



# Ask the Expert: Understanding User Authentication

Course Notes

*Ask the Expert: Understanding User Authentication Course Notes* was developed by NULL. Additional contributions were made by NULL. Editing and production support was provided by the Curriculum Development and Support Department.

SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration. Other brand and product names are trademarks of their respective companies.

**Ask the Expert: Understanding User Authentication Course Notes**

Copyright © 2017 SAS Institute Inc. Cary, NC, USA. All rights reserved. Printed in the United States of America. 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, or otherwise, without the prior written permission of the publisher, SAS Institute Inc.

---

Book code E71139, course code ATEPAUA, prepared date 31Aug2017.

ATEPAUA\_001

# Table of Contents

To learn more.....	iv
<b>Chapter 1 Exploring Authentication Mechanisms in the SAS® 9.4 Platform.....</b>	<b>1-1</b>
1.1 Exploring Initial Authentication to the SAS Metadata Server.....	1-3
Demonstration: Exploring Initial Authentication to the SAS Metadata Server.....	1-21
1.2 Exploring Authentication to Processing Servers and Data Servers.....	1-22
Demonstration: Monitoring SAS Servers and Sessions in SAS Management Console .....	1-49
Demonstration: (Optional) Configuring Access to a Database in SAS Management Console .....	1-52

## To learn more...



For information about other courses in the curriculum, contact the SAS Education Division at 1-800-333-7660, or send e-mail to [training@sas.com](mailto:training@sas.com). You can also find this information on the web at <http://support.sas.com/training/> as well as in the Training Course Catalog.

For a list of SAS books (including e-books) that relate to the topics covered in this course notes, visit <https://www.sas.com/sas/books.html> or call 1-800-727-0025. US customers receive free shipping to US addresses.

# Chapter 1 Exploring Authentication Mechanisms in the SAS<sup>®</sup> 9.4 Platform

<b>1.1</b>	<b>Exploring Initial Authentication to the SAS Metadata Server .....</b>	<b>1-3</b>
	Demonstration: Exploring Initial Authentication to the SAS Metadata Server .....	1-21
<b>1.2</b>	<b>Exploring Authentication to Processing Servers and Data Servers .....</b>	<b>1-22</b>
	Demonstration: Monitoring SAS Servers and Sessions in SAS Management Console.....	1-49
	Demonstration: (Optional) Configuring Access to a Database in SAS Management Console .....	1-52



# 1.1 Exploring Initial Authentication to the SAS Metadata Server

## Objectives

- Explore authentication mechanisms that are used in the SAS 9.4 Platform.
- Review initial host authentication to the metadata server.
- Review Integrated Windows authentication (IWA).
- Review metadata identities.
- Identify key internal accounts.
- Explore internal authentication.

4

Copyright © SAS Institute Inc. All rights reserved.



## SAS 9.4 Authentication Mechanisms

*Authentication* is the process of verifying the identity of a person or process for security purposes.

<b>External</b>	<ul style="list-style-type: none"> <li>• Host authentication (credential-based)</li> <li>• Direct LDAP authentication</li> <li>• Integrated Windows authentication</li> <li>• Web authentication</li> </ul>
<b>Internal</b>	<ul style="list-style-type: none"> <li>• SAS internal authentication</li> <li>• SAS token authentication</li> </ul>

5

Copyright © SAS Institute Inc. All rights reserved.



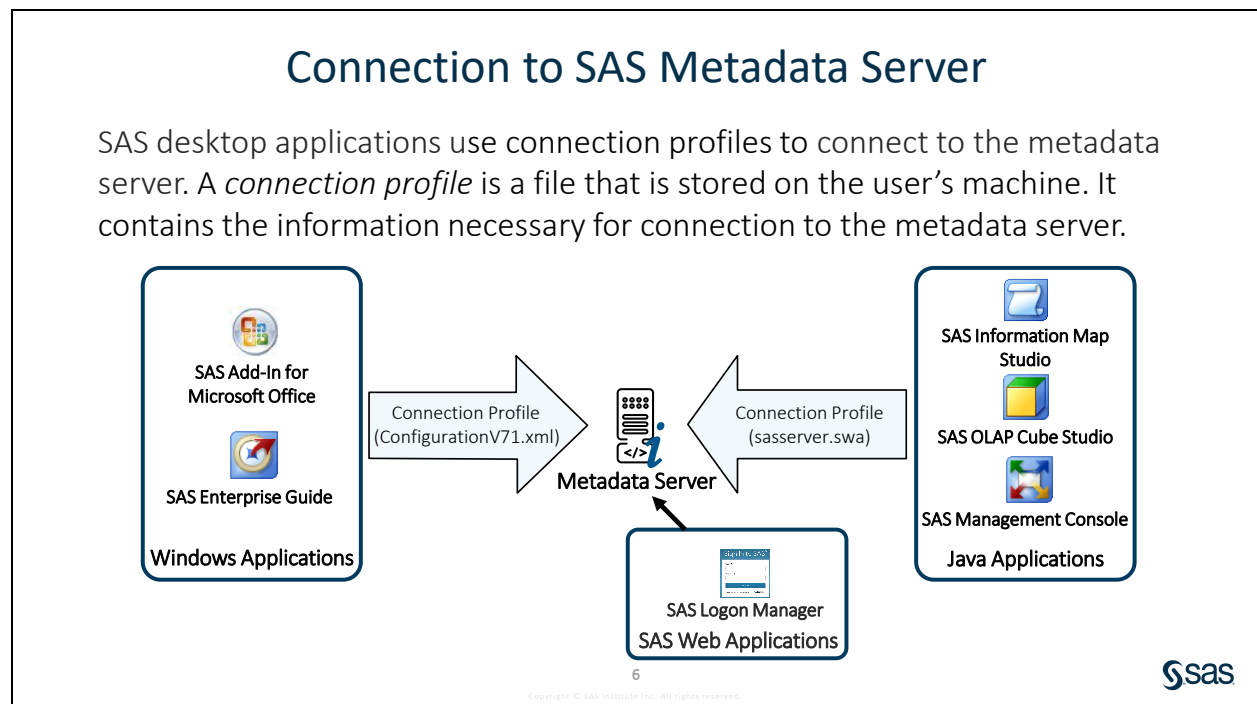
To provide authentication, the platform cooperates with systems such as the host environment, the web realm, and third-party databases.

A supporting feature of internal authentication mechanisms unifies the SAS realm and provides a degree of independence from your general computing environment.

Credential management provides single sign-on through the reuse of cached credentials or the retrieval of stored passwords.

Pluggable authentication modules (PAM) extend UNIX host authentication.

Trust relationships facilitate communication to the metadata server by permitting one privileged account to connect on behalf of other users (trusted user) or by accepting requests that use a proprietary protocol (trusted peer).

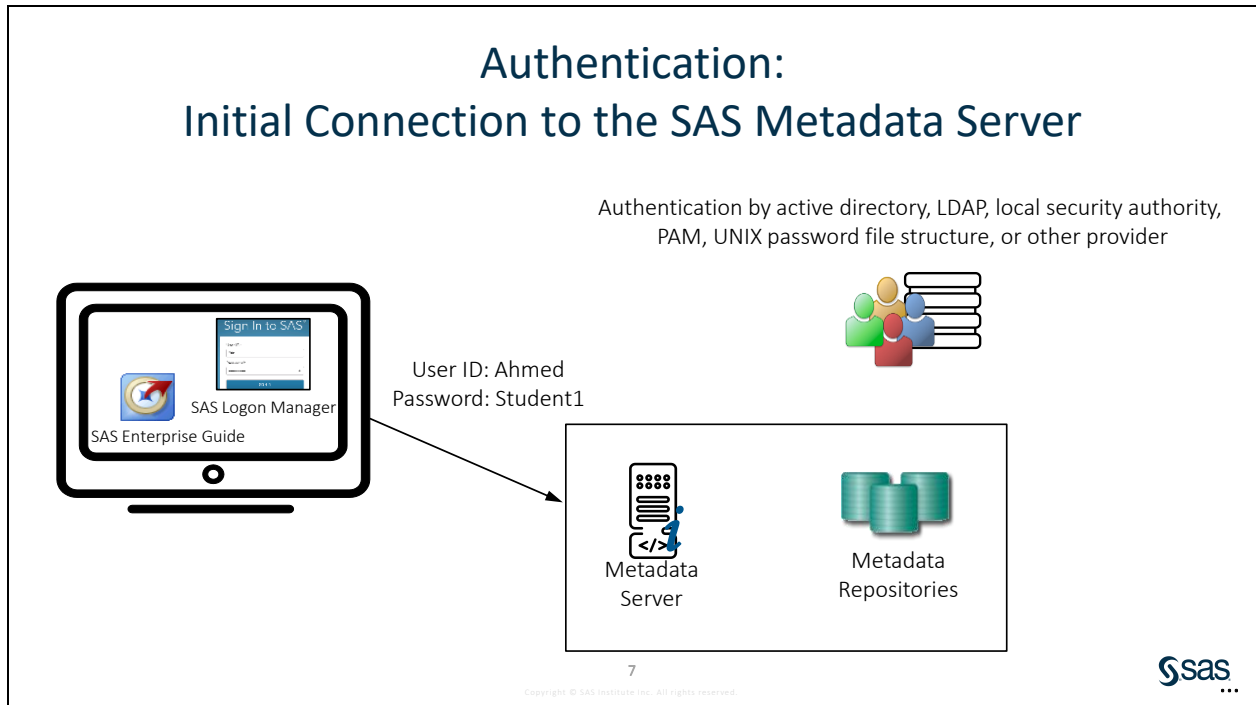


In most cases, users access and update metadata using SAS applications, including SAS Management Console, SAS Environment Manager, SAS Data Integration Studio, and SAS Enterprise Guide. Web-based applications need only a web browser. The connection profile is built into the web application.

You can also access and manage SAS Metadata through programmatic interfaces, including the METADATA and METALIB procedures, DATA step functions, and the batch tools for metadata management. The tools are documented in *SAS® 9.4 Intelligence Platform: System Administration Guide*.

Other parts of the SAS platform also communicate with the metadata server, including SAS spawners, SAS servers, and SAS middle-tier applications.

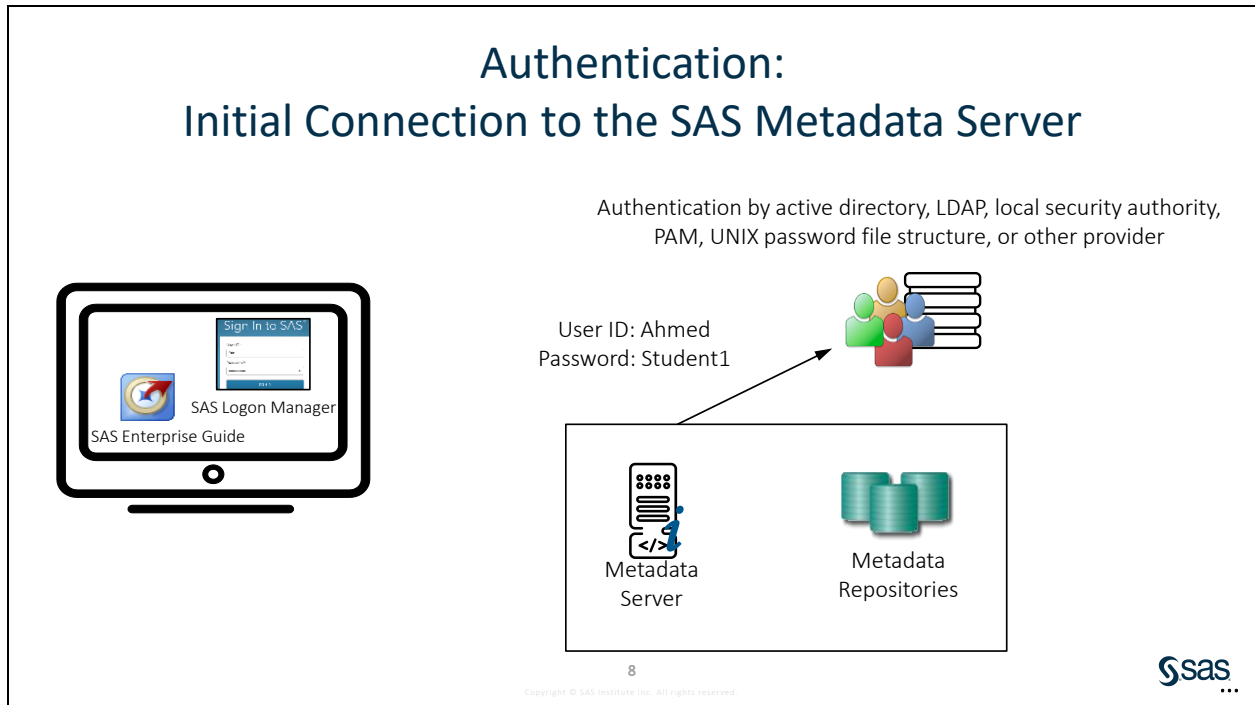




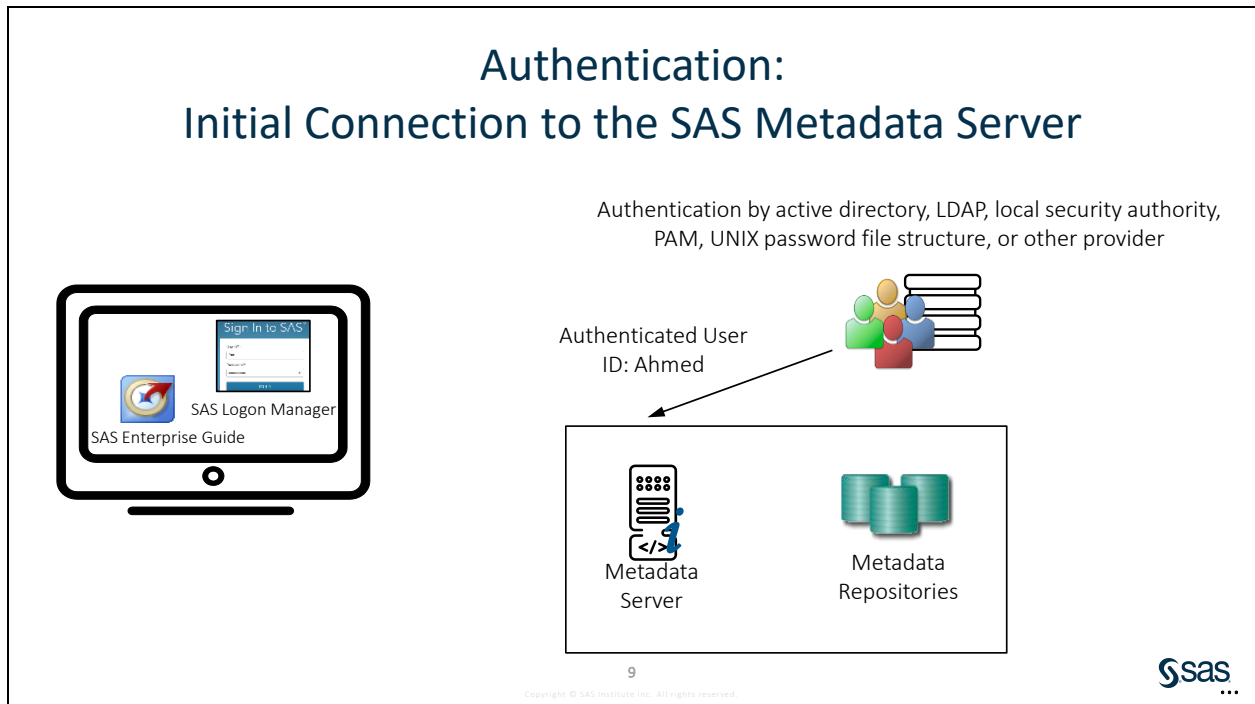
In the verification phase, the system ensures that the user is who he or she claims to be. For example, this credential-based host authentication method might be used:

1. The client prompts the user for an ID and password.
2. The user enters credentials that are known to the metadata server's host.
3. The client sends the credentials to the metadata server.

**Note:** An alternative to providing credentials is to use Integrated Windows Authentication.



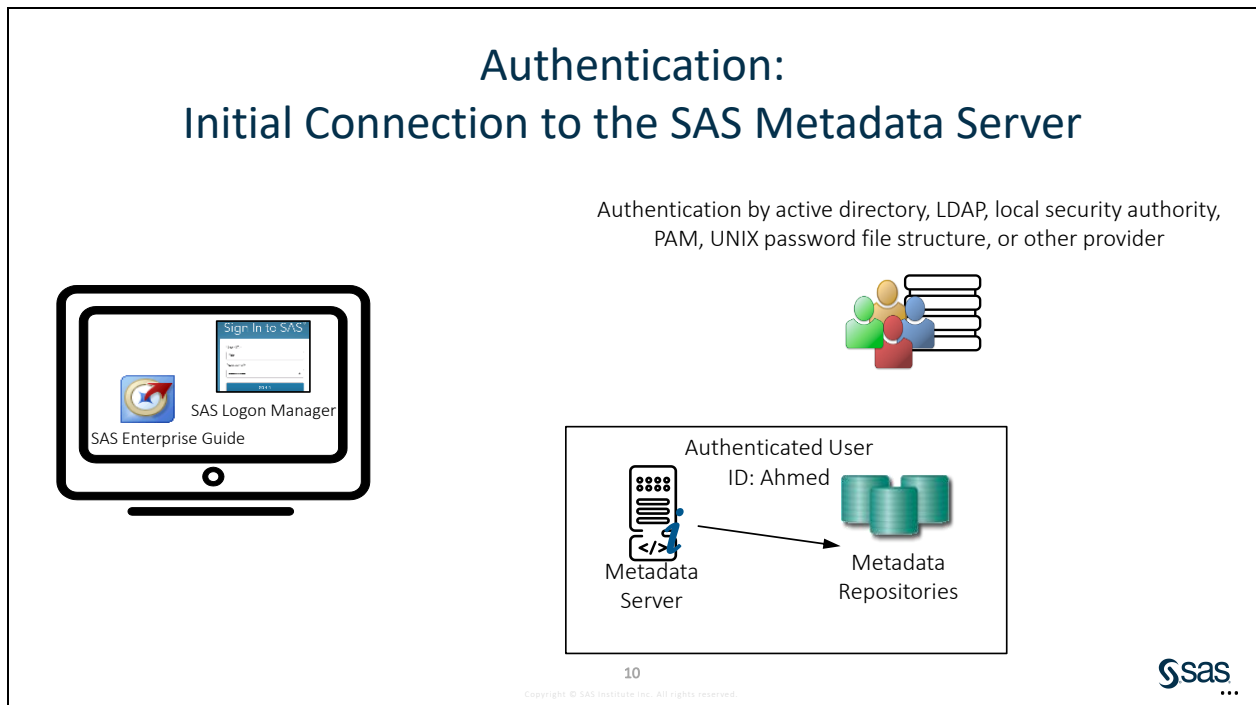
By default, the metadata server passes the credentials to its host. If the accounts are local, they are verified by the host. The host can also be configured to pass the authentication request to LDAP or the Microsoft Active Directory.



The authentication provider verifies that the credentials are valid and returns the fully qualified user ID to the metadata server.

**Note:** The authentication provider does not return the password to the metadata server.

**Note:** The form of the fully qualified user ID varies depending on the authentication provider.



In the SAS identity phase, the system resolves the authenticated user ID to a particular SAS identity. In this phase, SAS examines its copies of user IDs in an attempt to find one that matches the authenticated user ID. One of the following outcomes occurs:

- A matching user ID is found, so a connection is established under the owning identity. The owning identity is the user or group whose definition includes a login with the matching user ID.

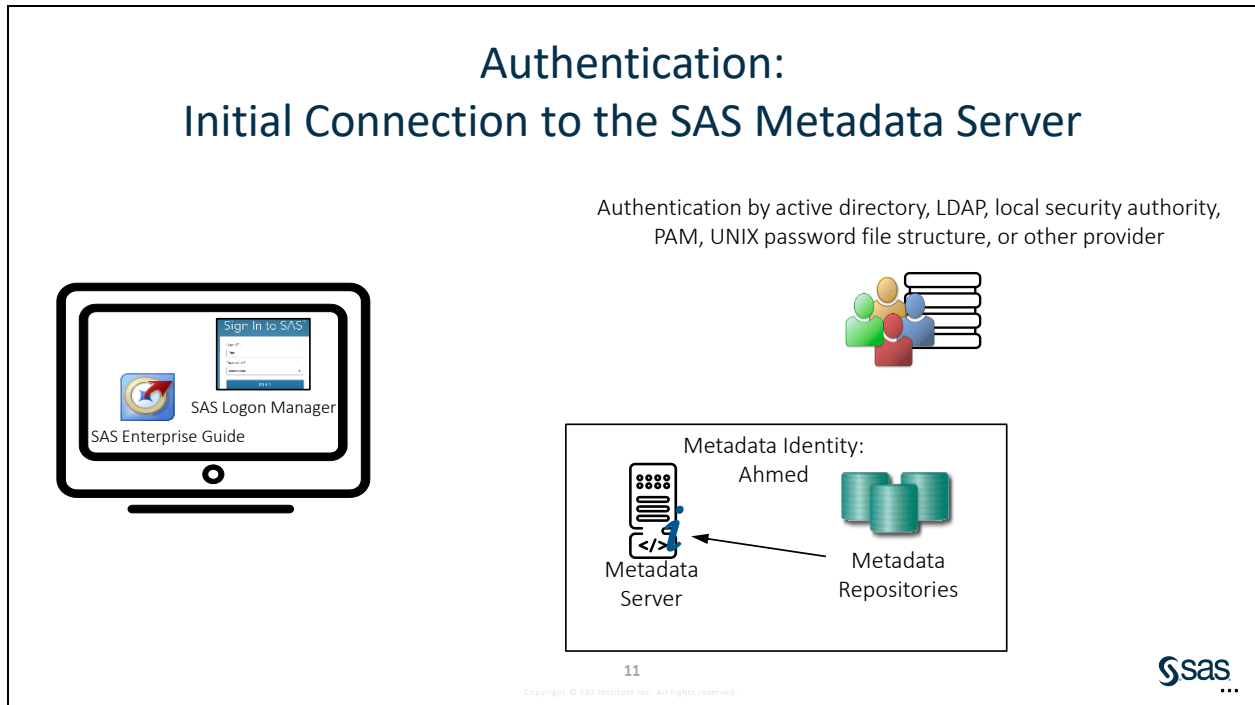
**Note:** Integrity constraints ensure that there is not more than one owning identity.

**Note:** Not all applications enable a group identity to log on.

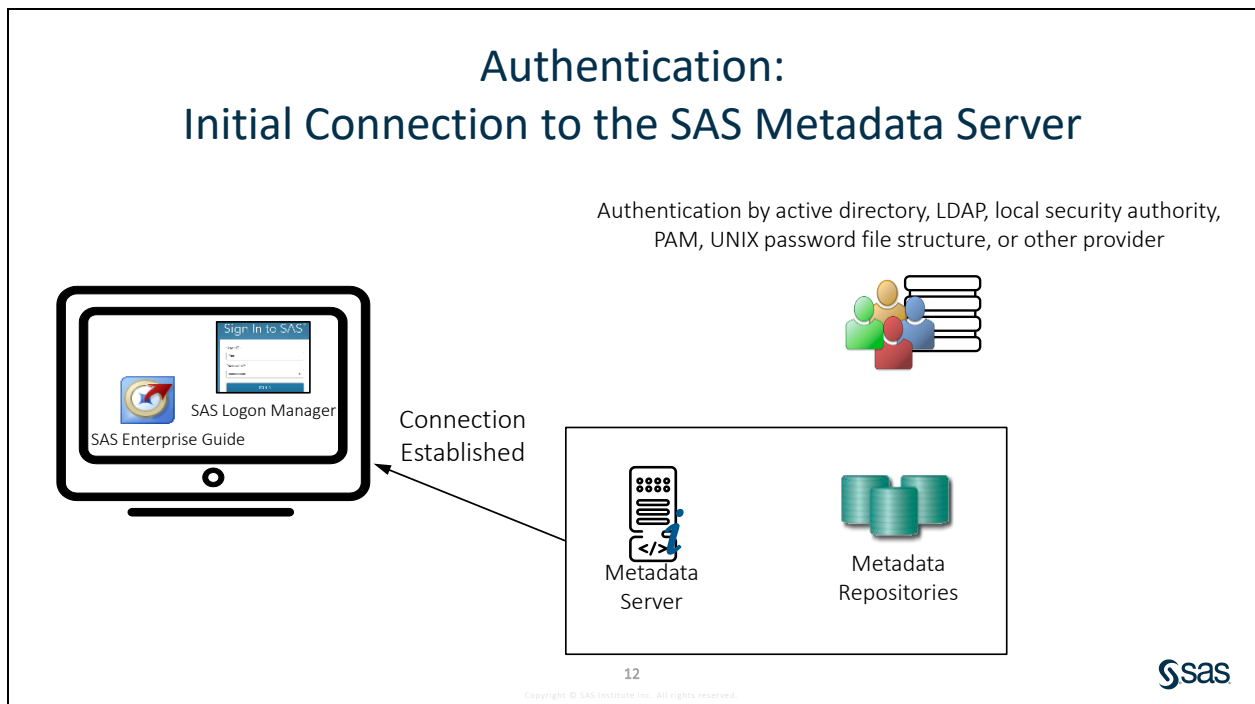
- No matching user ID is found, so a connection is established under the generic PUBLIC identity. In the metadata layer, the user is a PUBLIC-only user.

