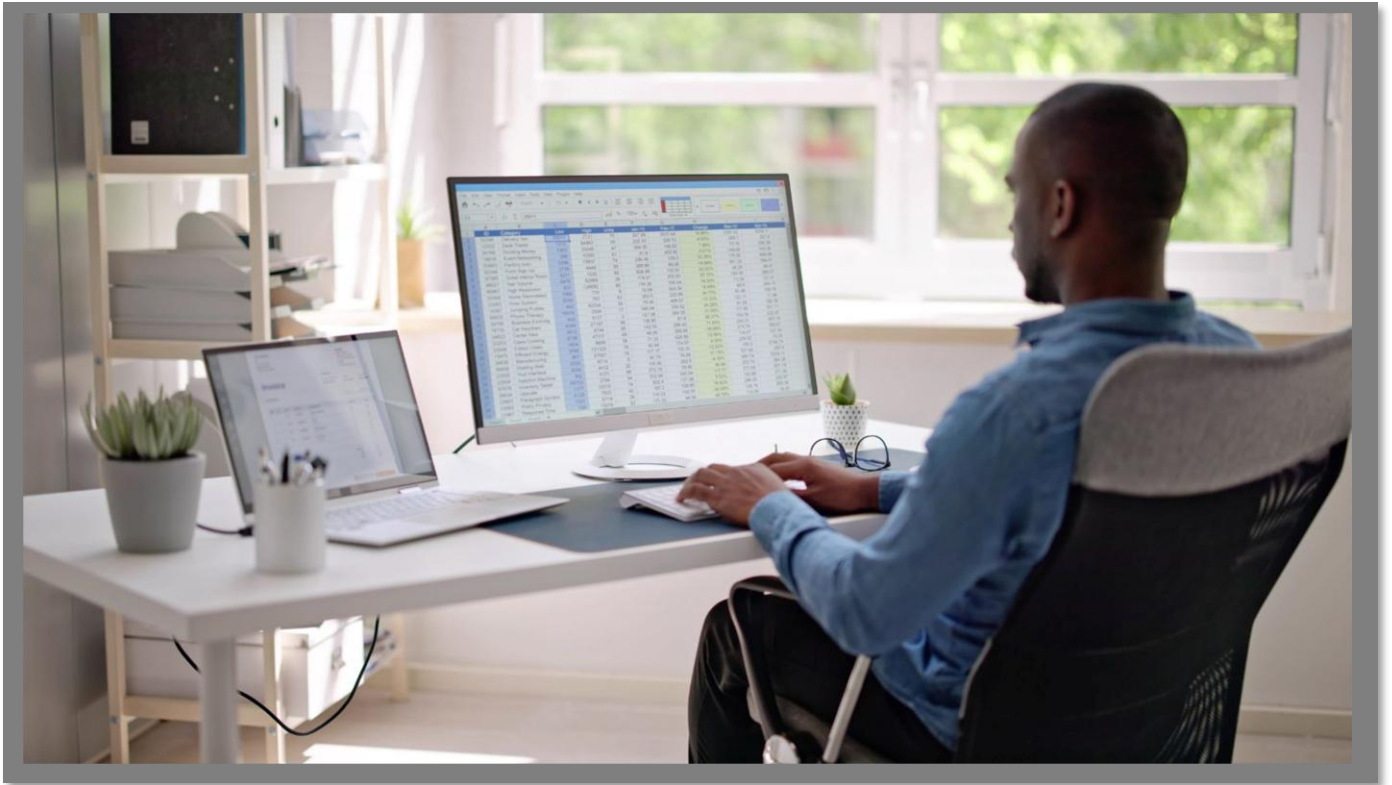


Amazon S3 CONFIGURATION GUIDE



Arbutus Connectors

Contents

A. Introduction	1
B. About Amazon S3	1
C. Determining if the Connector exists prior to configuring	2
D. Configuring the Connector after it has been installed	3
E. Editing the DSN properties – the Basic and Advanced tabs .	6
E1. Editing the DSN properties in the Basic tab	6
E2. Editing the DSN properties in the Advanced tab	19
F. Other questions and/or request for assistance	21

Arbutus Connectors

Arbutus Connector – Amazon S3

A. Introduction

The purpose of this Guide is to provide assistance with configuring the Arbutus Amazon S3 Connector using the ODBC Data Source Administrator. The configuration process can involve several technical steps that require a good understanding of IT systems and database management.

To make the most of this guide, it's advisable to have a good understanding of database connectivity, driver installation, and system settings. The ODBC Data Source Administrator, which is used as part of the configuration process, allows for the setup and management of data sources, enabling applications to access data from various database systems.

Due to the complexity and potential impact of these configurations, it is recommended that only those individuals with IT or database expertise undertake this task. In addition, it should also be understood that each client's network environment is different. A one-size-fits-all approach is rarely effective, as what works well in one environment may not be suitable in another.

