Paper 10962-2016

SAS® Metadata Security 201: Security Basics for a New SAS Administrator

Charyn Faenza, F.N.B. Corporation

ABSTRACT

The purpose of this paper is to provide an overview of SAS® metadata security for new or inexperienced SAS administrators. The focus of the discussion is on identifying the most common metadata security objects such as access control entries (ACEs), access control templates (ACTs), metadata folders, authentication domains, etc. and describing how these objects work together to secure the SAS environment. Based on a standard SAS® Enterprise Office Analytics for Midsize Business installation in a Windows environment, this paper walks through a simple example of securing a metadata environment, which demonstrates how security is prioritized, the impact of each security layer, and how conflicts are resolved.

INTRODUCTION

For most people, when they imagine a SAS Administrator, they envision someone who is highly technical and has many years of SAS experience, with extensive knowledge of the servers, network, etc. that their firm's SAS installation runs on. They often assume their administrator has a degree in computer science and has attended a great deal of SAS training to prepare them to support users, managers and their IT peers. While undoubtedly, there are many administrators that fit this profile, in my experience there are just as many administrators that have Finance, Marketing or Statistical backgrounds from a multitude of many diverse industries. These so-called "Accidental Admins" often become administrators simply because there is no-one else to do the job!

Fortunately, there is plenty of support material to be found online, including technical manuals on the SAS Support site, conversations on the SAS Community forums and many user-written white papers, such as this one. On the other hand, the material can be daunting if you are just starting out due to the technical nature of the role. Throughout the course of this paper we focus one of the most important aspects of an administrator's job – Security. For some, the mere idea of having to architect and maintain SAS security in their environment is a source of anxiety. Armed with a better understanding of commonly used security terms and how security works in a basic SAS environment, any new administrator, accidental or otherwise, will be able to use all of the resources at their disposal to approach this critical task with confidence.

SAS CONFIGURATIONS & THE FILE SYSTEM

The environment that is used for this example is a standard installation of SAS 9.4 Office Analytics on a single Windows Server using Windows Active Directory as the authentication provider. It is important to note, however, that SAS metadata security is highly customizable so the configurations and examples in this paper are presented only for illustrative purposes. Individual configurations and recommended settings will vary based upon a site's unique requirements.

This paper focuses on managing the SAS environment once it has been installed and configured. The reader should also bear in mind that there are several decisions made during the installation that have an impact on your security framework. You can find more information about the initial security configurations in the paper: *SAS® Metadata Security 101: A Primer for SAS Administrators and Users Not Familiar with SAS*.

THE THREE A'S OF SECURITY

The first place to start a discussion of security is a review of the three As: Authentication, Authorization and Audit. The *SAS® 9.4 Intelligence Platform: Security Administration Guide, Second Edition* defines the terms as follows:

Authentication:

> *The process of verifying the identity of a person or process for security purposes.*
>
> This answer the question "Who are you?"

Authorization:

> *The process of determining which users have which permissions for which resources. The outcome of the authorization process is an authorization decision that either permits or denies a specific action on a specific resource, based on the requesting user's identity and group memberships.*
>
> This answer the question "What are you allowed to do?"

Audit:

> *The process of logging of security-related events.*
>
> This answer the question "What did you do?"

Managing SAS auditing and logging functionality is out of scope for this paper; the topic is adequately covered in detail in numerous other papers. The scope of this discussion is on the subjects of Authentication and Authorization and basic tools used to administrate security.

SAS ADMINISTRATION TOOLS

As the name implies, SAS Management Console is the program administrators use to manage metadata objects in the SAS environment, including security (**Figure 1**).