**Note:** The matching process expects the SAS copy of the user ID to be qualified (if it is a Windows user ID).

**Note:** Not all applications enable a group identity to log on.



The metadata server determines which metadata identity owns the user ID. Based on the metadata identity, the metadata server can determine what level of access Ahmed has to the metadata. Access to the metadata server is set in the repository ACT (access control template). Only users with ReadMetadata and WriteMetadata in the repository ACT, named Default ACT by default, are enabled to connect to the metadata server.



The metadata server sends a credential handle to the application so that when the application requests information from the metadata server, it can pass the handle. The metadata server then knows the metadata identity of the user.

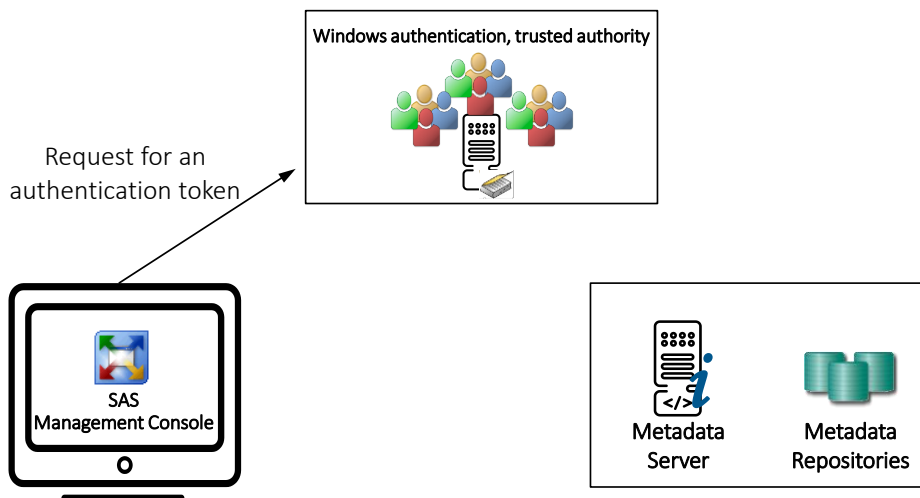
## SAS 9.4 Authentication Mechanisms

*Authentication* is the process of verifying the identity of a person or process for security purposes.

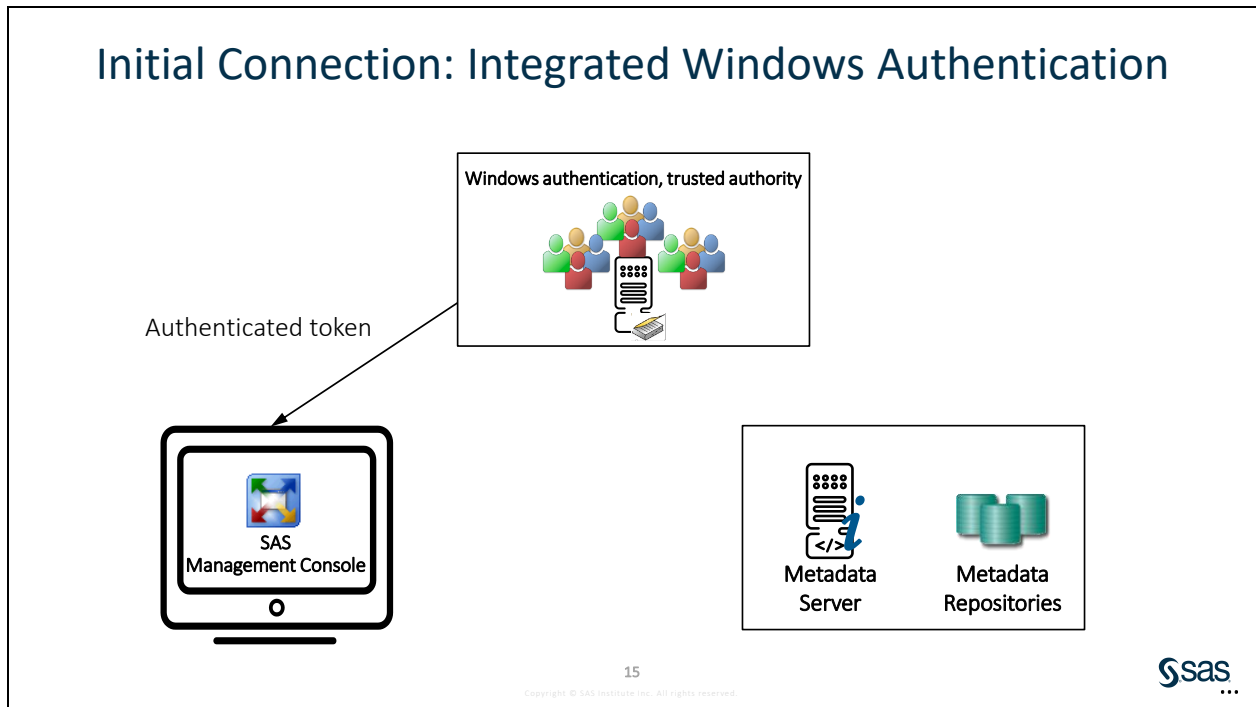
<b>External</b>	<ul style="list-style-type: none"> <li>• Host authentication (credential-based)</li> <li>• Direct LDAP authentication</li> <li>• Integrated Windows authentication</li> <li>• Web authentication</li> </ul>
<b>Internal</b>	<ul style="list-style-type: none"> <li>• SAS internal authentication</li> <li>• SAS token authentication</li> </ul>



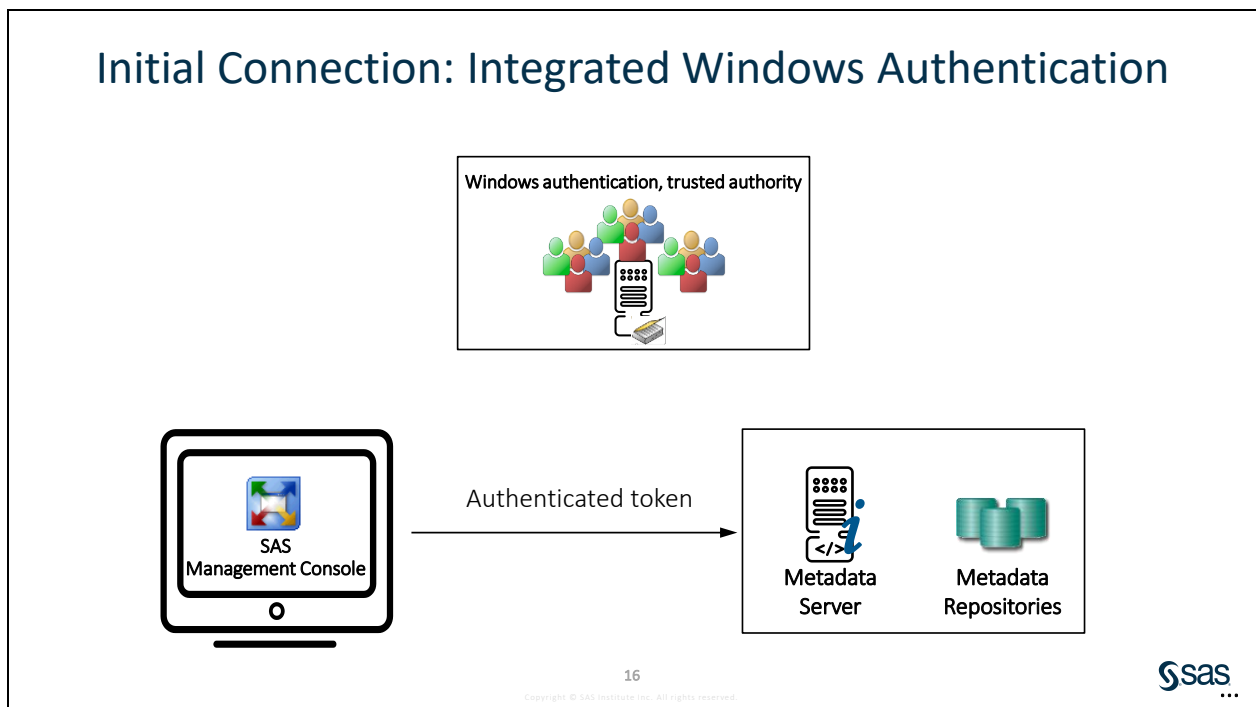
## Initial Connection: Integrated Windows Authentication



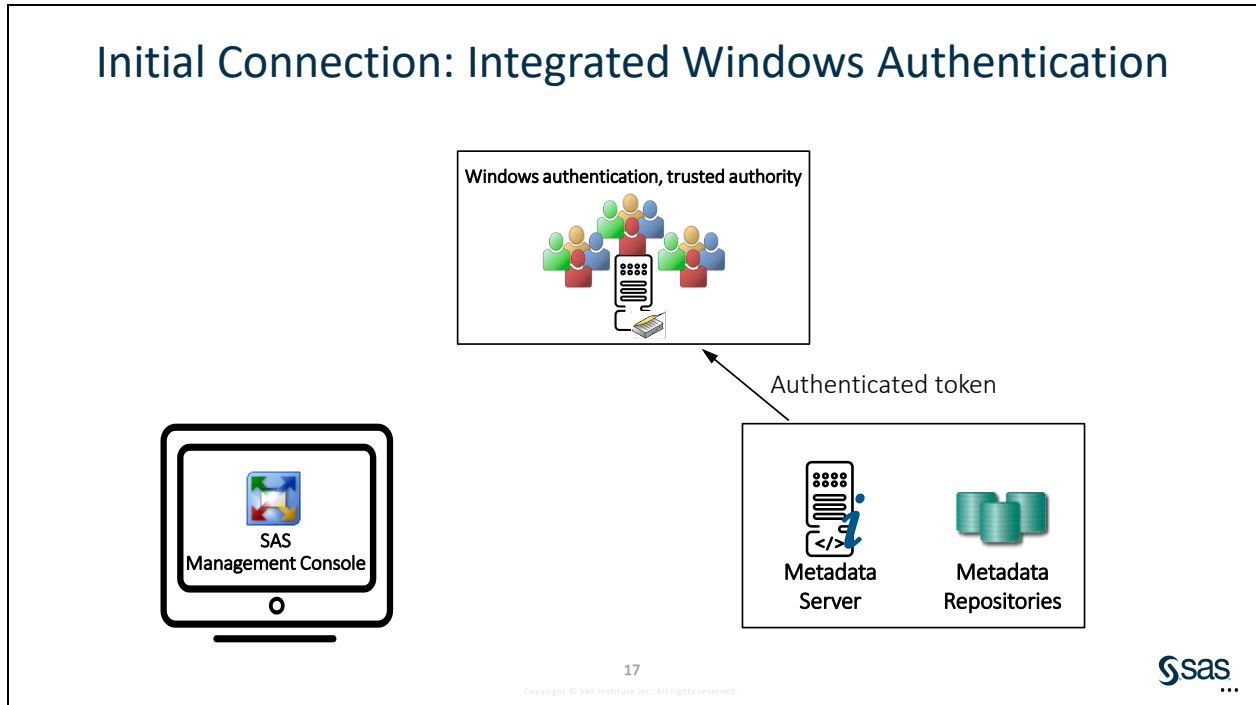
1. The client asks Windows for a token that represents the user who is currently logged on to the client computer.



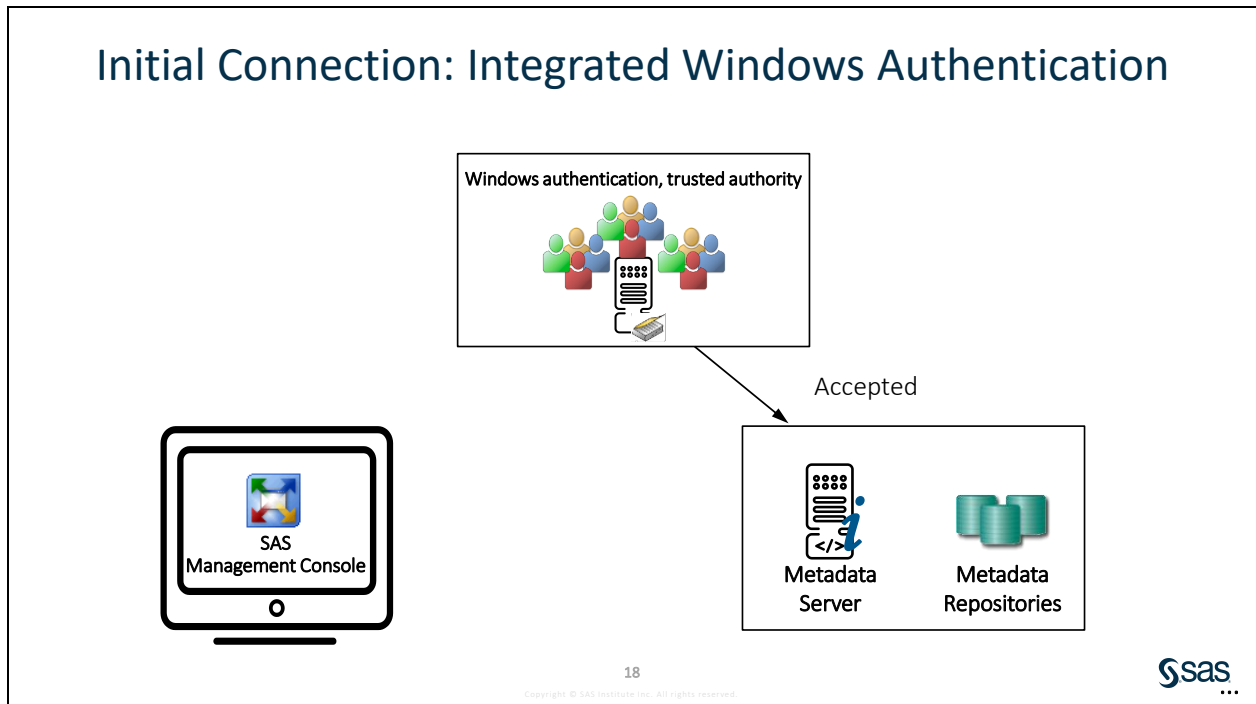
2. Windows provides the token to the client.



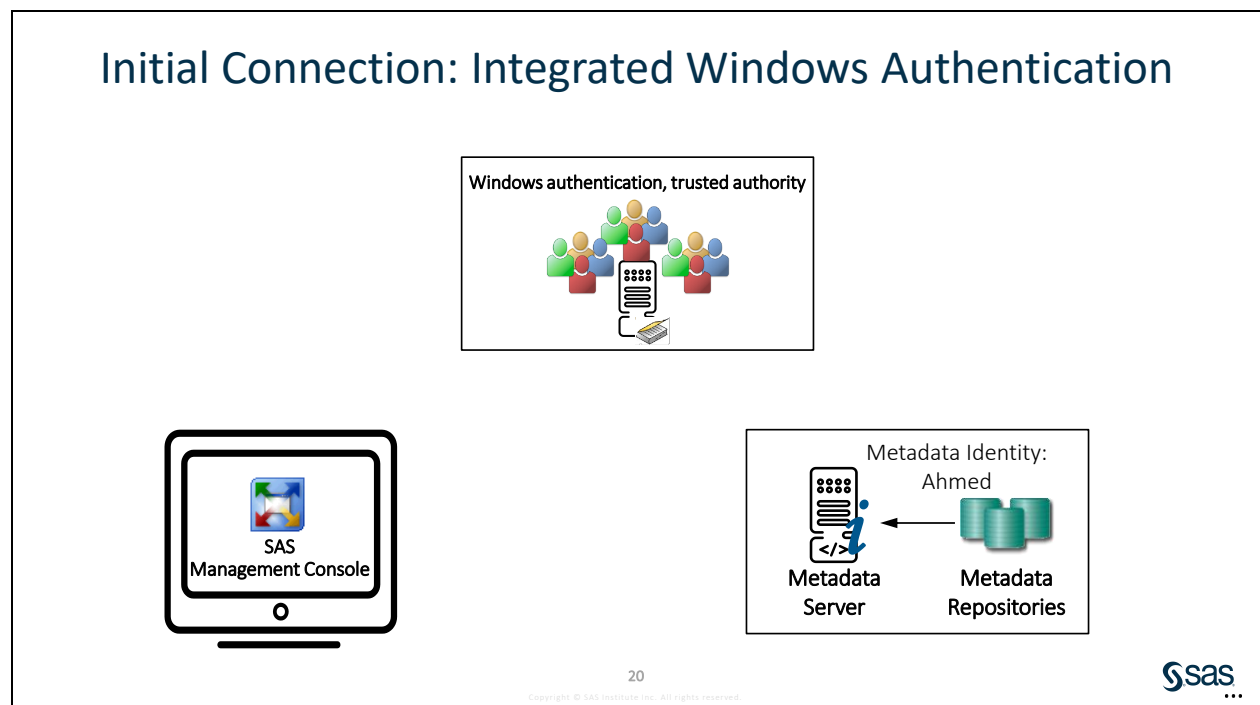
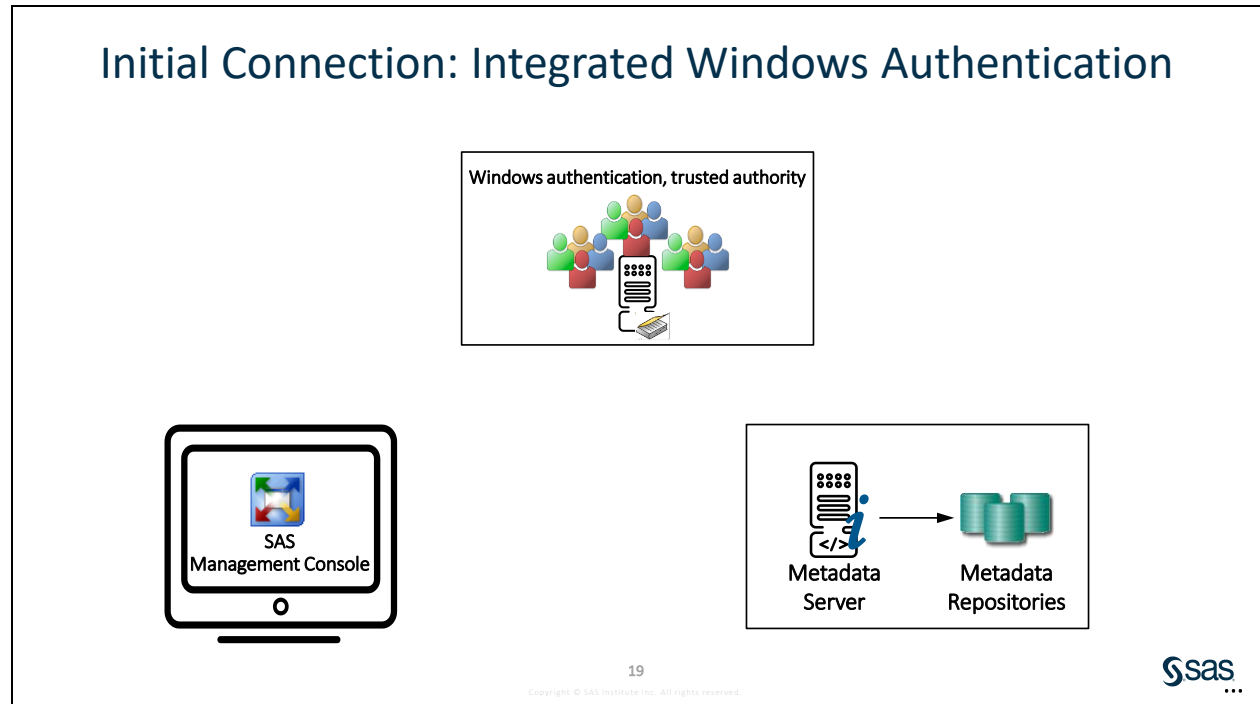
- The client sends the Windows token to the metadata server. Notice that only the token is sent. The user's password is not available to the metadata server.



- The metadata server sends the token back to Windows for verification.

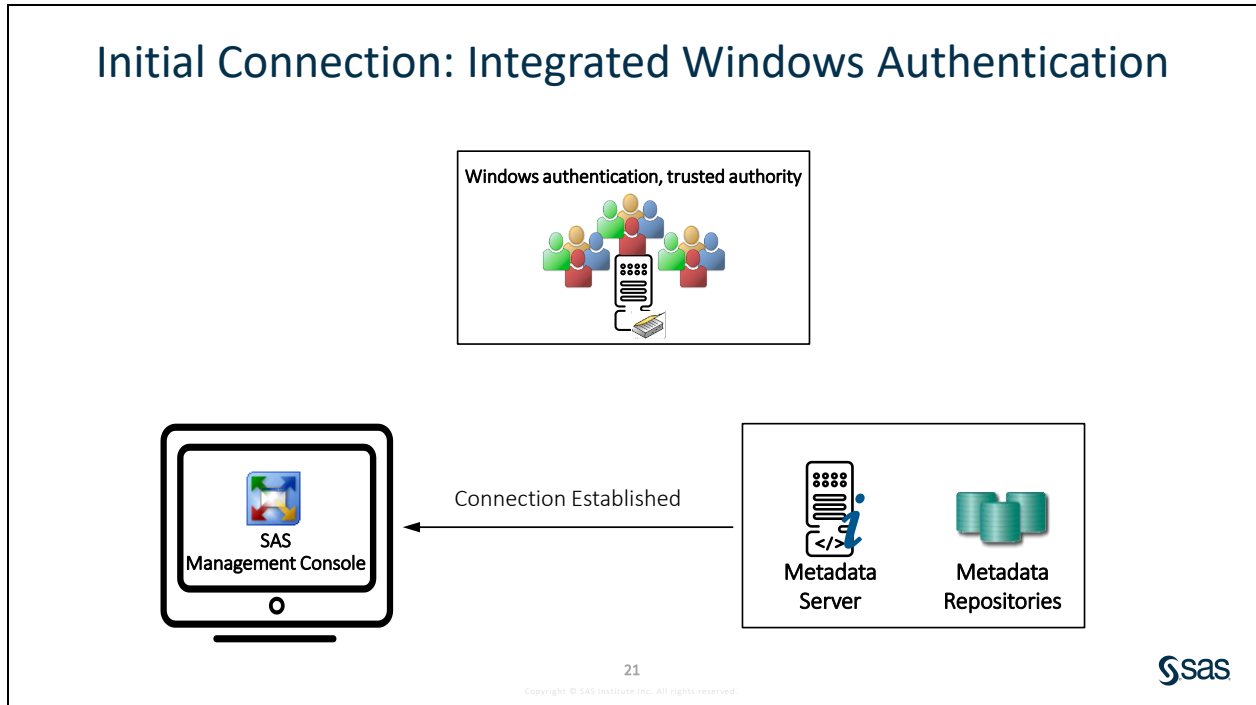


5. Windows tells the metadata server that the token is valid.





6. The metadata server identifies the user and verifies that the user was granted access to the metadata in the repository ACT.



7. The metadata server accepts the connection from the client.

**Note:** For initial connection to the metadata server, this represents the verification phase. The identification phase is essentially the same in all authentication models. After verification, the authenticated token includes the user ID. The metadata server searches its logons for a match. An inbound logon is still required.

