ini file.

For example, if your odbc.ini and odbcinst.ini files are located in /usr/local/odbc and your simba.athenaodbc.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 SIMBA_ATHENA_ODBC_INI=/etc/simba.athenaodbc.ini
```

For unixODBC:

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

export SIMBA ATHENA ODBC INI=/etc/simba.athenaodbc.ini

To locate the simba.athenaodbc.ini file, the driver uses the following search order:

- 1. If the SIMBA_ATHENA_ODBC_INI 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 simba.athenaodbc.ini.
- 3. The driver searches the current working directory of the application for a file named simba.athenaodbc.ini.
- 4. The driver searches the home directory for a hidden file named .simba.athenaodbc.ini (prefixed with a period).
- 5. The driver searches the /etc directory for a file named simba.athenaodbc.ini.

Configuring ODBC Connections on a Non-Windows Machine

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

- Creating a Data Source Name on a Non-Windows Machine on page 31
- Configuring a DSN-less Connection on a Non-Windows Machine on page 34
- Configuring Authentication on Non-Windows Machines on page 36
- Configuring Proxy Connections on Non-Windows Machines on page 40
- Configuring Query Result Encryption on a Non-Windows Machine on page 41
- Configuring Logging Options on a Non-Windows Machine on page 42
- Testing the Connection on a Non-Windows Machine on page 44

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 34.

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 start of the file name to make the file visible while you are editing it.

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

For example, on a macOS machine:

```
[ODBC Data Sources]
Sample DSN=Simba Athena ODBC Driver
```

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

```
[ODBC Data Sources]
Sample DSN=Simba Athena 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/simba/athenaodbc/lib/libathenaodbc_
sbu.dylib
```

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

```
Driver=/opt/simba/athenaodbc/lib/32/libathenaodbc_
sb32.so
```

b. Set the AwsRegion property to the AWS region of the Athena instance that you want to connect to.

For example:

```
AwsRegion=us-east-2
```

Note:

For a list of valid regions, see the "Athena" section in the *AWS Regions and Endpoints* documentation:

http://docs.aws.amazon.com/general/latest/gr/rande.html#athena.

c. Set the S3OutputLocation property to the path of the Amazon S3 location where you want to store query results, prefixed by s3://.

For example, to store results in a folder named "test-folder-1" inside an S3 bucket named "query-results-bucket", you would specify the following:

```
S3OutputLocation=s3://query-results-bucket/test-folder-1
```

- d. Configure authentication by specifying the authentication mechanism to use and providing your credentials. For more information, see Configuring Authentication on Non-Windows Machines on page 36.
- e. Optionally, configure the driver to connect to Athena through a proxy server. For more information, see Configuring Proxy Connections on Non-Windows Machines on page 40.
- f. Optionally, configure encryption for your query results. For more information, see Configuring Query Result Encryption on a Non-Windows Machine on page 41.
- g. Optionally, set additional key-value pairs as needed to specify other optional connection settings. For detailed information about all the configuration options supported by the Simba Athena ODBC Driver, see Driver Configuration Options on page 58.
- 4. Save the odbc.ini configuration file.

Note:

If you are storing this file in its default location in the home directory, then prefix the file name with a period (.) so that the file becomes hidden. If you are storing this file in another location, then save it as a non-hidden file (without the prefix), and make sure that the ODBCINI environment variable specifies the location. For more information, see Specifying the Locations of the Driver Configuration Files on page 29.

For example, the following is an odbc.ini configuration file for macOS containing a DSN that connects to Athena using IAM credentials:

```
[ODBC Data Sources]
Sample DSN=Simba Athena ODBC Driver
[Sample DSN]
Driver=/Library/simba/athenaodbc/lib/libathenaodbc_sbu.dylib
AuthenticationType=IAM Credentials
UID=ABCABCABC123ABCABC45
PWD=bCD+E1f2Gxhi3J4klmN/OP5QrSTuvwXYzabcdEF
AwsRegion=us-east-2
S3OutputLocation=s3://simba-athena-results/
```

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 Athena using IAM credentials:

```
[ODBC Data Sources]
Sample DSN=Simba Athena ODBC Driver 32-bit
[Sample DSN]
Driver=/opt/simba/athenaodbc/lib/32/libathenaodbc_sb32.so
```

```
AuthenticationType=IAM Credentials
UID=ABCABCABC123ABCABC45
PWD=bCD+E1f2Gxhi3J4klmN/OP5QrSTuvwXYzabcdEF
AwsRegion=us-east-2
S3OutputLocation=s3://simba-athena-results/
```

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.



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

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

For example:

```
[ODBC Drivers]
Simba Athena 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/simba/athenaodbc/lib/libathenaodbc_
sbu.dylib

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

Driver=/opt/simba/athenaodbc/lib/32/libathenaodbc_ sb32.so

b. Optionally, set the <code>Description</code> property to a description of the driver.

For example:

Description=Simba Athena 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 29.

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

```
[ODBC Drivers]
Simba Athena ODBC Driver=Installed
[Simba Athena ODBC Driver]
Description=Simba Athena ODBC Driver
Driver=/Library/simba/athenaodbc/lib/libathenaodbc_sbu.dylib
```

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

```
[ODBC Drivers]
Simba Athena ODBC Driver 32-bit=Installed
Simba Athena ODBC Driver 64-bit=Installed
[Simba Athena ODBC Driver 32-bit]
Description=Simba Athena ODBC Driver (32-bit)
Driver=/opt/simba/athenaodbc/lib/32/libathenaodbc_sb32.so
[Simba Athena ODBC Driver 64-bit]
Description=Simba Athena ODBC Driver (64-bit)
Driver=/opt/simba/athenaodbc/lib/64/libathenaodbc_sb64.so
```

You can now connect to your data store by providing your application with a connection string where the <code>Driver</code> property is set to the driver name specified in the <code>odbcinst.ini</code> 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 46.

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

- Configuring Authentication on Non-Windows Machines on page 36
- Configuring Proxy Connections on Non-Windows Machines on page 40
- Configuring Query Result Encryption on a Non-Windows Machine on page 41

For detailed information about all the connection properties that the driver supports, see .