B. About Amazon S3

Amazon S3 (Simple Storage Service) is a cloud storage service provided by Amazon Web Services (AWS). It allows users to store and retrieve any amount of data at any time, offering high scalability, robust security features and encryption to protect your data, and performance for a wide range of applications. Amazon S3 integrates seamlessly with other AWS services, e.g., Amazon Redshift and Amazon DynamoDB. However, each of them, including Amazon S3, serve distinct purposes within AWS.

Data in Amazon S3 is stored as objects, consisting of data (binary files, text, images, backups, etc.), metadata, and unique key. Objects are stored in buckets, which are globally unique logical containers for organizing data. Each object in a bucket is identified by a unique key (filename/path-like structure).

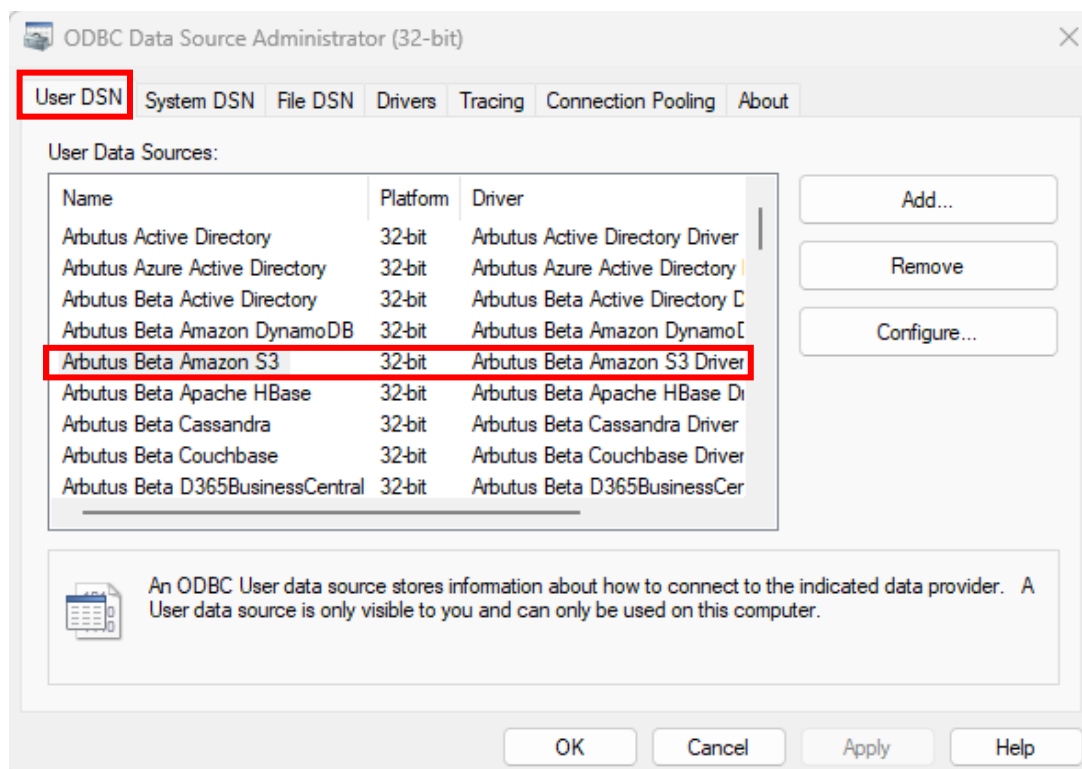
Arbutus Connectors

C. Determining if the Connector exists prior to configuring

Installation of the Arbutus Amazon S3 Connector is done at the time of installing the Arbutus software. For more information on this, please see the **Overview Guide Document**.

Once the Connector has been installed, the next step is to configure it.

Prior to configuring it, you can check to see if the Connector has been installed by opening the **32-bit ODBC Data Source Administrator**, pictured below, and clicking the **User DSN** tab. Included below is information on how you can access the **ODBC Data Source Administrator**.



Arbutus Connectors

- If the Arbutus Amazon S3 Connector appears in the list, it can be considered as installed.
- If it is not listed, it is likely that you did not select it during the installation or modification of the Arbutus software. In this case, it is recommended to reinstall the Arbutus software and choose the **Modify** option when prompted. For more details, please refer to the **Overview Guide Document**.

Below is the file path to access and run the **ODBC Data Source Administrator** application:

C:\Windows\SysWOW64\odbcad32.exe

Alternative, you can also try locating and opening the **ODBC Data Source Administrator** application by doing a search on your desktop application.

D. Configuring the Connector after it has been installed

Once you have verified that the Arbutus Connector has been installed, it is time to configure it.

This process is done using the **ODBC Data Source Administrator**. It can be described as “**editing the DSN configuration**”.

DSN, Drivers, and Data Sources

What is a DSN? DSN stands for Data Source Name, and is a unique name used to create a data connection to a database using open database connectivity (ODBC).

A DSN is a data structure that contains the information required to connect to a database. It is essentially a string that identifies the source database, including the driver details, the database name, and often authentication credentials and other necessary connection parameters. DSNs facilitate a standardized method for applications to access databases without needing hard-coded connection details, enhancing flexibility and scalability in database management.