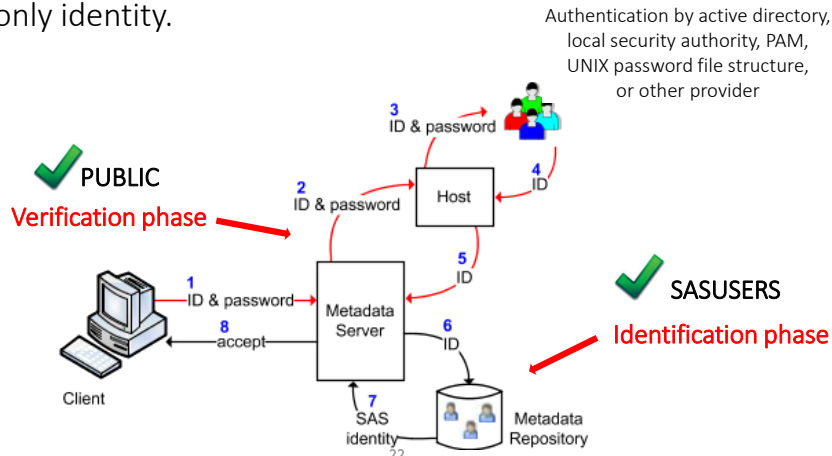
**Note:** There are limitations to IWA for servers on UNIX. To use IWA on UNIX platforms, consider the following:

- For SAS 9.4M1 on all platforms, you must purchase, install, and configure an additional third-party product (Quest Authentication Services 4.0).
- For SAS 9.4M2 on Linux platforms, you must ensure that a shared library that implements the GSSAPI with Kerberos 5 extensions is installed and configured to enable authentication against your Active Directory domain or Kerberos realm. Quest Authentication Services fulfills this requirement, as do the krb5 packages that are provided in supported operating system distributions and in various third-party solutions.
- When you use IWA on UNIX, only Kerberos connections are supported. (There is no support for NTLM on UNIX.) If you use IWA for a UNIX workspace server that makes outbound Kerberos requests, the service principal account in Active Directory must have the **trusted for delegation to all services** privilege.

For additional information about Integrated Windows Authentication, refer to *SAS® 9.4 Platform Intelligence: Security Administration Guide*.

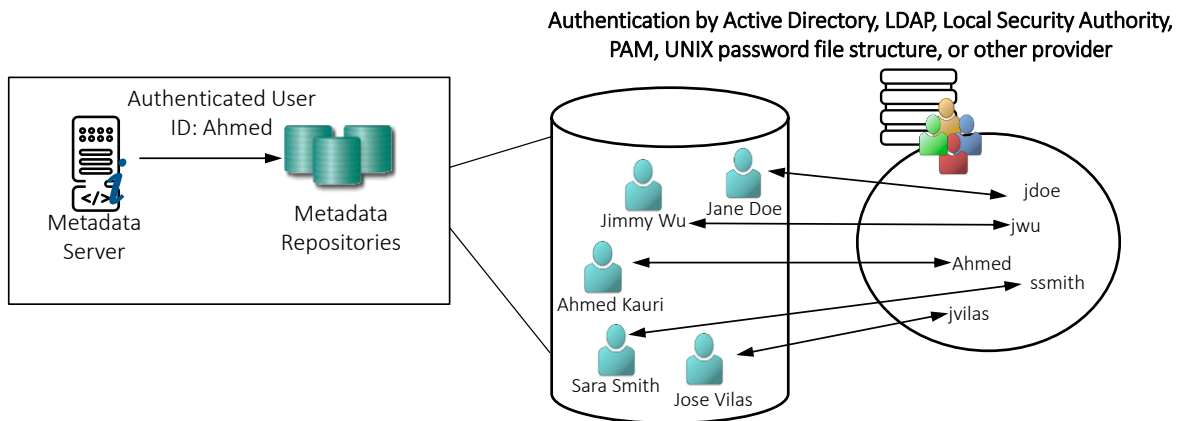
## Initial Connection to the SAS Metadata Server

Only the verification phase varies. The SAS identity phase is always the same. You need a well-formed user definition for each user who is not a PUBLIC-only identity.



## Metadata Identities

A user's metadata identity includes a copy of the external account that the user uses to log on to SAS applications.



In general, each SAS user has identity information in two distinct realms:

- In an authentication provider, the user has an account that can access the metadata server.
- In the SAS Metadata, the user has a definition that includes a copy of the account ID with which the user accesses the metadata server.

Coordination between these two realms establishes a unique SAS identity for each user. Each SAS identity is based on a match between the following two values:

- the account ID with which the user authenticates
- the account ID that is listed in the user's metadata definition

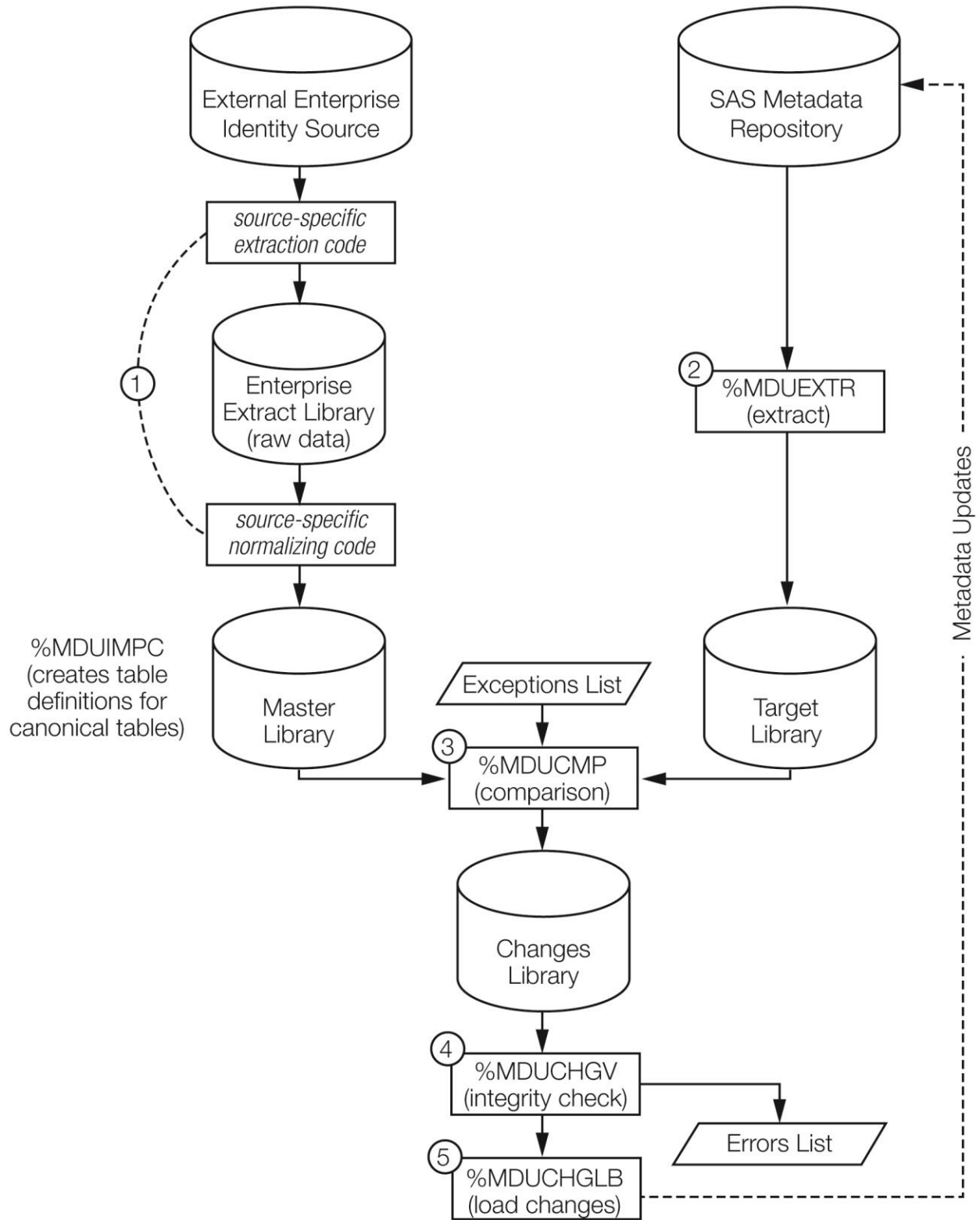
## Creating Users and Groups

Here are two ways to define user and group identities:

- manually, using the User Manager plug-in in SAS Management Console or in SAS Environment Manager Administration
- using the user import macros that are supplied by SAS to import identity information from an authentication provider


Documentation: *SAS® 9.4 Intelligence Platform: Security Administration Guide*  
⇒ Appendix ⇒ User Import Macros

There are other programmatic methods that can be used to create metadata identities.



## SAS 9.4 Authentication Mechanisms

*Authentication* is the process of verifying the identity of a person or process for security purposes.

<b>External</b>	<ul style="list-style-type: none"> <li>• Host authentication (credential-based)</li> <li>• Direct LDAP authentication</li> <li>• Integrated Windows authentication</li> <li>• Web authentication</li> </ul>
<b>Internal</b>	<ul style="list-style-type: none"> <li>• SAS internal authentication </li> <li>• SAS token authentication</li> </ul>

25

Copyright © SAS Institute Inc. All rights reserved.



A supporting feature of internal authentication mechanisms unifies the SAS realm and provides a degree of independence from your general computing environment.

## Internal Accounts

<b>SAS Administrator</b> <b>sasadm@saspw</b>	Has all capabilities provided by the metadata server regardless of metadata permission settings, due to membership of the Metadata Server: Unrestricted role.
<b>SAS Trusted User</b> <b>sastrust@saspw</b>	A service identity that can act on behalf of other users.
<b>SAS Environment Manager Service Account</b> <b>sasevs@saspw</b>	This account is required for communications between the SAS Environment Manager agent and the SAS Environment Manager server. It also enables SAS Environment Manager plug-ins to access the SAS Metadata Server.
<b>SAS Anonymous Web User</b> <b>webanon@saspw</b>	A service identity that functions as a surrogate for users who connect without supplying credentials.

26

Copyright © SAS Institute Inc. All rights reserved.



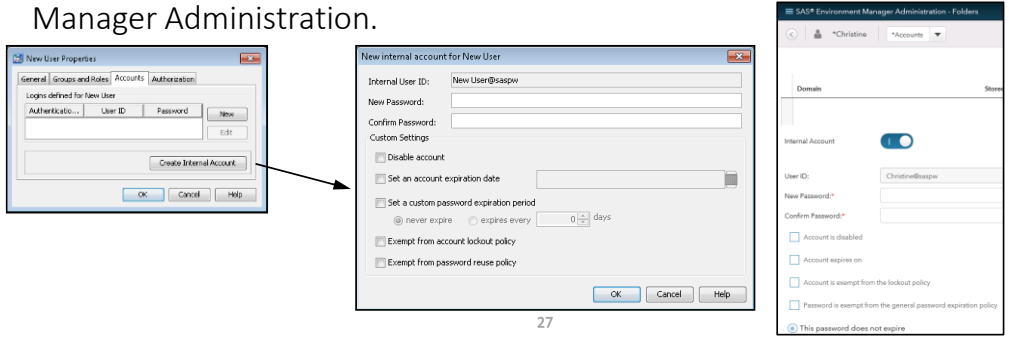
The SAS Anonymous Web User (webanon) is an optional account that can be used to grant web clients anonymous access to certain SAS Web Infrastructure Platform applications (SAS BI Web Services and SAS Stored Process Web Application). This anonymous account is configured with the SAS Deployment Wizard and is applicable only when SAS authentication is being used.

If web authentication is used, the web application server processes authentication requests, and this anonymous account has no effect.

For more information, see “Public Access and Anonymous Access” in *SAS® 9.4 Intelligence Platform: Security Administration, Second Edition*.


## Internal Accounts

- Internal accounts are primarily used to connect to the metadata server. They exist only in the metadata.
- They are authenticated by the metadata server.
- These accounts are created by the SAS Deployment Wizard and by the User Manager plug-in in SAS Management Console or in SAS Environment Manager Administration.



27

Copyright © SAS Institute Inc. All rights reserved.



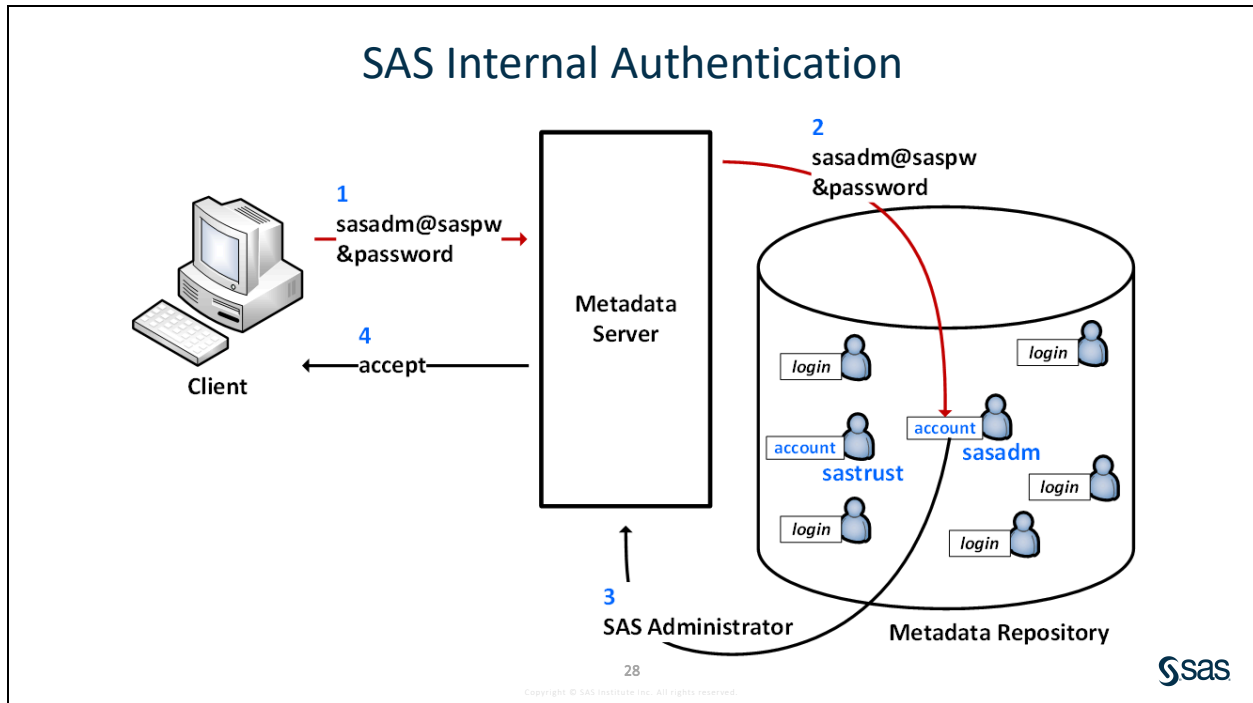
By initial policy, these server-level settings for internal account policies are in effect:

- Accounts do not expire and are not suspended due to inactivity.
- Passwords must be at least six characters, do not have to include mixed case or numbers, and do not expire.
- The five most recent passwords for an account cannot be reused for that account.
- There is no mandatory time delay between password changes.
- After three failed attempts to log on, an account is locked. If an account is locked because of logon failures, further log on attempts cannot be made for one hour.
- For an account that has a password expiration period, there is a forced password change on the first use after the password is reset by someone other than the account owner.
- An internal account has the format *userID@saspw*.

If you need to unlock an internal account and you have the necessary host access, do the following:

1. Edit the `adminUsers.txt` file to create a new unrestricted user by adding the fully qualified user ID preceded by an asterisk. Restart the metadata server for the change to take effect.
2. Log on to SAS Management Console with the new unrestricted user and unlock the account.
3. Verify that the account is unlocked by logging on to SAS Management Console with the account.

Remove the unrestricted user that you added from the `adminUsers.txt` file and restart the metadata server.



### Internal Authentication

1. At a logon prompt, **sasadm@saspw** and a password are entered. The client sends those credentials to the metadata server for verification.
2. The metadata server recognizes that the ID is for an internal account (because the ID has the @saspw suffix), so the metadata server checks the credentials against its list of internal accounts.
3. After validating the ID and password, the metadata server accepts the client connection. The connection is accepted using the SAS identity that is associated with the internal account.

Internal authentication alone is not sufficient to allow a user access to a standard workspace server because a host account is required.



Internal accounts are not designed to be used as end users.



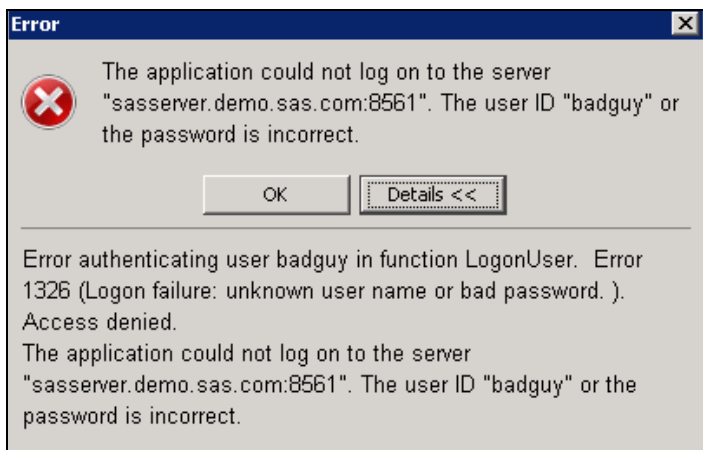




## Exploring Initial Authentication to the SAS Metadata Server

This demonstration illustrates the initial authentication process to the metadata server.

1. Log on to SAS Management Console as **Ahmed**.
2. Describe the connection profile.
3. Look at Ahmed's properties under the User Manager plug-in.
4. Connect to the metadata server under the server manager plug-in, so that you can check the metaserver log.
5. Open another instance of SAS Management Console as **badguy**.



6. Go back to the other instance of SAS Management Console and look at the log.
7. Then go into SAS Environment Manager and look at the event.
8. Go to Environment Manager Administration and show the SAS Administrator identity (or SAS Management Console).
9. Log on to SAS Enterprise Guide as Christine. See the difference. She has an operating system account.
10. Log on to SAS Studio as any of these people.

**End of Demonstration**

# 1.2 Exploring Authentication to Processing Servers and Data Servers

## Objectives

- Explore how users authenticate to the workspace server.
- Explore SAS Token Authentication that is used for the pooled workspace server and the stored process server.
- Identify the role of the object spawner.
- Examine the process of authentication to third-party database servers.
- Identify when outbound logons are needed.
- Explore authentication domains.
- Review credential management.


31

Copyright © SAS Institute Inc. All rights reserved.



## Authentication to Processing Servers

Whether users enter their own code, execute a stored process, or enable SAS applications to generate code for them, the code is executed on a SAS server. Each server type has different capabilities.

 <b>Workspace Server</b>	<ul style="list-style-type: none"> <li>• Host authentication (credential-based), by default</li> <li>• Integrated Windows Authentication</li> <li>• (SAS token authentication)</li> </ul>
<b>Pooled Workspace Server</b> <b>Stored Process Server</b>	<ul style="list-style-type: none"> <li>• SAS token authentication</li> </ul>

32

Copyright © SAS Institute Inc. All rights reserved.



**Note:** You can convert a standard workspace server to use SAS Token Authentication.

**Note:** You can convert a standard workspace server to use Integrated Windows Authentication.

## SAS Workspace Server

Most code that is generated by SAS applications is executed on a workspace server.

A *workspace server* is a SAS session that executes SAS code to access data libraries, perform tasks using the SAS language, and retrieve results.

By default, the following events occur:

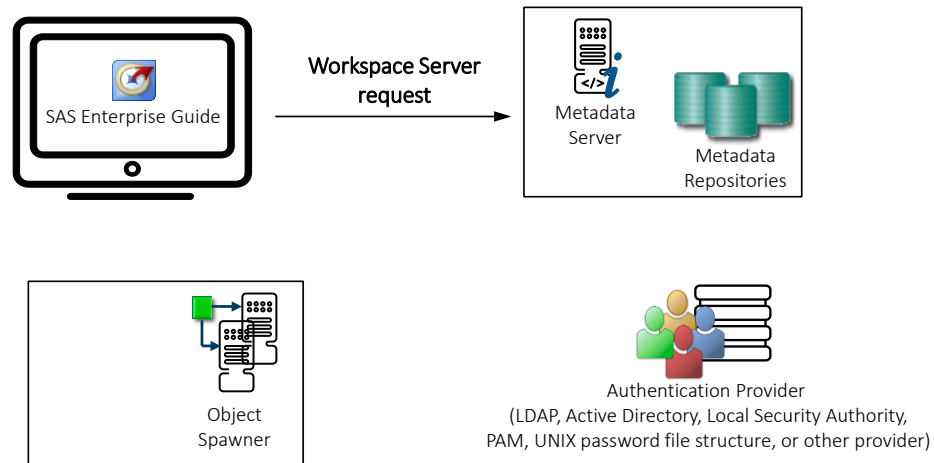
- The object spawner launches a workspace server under the user's credentials.
- The user's credentials are authenticated by the host operating system.
- The workspace server is shut down when the client application is shut down.

33

Copyright © SAS Institute Inc. All rights reserved.



## Connecting to a SAS Workspace Server

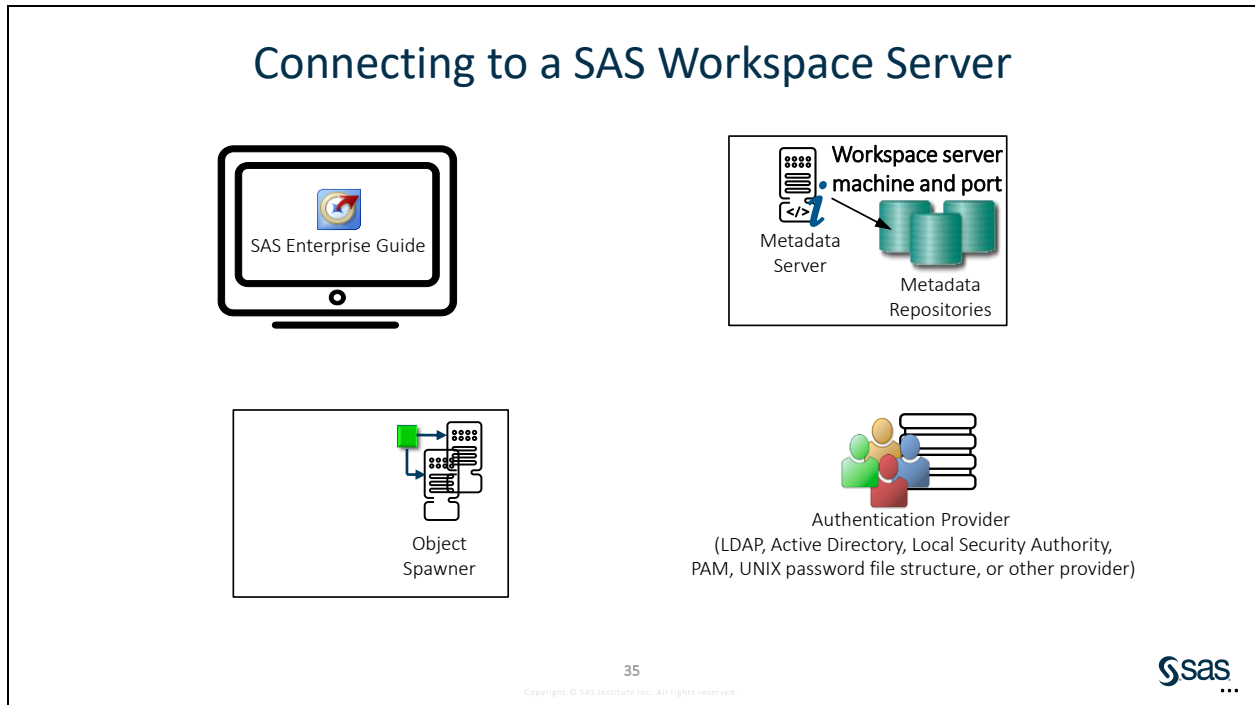


34

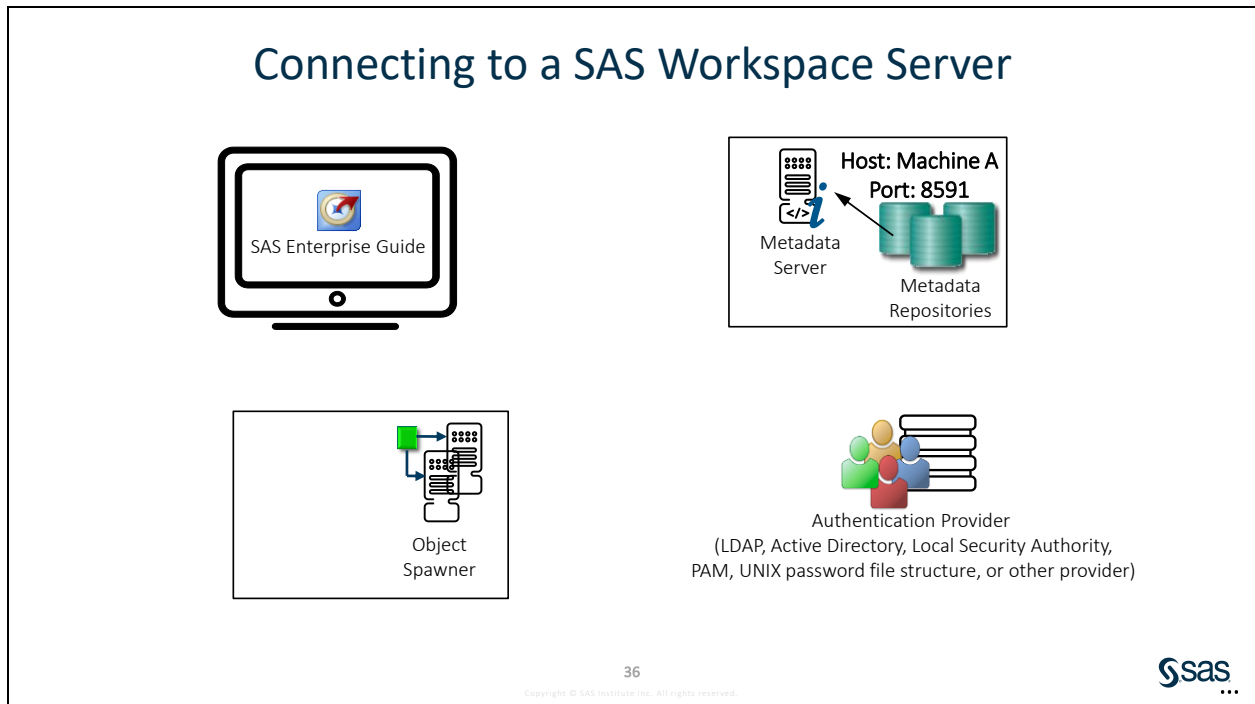
Copyright © SAS Institute Inc. All rights reserved.