Configuring Authentication on Non-Windows Machines

To access data from Athena, you must authenticate the connection. You can configure the Simba Athena ODBC Driver to provide your credentials and authenticate the connection using one of the following methods:

- Using the Default Credentials Provider Chain on Non-Windows Machines on page 36
- Using IAM Credentials on Non-Windows Machines on page 37
- Using an IAM Profile on Non-Windows Machines on page 37
- Using an Instance Profile on Non-Windows Machines on page 38
- Using the Active Directory Federation Services (AD FS) Credentials Provider on a Non-Windows Machine on page 38
- Using the Okta Service on a non-Windows machine on page 39

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.

Using the Default Credentials Provider Chain on Non-Windows Machines

You can configure the driver to authenticate the connection using credentials that are stored in one of the locations in the default credentials provider chain. The driver looks for a valid access key and secret key pair by checking the following locations, in the following order:

- 1. The AWS credentials file stored in the ~/.aws/credentials directory.
- 2. The AWS_ACCESS_KEY_ID and AWS_SECRET_ACCESS_KEY system environment variables.
- 3. The instance profile from the Amazon EC2 Instance Metadata Service.

For detailed information about configuring default credentials, see "Providing AWS Credentials" in the *AWS SDK for C++ Developer Guide*: http://docs.aws.amazon.com/sdk-for-cpp/v1/developer-guide/credentials.html.

To configure authentication using the default credentials provider chain on a non-Windows machine:

> Set the AuthenticationType property to Default Credentials.

Using IAM Credentials on Non-Windows Machines

You can configure the driver to authenticate the connection using an access key and a secret key that is specified directly in the connection information.

If you are using temporary credentials, which are only valid for a limited amount of time, then you must also provide a session token. For more information, see "Temporary Security Credentials" in the AWS Identity and Access Management User Guide: http://docs.aws.amazon.com/IAM/latest/UserGuide/id_credentials_temp.html.

To configure authentication using IAM credentials on a non-Windows machine:

- 1. Set the AuthenticationType property to IAM Credentials.
- 2. Set the UID property to the access key provided by your AWS account.
- 3. Set the PWD property to the secret key provided by your AWS account.
- 4. If you are using temporary credentials, set the SessionToken property to the session token generated by the AWS Security Token Service.

Using an IAM Profile on Non-Windows Machines

You can configure the driver to authenticate the connection using credentials that are associated with an IAM profile in a credentials file.

By default, the driver uses the credentials associated with a profile named <code>default</code> in the credentials file found in the ~/.aws/credentials directory. To use a different profile, specify the profile name in your connection settings. To use a different credentials file, set the AWS_SHARED_CREDENTIALS_FILE system environment variable to the full path of your credentials file.

For information about the format of a credentials file, see the "AWS Credentials File Format" section from the "Working with AWS Credentials" page in the AWS SDK for

Java Developer Guide: http://docs.aws.amazon.com/sdk-for-java/v1/developer-quide/credentials.html.

To configure authentication using an IAM profile on a non-Windows machine:

- 1. Set the AuthenticationType property to IAM Profile.
- 2. Set the AWSProfile property to the name of the profile to use.

Using an Instance Profile on Non-Windows Machines



Because Amazon EC2 instances are not available for macOS at this time, the macOS version of the Simba Athena ODBC Driver cannot use this authentication method.

You can configure the driver to authenticate the connection using credentials that have been loaded from the Amazon EC2 Instance Metadata Service into an instance profile.

Instance profiles contain authorization information such as roles, permissions, and credentials, and are automatically created by Amazon EC2 for each IAM role that is defined for an EC2 instance. For more information, see "IAM Roles for Amazon EC2" in the *Amazon Elastic Compute Cloud User Guide for Linux Instances*: http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/iam-roles-for-amazon-ec2.html.

To configure authentication using an instance profile on a non-Windows machine:

> Set the Authentication Type property to Instance Profile.

Using the Active Directory Federation Services (AD FS) Credentials Provider on a Non-Windows Machine

You can configure the driver to authenticate the connection using credentials obtained from the AD FS credentials provider. To do this, you must specify information about the AD FS service, such as the host and port of the server where the service is hosted.

To configure authentication using AD FS on a non-Windows machine:

- Set the AuthenticationType property to ADFS.
- 2. To specify your credentials for accessing the AD FS server, do the following:
 - a. Set the UID property to the user name that you use to access the AD FS server. You can include the domain name using the format [DomainName] \ [UserName].

- b. Set the PWD property to the password corresponding to the user name that you provided in the previous step.
- 3. To specify AD FS service information, do the following:
 - a. Set the IdP Host property to the host name of the AD FS service.

! Important:

The host name cannot include any slashes (/).

b. Optionally, set the IdP_Port property to the number of the port that the AD FS service host uses to listen for requests.

Note:

The exact port number that you need to specify may differ depending on the AD FS server configuration. If you are not sure which port to specify, contact your system administrator.

- 4. Optionally, set the Preferred_Role property to the Amazon Resource Name (ARN) of the role that you want to assume when authenticated through AD FS.
- 5. If the AD FS service must be accessed through an HTTP proxy, set the UseProxyForIdP property to 1. For information about configuring the proxy connection, see Configuring Proxy Connections on Non-Windows Machines on page 40.
- 6. Optionally, if you do not want the driver to verify the AD FS server certificate, set the SSL_Insecure property to false.

Using the Okta Service on a non-Windows machine

You can configure the driver to authenticate the connection using credentials obtained from the Okta credentials provider. To do this, you must specify information about the Okta service, such as the host name of the Okta service and the Okta application ID.

To configure authentication using Okta on a non-Windows machine:

- 1. **Set the** AuthenticationType **property to** Okta.
- 2. Set the UID property to the user name associated with your Okta account.
- 3. Set the PWD property to the password associated with your Okta user name.
- 4. Set the IdP Host property to the host name of the Okta service.
- 5. Set the App_ID property to the Okta-supplied ID associated with your Athena application.
- 6. Optionally, set the Preferred_Role property to the Amazon Resource Name (ARN) of the role that you want to assume when authenticated through Okta.
- 7. Optionally, set the App Name property to the name of your Okta application.