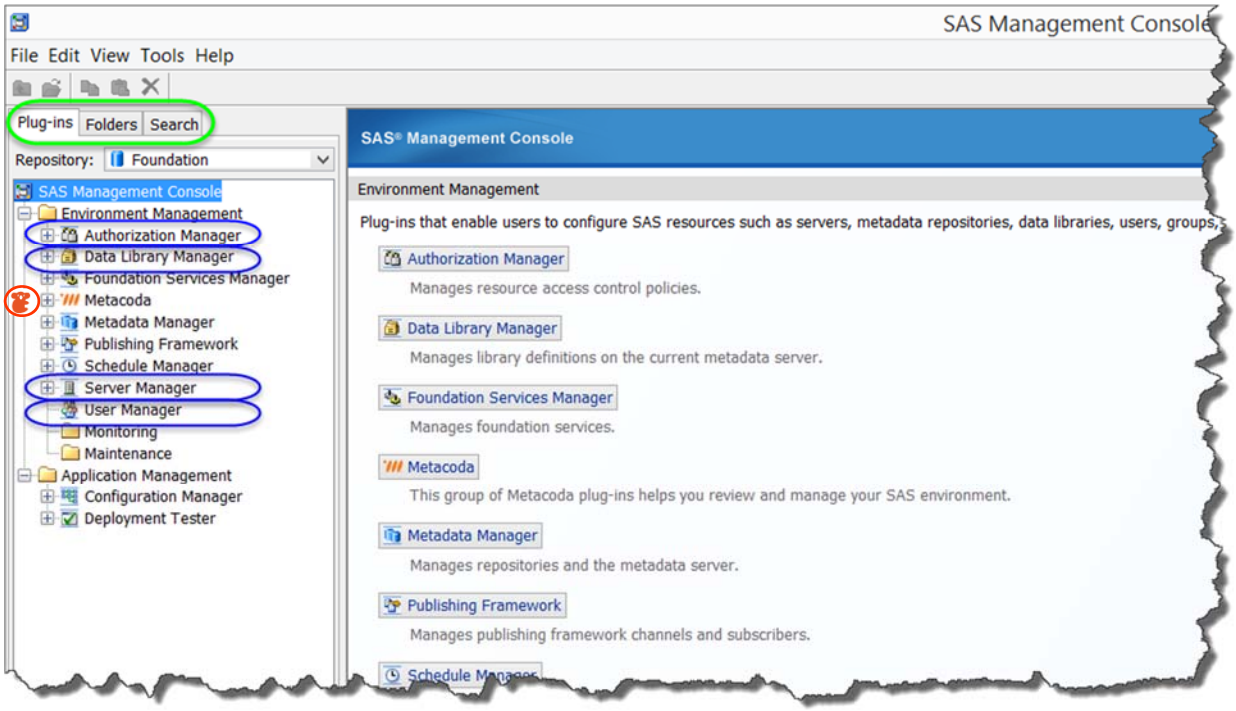


Figure 1 - SAS Management Console Interface

The interface for Management Console includes three views: **Plug-Ins**, **Folders**, and **Search**. Administrators should take the time to become familiar with each of these views and their purpose.

SAS Management Console Plug-Ins

The **Plug-Ins** tab provides administrators a list of modules that they can use to create or maintain metadata for a specific type of resource. There are several common out-of-the-box plug-ins:

- Authorization Manager – used to control users and group access
- Data Library Manager – used to create and maintain library definitions
- Foundation Services Manager – used to view and modify deployment configurations
- Metadata Manager – used to administrate the SAS Metadata Server (including backing up and restoring metadata repositories)
- Server Manager – used to perform administrative tasks related to SAS servers
- User Manager – used to create and maintain users, groups, and roles

The number and type of plug-ins available vary based upon the products deployed in the SAS environment. In **Figure 1**, the plug-ins referred to in this paper are circled in blue. Additionally, third party plug-ins that can be added to SAS Management Console for additional metadata management functionality, such as the instance of **Metacoda** visible in the figure above.

SAS Management Console Folders

The **Folders** tab allows administrators to view and manage the hierarchy of SAS metadata folders. SAS folders are metadata objects that are created to help organize and secure other metadata objects such as libraries, tables, reports, etc., as well SAS content that is not directly accessed by the SAS user. While it is possible to secure virtually any metadata object (i.e. individual tables, users, etc.), it is a good practice to secure objects at the metadata folder level whenever possible.