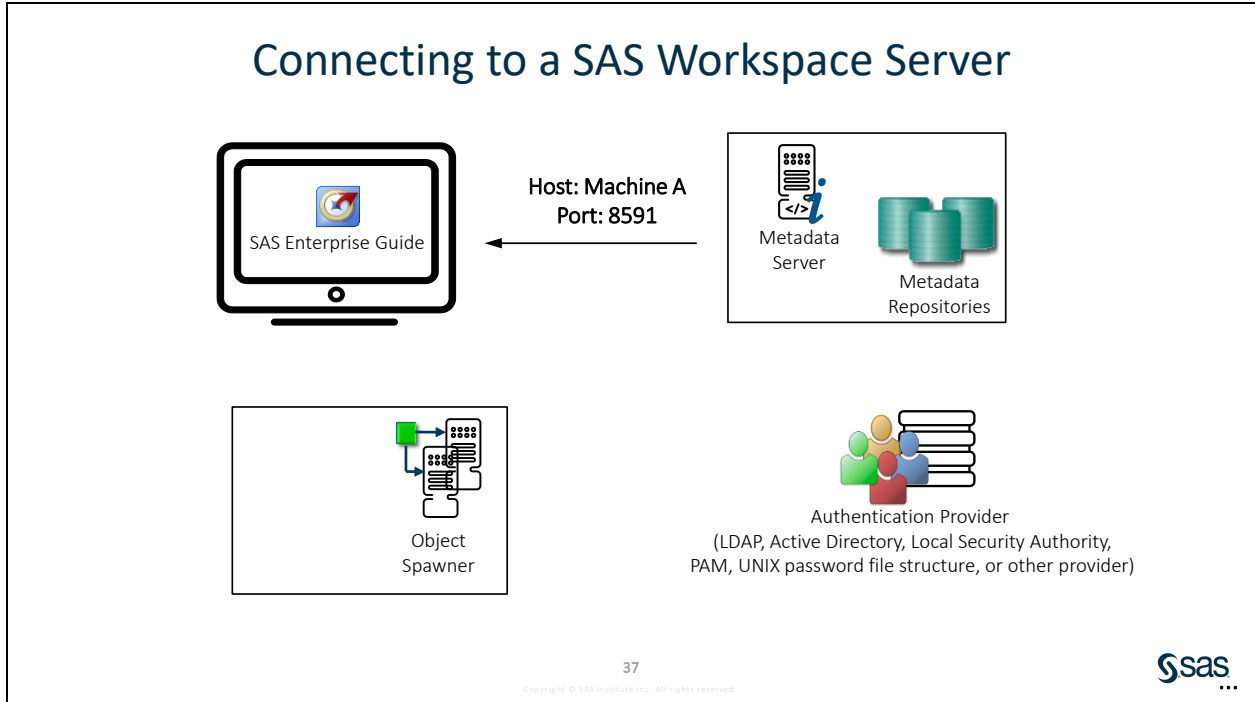
Using the established connection to the metadata server, SAS Enterprise Guide requests access to a workspace server.



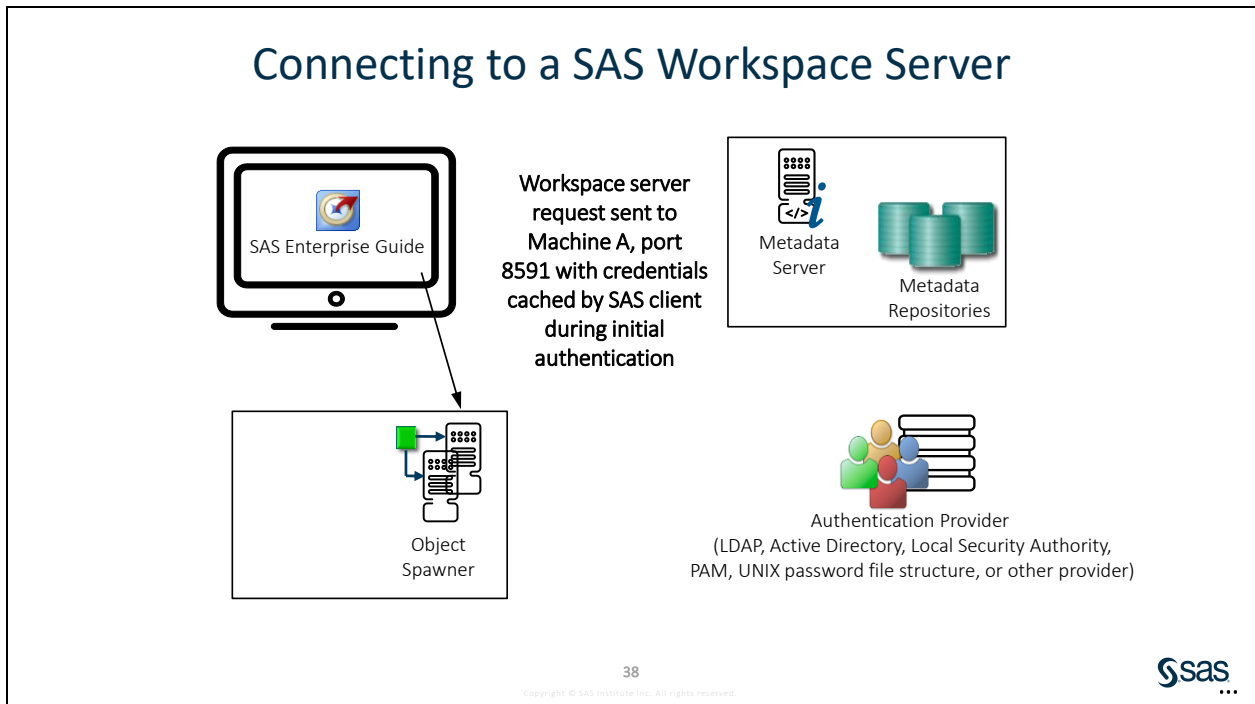
The metadata server searches the metadata for the workspace server in question.



The metadata server retrieves the name of the machine that hosts the workspace server, the port on which the object spawner listens for request for this server, and the authentication domain that is associated with the workspace server.

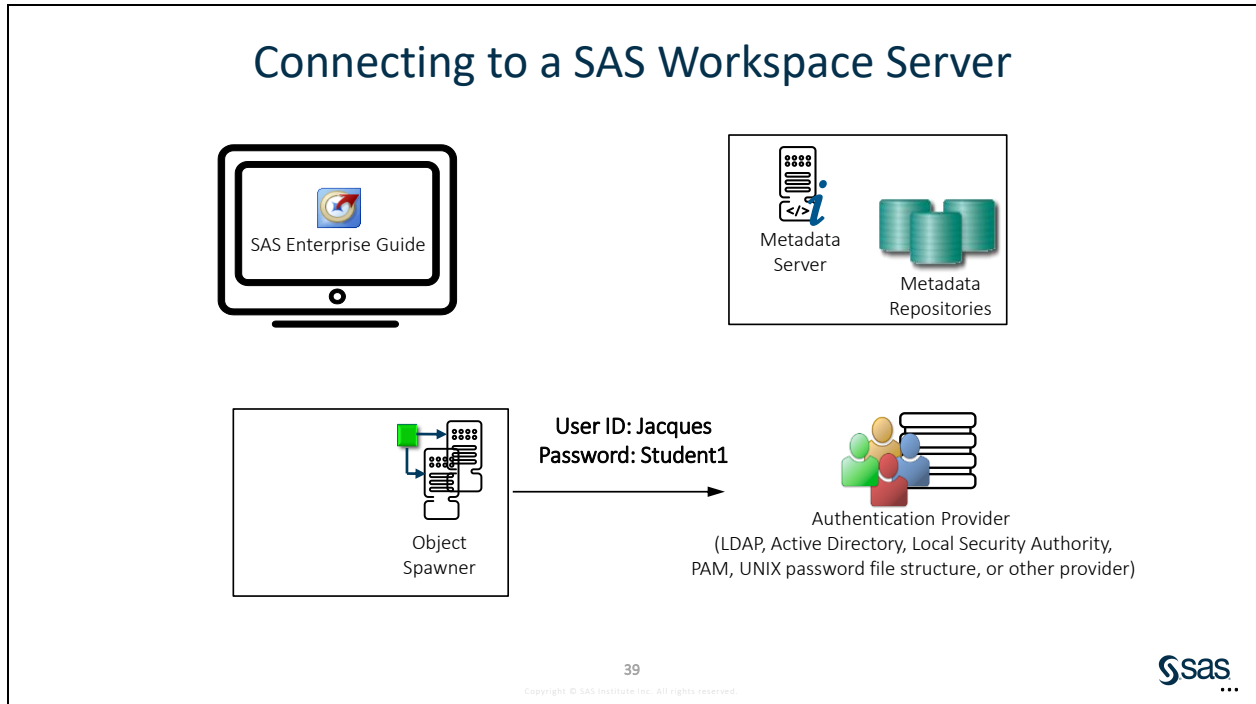


The connection information is returned to SAS Enterprise Guide.

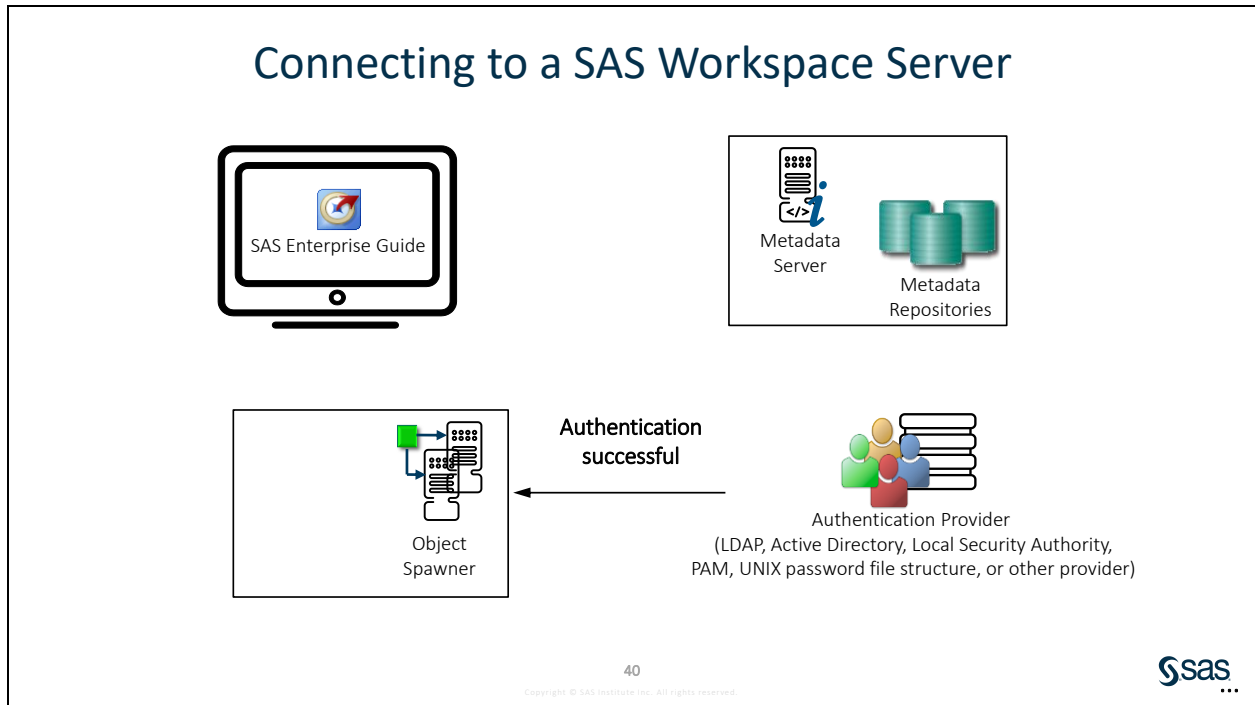


SAS Enterprise Guide uses the connection information to make the request for a workspace server. If the authentication domain for the server matches that of the initial inbound login, SAS Enterprise Guide passes along the credentials as well.

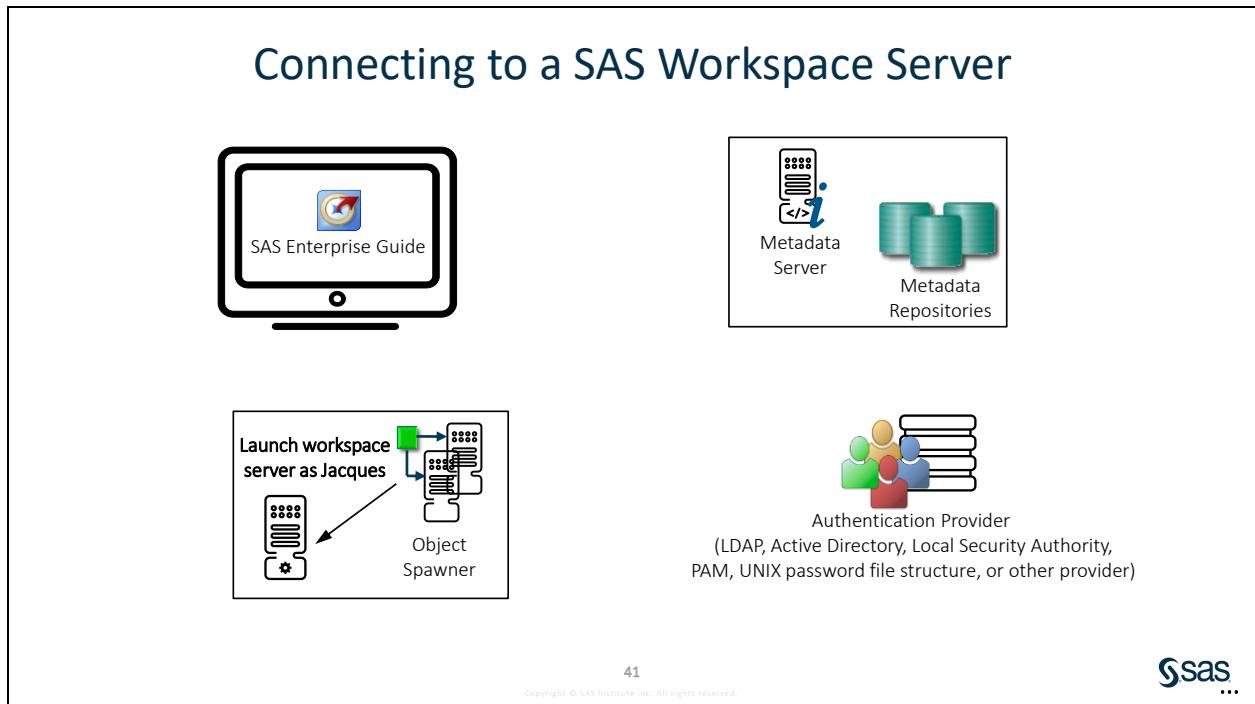
**Note:** If the server is assigned a different authentication domain, SAS Enterprise Guide searches its in-memory list of credentials for Jacques for credentials with the appropriate authentication domain. If none is found, SAS Enterprise Guide queries the metadata server for credentials for Jacques for that particular authentication domain (outbound login). If none is found, Jacques is prompted for credentials.



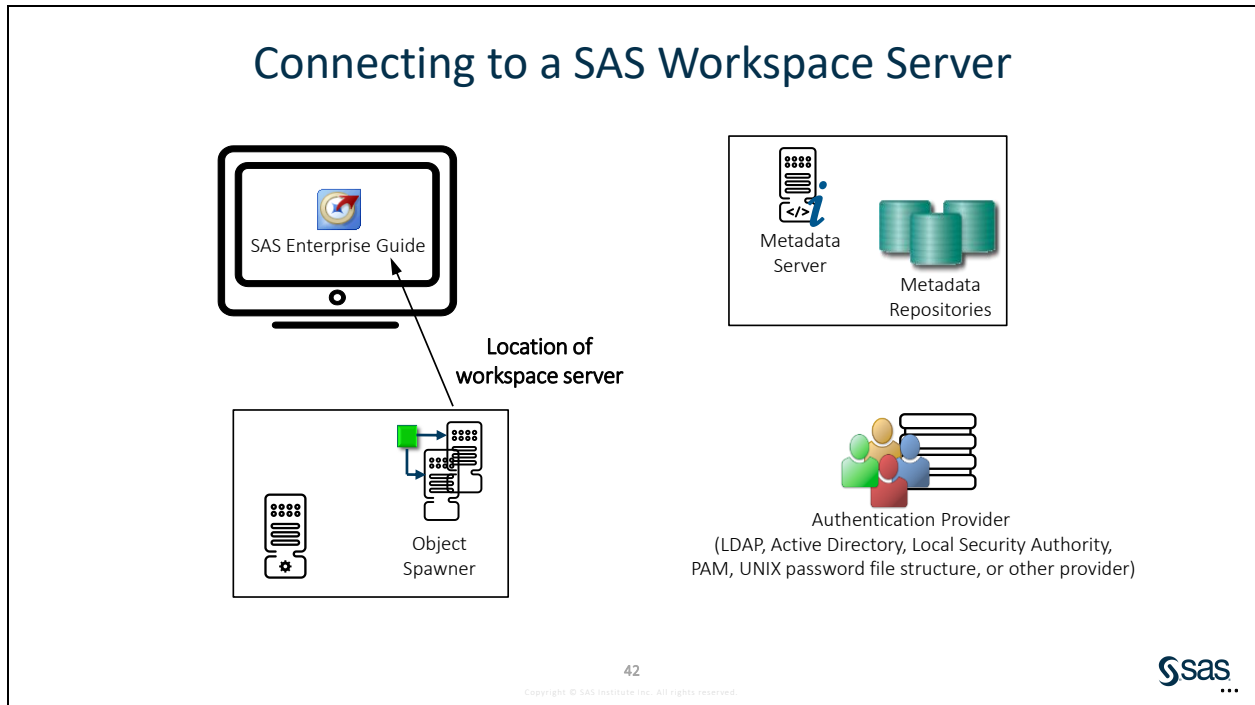
The object spawner sends Jacques' credentials to its authentication provider. The default authentication provider is the host.



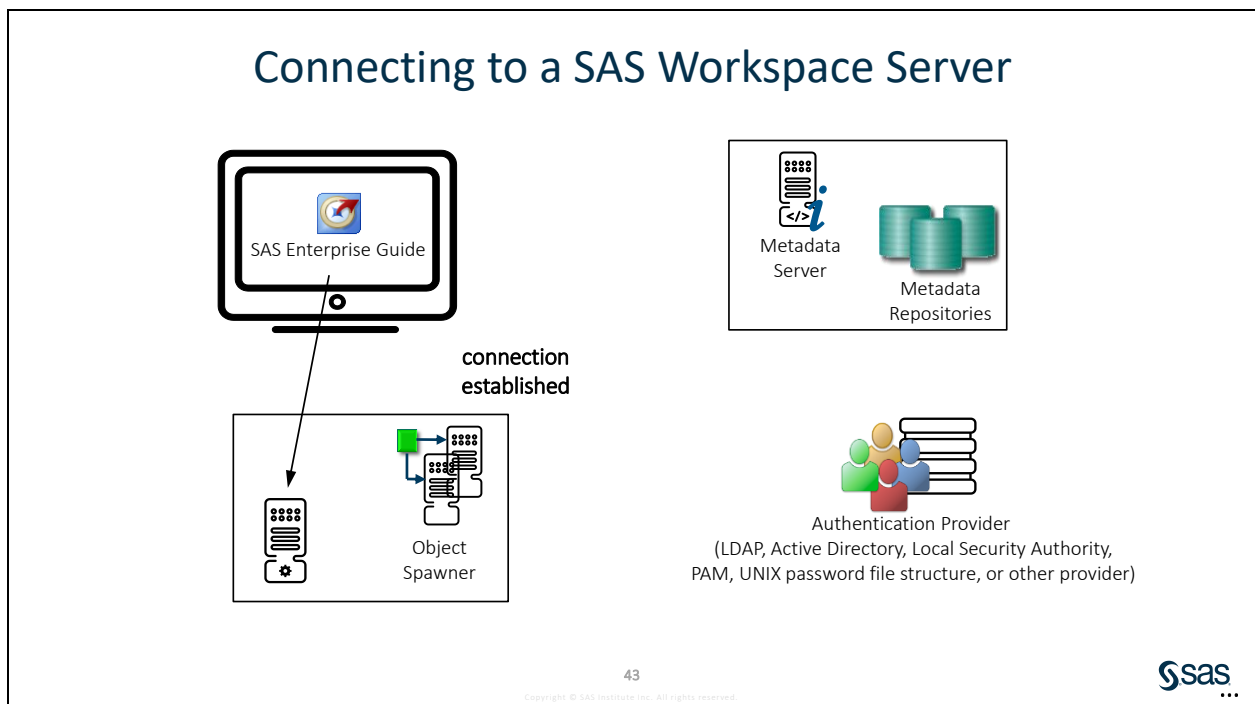
The authentication provider verifies that the credentials are valid.



The object spawner launches the workspace server. It uses the launch command that was retrieved from the metadata at start-up. The workspace server runs under the credentials that are provided by SAS Enterprise Guide and authenticated by the host.

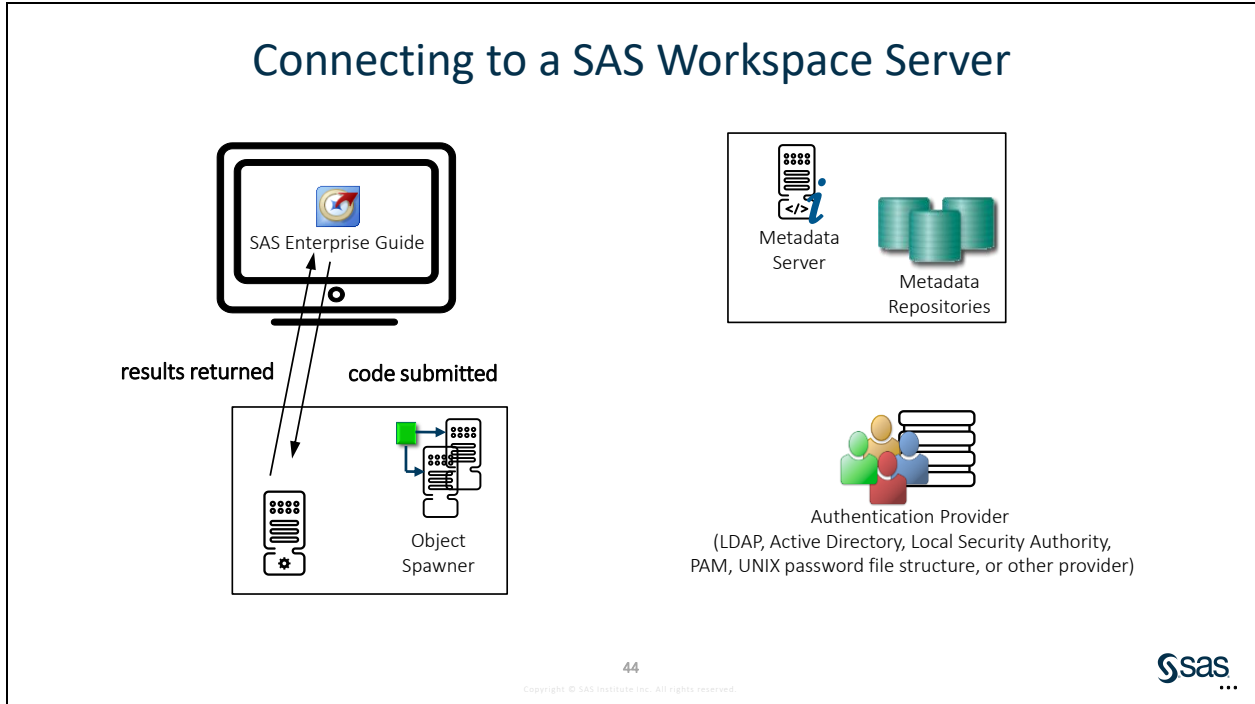


The object spawner provides SAS Enterprise Guide with a TCP connection to the workspace server session.