Configuring Proxy Connections on Non-Windows Machines

You can configure the driver to connect through a proxy server instead of connecting directly to the Athena service.

! Important:

If you are connecting to Athena through a proxy server, make sure that the proxy server does not block port 444. The result set streaming API uses port 444 on the Athena server for outbound communications. For more information, see Use Result Set Streaming on page 75.

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 a proxy connection on a non-Windows machine:

- 1. To enable proxy connections, set the UseProxy property to 1.
- 2. Set the ProxyHost property to the IP address or host name of your proxy server.
- 3. Set the ProxyPort property to the number of the TCP port that the proxy server uses to listen for client connections.
- 4. Set the ProxyUID property to your user name for accessing the proxy server.
- 5. Set the ProxyPWD property to your password for accessing the proxy server.

If the proxy server is configured to intercept SSL-encrypted connections, then in addition to specifying the proxy server information described above, you must also export the proxy server's root certificate onto your machine and configure the driver to use it.

To export and specify the proxy server's root certificate:

1. Export the proxy's certificate as a .pem file. You can do this using OpenSSL.

If necessary, you can export the certificate as another format, such as .cer, and convert that file into a .pem file.

For example:

openssl x509 -inform der -in certificate.cer -out certificate.pem

2. In your connection string or DSN (in the odbc.ini file), set the TrustedCerts property to the full path and name of .pem file containing the proxy server's root certificate.

Configuring Query Result Encryption on a Non-Windows Machine

You can configure the Simba Athena ODBC Driver to encrypt your query results using any of the encryption protocols that Athena supports.

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 query result encryption on a non-Windows machine:

1. Set the S3OutputEncOption property to one of the following values.



For detailed information about these encryption options, see "Configuring Encryption Options" in the *Amazon Athena User Guide*:

http://docs.aws.amazon.com/athena/latest/ug/encryption.html.

Option Name	Description
NOT_SET	The driver does not encrypt the data.
SSE_S3	The driver uses server-side encryption with an Amazon S3-managed key.
SSE_KMS	The driver uses server-side encryption with an AWS KMS-managed key.
CSE_KMS	The driver uses client-side encryption with an AWS KMS-managed key.

2. If you specified SSE_KMS or CSE_KMS in the previous step, then set the S3OutputEncKMSKey property to the KMS customer key to use for encrypting data.

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

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

To enable logging on a non-Windows machine:

- 1. Open the simba.athenaodbc.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.



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

5. Set the LogFileSize key to the maximum size of each log file in bytes.



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

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

The Simba Athena ODBC Driver produces the following log files at the location you specify using the LogPath key:

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

If you set the <code>UseLogPrefix</code> property to 1, then each file name is prefixed with <code>[UserName]_[ProcessID]_</code>, where <code>[UserName]</code> is the user name associated with the connection and <code>[ProcessID]</code> is the process ID of the application through which the connection is made. For more information, see <code>UseLogPrefix</code> on page 78.

To disable logging on a non-Windows machine:

- 1. Open the simba.athenaodbc.ini configuration file in a text editor.
- 2. Set the LogLevel key to 0.
- 3. Save the simba.athenaodbc.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.



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:

- 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 46.

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 <u>Driver Configuration Options</u> on page 58.

DSN Connection String Example

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

DSN=[DataSourceName]

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

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

DSN-less Connection String Examples

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

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

- [CertificateStorePath] is the complete path to the proxy's certificate.
- [CredProviderHost] is the host name of your credentials provider service.
- [ProxyHost] is the IP address of the proxy server you are connecting through.
- [Proxy Password] is the password associated with the application's proxy user ID.
- [ProxyUserID] is the ID your application uses to log into the proxy server.
- [S3Path] is the path of the Amazon S3 location where you want to store query results, prefixed by s3://.
- [Region] is the AWS region of the Athena instance that you want to connect to.

- [VPC Endpoint] is the endpoint URL for your Virtual Private Cloud.
- [YourAccessKey] is the access key provided by your AWS account.
- [YourCredProviderPassword] is your password for your credentials provider service.
- [YourCredProviderUserName] is your user name for your crendetials provider service.
- [YourProfileName] is the name of the IAM profile to use for authentication.
- [YourSecretKey] is the secret key provided by your AWS account.

Connecting to Athena Using the Default Credentials Provider Chain

The following is the format of a DSN-less connection string for connecting to Athena using the default credentials provider chain:

```
Driver=Simba Athena ODBC Driver; AwsRegion=
[Region]; S3OutputLocation=
[S3Path]; AuthenticationType=Default Credentials;
```

For example:

```
Driver=Simba Athena ODBC Driver; AwsRegion=us-east-
2; S3OutputLocation=s3://query-results-bucket/test-folder-
1; AuthenticationType=Default Credentials;
```

Connecting to Athena Using a VPC Endpoint

The following is the format of a DSN-less connection string for connecting to Athena using ta VPC Endpoint:

```
Driver=Simba Athena ODBC Driver; EndpointOverride=[VPC
Endpoint]; AwsRegion=[Region]; S3OutputLocation=
[S3Path]; AuthenticationType=IAM Profile; AWSProfile=
[YourProfileName];
```

For example:

```
Driver=Simba Athena ODBC Driver;EndpointOverride=vpce-
123456abc.athena.us-east-
1.vpce.amazonaws.com:443;AwsRegion=us-east-
2;S3OutputLocation=s3://query-results-bucket/test-folder-
1;AuthenticationType=AuthenticationType=IAM Profile;AWSProfile=simba;
```

Connecting to Athena Using IAM Credentials

The following is the format of a DSN-less connection string for connecting to Athena using IAM credentials:

```
Driver=Simba Athena ODBC Driver; AwsRegion=
[Region]; S3OutputLocation=
[S3Path]; AuthenticationType=IAM Credentials; UID=
[YourAccessKey]; PWD=[YourSecretKey];
```