Arbutus Connectors

- **Drivers** are the components that process ODBC requests and return data to the application. If necessary, drivers modify an application's request into a form that is understood by the data source. The **Drivers** tab in the **ODBC Data Source Administrator** dialog box lists all drivers installed on your computer, including the name, version, company, file name, and file creation date of each driver.
- **Data sources** are the databases of files accessed by a driver and are identified by a data source name (DSN). You use the ODBC Data Source Administrator to add, configure, and delete data sources from your system.

All ODBC connections require that a DSN be configured to support the connection. When a client application wants to access an ODBC-compliant database, it references the database using the DSN.

The types of DSNs are:

- **User DSN** – User DSNs are local to a computer and can be used only by the current user. They are registered in the HKEY_Current_USER registry subtree.
- **System DSN** – System DSNs are local to a computer rather than dedicated to a user. The system or any user with privileges can use a data source set up with a system DSN. System DSNs are registered in the HKEY_LOCAL_MACHINE registry subtree.
- **File DSN** – File DSNs are file-based sources that can be shared among all users who have the same drivers installed and therefore have access to the database. These data sources need not be dedicated to a user nor be local to a computer. File data source names are identified by a file name with a .dsn extension.

User and system data sources are collectively known as *machine* data sources because they are local to a computer.

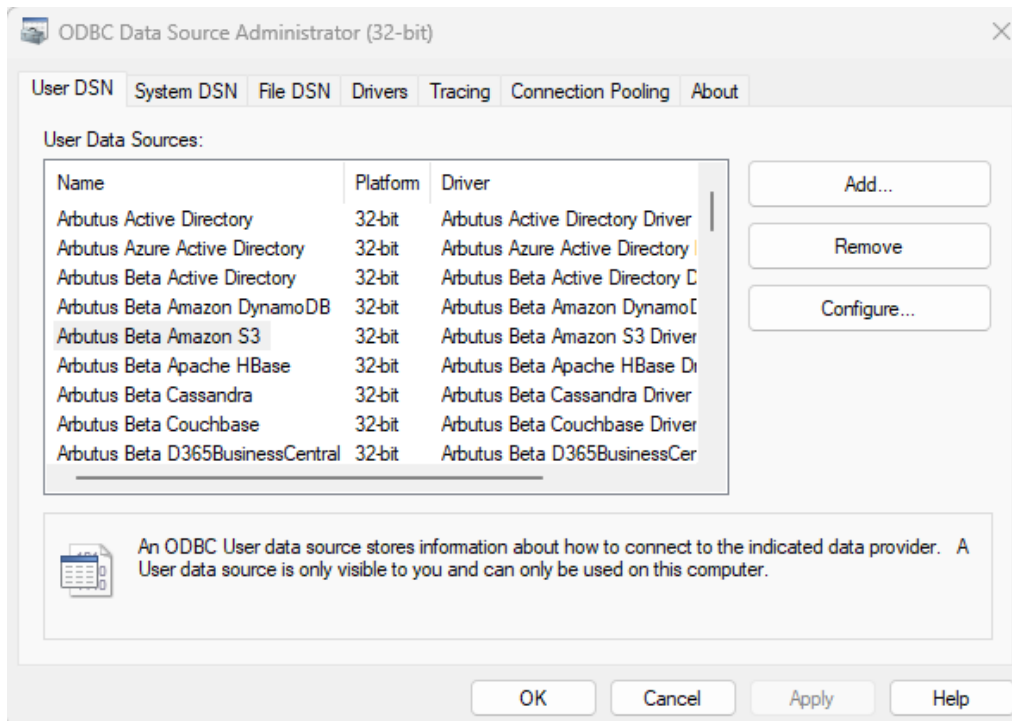
Each of these DSNs has a tab in the **ODBC Data Source Administrator** dialog.

The Arbutus ODBC Driver for Amazon S3 enables real-time access to Amazon S3 data, directly from any applications that support ODBC connectivity, the most widely supported interface for connecting applications with data.

Arbutus Connectors

Follow these steps to edit the DSN configuration and configure the Connector.