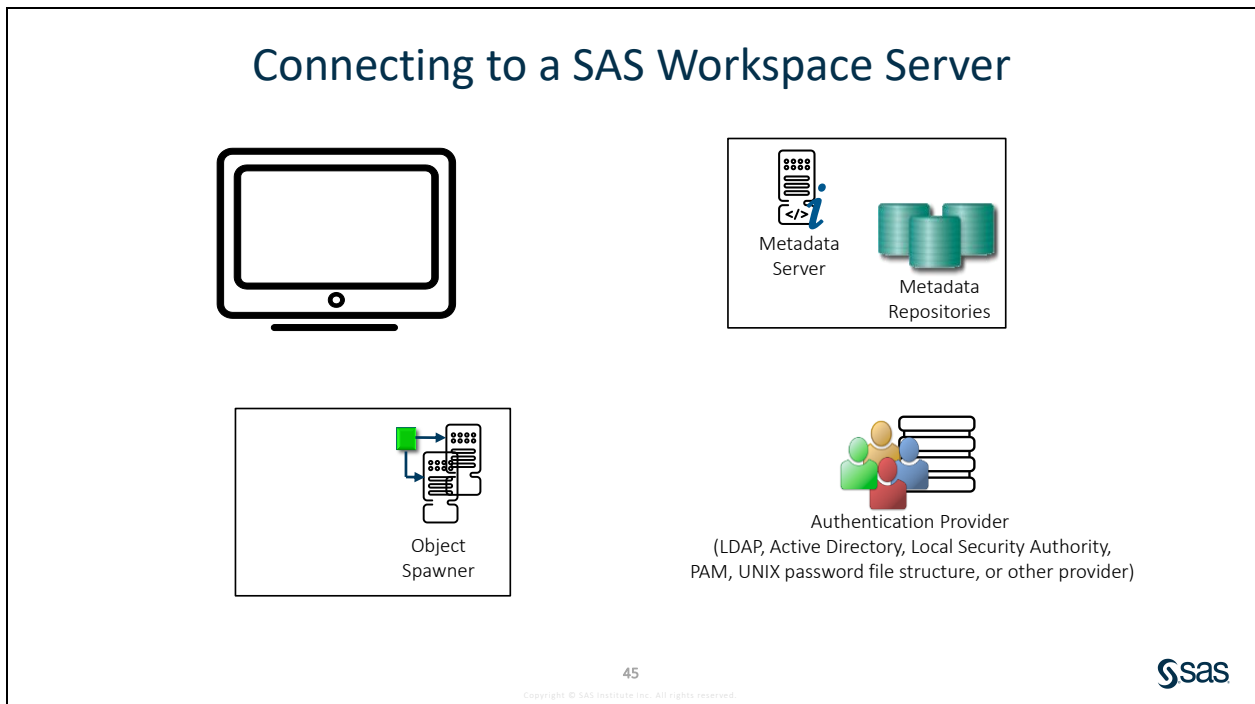


SAS Enterprise Guide communicates directly with the workspace server.





SAS Enterprise Guide submits one or more requests for processing. Results are returned to SAS Enterprise Guide as appropriate.



After Jacques closes SAS Enterprise Guide, the workspace server session ends.

**Note:** The connection could close earlier if there is a TCP time-out.

## Authentication to Processing Servers

Whether users enter their own code, execute a stored process, or enable SAS applications to generate code for them, the code is executed on a SAS server. Each server type has different capabilities.

<b>Workspace Server</b>	<ul style="list-style-type: none"> <li>• Host authentication (credential-based), by default</li> <li>• Integrated Windows Authentication</li> <li>• (SAS token authentication)</li> </ul>
<b>Pooled Workspace Server</b>	<ul style="list-style-type: none"> <li>• SAS token authentication</li> </ul>
<b>Stored Process Server</b>	

46

Copyright © SAS Institute Inc. All rights reserved.



## SAS Token Authentication

*SAS token authentication* is when the metadata server generates and validates a single-use identity token for each authentication event. This enables the following SAS processing servers to accept users who are already connected to the metadata server:

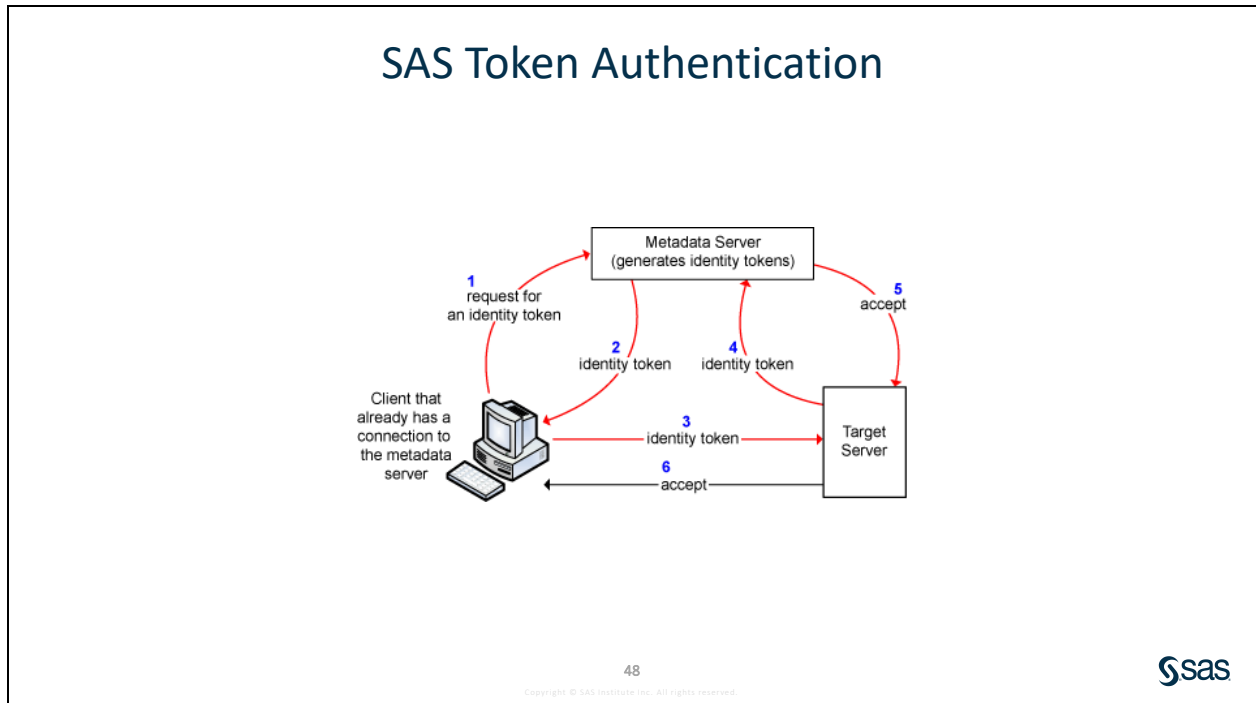
- OLAP server
- stored process server
- pooled workspace server

The workspace server can also use SAS Token Authentication.

47

Copyright © SAS Institute Inc. All rights reserved.





SAS Token Authentication is when the metadata server generates and validates a single-use identity token for each authentication event. This enables participating SAS servers to accept users who are already connected to the metadata server.

1. The user initiates a request that requires access to a target server (for example, a request in SAS Enterprise Guide to open a cube that is associated with the OLAP server). Using the existing connection to the metadata server, the client requests an identity token for the target server.
2. The metadata server generates the token and returns it to the client.
3. The client sends the token to the target server.
4. The target server sends the token back to the metadata server for validation.
5. The metadata server validates the token and returns an acceptance message and a representation of the user to the target server.
6. The target server accepts the connection.

The benefits of SAS token authentication are listed here:

- Individual, external accounts for credential-based authentication are not required.
- SAS copies of individual, external passwords do not need to be stored in the metadata.
- Reusable credentials are not transmitted across the network.
- Metadata layer evaluations are based on the requesting user's identity.

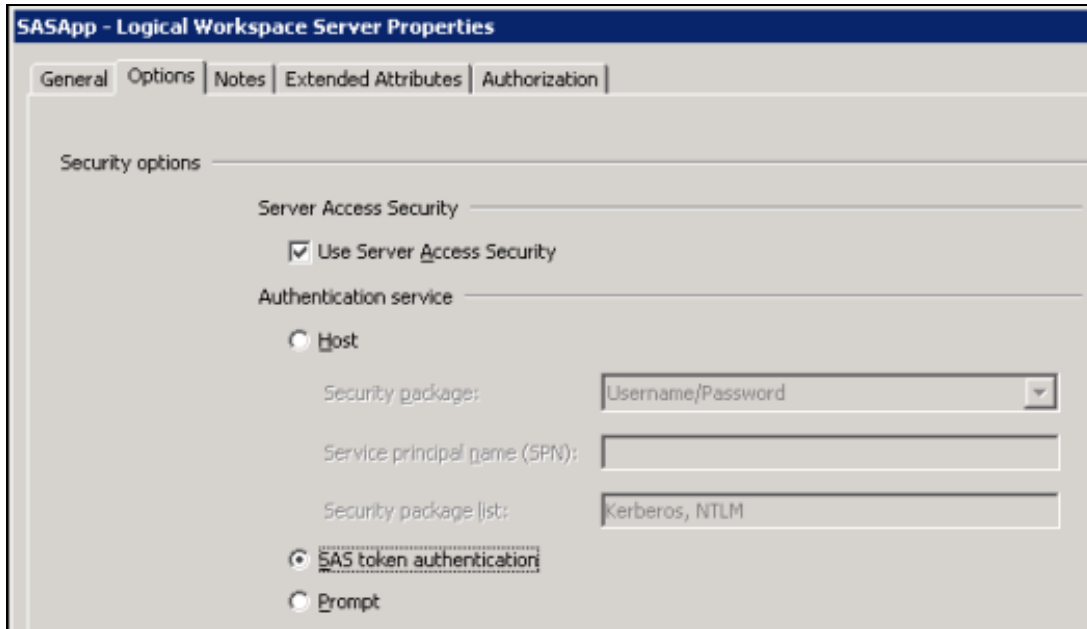
The limitations of using SAS token authentication are as follows:

- Host access is based on a shared login, if it is implemented for use on a standard workspace server.
- It is available only for metadata-aware connections to the target server.
- This authentication is not available for access to third-party database servers.

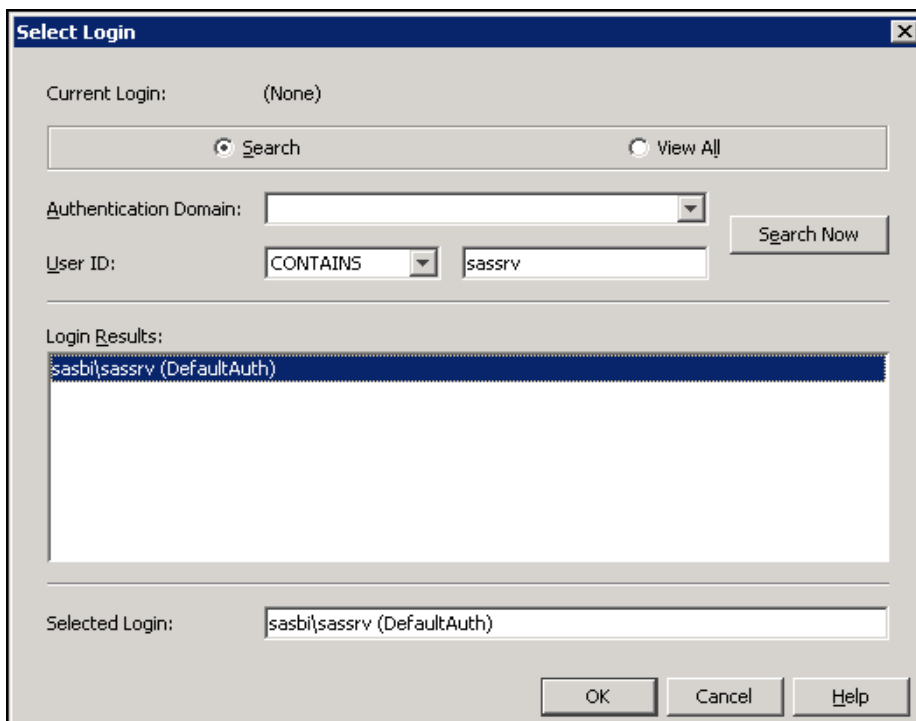
Because SAS token authentication essentially uses a shared login (typically, sassrv), host access to resources is based on the access rights that are associated with that account.

Converting a standard workspace server to use SAS token authentication requires some changes to the server's metadata.

In the Properties window for the logical workspace server, select **SAS token authentication** on the Options tab.



In the Properties window for the physical workspace server, select **Launch credentials** on the Options tab.



## Workspace Server Pooling

- In pooling, a set of workspace server processes are
- made available to process certain types of requests
  - reused for subsequent requests
  - owned by a shared identity.



The primary purpose of workspace server pooling is to enhance performance by avoiding the time associated with launching workspace servers on demand.



In general, pooling is used when a relational information map is queried, processed, opened, or used indirectly through a report.

49

Copyright © SAS Institute Inc. All rights reserved.



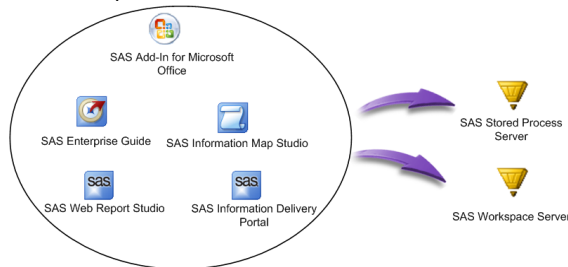
## SAS Stored Processes

A *SAS Stored Process* has the following characteristics:

- consists of a SAS program that is hosted on a server or in metadata, and includes a metadata definition that describes how the stored process should execute



- is typically executed on a stored process server but can also be executed on a workspace server



50

Copyright © SAS Institute Inc. All rights reserved.



The stored process metadata properties determine which type of server the stored process is executed on, where the source code is stored, and the type of output that is produced.

**New Stored Process**

**Execution**  
Specify the file, execution environment and result type for the stored process.

Application server:  
<Select a server>

Server type:

- Default server  
Select this option to allow the client application to specify the server.
- Stored process server only  
Select this option if the stored process uses sessions or if it uses replay (for example, to produce graphics in streaming output).
- Workspace server only  
Select this option if the stored process must be run under the client identity.

---

Source code location and execution:

- Allow execution on other application servers (store source code in metadata)
- Allow execution on selected application server only
  - Store source code in metadata
  - Store source code on application server

Source code repository: [dropdown] [Manage...]

Source file: [text box]

[Edit Source Code...]

---

Result capabilities:  Stream  Package

## SAS Stored Process Server

SAS Stored Process Servers interact with SAS by executing stored processes.

- Each stored process server handles multiple users.
  - It is reused for subsequent requests.
  - It is owned by a shared identity.
- This server includes load-balancing settings that the object spawner uses to distribute requests between the server processes.

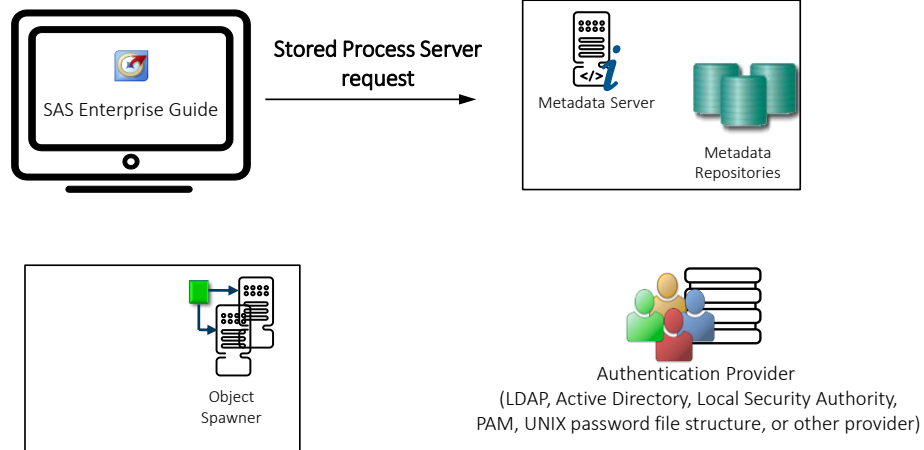


51

Copyright © SAS Institute Inc. All rights reserved.



## Connecting to a Stored Process Server

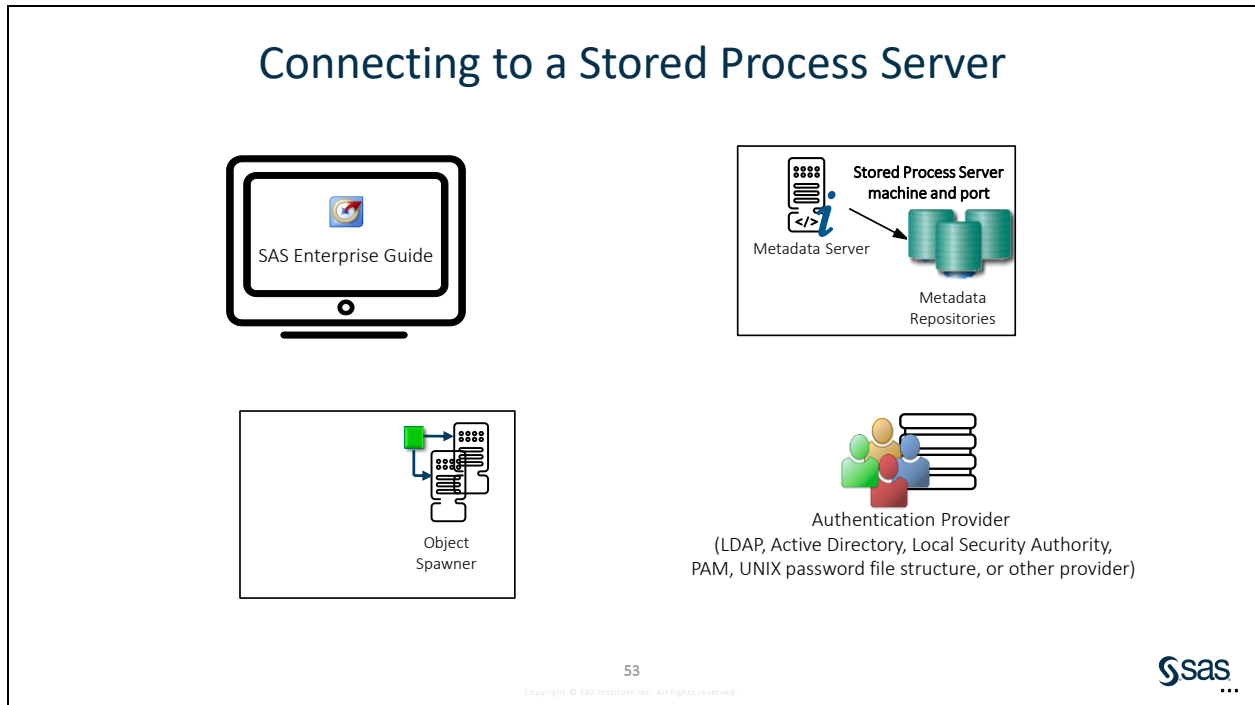


52

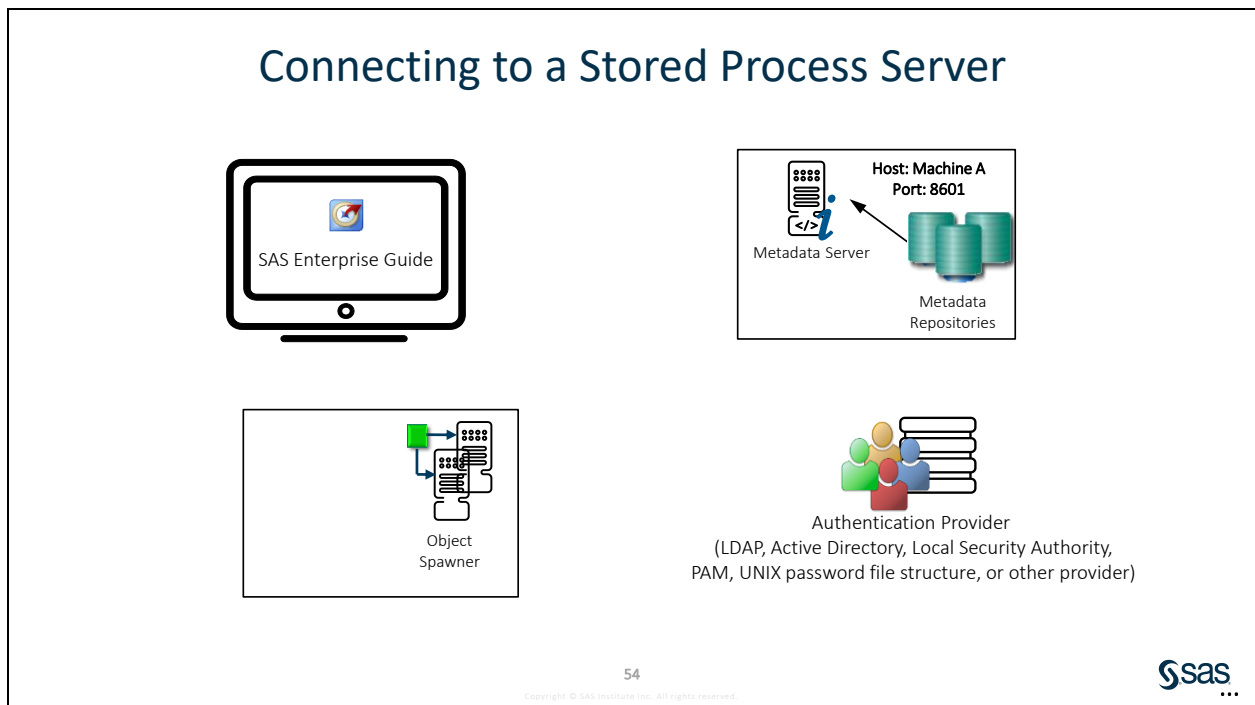
Copyright © SAS Institute Inc. All rights reserved.



Using the established connection, SAS Enterprise Guide requests access to a stored process server.