For example:

```
Driver=Simba Athena ODBC Driver; AwsRegion=us-east-
2;S3OutputLocation=s3://query-results-bucket/test-folder-
1;AuthenticationType=IAM Credentials; UID=ABCABCABC123ABCABC4
5;PWD=abCD+E1f2Gxhi3J4klmN/OP5QrSTuvwXYzabcdEF;
```

Connecting to Athena Using an IAM Profile

The following is the format of a DSN-less connection string for connecting to Athena using an IAM profile:

```
Driver=Simba Athena ODBC Driver; AwsRegion=
[Region]; S3OutputLocation=
[S3Path]; AuthenticationType=IAM Profile; AWSProfile=
[YourProfileName];
```

For example:

```
Driver=Simba Athena ODBC Driver; AwsRegion=us-east-
2; S3OutputLocation=s3://query-results-bucket/test-folder-
1; AuthenticationType=IAM Profile; AWSProfile=simba;
```

Connecting to Athena Using an Instance Profile

The following is the format of a DSN-less connection string for connecting to Athena using an instance profile from the Amazon EC2 Instance Metadata Service:

```
Driver=Simba Athena ODBC Driver; AwsRegion=
[Region]; S3OutputLocation=
[S3Path]; AuthenticationType=Instance Profile;
```

For example:

```
Driver=Simba Athena ODBC Driver; AwsRegion=us-east-
2; S3OutputLocation=s3://query-results-bucket/test-folder-
1; AuthenticationType=Instance Profile;
```

Connecting to Athena Using the AD FS Credentials Provider

The following is the format of a DSN-less connection string for connecting to Athena using credentials provided by the AD FS service. If you are connecting to Athena from a Windows machine, the UID and PWD properties are optional.

```
Driver=Simba Athena ODBC Driver; AwsRegion=
[Region]; S3OutputLocation=
[S3Path]; AuthenticationType=ADFS; IdP_Host=
[CredProviderHost]; UID=[YourCredProviderUserName]; PWD=
[YourCredProviderPassword];
```

For example:

```
Driver=Simba Athena ODBC Driver; AwsRegion=us-east-
2; S3OutputLocation=s3://query-results-bucket/test-folder-
1; AuthenticationType=ADFS; IdP_
Host=example.adfs.server; UID=HOME\jsmith; PWD=simba12345;
```

Connecting to Athena Using the Okta Credentials Provider

The following is the format of a DSN-less connection string for connecting to Athena using credentials provided by the Okta credentials service.

```
Driver=Simba Athena ODBC Driver; AwsRegion=
[Region]; S3OutputLocation=
[S3Path]; AuthenticationType=Okta; IdP_Host=
[CredProviderHost]; UID=[YourCredProviderUserName]; PWD=
[YourCredProviderPassword]; App_ID=[YourOktaAppId];
```

For example:

```
Driver=Simba Athena ODBC Driver; AwsRegion=us-east-2; S3OutputLocation=s3://query-results-bucket/test-folder-1; AuthenticationType=Okta; IdP_Host=dev-123456.okta.com:UID=jsmith@example.com; PWD=simba12345; App_ID=12abc123456789/123;
```

Connecting to Athena Using a Proxy Server

The following is the format of a DSN-less connection string for connecting to Athena using a proxy server:

```
Driver=Simba Athena ODBC Driver; AwsRegion=
[Region]; S3OutputLocation=
[S3Path]; AuthenticationType=Default
Credentials; UseProxy=1; ProxyScheme=HTTPS; ProxyHost=
[ProxyHost]; ProxyPort=[Port]; ProxyUID=
[ProxyUserID]; ProxyPWD=[ProxyPassword];
```

For example:

```
Driver=Simba Athena ODBC Driver; AwsRegion=us-east-
2; S3OutputLocation=s3://query-results-bucket/test-folder-
1; AuthenticationType=Default
Credentials; UseProxy=1; ProxyScheme=HTTPS; ProxyHost=123.456.7
89.012; ProxyPort=8080; ProxyUID=simba; ProxyPWD=simba;
```

Connecting to Athena Using a Proxy Server on a Non-Windows Machine With Trusted Certificate

The following is the format of a DSN-less connection string for connecting to Athena using a proxy server:

```
Driver=Simba Athena ODBC Driver; AwsRegion=
[Region]; S3OutputLocation=
[S3Path]; AuthenticationType=Default
Credentials; UseProxy=1; ProxyScheme=HTTPS; ProxyHost=[Proxy Host]; ProxyPort=[Port]; ProxyUID=[Proxy User ID]; ProxyPWD=
[Proxy Password]; TrustedCerts=[CertificateStorePath];
```

For example:

```
Driver=Simba Athena ODBC Driver; AwsRegion=us-east-
2;S3OutputLocation=s3://query-results-bucket/test-folder-
1;AuthenticationType=Default
Credentials;UseProxy=1;ProxyScheme=HTTPS;ProxyHost=123.456.7
89.012;ProxyPort=8080;ProxyUID=simba;ProxyPWD=simba;TrustedCerts=/disk/dir/certificates.pem;
```

Example: Using Workgroups

A workgroup is an Athena feature that enables you to control the data access and costs associated with running queries. For more information, see "Using Workgroups to Control Query Access and Costs" in the Amazon Athena User Guide: https://docs.aws.amazon.com/athena/latest/ug/manage-queries-control-costs-withworkgroups.html.

To use a workgroup when connecting to Athena through the Simba Athena ODBC Driver, either specify a workgroup in the DSN, write a connection URL that sets the Workgroup property to the name of your workgroup. For example, to use a workgroup named SimbaAdmins:

Specifying a workgroup does not change the way that you run SQL statements or make ODBC API calls. The driver passes the workgroup name to Athena, and all workgroup handling takes place in the Athena service instead of in the driver.

Features

For more information on the features of the Simba Athena ODBC Driver, see the following:

- Catalog and Schema Support on page 52
- File Formats on page 52
- Data Types on page 52
- Result Set Streaming Support on page 56
- Query Execution Polling on page 56
- Security and Authentication on page 57

Catalog and Schema Support

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

Amazon Athena provides catalogs that enable you to access the data source that is being queried. These catalogs contain databases, which correspond to ODBC schemas, and each database contains data that has been organized into tables.