1. First open the **ODBC Data Source Administrator**.



2. Click the **User DSN** tab.

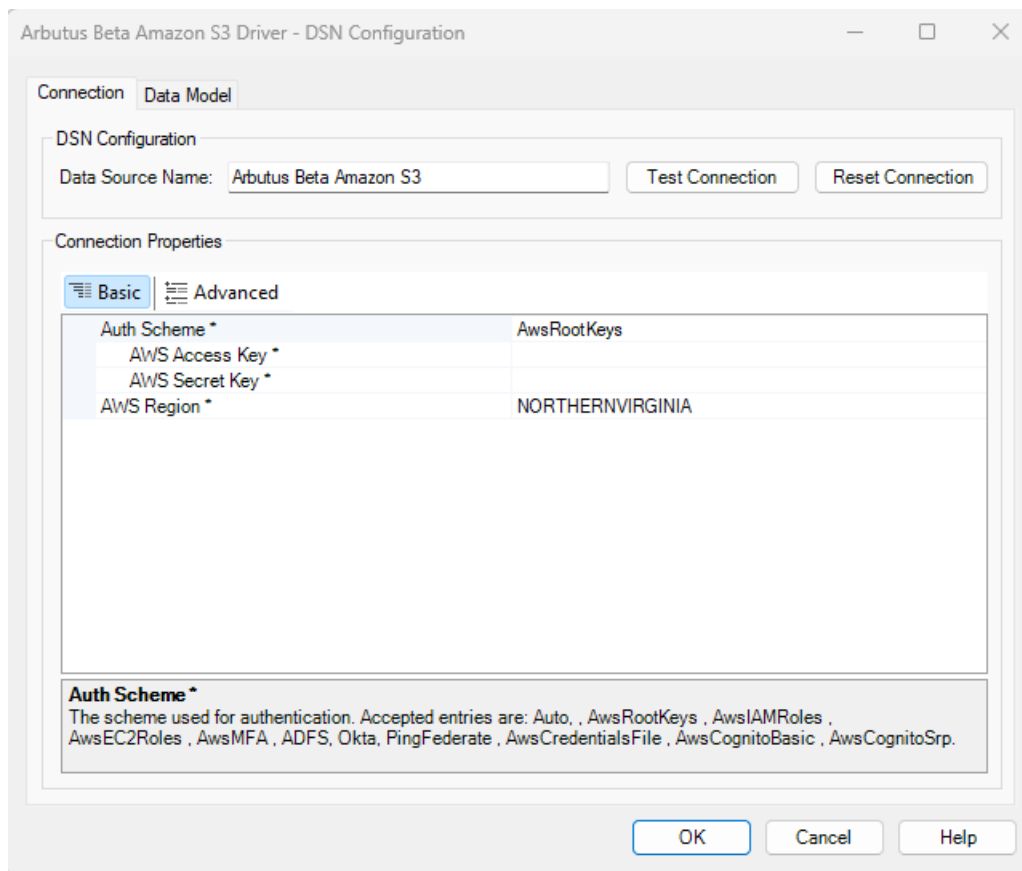
Selected data connectors are installed as **User DSN's** in Window's 32 Bit **ODBC Data Source Administrator**.

Also, each of the data connector's names is prefaced with Arbutus, for example, **Arbutus Amazon S3**.

3. Select the Arbutus Connector, in this case it is **Arbutus Amazon S3**.
4. Click **Configure**.

Arbutus Connectors

This opens the **Arbutus Amazon S3 Driver – DSN Configuration** dialog.



E. Editing the DSN properties – the Basic and Advanced tabs

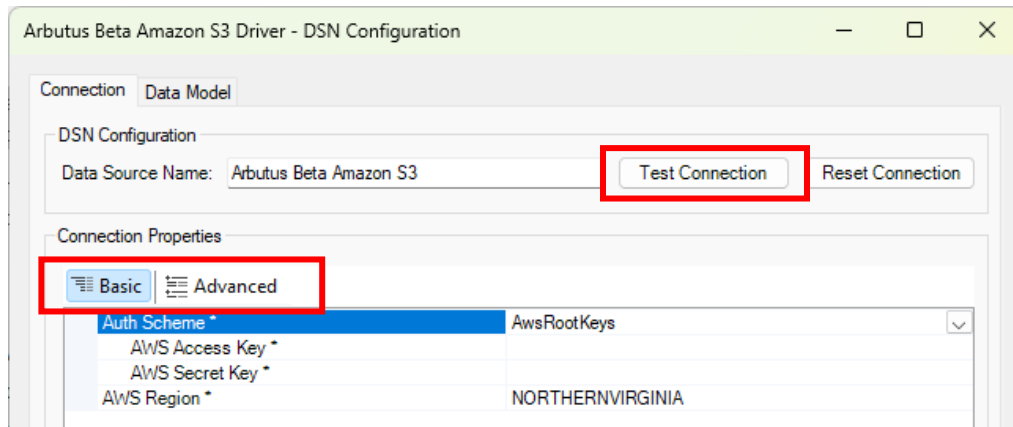
With the DSN Configuration dialog open, the next step is to edit the DSN properties, where necessary, in the **Basic** and **Advanced** tabs. For example, editing the **AWS Region** property (per screenshot below) to match the region of your Amazon Web Services.

E1. Editing the DSN properties in the Basic tab

The properties listed in the **Basic** tab are typically the ones that are most commonly used, and as such are designed to be more user-friendly and straightforward, allowing you to quickly make changes without needing in-depth technical knowledge.

Arbutus Connectors

Once you have completed editing the properties in the **Basic** tab, you can go ahead and try testing the connection to the Amazon S3 system by clicking the **Test Connection** button, as highlighted in the screenshot below.