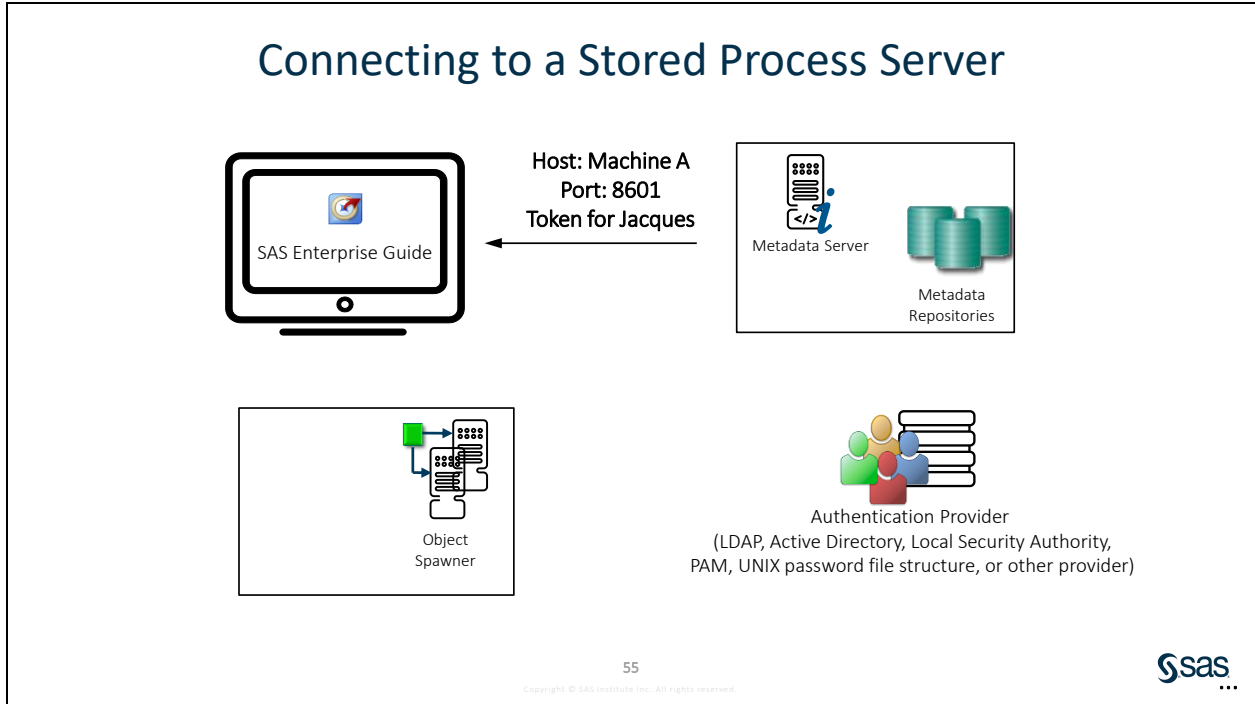
The metadata server searches the metadata for the stored process server in question.



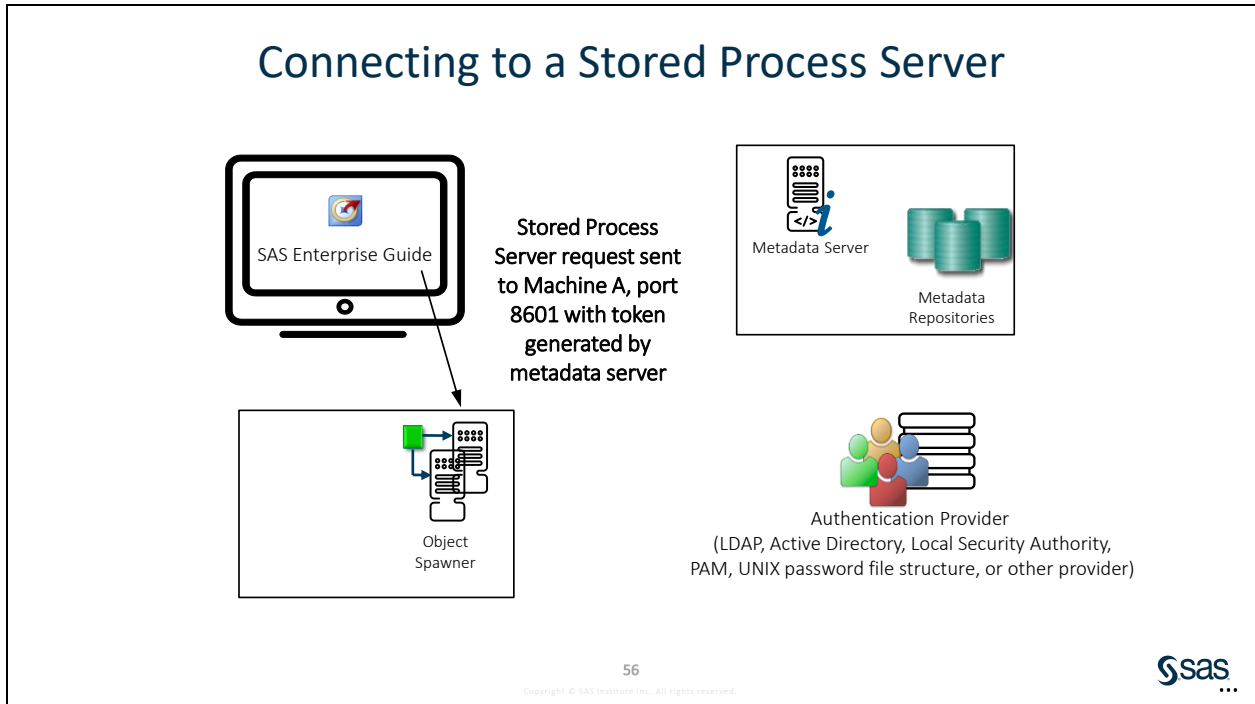
The metadata server retrieves the machine name that hosts the stored process server, the port on which the object spawner listens for request for this server, and a token.

**Note:** A SAS identity token is a single-use, proprietary software representation of an identity.

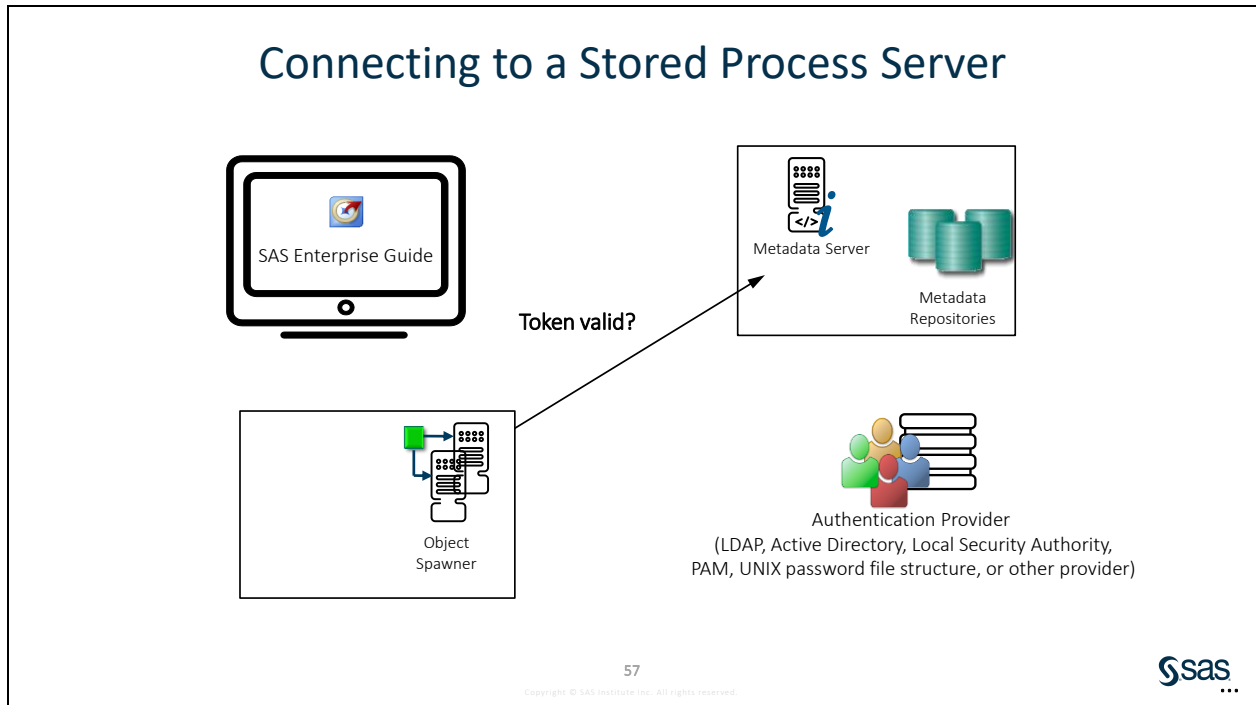




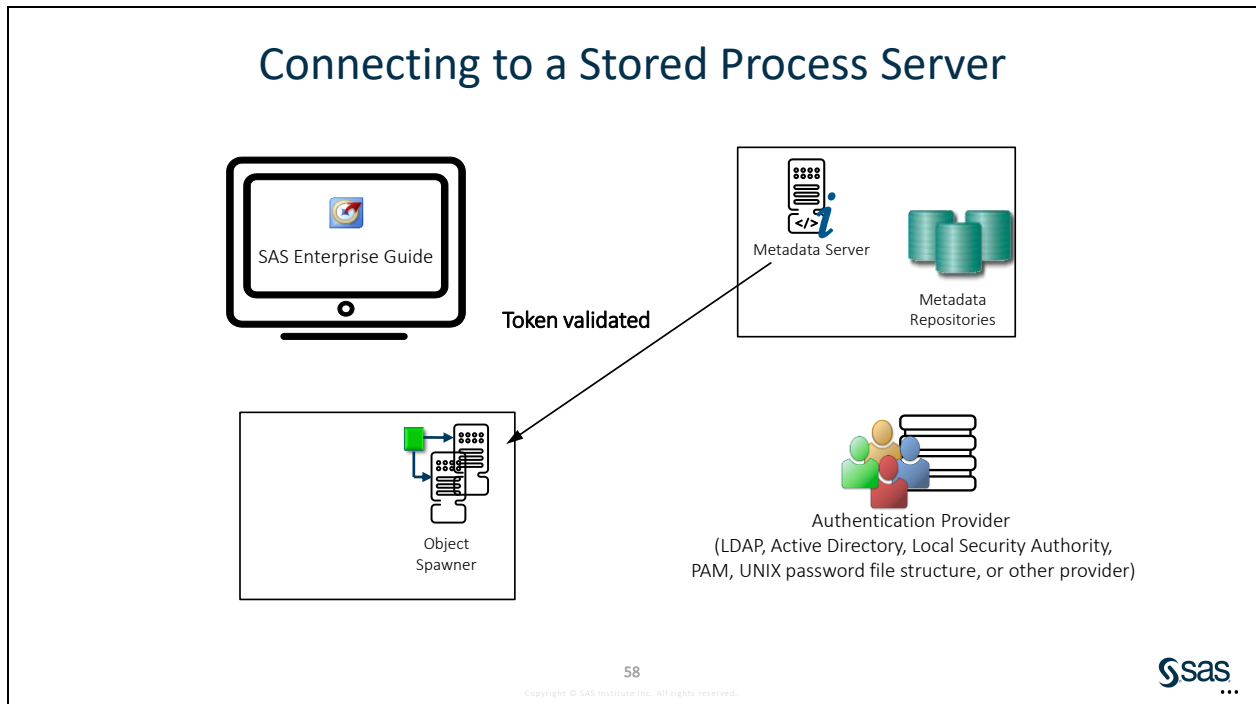
The connection information is returned to SAS Enterprise Guide.



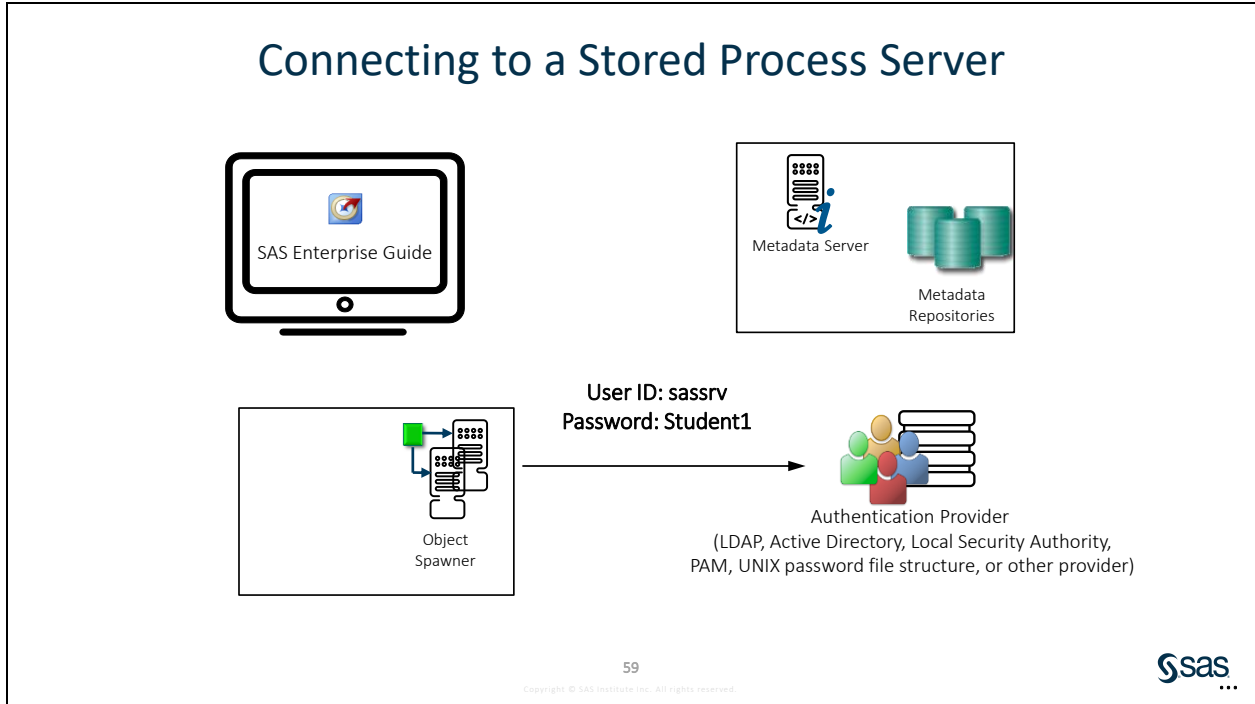
SAS Enterprise Guide uses the connection information and the token that is provided by the metadata server to make the request for a stored process server.



The object spawner sends the token to the metadata server for verification.

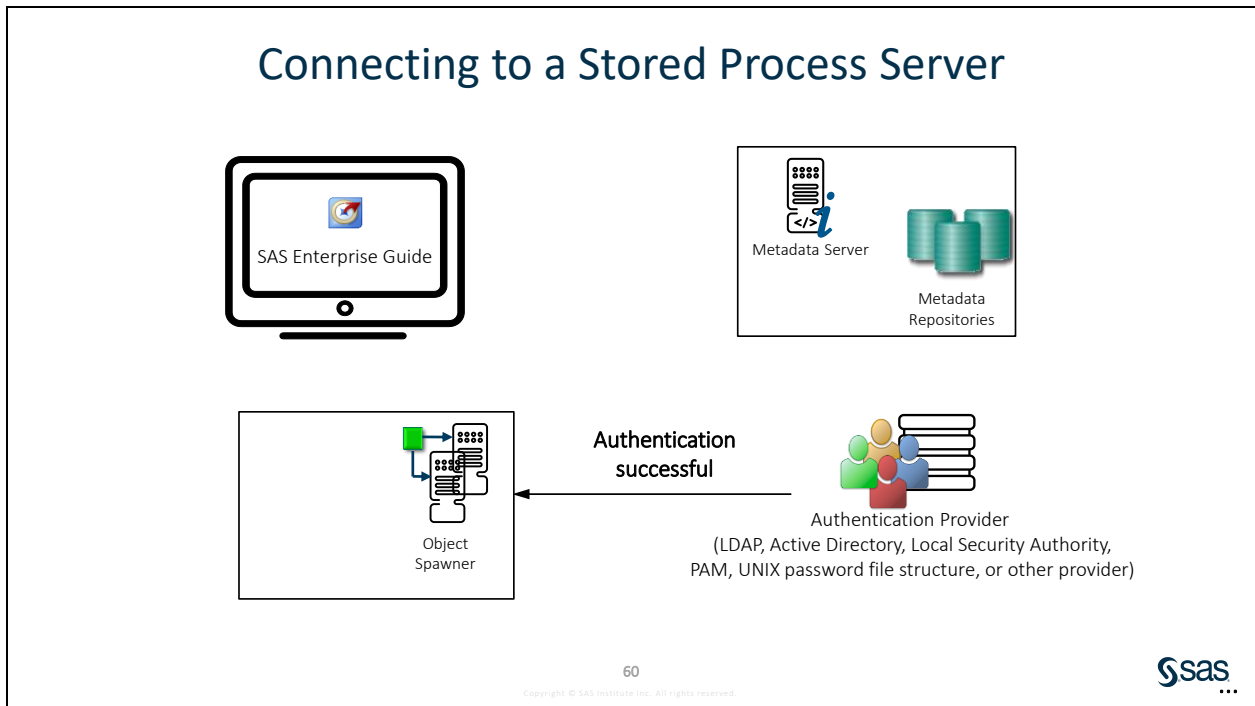


The metadata server verifies that the token is valid.

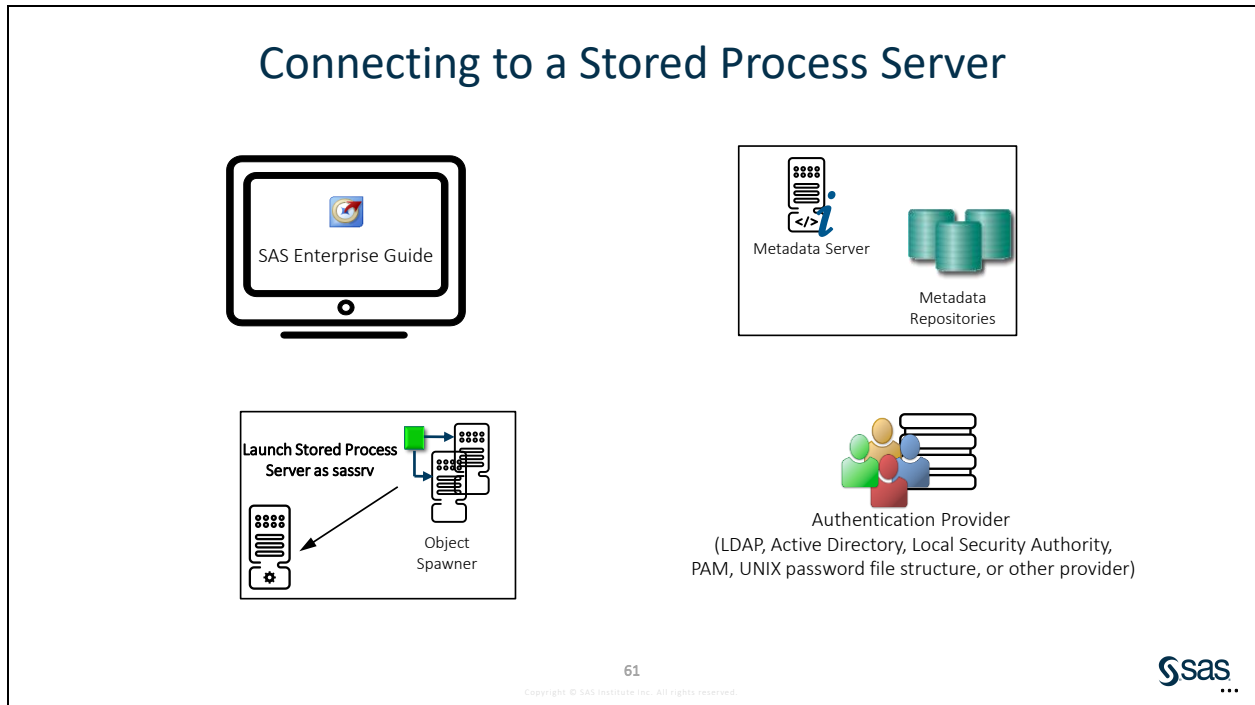


If no stored process server is currently available and more can be spawned, the object spawner sends the shared credentials, typically `sassrv`, to the host for authentication.

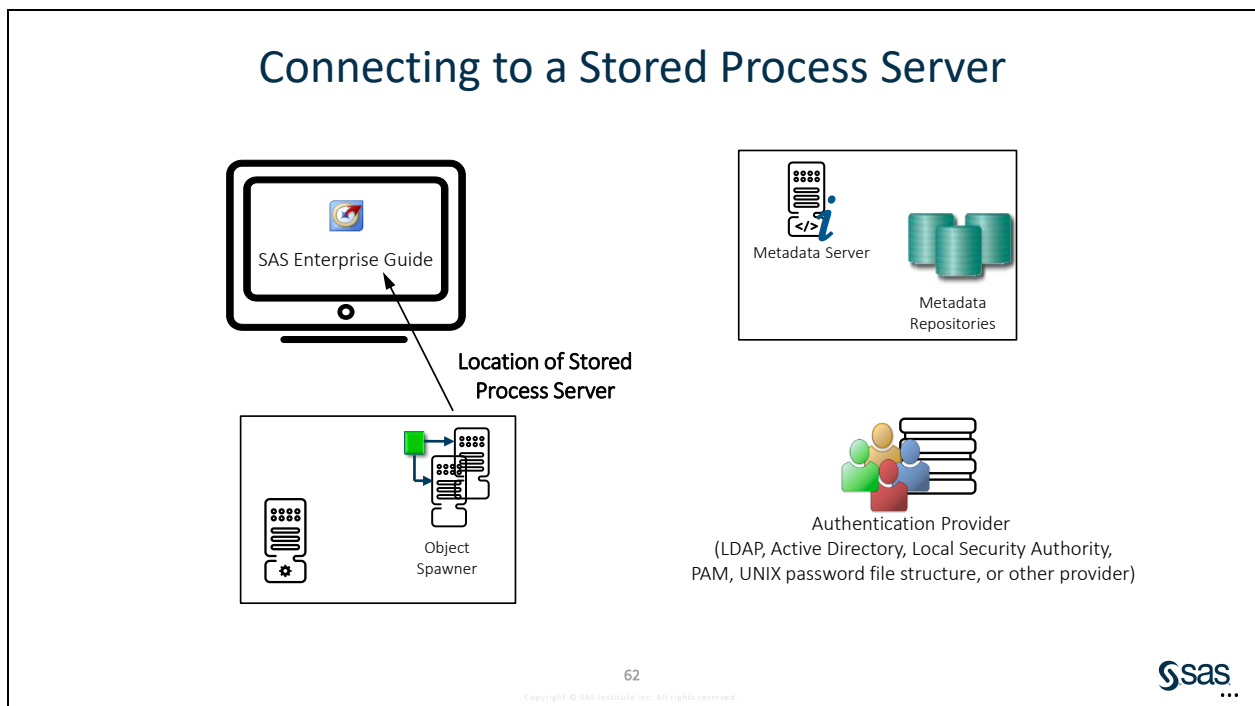
**Note:** During its own start-up, the object spawner not only retrieves the launch command for the stored process server from the metadata, but also the shared credentials, user ID, and password.



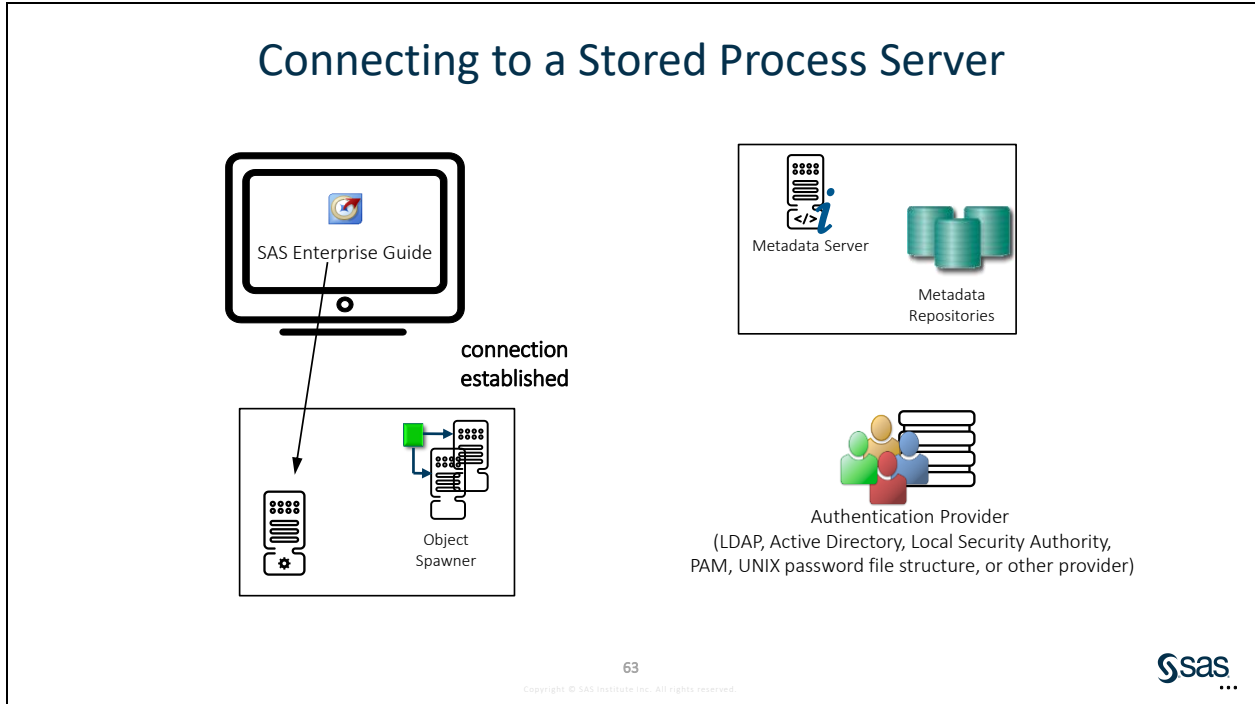
The authentication provider authenticates the credentials.