A closer look at the relationship between the physical file folder structure and the metadata file folder structure is shown in **Figure 2**. To make things less confusing for the users, the metadata folders in our sample environment is set up with the same structure and naming conventions as the physical folders; however, this is not required, and is often not the case. For example, the metadata folder called IT could be renamed as Information Technology in metadata with the metadata library and its tables in the new Information Technology folder pointing to the **F: /Files/IT** location.

SAS security is only applied to the object's metadata file structure. SAS Administrators should bear I mind that this is not the same as securing the physical table on the file system. The security of the file system is the responsibility of the host's authentication provider, for example, Active Directory.

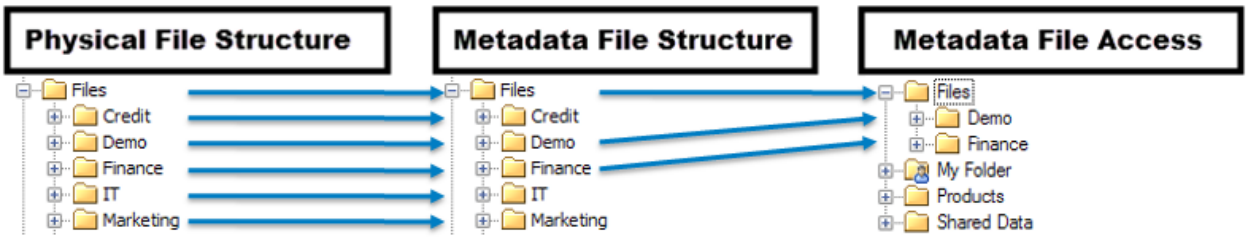


Figure 2 - Physical Folders versus Metadata Folder View

Some of the common tasks that an administrator performs on SAS metadata folders include:

1. Creating the folder structure for the organization. While users can, and do, create their own folders, administrators can control, though security, the metadata locations users may do this in order to maintain an orderly environment.
2. Setting permissions on the folders and objects. In the SAS Management Console, the Folders tab is where the majority of the security is applied.
3. Importing and exporting metadata and associated files. Using the Folders tab, specific collections of data can be imported or exported as needed, for example, when promoting content to production.

SAS Management Console Search

The **Search** tab (**Figure 3**) gives administrators the ability to search through the SAS metadata folders to locate specific objects. Once located the objects are displayed in the search results window and basic properties can be reviewed.