In the **Basic** tab, there are **two** main properties to review:

1. **Auth Scheme** – click the dropdown to select from the list the appropriate scheme used for authentication. The options available for selection are as follows:
 - **AwsRootKeys** – select this if you are using the **root account credentials** for authentication. This method involves using the access key and secret key associated with your AWS root account.

However, it is **important to note** that using root credentials is generally discouraged for regular operations due to security risks. It's recommended to use IAM roles or temporary credentials for better security practices.

Selecting **AwsRootKeys** requires you to specify the following:

- **AWS Access Key** – this is your AWS account access key. This value is accessible from your AWS security credentials page.

Arbutus Connectors

To access this value, follow these steps:

- a. Sign into the AWS Management console with the credentials for your root account.
 - b. Select your account name or number and select **My Security Credentials** in the menu that is displayed.
 - c. Click **Continue to Security Credentials** and expand the **Access Keys** section to manage or create root account access keys.
- [AWS Secret Key](#) - Your AWS account secret key. This value is accessible from your AWS security credentials page.

See the steps listed above on how to access this value.

The default value is **AwsRootKeys**.

- [AwsIAMRoles](#) – select this if you are using **AWS Identity and Access Management (IAM) roles** for authentication. This method allows your application to authenticate to Amazon S3 using IAM roles assigned to AWS resources, such as EC2 instances or Lambda functions.

Selecting **AwsIAMRoles** requires you to specify the following:

- [AWS Access Key](#) – for more information, please see this same property listed and described above (under the **AwsRootKeys** section).
- [AWS Secret Key](#) - for more information, please see this same property listed and described above (under the **AwsRootKeys** section).

Arbutus Connectors

- [AWS Role ARN](#) – this is the Amazon Resource Name (ARN) of the role to use when authenticating.

When authenticating outside of AWS, it is common to use a Role for authentication instead of your direct AWS account credentials. Entering the **AWS Role ARN** will cause the ODBC Driver for Amazon S3 to perform a role based authentication instead of using the **AWS Access Key** (see above) and **AWS Secret Key (see above)** directly. The **AWS Access Key** and **AWS Secret Key** must still be specified to perform this authentication. You cannot use the credentials of an AWS root user when setting **Role ARN**. The **AWS Access Key** and **AWS Secret Key** must be those of an IAM user.

- [AWS External Id](#) – this is a unique identifier that might be required when you assume a role in another account.
- [AwsEC2Roles](#) – select this if you are using **AWS EC2 instance roles** for authentication. This method allows your application to authenticate to Amazon S3 using the IAM role assigned to an EC2 instance.
- [AwsMFA](#) – select this if you are using **Multi-Factor Authentication (MFA)** for added security. This method requires an additional authentication factor, typically a one-time password (OTP) generated by an MFA device, along with your standard AWS credentials.

Arbutus Connectors

Selecting **AwsMFA** requires you to specify the following:

- **MFA Serial Number** – this is the serial number of the MFA device if one is being used.

You can find the device for an IAM user by going to the AWS Management Console and viewing the user's security credentials.

- **MFA Token** – this is the temporary token available from your MFA device.

If MFA is required, this value will be used along with the **MFA Serial Number** to retrieve temporary credentials to login. The temporary credentials available from AWS will only last up to 1 hour by default (see **Temporary Token Duration** below). Once the time is up, the connection must be updated to specify a new MFA token so that new credentials may be obtained.

- **Temporary Token Duration** – this is the amount of time (in seconds) a temporary token will last.

Temporary tokens are used with both MFA and Role based authentication. Temporary tokens will eventually time out, at which time a new temporary token must be obtained. For situations where MFA is not used, this is not a big deal. The ODBC Driver for Amazon S3 will internally request a new temporary token once the temporary token has expired.

However, for MFA required connection, a new **MFA Token** (see above) must be specified in the connection to retrieve a new temporary token. This is a more intrusive issue since it requires an update to the connection by the user. The maximum and minimum that can be specified will depend largely on the connection being used.

Arbutus Connectors

For Role based authentication, the minimum duration is 900 seconds (15 minutes) while the maximum is 3600 (1 hour). Even if MFA is used with role based authentication, 3600 is still the maximum.

For MFA authentication by itself (using an IAM User or root user), the minimum is 900 seconds (15 minutes), the maximum is 129600 (36 hours).

The default value is **3600** seconds.

- [AwsCredentialsFile](#) – select this if you are using a **credentials file** for authentication. This method involves storing your AWS access key and secret key in a file, which the driver can then use to authenticate.
- [OKTA](#) – select this if you are using **Okta** for authentication. This method allows you to leverage Okta's identity and access management capabilities, including single sign-on (SSO) and multi-factor authentication (MFA).

Note:

Okta is a widely used cloud-based identity and access management (IAM) service that provides secure identity management and Single Sign-On (SSO) solutions. It helps organizations manage user authentication and authorization across various applications and services, ensuring secure access and streamlined user experience.