By default, if your query statement does not specify a catalog, the driver queries the data under the catalog named AwsDataCatalog. You can use the Catalog connection property to specify a different default catalog for your queries. For more information, see Catalog on page 61.

File Formats

The Simba Athena ODBC Driver supports all the file formats that Athena supports, which include the following:

- Avro
- Comma-Separated Values (CSV)
- JavaScript Object Notation (JSON)
- Optimized Row Columnar (ORC)
- Parquet

Data Types

The Simba Athena ODBC Driver supports many common data formats, converting between Athena data types and SQL data types.

The following table lists the supported data type mappings.

Athena Type	SQL Type
ARRAY	 SQL_VARCHAR if the Use SQL Unicode Types option (the UseSQLUnicodeTypes property) is disabled. SQL_WVARCHAR if the Use SQL Unicode Types option (the UseSQLUnicodeTypes property) is enabled.
BIGINT	SQL_BIGINT
BINARY	SQL_VARBINARY
BOOLEAN	SQL_BIT
CHAR	 SQL_CHAR if the Use SQL Unicode Types option (the UseSQLUnicodeTypes property) is disabled. SQL_WCHAR if the Use SQL Unicode Types option (the UseSQLUnicodeTypes property) is enabled.
DATE Note: Not supported for Parquet files.	 SQL_TYPE_DATE if the application uses ODBC version 3.00 or later. SQL_DATE if the application uses an ODBC version earlier than 3.00.
DECIMAL (p, s)	SQL_DECIMAL
DOUBLE	SQL_DOUBLE
FLOAT	SQL_REAL

Athena Type	SQL Type
INTEGER Note: Although Athena reports integer data as type INT, the driver reports integer data as type INTEGER to ensure compatibility with standard BI tools. For more information, see	SQL_INTEGER
Integer Support on page 55. MAP	 SQL_VARCHAR if the Use SQL Unicode Types option (the UseSQLUnicodeTypes property) is disabled. SQL_WVARCHAR if the Use SQL Unicode Types option (the UseSQLUnicodeTypes property) is enabled.
SMALLINT	SQL_SMALLINT
STRING	 SQL_VARCHAR if the Use SQL Unicode Types option (the UseSQLUnicodeTypes property) is disabled. SQL_WVARCHAR if the Use SQL Unicode Types option (the UseSQLUnicodeTypes property) is enabled.
STRUCT	 SQL_VARCHAR if the Use SQL Unicode Types option (the UseSQLUnicodeTypes property) is disabled. SQL_WVARCHAR if the Use SQL Unicode Types option (the UseSQLUnicodeTypes property) is enabled.

Athena Type	SQL Type	
TIMESTAMP	 SQL_TYPE_TIMESTAMP if the application uses ODBC version 3.00 or later. SQL_TIMESTAMP if the application uses an ODBC version earlier than 3.00. 	
TINYINT	SQL_TINYINT	
VARCHAR	 SQL_VARCHAR if the Use SQL Unicode Types option (the UseSQLUnicodeTypes property) is disabled. SQL_WVARCHAR if the Use SQL Unicode Types option (the UseSQLUnicodeTypes property) is enabled. 	

Integer Support

Athena combines two different implementations of the integer data type:

- In Data Definition Language (DDL) queries, Athena uses the INT data type from Apache Hive.
- In all other queries, Athena uses the INTEGER data type from Presto.

To support the CAST queries that are used in many BI tools, the driver reports integer data as type INTEGER even though Athena reports the data as type INT.

Be aware that, when executing DDL queries, you must specify integer data using INT as the data type.



Athena supports some but not all DDL statements. For a list of the supported DDL statements, see "SQL and HiveQL Reference" in the *Amazon Athena API Reference*: http://docs.aws.amazon.com/athena/latest/ug/language-reference.html.

Result Set Streaming Support

The driver uses the result set streaming API to improve the performance in fetching query results. To take advantage of this feature you must include and allow the athena: GetQueryResultsStream action in your IAM policy statement. For details on managing Athena IAM policies, see https://docs.aws.amazon.com/athena/latest/ug/access.html.

This is configured using the Use Result Set Streaming option (the UseResultSetStreaming property). For more information, see Use Result Set Streaming on page 75.

Query Execution Polling

When a query is run, the driver polls the Athena server for the query results until they are returned. The driver starts by polling the server frequently, and then increasing the interval of time between polls (decreasing the polling rate) as the query continues to run. You can configure the polling rate of the driver by setting the following properties:

- Max Query Execution Polling Interval on page 66
- Min Query Execution Polling Interval on page 67
- Query Execution Polling Interval Multiplier on page 71

The driver starts by polling the server every minimum interval, which is the number of milliseconds specified in Min Query Execution Polling Interval. The driver then increases the polling interval using the multiplier specified in Query Execution Polling Interval Multiplier until the interval specified in Max Query Execution Polling Interval is reached. The driver then continues to poll the server using this maximum interval until the query results are returned.

For example, the driver uses the following default settings for query execution polling:

- Min Query Execution Polling Interval: 5
- Max Query Execution Polling Interval: 1800000
- Query Execution Polling Interval Multiplier: 2

Using these settings, the driver polls the server 5ms after the query begins to run, and then doubles the interval between polls after each subsequent poll. In other words, the driver polls the server at these intervals: 5ms after the query starts to run, 10ms after the first poll, 20ms after the second poll, and so on until the 1800000ms polling interval is reached. The driver then continues to poll the server every 1800000ms until the query results are returned.

Security and Authentication

To protect data from unauthorized access, Athena requires all connections to be authenticated using IAM credentials (an access key and a secret key), and uses the SSL protocol that is implemented in Amazon Web Services. The Simba Athena ODBC Driver protects your data by providing support for these authentication protocols and further obscuring data from unwanted access by providing encryption options for your query results.

The driver can authenticate your connection using IAM credentials from any of the following sources:

- A default credentials provider chain
- An IAM profile
- An instance profile
- The DSN or connection string settings
- The Active Directory Federation Services (AD FS) credentials provider
- Okta service credentials provider

For detailed configuration instructions, see Configuring Authentication on Windows on page 12 or Configuring Authentication on Non-Windows Machines on page 36.