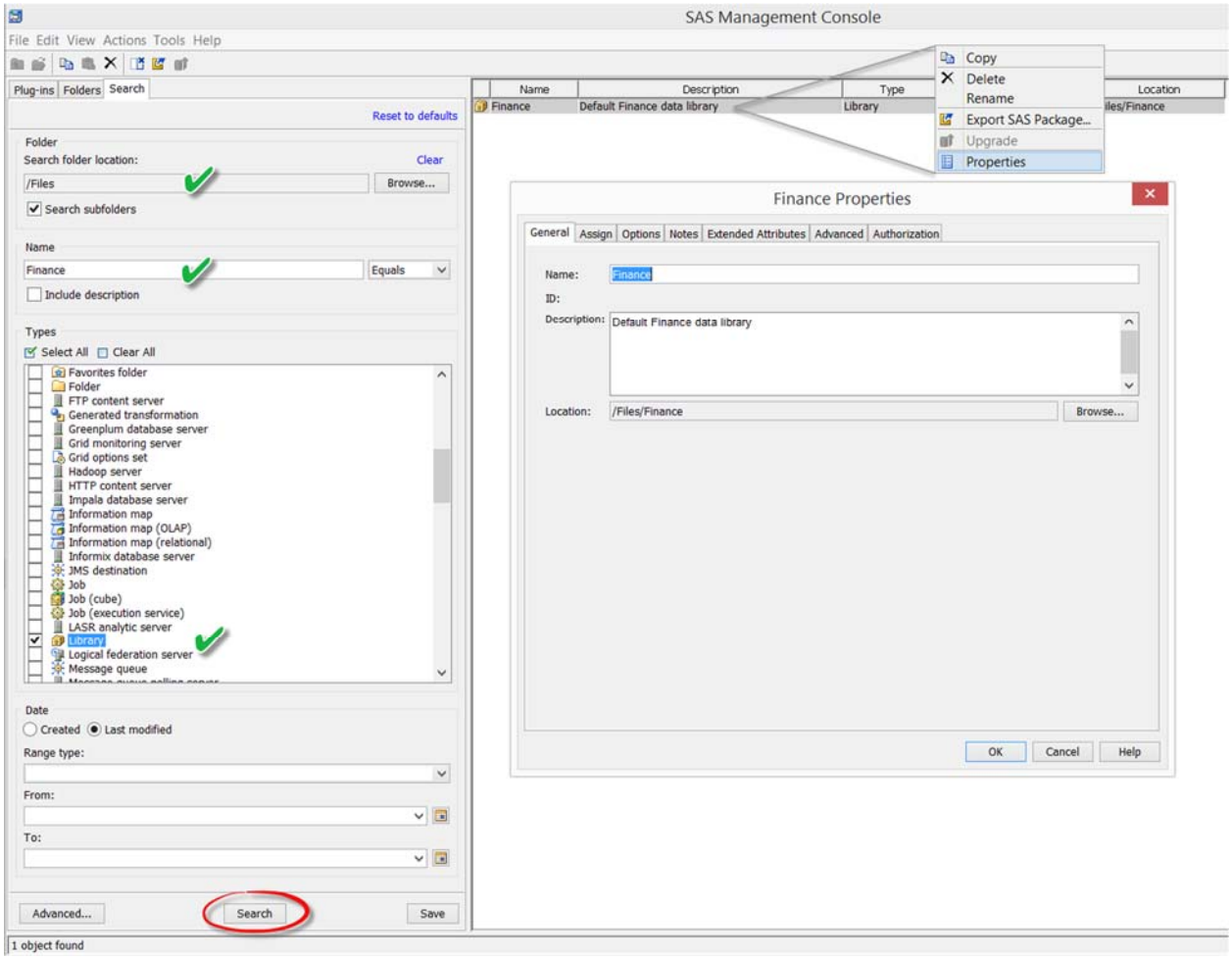


Figure 3 - SAS Management Console Search

METADATA SECURITY TERMS

In addition to the interface, there are several security terms an administrator should be familiar with.

User Metadata Identity

Often referred to as the SAS user or SAS identity, the user metadata identity is the user definition that is set up for an individual in SAS Management Console and associated with the user's login information. For example, a later example demonstrates how the user metadata identity **KFrog** is associated with the employee **Kermit**.

Group Metadata Identity

An administrator can register and define a user group in SAS Management Console, an associate one or more user metadata identities and their login definitions to the group. Grouping multiple identities allows that administrator to apply and manage security permissions to several individuals by assigning the permissions at the group level. These groups include groups created to manage the environment as well as two system groups:

- Public – A system group that includes everyone that can log on
- SASUSERS – A system group that includes everyone that is a registered user

In the subsequent examples, the SAS user **KFrog** is a member of the Sesame Street group.