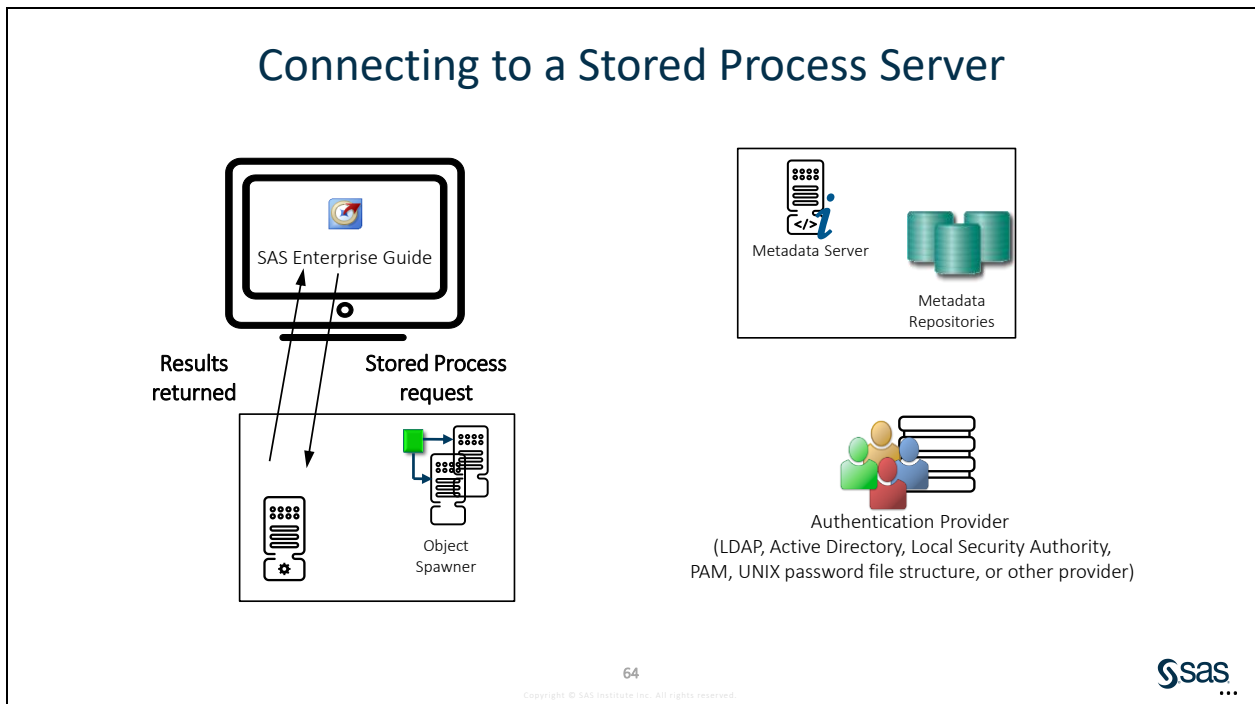
The object spawner launches the stored process server. It uses the launch command that it retrieved from the metadata at start-up. The stored process server runs under shared credentials.



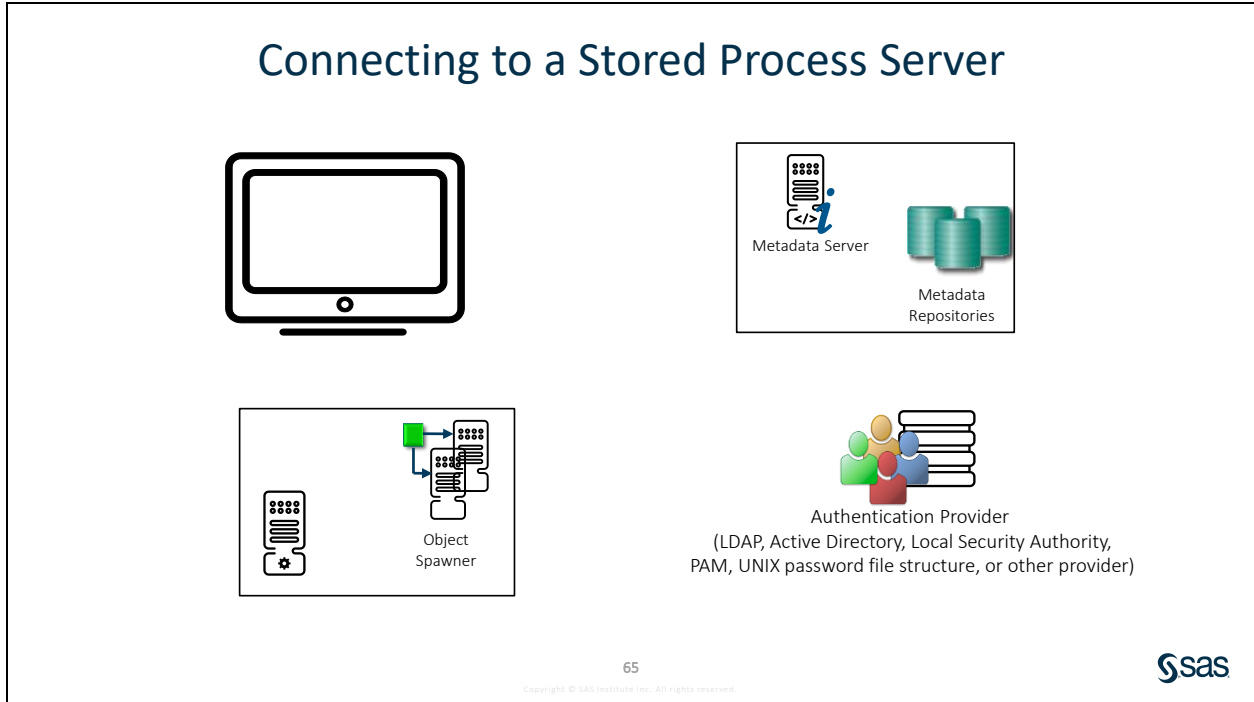
The object spawner provides SAS Enterprise Guide with a TCP connection to the stored process server. During the execution of the stored process, metadata server requests are done as an individual user, and operating system requests are done as the shared account.



SAS Enterprise Guide communicates directly with the stored process server. SAS Enterprise Guide submits a request to execute a stored process.



The results from the stored process are returned to SAS Enterprise Guide as appropriate.



After the execution of the stored process is complete, the stored process server is available for reuse by other requests from the same or a different user.

## SAS Object Spawners

Workspace servers and stored process servers are initialized by the SAS Object Spawner.

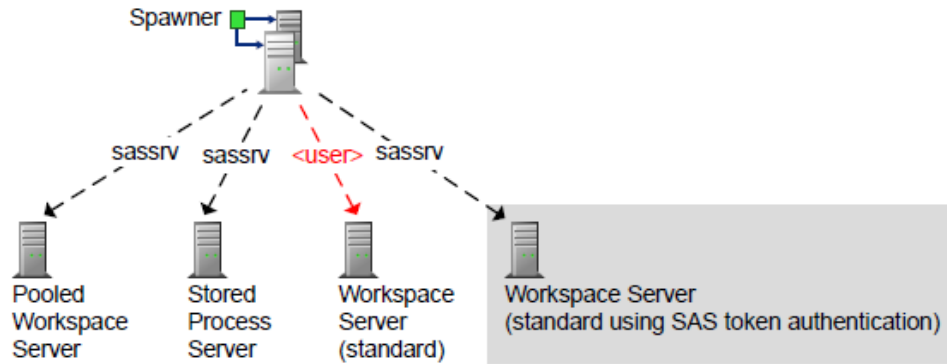
An object spawner does the following:

- runs on each machine where you want to run a workspace server or stored process server
- listens for requests and launches servers, as necessary

The screenshot shows the 'Object Spawner - sasserver Properties' dialog box. The 'Servers' tab is active, displaying a list of servers. The 'Available servers' list is empty, while the 'Selected servers' list contains: Operating System Services - sasserver, SASApp - Pooled Workspace Server, SASApp - Stored Process Server, SASApp - Visual Process Orchestration Design Server, and SASApp - Workspace Server. The SAS logo is in the bottom right corner.

## SAS Object Spawners

When the object spawner starts, it retrieves information about how to launch the servers. (It connects to the metadata server by reading its start-up file, which is named metadataConfig.xml.)



67

Copyright © SAS Institute Inc. All rights reserved.



If changes are made to the server or spawner configurations, the spawner can be refreshed to pick up and apply these new changes. The refresh reinitializes the spawner and forces it to reread its configuration in the metadata. As part of this refresh, the spawner quiesces any servers that it started. The servers shut down when their clients complete their work.

To refresh an object spawner, follow these steps:

1. Expand the **Server Manager** node ⇒ **Object Spawner**. Then right-click the **Object Spawner** machine name node.
2. From the pop-up menu, select **Connect**.
3. Right-click the **Object Spawner** node again. From the pop-up menu, select **Refresh Spawner**.
4. In the confirmation dialog box, click **Yes**.

**Note:** When an object spawner manages more than one SAS Application Server context, you can refresh a specific application server by selecting **Refresh Application Server**.

## Connection to DBMS Data Libraries

Three authentications and permissions take place when you access DBMS data.

- metadata authentication
- SAS Workspace Server authentication
- DBMS authentication

68

Copyright © SAS Institute Inc. All rights reserved.



Three authentications and permissions occur when you access DBMS data.

*Metadata authentication* is the first. This is mainly for the metadata server to know who is requesting the data and verify that the user has metadata permissions to the data.

*Workspace server authentication* is the second authentication. If metadata permissions enable the user to access the workspace server, then the metadata server retrieves and passes the user's credentials to the host operating system of the SAS Workspace Server for authentication (via the object spawner).

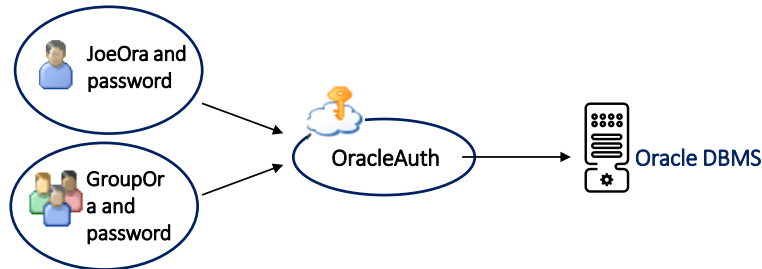
When the first two authentications and authorizations are met, the metadata server fetches the corresponding metadata-stored DBMS credentials to pass to the DBMS for authentication. (These credentials must be stored in metadata via groups for shared credentials or at the individual user level, except when you use SQL Server Windows Integration Authentication).

Next, the *DBMS system* controls which data the credentials have permission to access. SAS cannot and does not override the DBMS permissions on DBMS data. However, SAS can add or enhance DBMS data permissions through metadata permissions.



## Seamless Connection to DBMS Data Libraries

To enable clients to seamlessly obtain user credentials for disparate systems for outbound use, logons are stored in the metadata: **User ID, Password, and Authentication Domain.**



**Note:** An example of outbound use is a DBMS or workspace server on a machine with separate authentication from where the metadata server resides.

69

Copyright © SAS Institute Inc. All rights reserved.



Joe's second logon provides seamless access to Oracle using an individual account. This logon includes a password and must be in the Oracle server's authentication domain. The ETL group's logon is a shared logon for the Oracle server. Joe's personal Oracle logon has a higher priority.

**Note:** If you choose to store passwords for the workspace server, the relationships would be comparable to the depiction of the Oracle DBMS, OracleAuth authentication domain, and Oracle logons. For example, you might put the workspace server in WorkspaceAuth and create individual and group logons in that authentication domain.

## Outbound Logons

Outbound logons can be defined on the Accounts tab of individual and group identities and must include these items:

- a fully qualified external account
- password
- authentication domain

Authentication Domain	User ID	Password
OraAuth	oragroup	*****

Authentication Domain	User ID	Password
DefaultAuth	sasserver\Eric	*****
OraAuth	oraID	*****

70

Copyright © SAS Institute Inc. All rights reserved.



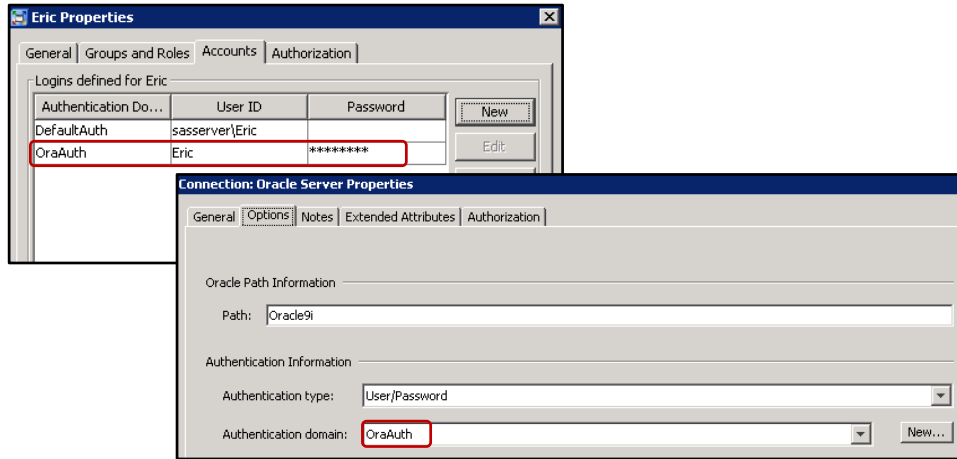
Clients use authentication domain assignments to determine which credentials are valid for which servers. The target server validates the client-supplied credentials against its authentication provider.

In most deployments of the platform for SAS Business Analytics, passwords for external accounts need to be stored in the metadata to support **only** these types of access:

- seamless access to an external database
- seamless access to the standard workspace server in a mixed provider environment where Integrated Windows Authentication and SAS token authentication is not applicable

## Authentication Domains

An *authentication domain* is a SAS metadata object that pairs logons with the server definitions where those credentials are correctly authenticated.



For example, an Oracle server definition and the SAS copies of Oracle credentials (outbound logons) have the same authentication domain value (for example, OracleAuth) if those credentials authenticate on that Oracle server. Authentication domains can be managed using the Server Manager plug-in or the User Manager plug-in. Right-click the plug-in and select **Authentication Domains**.

## Credential Management

Each client application maintains an in-memory list of credentials (*user context*) for each connected user. The list includes the following:

- credentials provided when the application is launched (*cached credentials*)
- credentials provided interactively during the session (*prompting*)
- retrieval of credentials from metadata, either from the user's account properties or from a group's account properties in the user's identity hierarchy

### Example: Contents of a *User Context*

User ID	Password	Authentication Domain
myWINID		DefaultAuth
GroupDBMSid	*****	DBMSauth

72

Copyright © SAS Institute Inc. All rights reserved.



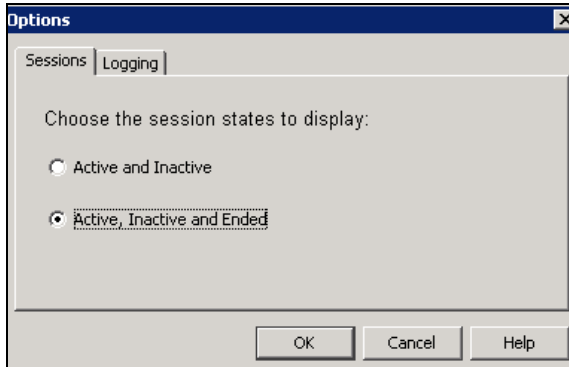
**Note:** Credentials from a user or group's metadata definition are not included in the initial list that is created when a user logs on. Instead, such credentials are added to the list dynamically (when and if they are needed during the user's session).



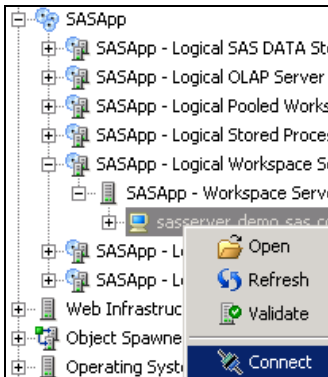
## Monitoring SAS Servers and Sessions in SAS Management Console

This demonstration illustrates how to monitor SAS servers and sessions in SAS Management Console.

1. In SAS Management Console, right-click the **Server Manager** plug-in and select **Options**. Select **Active, Inactive and Ended** and click **OK**.

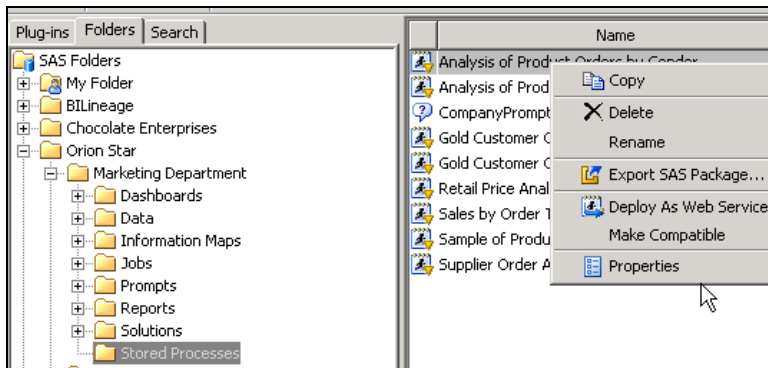


2. Expand the **Server Manager** plug-in. Then select **SASApp** ⇒ **SASApp - Logical Workspace Server** ⇒ **SASApp - Workspace Server** ⇒ **sasserver.demo.sas.com**. Right-click **sasserver.demo.sas.com** and select **Connect**.

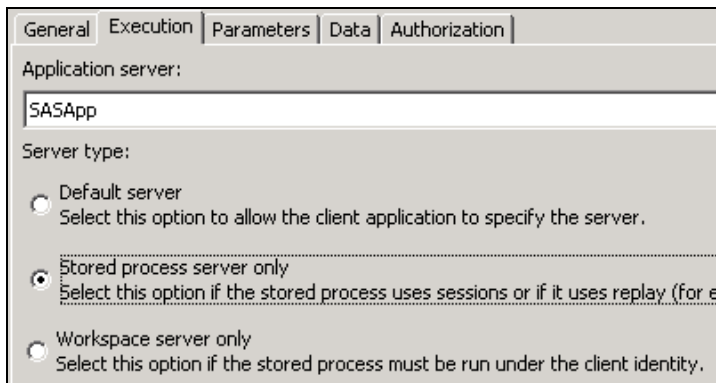


3. Connect to the stored process server. Expand **SASApp - Logical Stored Process Server** ⇒ **SASApp - Stored Process Server**. Right-click **sasserver.demo.sas.com** and select **Connect**. Notice that the tabs become active when you are connected.

- On the Folders tab, navigate to **Orion Star** ⇒ **Marketing Department** ⇒ **Stored Processes**. Right-click **Analysis of Product Orders by Gender**.

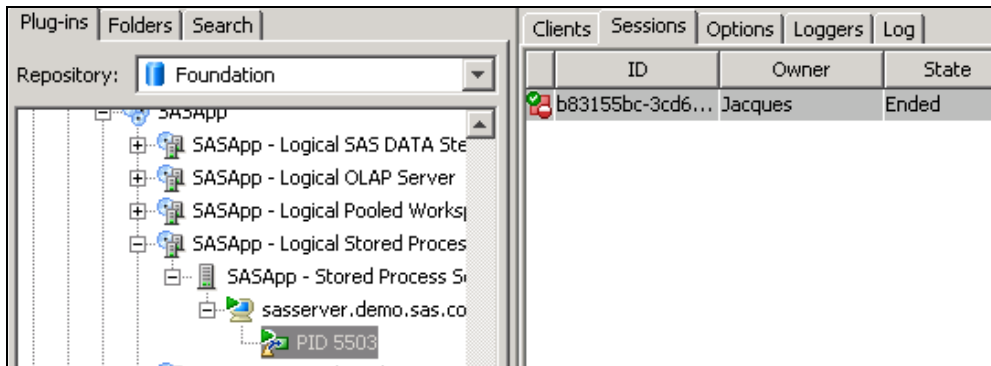


- On the Execution tab, select **Stored process server only**. Click **OK**.



- Start a SAS Enterprise Guide session. Select **Start** ⇒ **All Programs** ⇒ **SAS** ⇒ **SAS Enterprise Guide 7.1**. Close the Welcome window.
- In the Server list, expand **Servers** ⇒ **SASApp**.
- Locate the process that is running under Jacques' credentials. What is the process ID?
- In SAS Enterprise Guide, select **File** ⇒ **Open** ⇒ **Stored Process**. Navigate to **Orion Star** ⇒ **Marketing Department** ⇒ **Stored Processes**. Select **Analysis of Product Order by Gender**. Click **Open**.
- Highlight the stored process in the Process Flow window. Select **Run** ⇒ **Run Analysis of Product Order by Gender**.  
Return to SAS Management Console. What is the process ID? **The process ID varies.**  
Who is the process owner? **sassrv**
- Expand **sasserver.demo.sas.com** and select the process ID. Click the **Sessions** tab.  
Are any sessions listed? If not, why not? **The session is listed while the stored process executes. (That might be too fast to see.)**
- Return to SAS Enterprise Guide and rerun the stored process. While the stored process executes, return to SAS Management Console and select the stored process server **PID**.

Was a new process started? **No, the process was reused.**



**End of Demonstration**

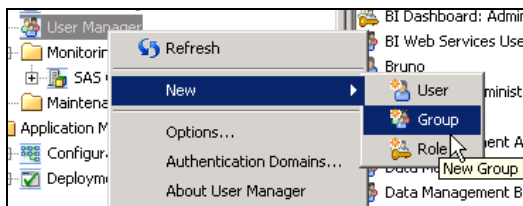


## (Optional) Configuring Access to a Database in SAS Management Console

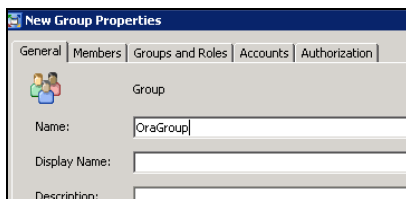
This demonstration illustrates how to create a group for the purposes of storing credentials that access a database server, define a database server, and register a library in SAS Management Console.

1. In SAS Management Console, define a group that stores credentials that authenticate to a database server.

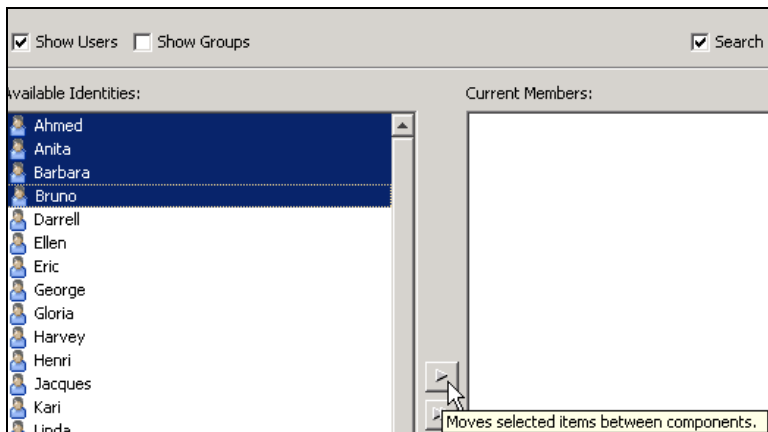
Right-click the **User Manager** plug-in and select **New** ⇒ **Group**.



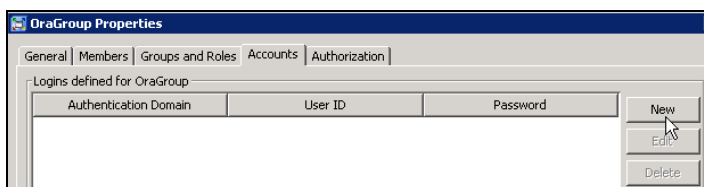
2. On the General tab, enter the group name **OraGroup**.



3. Click the **Members** tab. Clear **Show Groups**. Add the first four users that are listed by holding down the Shift key while you highlight the names. Click the arrow facing to the right.



4. Click the **Accounts** tab and click **New**.





5. Enter the following:
  - **oracleid** for the user ID
  - **Student1** for the password twice

Click **New** next to the **Authentication Domain** field to create a new authentication domain that can also be attached to the registered database server and libraries.