Additionally, the driver automatically applies SSL encryption to all connections. SSL encryption protects data and credentials when they are transferred over the network, and provides stronger security than authentication alone.

For query results, the Simba Athena ODBC Driver supports all the encryption options that Athena supports. For detailed information about the supported encryption options, see "Configuring Encryption Options" in the *Amazon Athena User Guide*: http://docs.aws.amazon.com/athena/latest/ug/encryption.html. For information about configuring encryption in the driver, see Creating a Data Source Name on Windows on page 10 or Configuring Query Result Encryption on a Non-Windows Machine on page 41.

Driver Configuration Options

Driver Configuration Options lists the configuration options available in the Simba Athena 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 machine, the fields and buttons described below are available in the following dialog boxes:

- Simba Athena ODBC Driver DSN Setup
- Authentication Options
- Advanced Options
- Logging Options

When using a connection string or configuring a connection from a non-Windows machine, use the key names provided below.

Configuration Options Appearing in the User Interface

The following configuration options are accessible via the Windows user interface for the Simba Athena ODBC Driver, or via the key name when using a connection string or configuring a connection from a Linux or macOS computer:

- Authentication Type on page 59
- AWS Profile on page 60
- AWS Region on page 60
- Binary Column Length on page 60
- Catalog on page 61
- Encryption Options on page 61
- Endpoint Override on page 61
- IdP Host on page 62
- IdP Port on page 62
- KMS Key on page 62
- Log Level on page 63
- Log Path on page 64
- Max Catalog Name Length on page 64

- Okta App ID on page 68
- Okta App Name on page 68
- Password on page 68
- Preferred Role on page 69
- Proxy Host on page 69
- Proxy Password on page 70
- Proxy Port on page 70
- Proxy Username on page 70
- Query Execution Polling Interval Multiplier on page 71
- Rows To Fetch Per Block on page 71
- S3 Output Location on page 72
- Schema on page 72

- Max Column Name Length on page 65
- Max Complex Type Column Length on page 65
- Max File Size on page 65
- Max Number Files on page 66
- Max Query Execution Polling Interval on page 66
- Max Schema Name Length on page 67
- Max Table Name Length on page 67
- Min Query Execution Polling Interval on page 67

- Session Token on page 73
- SSL Insecure on page 73
- Streaming Endpoint Override on page 73
- String Column Length on page 74
- Use HTTP Proxy For IdP Host on page 74
- Use Proxy on page 74
- Use Result Set Streaming on page 75
- Use SQL Unicode Types on page 75
- User on page 76
- Workgroup on page 76

Authentication Type

Key Name	Default Value	Required
AuthenticationType	IAM Credentials (IAM Credentials)	Yes

Description

This option specifies how the driver authenticates the connection to Athena.

- Default Credentials (Default Credentials): The driver authenticates the connection using credentials that are stored in one of the locations in the default credentials provider chain.
- IAM Credentials (IAM Credentials): The driver authenticates the connection using an access key and a secret key that is specified directly in the connection information.
- IAM Profile (IAM Profile): The driver authenticates the connection using credentials that are associated with an IAM profile in a credentials file.
- Instance Profile (Instance Profile): The driver authenticates the connection using credentials that have been loaded from the Amazon EC2 Instance Metadata Service into an instance profile.
- ADFS (ADFS): The driver authenticates the connection using credentials provided by the Active Directory Federation Services (AD FS) credential provider.

• Okta (Okta): The driver authenticates the connection using credentials provided by the Okta credential provider.

AWS Profile

Key Name	Default Value	Required
AwsProfile	default	No

Description

The name of the profile to use from the credentials file. This setting is applicable only when Authentication Type is set to IAM Profile (the AuthenticationType property is set to IAM Profile).

AWS Region

Key Name	Default Value	Required
AwsRegion	None	Yes

Description

The AWS region of the Athena instance that you want to connect to.

For a list of valid regions, see the "Athena" section in the *AWS Regions and Endpoints* documentation: http://docs.aws.amazon.com/general/latest/gr/rande.html#athena.

Binary Column Length

Key Name	Default Value	Required
BinaryColumnLength	32767	No

Description

The maximum data length for BINARY columns.

Catalog

Key Name	Default Value	Required
Catalog	AwsDataCatalog	No

Description

The default catalog used for query execution. If no catalog is specified in the SQL query, the Athena driver executes the query against the catalog specified.

Encryption Options

Key Name	Default Value	Required
S3OutputEncOption	NOT_SET (NOT_SET)	Yes

Description

The encryption protocol that the driver uses to encrypt your query results.

- NOT_SET (NOT SET): The driver does not encrypt the data.
- SSE_S3 (SSE_S3): The driver uses server-side encryption with an Amazon S3-managed key.
- SSE_KMS (SSE_KMS): The driver uses server-side encryption with an AWS KMS-managed key.
- CSE_KMS (CSE_KMS): The driver uses client-side encryption with an AWS KMS-managed key.

For detailed information about these encryption options, see "Configuring Encryption Options" in the *Amazon Athena User Guide*:

http://docs.aws.amazon.com/athena/latest/ug/encryption.html.

Endpoint Override

Key Name	Default Value	Required
EndpointOverride	None	Yes, if not using the default Athena endpoint.

Description

The endpoint for the Athena instance the driver connects to if not using the default endpoint. If this property is not set, the driver attempts to connect to the default Athena endpoint.

IdP Host

Key Name	Default Value	Required
IdP_Host	None	Yes, if authenticating through AD FS or Okta.

Description

The host name of the AD FS or Okta service that you use to authenticate the connection. The host name cannot include any slashes (/).

IdP Port

Key Name	Default Value	Required
IdP_Port	443	No

Description

The number of the port that the AD FS or Okta service host uses to listen for requests.

The port number to specify may differ depending on the service's server configuration. If you are not sure which port to specify, contact your system administrator.

KMS Key

Key Name	Default Value	Required
S30utputEncKMSKey	None	Yes, if using SSE_KMS or CSE_KMS encryption.