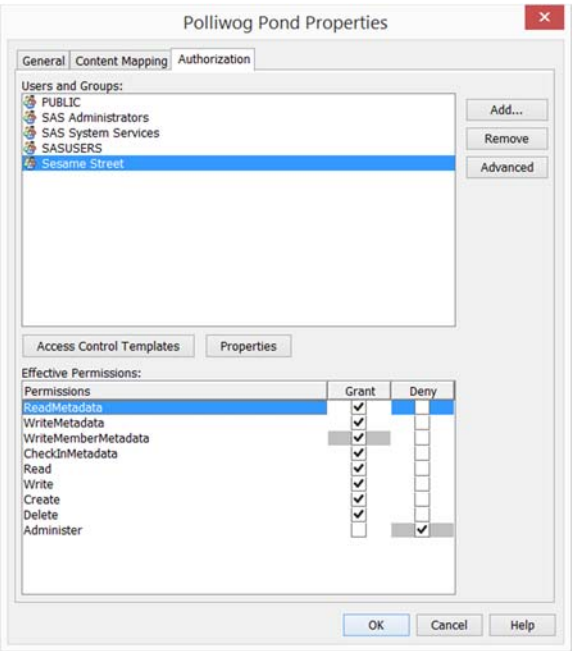


Figure 4 - Access Control Entries on the Polliwog Pond Metadata Folder Object

Access Control Entry (ACE)

The term **ACE** stands for Access Control Entry. The explicit grant or denial of a user or group on an object is called an Access Control Entry. This method of securing an object is the simplest and most direct way to apply security; however, in a large or complex environment, this can quickly become very difficult to maintain. For example, let's suppose you need to provide read access to report writers from the Marketing, Finance, and Sales departments to data located in 10 different metadata folders for various databases. Assuming you have created a group for each department's report writers, that is still 30 unique ACEs.

Clearly, as the number of users, groups, and objects increases, the number of items that need to be maintained expands quickly. Fortunately, there is a better way!

Access Control Template (ACT)

The term **ACT** stands for Access Control Template, which is a security template that holds the pre-defined permission patterns for user and groups. When these templates are applied to objects the security settings for each of the groups in the template are established on the object. In the example above the template would contain the three groups (Marketing, Finance, and Sales) with each set to grant read-only access. By applying the template to the folders instead of the groups, the administrator's maintenance is reduced from three entries to one.

Additionally, because the template was used to secure the folders, any changes to the permissions made on the template are automatically passed on to each of the metadata objects the template secures. For example, adding a new reporting group for the Production department to the ACT above applies the permissions for the Production department to all of the databases secured under the ACT instead of having to apply the group to each database folder individually.

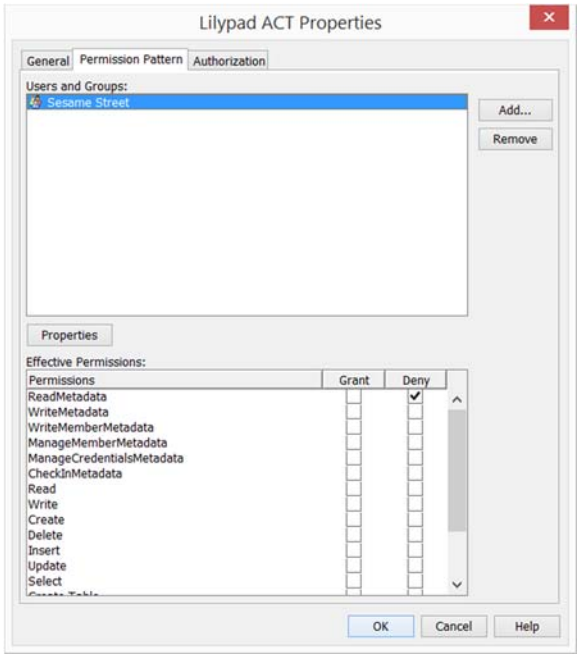


Figure 5 - Access Control Template

SAS Authentication Domain

A **SAS Authentication Domain** is the name of a reference that pairs the user's login credentials with the servers the users need to access. All users start with one authorization domain, typically the **DefaultAuth**, which connects them to their primary network's authorization domain, for example, Active Directory. Additional authorization domains are frequently used to provide access to external databases, where providing each user with a unique user name and password, or surfacing a system access user name and password and sharing it amongst business users would be undesirable