Selecting **OKTA** requires you to specify the following:

- [User](#) – this is the identity provider (IDP) user used to authenticate the IDP (OKTA) via SSO. This refers to the user credentials that are used to authenticate with the Identity Provider (IDP) through Single Sign-On.

The IDP user is the user account that exists within the Identity Provider, i.e. OKTA.

Arbutus Connectors

The IDP is responsible for verifying the identity of the user.

Together with **Password** (see below), this field is used to authenticate in SSO connections against the Amazon S3 server.

- **Password** – this is the password used to authenticate the IDP user (see above for more information) user via SSO. The **User** (see above) and **Password** are together used in SSO connections to authenticate with the server.
- **SSO Login URL** – this is the identity provider's login URL.
- **SSO Properties** - additional properties required to connect to the identity provider in a semicolon-separated list. **SSO Properties** is used in conjunction with the **AWS Role ARN** and **AWS Principal ARN**.

Shown below is an example of a semi-colon separated list:

```
Okta; AWSRegion=Ireland;  
User=user@abc_company.com;  
Password=CH8WerW121235647iCa6;  
SSOLoginURL='https:// the-identity-provider's-login-URL';  
SSOProperties='ApiToken=01230GGG2ceAnm_tPAf4MhiM  
ELXZ0L0N1pAYrO1VR-hGQSf;';  
AWSRoleArn=arn:aws:iam::1234:role/Okta_SSO;  
AWSPrincipalARN=arn:aws:iam::1234:saml-  
provider/OktaProvider;
```

Arbutus Connectors

- [Use Lake Formation](#) - this is a True/False selection. Select the appropriate value, based on following determination:
 - When this property is set to true, **AWS Lake Formation** service will be used to retrieve temporary credentials, which enforce access policies against the user based on the configured Identity Access Management (IAM) role.

The service can be used when authenticating through OKTA, ADFS, AzureAD, PingFederate, while providing a SAML (Single Assertion Markup Language) assertion – a SAML assertion is the message that tells a service provider that a user is signed in.

The default value is **False**.

- [Temporary Token Duration](#) - for more information, please see this same property listed and described above (under the **AwsMFA** section).
- [TemporaryCredentials](#) – select this when you need to use temporary security credentials to access Amazon S3. For example, when you need temporary (short-term) access to resources without using long-term credentials.

Selecting **TemporaryCredentials** requires you to specify the following:

- [AWS Access Key](#) - for more information, please see this same property listed and described above (under the **AwsRootKeys** section).
- [AWS Secret Key](#) - for more information, please see this same property listed and described above (under the **AwsRootKeys** section).

Arbutus Connectors

- [AWS Session Token](#) – this is your AWS session token. This value can be retrieved in different ways. See [this link](#) for more information.
- [PingFederate](#) – select this when you need to use PingFederate for authentication. As an example, PingFederate is typically useful when you want to enable single sign-on (SSO) capabilities, allowing users to authenticate once and gain access to multiple systems without needing to log back in again.

Note:

[PingFederate is a highly regarded enterprise federation server that specializes in user authentication and providing standardized single sign-on \(SSO\) solutions.](#)

Selecting **PingFederate** requires you to specify the following:

- [User](#) - for more information, please see this same property listed and described above (under the **OKTA** section).
- [Password](#) - for more information, please see this same property listed and described above (under the **OKTA** section).
- [SSO Login URL](#) - For more information, please see this same property listed and described above (under the **OKTA** section).
- [SSO Properties](#) - for more information, please see this same property listed and described above (under the **OKTA** section).
- [SSO Exchange URL](#) – this is the URL used for consuming the SAML response and exchanging it for service specific credentials.

Arbutus Connectors

The ODBC Driver for Amazon S3 will use the URL specified here to consume a SAML response and exchange it for service specific credentials. The retrieved credentials are the final piece during the SSO connection that are used to communicate with Amazon S3.

- [Use Lake Formation](#) - for more information, please see this same property listed and described above (under the **OKTA** section).
- [AWS Principal ARN](#) – this is the ARN of the SAML Identity provider in your AWS account.

ARN = **A**mazons **R**esource **N**ame. This is a unique identifier used to specify AWS resources. It identifies the specific AWS entity (such as an IAM user, role, or service) that is allowed or denied access to a resource.

SAML = **S**ingle **A**ssertion **M**arkup **L**anguage assertion – a SAML assertion is the message that tells a service provider that a user is signed in.

- [AwsCognitoBasic](#) – select this when you need to use Amazon Cognito for authentication. As an example, AwsCognitoBasic is typically useful when you want to manage sign-up, sign-in, and access control using Amazon Cognito. Another useful scenario is when you need to authenticate users through social identify providers (like Google, Facebook).