**New Login Properties**

Enter Login information. Enter User IDs for Microsoft Windows in the format of domain\userid. See help for details.

User ID: oracleid

Password: \*\*\*\*\*

Confirm Password: \*\*\*\*\*

Authentication Domain: DefaultAuth New

OK Cancel Help

6. Enter **OraAuth**. Click **OK**.

**New Authentication Domain**

Name: OraAuth

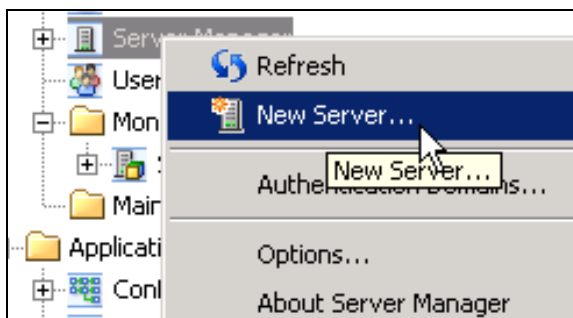
Description:

Outbound only  Trusted only

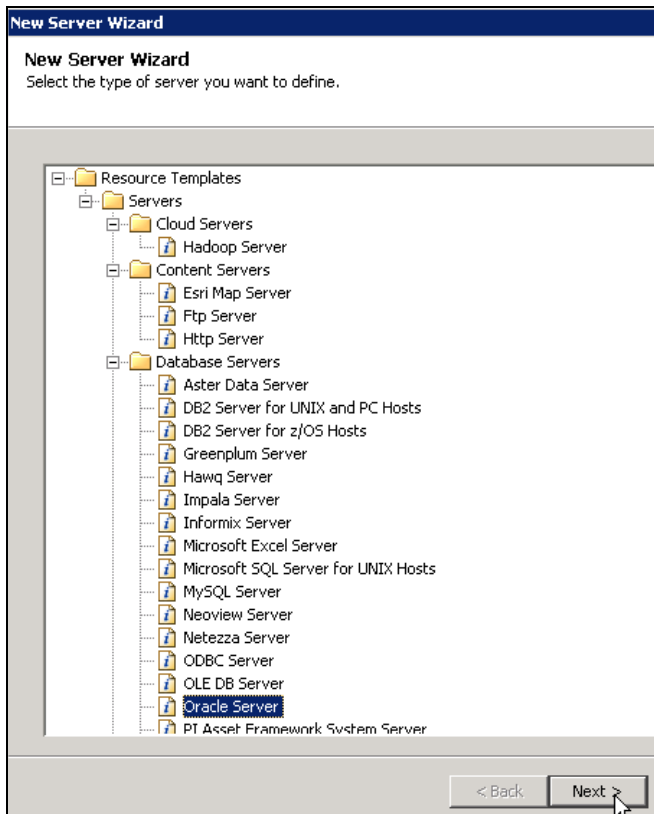
OK Cancel Help

7. Click **OK** to create the group.
8. Define the Oracle server.

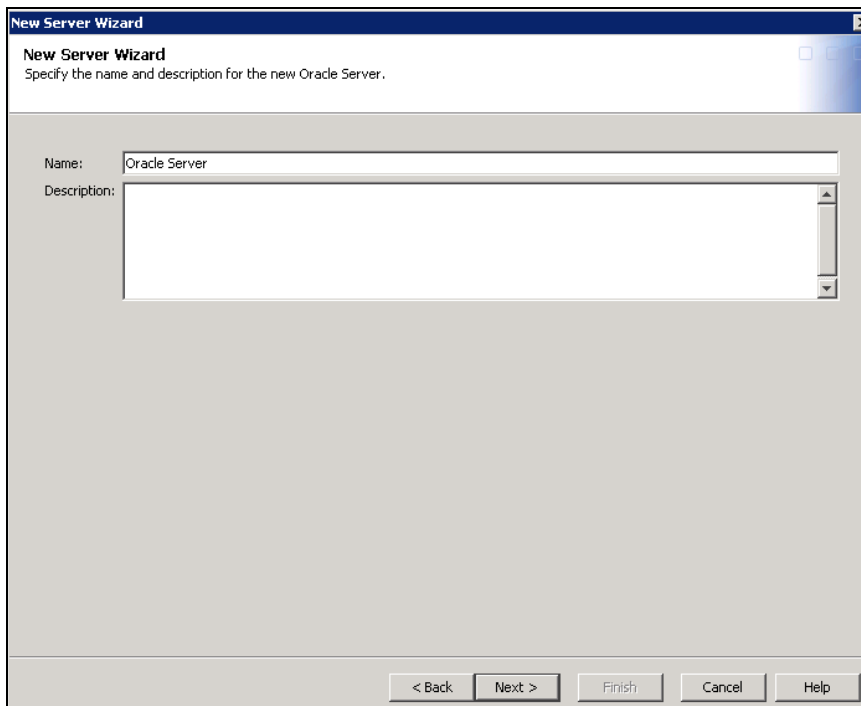
Right-click **Server Manager**. Select the **New Server** option to access the New Server Wizard.



9. Select **Oracle Server** from the Database Servers list. Click **Next**.



10. Enter an appropriate server name in the **Name** field: **Oracle Server**. You can supply an optional description. Click **Next**.



11. The server properties that are displayed in the window are default values and should not be changed. To change the **Associated Machine** property, click the down arrow at the right of the field and select the appropriate server from the drop-down list.

Click **Next**.

The screenshot shows the 'New Server Wizard' dialog box with the following fields and values:

- Major version number: 0
- Minor version number: 0
- Software version: (empty)
- Vendor: Oracle Corporation
- Associated Machine: sasserver.demo.sas.com (with a 'New...' button to the right)

Navigation buttons at the bottom: < Back, Next >, Finish, Cancel, Help.

12. Enter the following connection properties:

- **Path to the Oracle Server:** **newserver10G**. (This value is contained in the **tnsnames.ora** file that is generated during the Oracle installation. The file is stored in an Oracle installation directory such as `/opt/oracle/app/oracle/product/10.2.0/db_1/network/admin/tnsnames.ora`. The alias for the connection information is contained in this file.)
- **Authentication Domain:** Click the arrow at the right of the field and select the authentication domain that you created when you created the Oracle group. This enables the appropriate Oracle user ID and password to be used with this server.

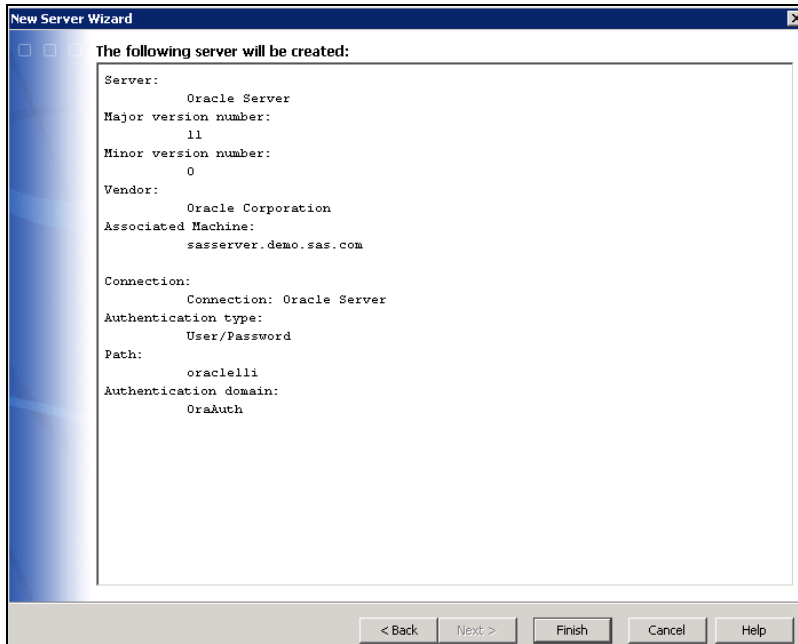
Click **Next**.

The screenshot shows the 'New Server Wizard' dialog box with the following fields and values:

- Oracle Path Information:** Enter the Path to this Oracle Server. Path: newserver10G
- Authentication Information:** Enter the authentication information needed to connect to this server.
  - Authentication type: User/Password
  - Authentication domain: OraAuth (with a 'New...' button to the right)

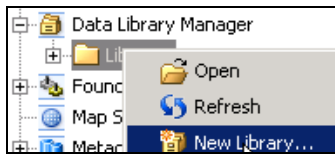
Navigation buttons at the bottom: < Back, Next >, Finish, Cancel, Help.

13. Click **Finish**.

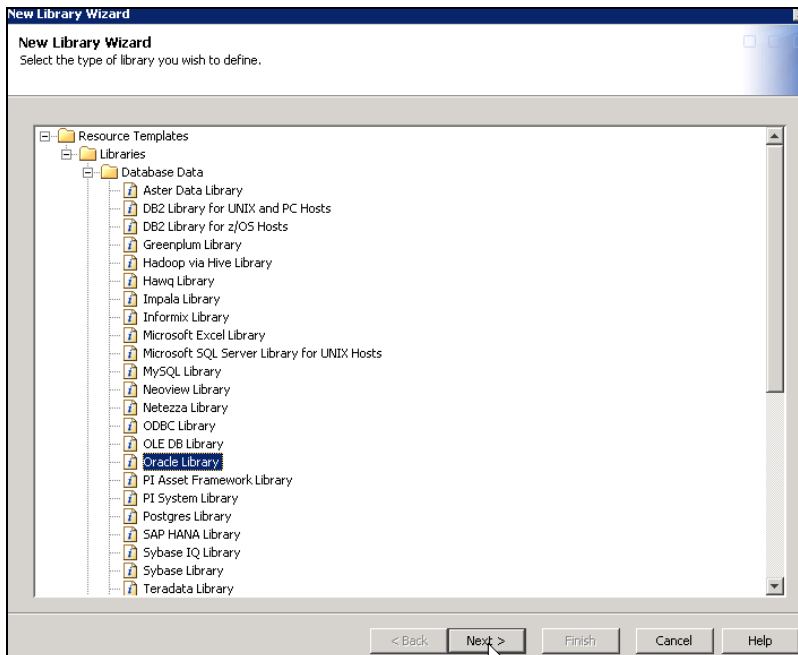


14. Define an Oracle Library.

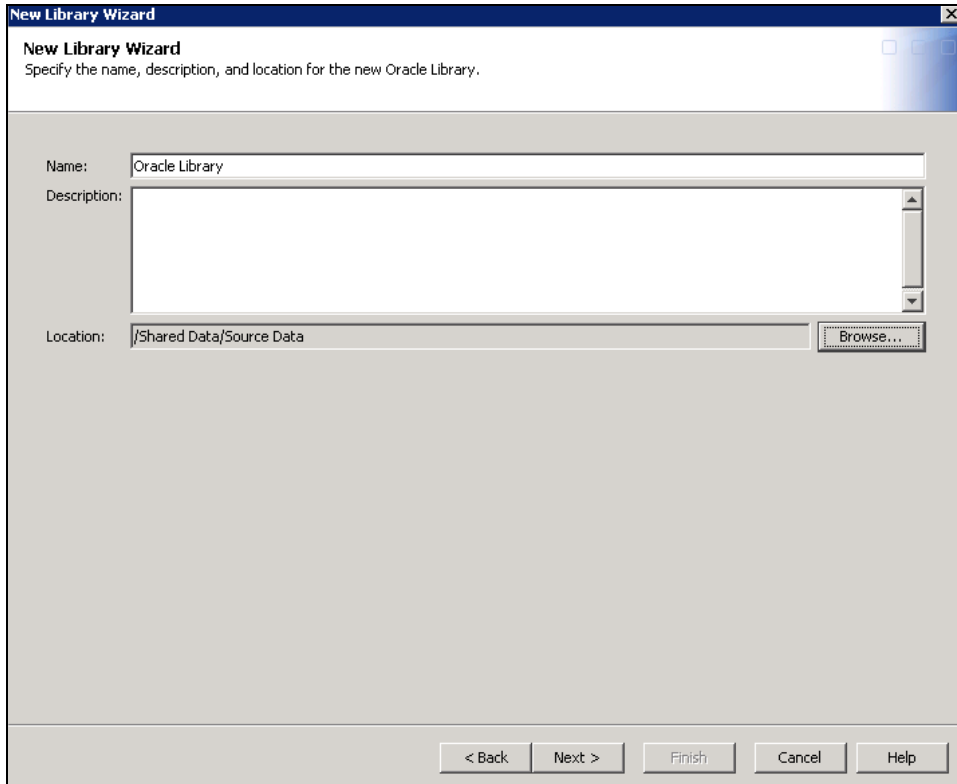
Expand the **Data Library Manager** plug-in. Right-click **Libraries** and select the **New Library** option to access the New Library Wizard.



15. Select **Oracle** from the Database Data list. Click **Next**.



16. Enter **Oracle Library** in the **Name** field. Click **Next**.



**New Library Wizard**  
Specify the name, description, and location for the new Oracle Library.

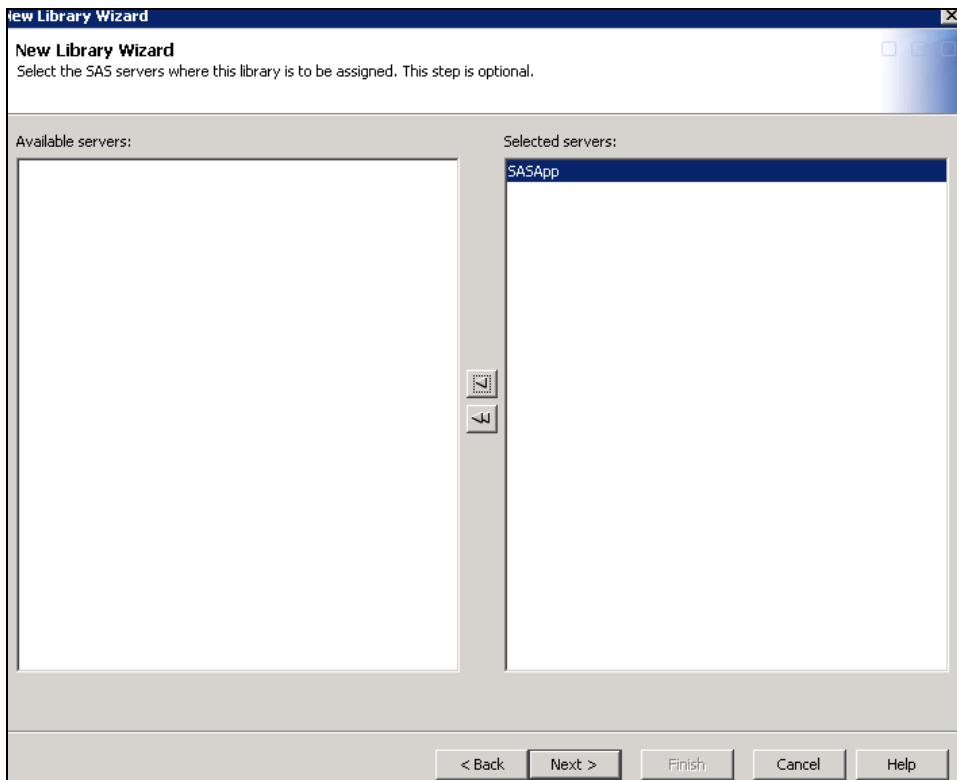
Name: Oracle Library

Description:

Location: /Shared Data/Source Data

< Back   Next >   Finish   Cancel   Help

17. Move **SASApp** over so that this library is assigned to the SASApp server context. Click **Next**.



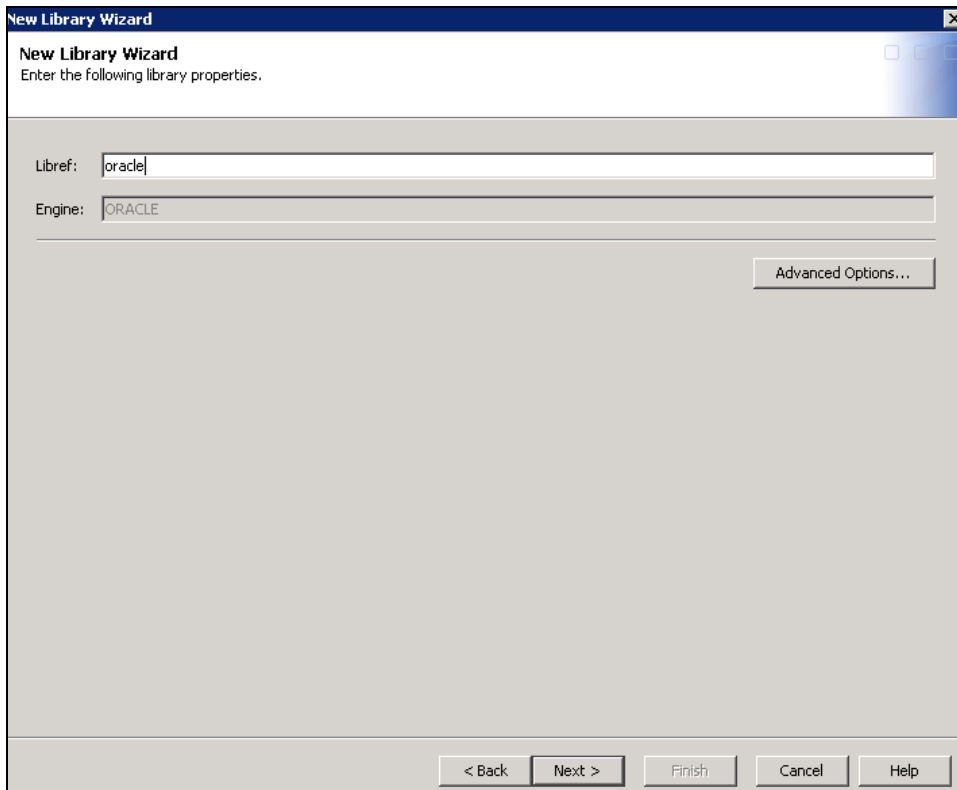
**New Library Wizard**  
Select the SAS servers where this library is to be assigned. This step is optional.

Available servers:

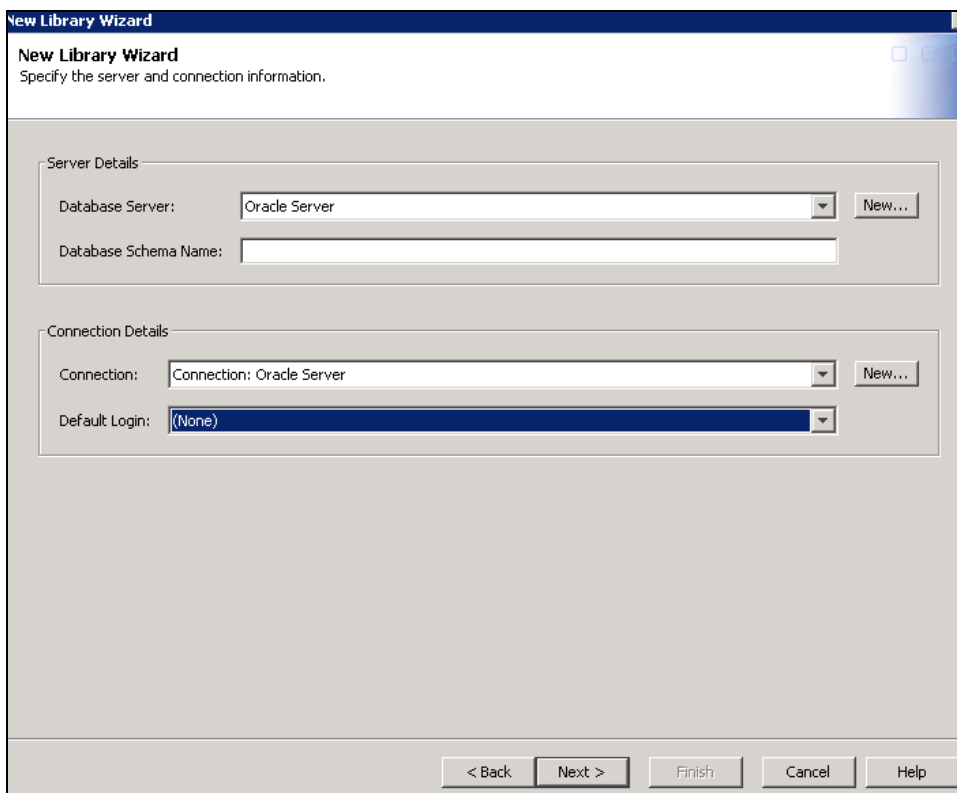
Selected servers:  
SASApp

< Back   Next >   Finish   Cancel   Help

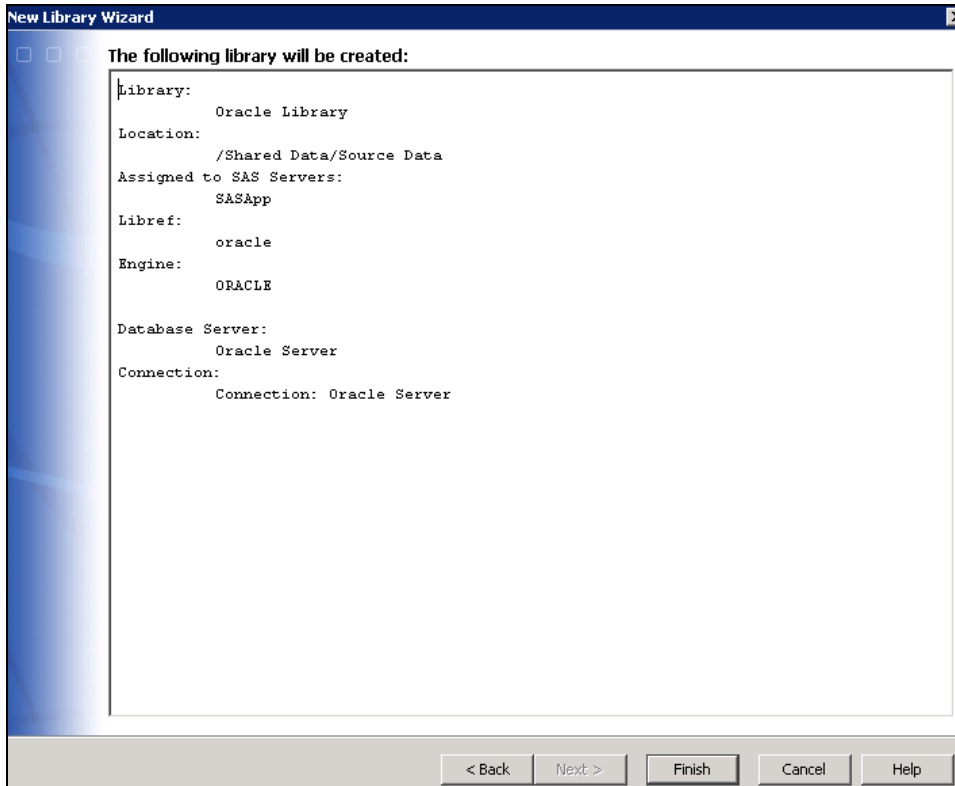
18. Enter **oracle** as the libref. Click **Next**.



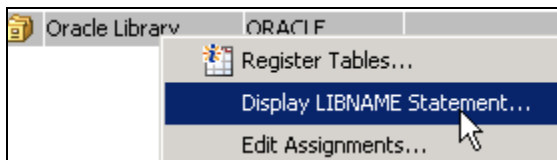
19. The database server is **Oracle Server**. For the database schema name, enter **Scott**. Click **Next**.



20. Click **Finish**.

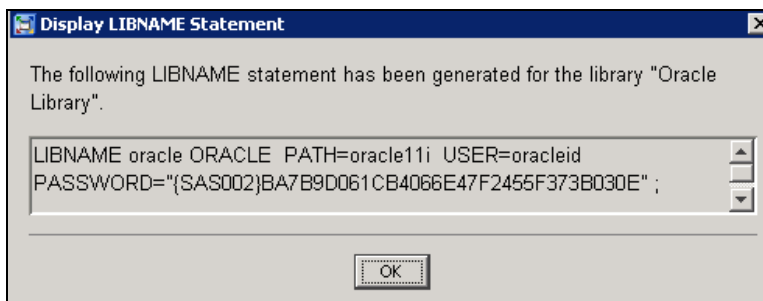


21. Right-click **Oracle Library** and select **Display LIBNAME Statement**.



22. The interface generated the LIBNAME statement that is processed when a user in that group accesses Oracle tables from this library, but they are not prompted.

**Note:** If you are logged on as the unrestricted user, you are prompted because the unrestricted user cannot retrieve passwords from metadata.



**End of Demonstration**