Description

The KMS customer key to use when encrypting query results using SSE_KMS or CSE_KMS encryption.

For detailed information about the supported encryption options, see "Configuring Encryption Options" in the *Amazon Athena User Guide*:

http://docs.aws.amazon.com/athena/latest/ug/encryption.html.

Log Level

Key Name	Default Value	Required
LogLevel	OFF (0)	No

Description

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

! Important:

- Only enable logging long enough to capture an issue. Logging decreases performance and can consume a large quantity of disk space.
- The settings for logging apply to every connection that uses the Simba Athena ODBC Driver, so make sure to disable the feature after you are done using it.
- This option is not supported in connection strings. To configure logging for the Windows driver, you must use the Logging Options dialog box. To configure logging for a non-Windows driver, you must use the simba.athenaodbc.ini file.

Set the property to one of the following values:

- OFF (0): Disable all logging.
- FATAL (1): Logs severe error events that lead the driver to abort.
- ERROR (2): Logs error events that might allow the driver to continue running.
- WARNING (3): Logs events that might result in an error if action is not taken.
- INFO (4): Logs general information that describes the progress of the driver.
- DEBUG (5): Logs detailed information that is useful for debugging the driver.
- TRACE (6): Logs all driver activity.

When logging is enabled, the driver produces the following log files at the location you specify in the Log Path (LogPath) property:

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

If you enable the UseLogPrefix connection property, the driver prefixes the log file name with the user name associated with the connection and the process ID of the application through which the connection is made. For more information, see UseLogPrefix on page 78.

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 simba.athenaodbc.ini file.

Max Catalog Name Length

Key Name	Default Value	Required
MaxCatalogNameLen	0	No

Description

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

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

Max Column Name Length

Key Name	Default Value	Required
MaxColumnNameLen	0	No

Description

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

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

Max Complex Type Column Length

Key Name	Default Value	Required
ComplexTypeColumnLength	65535	No

Description

The maximum data length for complex data types that the driver casts to SQL_VARCHAR. For example, ARRAY, MAP, and STRUCT data types.

Max File Size

Key Name	Default Value	Required
LogFileSize	20971520	No

Description

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

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

! Important:

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

Max Query Execution Polling Interval

Key Name	Default Value	Required
MaxQueryExecutionPollingInterval	1800000	No

Description

The maximum time, in milliseconds, to wait between attempts when polling the server for the query execution result. This value cannot be lower than the Min Query Execution Polling Interval setting.

For more information Query Execution Polling on page 56.

Max Schema Name Length

Key Name	Default Value	Required
MaxSchemaNameLen	256	No

Description

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

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

Max Table Name Length

Key Name	Default Value	Required
MaxTableNameLen	0	No

Description

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

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

Min Query Execution Polling Interval

Key Name	Default Value	Required
MinQueryExecutionPollingInterval	5	No

Description

The minimum value of the polling interval, in milliseconds. This value must be greater than 0 but less than Max Query Execution Polling Interval. A value of 0 or a negative value means that the default value of 5 is used.

For more information see Query Execution Polling on page 56.

Okta App ID

Key Name	Default Value	Required
App_ID	None	Yes, if authenticating through the Okta service.

Description

The Okta-provided unique ID associated with your Athena application.

Okta App Name

Key Name	Default Value	Required
App_Name	None	No

Description

The name of the Okta application that you use to authenticate the connection to Athena.

Password

Key Name	Default Value	Required
PWD	None	Yes, if Authentication Type is set to IAM Credentials, or if the Authentication Type is set to Okta or ADFS when connecting from a non-Windows machine.

Description

If Authentication Type is set to IAM Credentials (the AuthenticationType property is set to IAM Credentials), then set this property to the secret key provided by your AWS account.

If Authentication Type is set to ADFS or Okta (the AuthenticationType property is set to ADFS or Okta), then set this property to the password that you use to access the credentials service server.

On Windows machines, if you do not provide a password, the driver attempts to authenticate to the AD FS server using your Windows password over the NTLM protocol.

Preferred Role

Key Name	Default Value	Required
Preferred_Role	None.	No
	However, by default, the driver assumes the first role from the list returned in the SAML response from the identity provider.	

Description

The Amazon Resource Name (ARN) of the role that you want to assume when authenticated through AD FS or Okta.

Proxy Host

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

Description

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

Proxy Password

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

Description

The password that you use to access the proxy server.

Proxy Port

Key Name	Default Value	Required
ProxyPort	8080	Yes, if connecting through a proxy server.

Description

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

Proxy Username

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

Description

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

Query Execution Polling Interval Multiplier

Key Name	Default Value	Required
QueryExecutionPollingIntervalMultiplier	2	No

Description

The multiplier by which the driver increases the amount of time between polls, when polling the Athena server for query results. You cannot specify a value less than 2.

For more information see Query Execution Polling on page 56.

Rows To Fetch Per Block

Key Name	Default Value	Required
RowsToFetchPerBlock	10000 for result set streaming, 1000 for pagination	No

Description

The maximum number of rows to fetch per stream if using the result set streaming API.

Or, the maximum number of rows to fetch per page if using pagination.

See Use Result Set Streaming on page 75 for details on result set streaming.



Note:

While setting this option with a large value when using the result set streaming API can give you better fetch performance, it can also result in higher memory usage. This can be mitigated if the ODBC application can retrieve the result set from the driver quickly.

S3 Output Location

Key Name	Default Value	Required
S3OutputLocation	None	Yes, if the Workgroup property specifies a workgroup that is not configured with an output location.

Description

The path of the Amazon S3 location where you want to store query results, prefixed by s3://.

For example, to store Athena query results in a folder named "test-folder-1" inside an S3 bucket named "query-results-bucket", you would set this property to s3://query-results-bucket/test-folder-1.

For details on managing Athena S3 output in workgroups, see https://docs.aws.amazon.com/athena/latest/APIReference/API_ResultConfiguration.html#athena-Type-ResultConfiguration-OutputLocation.

Schema

Key Name	Default Value	Required
Schema	default	No

Description

The name of the database schema to use when a schema is not explicitly specified in a query. You can still issue queries on other schemas by explicitly specifying the schema in the query.