Arbutus Connectors

Selecting **AwsCognitoBasic** requires you to specify the following:

- [AWS Cognito Region](#) - click the dropdown to select from the list the hosting region for AWS Cognito. The regions available for selection are as follows:

OHIO, NORTHERNVIRGINIA, NORTHERNCALIFORNIA, OREGON, CAPETOWN, HONGKONG, HYDERABAD, JAKARTA, MALAYSIA, MELBOURNE, MUMBAI, OSAKA, SEOUL, SINGAPORE, SYDNEY, TOKYO, CENTRAL, CALGARY, BEIJING, NINGXIA, FRANKFURT, IRELAND, LONDON, MILAN, PARIS, SPAIN, STOCKHOLM, ZURICH, TELAVIV, BAHRAIN, UAE, SAOPAULO, GOVCLOUDEAST, GOVCLOUDWEST, ISOLATEDUSEAST, ISOLATEDUSEASTB, ISOLATEDUSWEST, ISOLATEDEUWEST

The default value may show as **NORTHERNVIRGINIA**.

- [AWS User Pool Id](#) – this is the User Pool Id. You can find this in AWS Cognito -> Manage User Pools -> select your user pool -> General settings -> Pool Id.
- [AWS User Pool Client App Id](#) – this is the User Pool Client App Id. You can find this in AWS Cognito -> Manage Identity Pools -> select your user pool -> General settings -> App clients -> App client Id.
- [AWS Identity Pool Id](#) – this is the Identity Pool Id. You can find this in AWS Cognito -> Manage Identity Pools -> select your identity pool -> Edit identity pool -> Identity Pool Id.
- [AWS User Pool Client App Secret](#) (optional) – this is the User Pool Client App Secret. You can find this in AWS Cognito -> Manage Identity Pools -> select your user pool -> General settings -> App clients -> App client secret.

Arbutus Connectors

- [AwsCognitoSrp](#) – select this when you need to use the Secure Remote Password (SRP) protocol for authentication with Amazon Cognito. This is typically useful in a scenario where you want to authenticate users securely using their username and password managed by Amazon Cognito. Notably, SRP is a secure password-based authentication protocol that ensures passwords are never sent over the network, providing an additional layer of security.

Selecting **AwsCognitoSrp** requires you to specify the following:

- [AWS Cognito Region](#) - for more information, please see this same property listed and described above (under the **AwsCognitoBasic** section).
- [AWS User Pool Id](#) - for more information, please see this same property listed and described above (under the **AwsCognitoBasic** section).
- [AWS User Pool Client App Id](#) - for more information, please see this same property listed and described above (under the **AwsCognitoBasic** section).
- [AWS Identity Pool Id](#) - for more information, please see this same property listed and described above (under the **AwsCognitoBasic** section).
- [AWS User Pool Client App Secret](#) - for more information, please see this same property listed and described above (under the **AwsCognitoBasic** section).

Arbutus Connectors

- **ADFS** – select this when you need to use Active Directory Federation Services (ADFS) for authentication. As an example, this is typically useful when you want to enable single sign-on (SSO) capabilities, allowing users to authenticate once and gain access to multiple systems without needing to log back in again.

Selecting **ADFS** requires you to specify the following:

- **User** – For more information, please see this same property listed and described above (under the **AwsCognitoBasic** section).
- **Password** – for more information, please see this same property listed and described above (under the **PingFederate** section).
- **SSO Login URL** – for more information, please see this same property listed and described above (under the **PingFederate** section).
- **SSO Properties** – for more information, please see this same property listed and described above (under the **PingFederate** section).
- **Use Lake Formation** – for more information, please see this same property listed and described above (under the **OKTA** section).
- **Temporary Token Duration** - for more information, please see this same property listed and described above (under the **OKTA** section).

Arbutus Connectors

2. **AWS Region** – click the dropdown to select from the list the hosting region for your Amazon Web Services. The regions available for selection are as follows:

OHIO, NORTHERNVIRGINIA, NORTHERNCALIFORNIA, OREGON, CAPETOWN, HONGKONG, HYDERABAD, JAKARTA, MALAYSIA, MELBOURNE, MUMBAI, OSAKA, SEOUL, SINGAPORE, SYDNEY, TOKYO, CENTRAL, CALGARY, BEIJING, NINGXIA, FRANKFURT, IRELAND, LONDON, MILAN, PARIS, SPAIN, STOCKHOLM, ZURICH, TELAVIV, BAHRAIN, UAE, SAOPAULO, GOVCLOUDEAST, GOVCLOUDWEST, ISOLATEDUSEAST, ISOLATEDUSEASTB, ISOLATEDUSWEST, ISOLATEDEUWEST