Session Token

Key Name	Default Value	Required
SessionToken	None	Yes, if you are using temporary security credentials.

Description

The session token generated by the AWS Security Token Service. This setting is applicable only when Authentication Type is set to IAM Credentials (the AuthenticationType property is set to IAM Credentials).

SSL Insecure

Key Name	Default Value	Required
SSL_Insecure	Disabled (false)	No

Description

This property indicates whether the ldp host server certificate should be verified.

- Enabled (true): The driver does not check the authenticity of the server certificate.
- Disabled (false): The driver checks the authenticity of the server certificate.

Streaming Endpoint Override

Key Name	Default Value	Required
StreamingEndpointOverride	None	No

Description

The endpoint for the Athena streaming service corresponding to the specified Athena instance. If this property is not set and Use Result set Streaming is set, the driver tries to connect to the 444 port of either the default Athena endpoint or Athena endpoint specified in the Endpoint Override connection property.

String Column Length

Key Name	Default Value	Required
StringColumnLength	255	No

Description

The maximum data length for STRING columns.

Use HTTP Proxy For IdP Host

Key Name	Default Value	Required
UseProxyForIdP	Disabled (0)	Yes, if authenticating through an AD FS service that must be accessed through an HTTP proxy.

Description

This option specifies whether the driver accesses the AD FS service through an HTTP proxy.

- Enabled (1): The driver accesses the AD FS service through 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. In order for these proxy settings to take effect, the Use Proxy option (or the UseProxy property) must also be enabled.
- Disabled (0): The driver accesses the AD FS service directly.

Use Proxy

Key Name	Default Value	Required
UseProxy	Clear (0)	No

Description

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

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

Use Result Set Streaming

Key Name	Default Value	Required
UseResultsetStreaming	1	No

Description

This property specifies whether the driver uses the AWS result set streaming API to fetch result sets.

- 1: The driver uses the result set streaming API.
- 0: The driver uses pagination logic for result set fetching.

See Rows To Fetch Per Block on page 71 to configure how many rows to fetch per stream.

Use SQL Unicode Types

Key Name	Default Value	Required
UseSQLUnicodeTypes	Clear (0)	No

Description

This option specifies the SQL types to be returned for string data types.

- Enabled (1): The driver returns SQL_WVARCHAR for ARRAY, MAP, STRING, STRUCT, and VARCHAR columns.
- Disabled (0): The driver returns SQL_VARCHAR for for ARRAY, MAP, STRING, STRUCT, and VARCHAR columns.

User

Key Name	Default Value	Required
UID	None	Yes, if Authentication Type is set to IAM Credentials, or if the Authentication Type is set to ADFS or Okta when connecting from a non-Windows machine.

Description

If Authentication Type is set to IAM Credentials (the AuthenticationType property is set to IAM Credentials), then set this property to the access key provided by your AWS account.

If Authentication Type is set to ADFS or Okta (the AuthenticationType property is set to ADFS or Okta), then set this property to the user name that you use to access the authentication server.

When using AD FS, you can include the domain name using the format <code>[DomainName] \setminus [UserName]</code>. On Windows machines, if you do not provide a user name when using AD FS, the driver attempts to authenticate to the AD FS server using your Windows user name over the NTLM protocol.

Workgroup

Key Name	Default Value	Required
Workgroup	primary	No

Description

The name of the workgroup to use when signing in to Athena.

A workgroup is an Athena feature that enables you to control the data access and costs associated with running queries. For more information, see "Using Workgroups to Control Query Access and Costs" in the Amazon Athena User Guide:

https://docs.aws.amazon.com/athena/latest/ug/manage-queries-control-costs-withworkgroups.html

Specifying a workgroup does not change the way that you run SQL statements or make ODBC API calls. The driver passes the workgroup name to Athena, and all workgroup handling takes place in the Athena service instead of in the driver.

Configuration Options Having Only Key Names

The following configuration options do not appear in the Windows user interface for the Simba Athena ODBC Driver. They are accessible only when you use a connection string or configure a connection on macOS or Linux.

- Driver on page 77
- ProxyScheme on page 78
- TrustedCerts on page 78

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

UseLogPrefix on page 78

Driver

Key Name	Default Value	Required
Driver	Simba Athena 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 (Simba Athena 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.

ProxyScheme

Key Name	Default Value	Required
ProxyScheme	HTTP	No

Description

The scheme to use to connect to the proxy server.

Set the property to one of the following values:

- HTTP: For connections using HTTP.
- HTTPS: For connections using HTTPS.

TrustedCerts

Key Name	Default Value	Required
TrustedCerts	The cacerts.pem file in the /lib subfolder within the driver's installation directory.	No

Description

The full path and name of the .pem file containing the root certificate of the proxy server.



This setting is applicable only when connecting from a non-Windows machine, and only when connecting through a proxy server that has SSL interception enabled.

UseLogPrefix

Key Name	Default Value	Required
UseLogPrefix	0	No

Description

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

Set the property to one of the following values:

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

For example, if you are connecting as a user named "jdoe" and using the driver in an application with process ID 7836, the generated log files would be named jdoe_7836_simbaathenaodbcdriver.log and jdoe_7836_simbaathenaodbcdriver_connection_[Number].log, where [Number] is a number that identifies each connection-specific log file.

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

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

- For a 32-bit driver installed on a 64-bit machine: HKEY_LOCAL_ MACHINE\SOFTWARE\Wow6432Node\Simba\Simba Athena ODBC Driver\Driver
- Otherwise: HKEY_LOCAL_MACHINE\SOFTWARE\Simba\Simba Athena ODBC Driver\Driver

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

To configure this option for a non-Windows driver, you must use the simba.athenaodbc.ini file.

Third-Party Trademarks

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.

Amazon Athena, Amazon S3, Amazon Simple Storage Service, Amazon Web Services, AWS, AWS Glue, and Amazon are trademarks or registered trademarks of Amazon Web Services, Inc. or its subsidiaries in Canada, United States and/or other countries.

All other trademarks are trademarks of their respective owners.