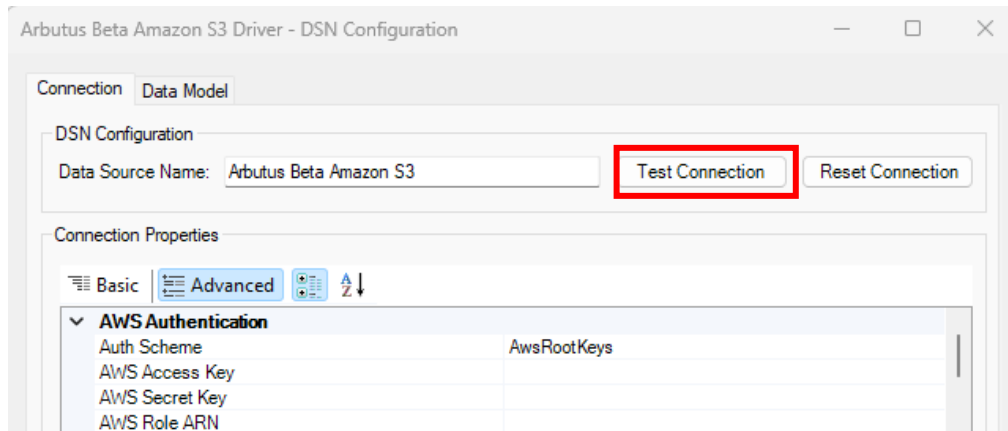
The default value may show as **NORTHERNVIRGINIA**.

E2. Editing the DSN properties in the **Advanced** tab

This tab includes more detailed and technical properties. It is intended for those users who need more control over the configuration and are comfortable with more complex options. The **Advanced** tab often includes properties that can fine-tune the behaviour of the system feature.

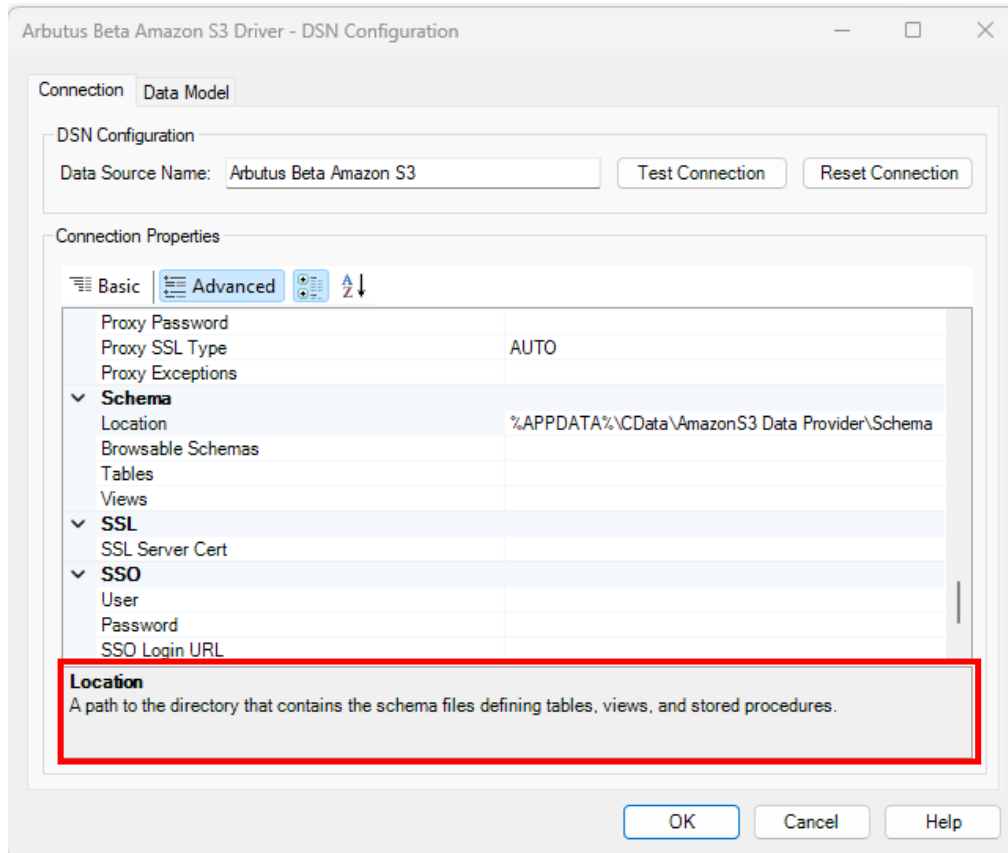
If you have already completed editing the properties in the **Basic** tab, as required, you do not necessarily need to also complete editing the properties in the **Advanced** tab. Instead, once you have completed editing the properties in the **Basic** tab, you may opt to proceed to testing the connection to the Amazon S3 system by clicking the **Test Connection** button.

Arbutus Connectors



There are a lot more properties included for editing in the **Advanced** tab.

However, it is useful to know that each property does provide a short description of it and as such serves as a guide in terms of what to edit and/or enter. This short description can be seen at the bottom of the **DSN Configuration** dialog box, as seen in the screenshot below.



Arbutus Connectors

If it is deemed necessary to complete some/all the properties in the Advanced tab, it is recommended that you refer to the description shown for any of the properties being edited and/or entered.

If required, more information on the properties listed in the **Advanced** tab can also be provided.

F. Other questions and/or request for assistance

There may be times when you need to consult with the technical support team at Arbutus Software. If so, please send an email request to support@ArbutusSoftware.com.

For more information, please refer to the [CONTACT US](#) page on our website.