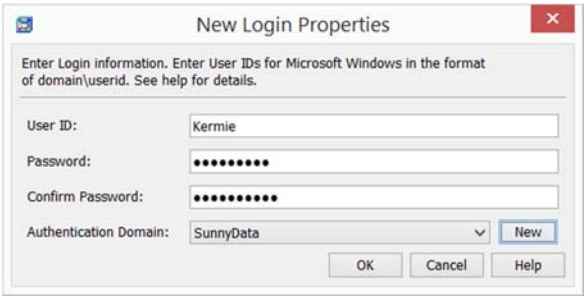


Figure 6 - Authentication Domain

In order to pass credentials to its users the administrator must set up an **Authentication Domain** for the target data source.

When a user attempts to access a data source, the SAS Metadata Server reviews the list of authentication domains associated with that users to determine if they have a log in to the desired data source.

If a match is found, the username and password stored in the authentication domain is used to authenticate the user.

To simplify the process for both users and administrators, the authentication domain can be assigned to a user group instead of a user, provided this is consistent with the security policies of the company. Any user that needs to access that resources is then simply added to the group

Relationship Networks

The *SAS® 9.4 Intelligence Platform: Security Administration Guide, Second Edition* identifies two paths by which permissions can be conveyed, referred to as relationship networks.

Relationship Network: Object Inheritance

The first relationship network, **object inheritance**, is based upon the relationship between of the permissions on the object and the permissions on the objects it is related to. In object inheritance, an object may be affected by another object's permissions and its permissions may affect many other objects. For example, a folder may inherit permissions from a parent folder and, likewise, it passes permissions to any objects (folders, libraries, tables, etc.) it contains. The inherited security settings are referred to as indirect permissions settings. Settings that are applied on an object are called direct settings. To resolve conflicts between multiple direct and indirect settings, the following order of precedence is followed:

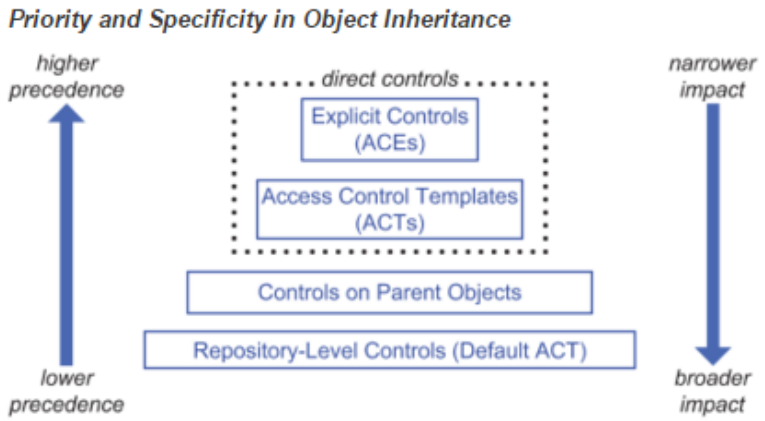


Figure 7 - Object Inheritance (SAS® 9.4 Intelligence Platform: Security Administration Guide, Second Edition)

Relationship Network: Identity Inheritance

The second relationship network is the **identity relationship**. This network governs the identity that the permissions are being granted to (or denied from). Similar to the object inheritance network, identities can affect, and be affected by, other identities. These relationships start with the primary identity (usually a specific user) and typically include a number of groups that have varying levels of precedence. The resulting order of precedence is called the **identity hierarchy** and, like the object inheritance order of precedence, it creates a standard by which permissions conflicts are resolved:

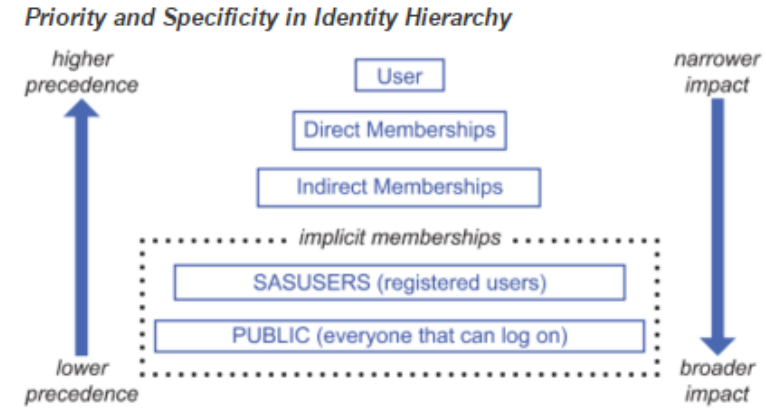


Figure 8 – Identity Hierarchy (SAS® 9.4 Intelligence Platform: Security Administration Guide, Second Edition)

It is important to note that in **Figure 7** and **Figure 8**, impact and precedence have an inverse relationship. That is, controls that have a higher precedence, typically affect a smaller set of objects than controls with a lower precedence. In both the object inheritance path and the identity hierarchy, it is the controls that have the greatest level of specificity that have the greatest precedence. More simply stated, the general rule of thumb is: The closer the security is applied to an object or user, the higher the priority when there is a conflict.

AUTHORIZATION DECISIONS

SAS uses both the object inheritance path and the identity hierarchy when determining whether or not permissions are applied to an object. The *SAS® 9.4 Intelligence Platform: Security Administration Guide, Second Edition* outlines each step that is taken in the access control evaluation process. Administrators should familiarize themselves with these rules prior to starting their security framework plans. They are listed below for reference:

1. The relative precedence of each access control is based on where it set, who it is assigned to, and how it is set. The following list summarizes how the metadata server evaluates all relevant access controls to reach an authorization decision:
   - Direct controls (permissions that are set directly on the target object) are examined. Any conflicts that arise from group membership are resolved by the identity hierarchy. For example, an explicit control that is assigned to a user overrides a conflicting explicit control that is assigned to a group to which the user belongs.
   - If there is a conflict between an explicit control and an ACT setting at the same level in the identity hierarchy, then the explicit control takes precedence.
   - If there is a conflict between two explicit controls (or two ACT settings) at the same level in the identity precedence hierarchy, then the outcome is a denial.
   - If one or more permission conditions have been defined, then the condition that is assigned at the highest level of identity precedence is applied. Other conditions that also apply to a user because of group memberships do not provide additional, cumulative access (unless there are multiple tied groups at the highest level of identity precedence).
   - If there are no relevant direct controls, then the evaluation process continues.

2. The step 1 evaluation process is applied to the immediate parent of the target object. For example, the immediate parent of a report is its folder. If no direct controls are found, each successive parent object is examined in turn.

   Note: In the unusual circumstance in which an object has more than one immediate parent, a grant from any inheritance path is sufficient to provide access.

3. If no direct controls are found on the object or on any of its parent objects, the permission pattern of the repository ACT (Default ACT) is examined. The repository ACT serves as the inheritance parent of last resort.
   - If the repository ACT grants or denies the requested permission, then that grant or denial is determinative.
   - If the repository ACT neither grants nor denies the permission, then the permission is denied.
   - If there is no repository ACT, then the permission is granted. You should always have a designated repository ACT.

ESSENTIAL SECURITY TASKS

The following section addresses basic security tasks that are commonly performed by all administrators. These tasks can be categorized into three distinct areas: Library Management, Folder Management, and User Management. In an existing SAS environment, administrators typically work within an established security framework to add and modify users, groups, folders, and libraries.

On the other hand, when the administrator is creating a new environment, or restructuring an existing environment, the first and most crucial step is to draw up the security model. This model should be reviewed and testing prior to changing the system. Generally speaking, the order in which tasks are performed when implementing a new security framework is:

1. Create Default ACT
2. Create folder hierarchy
3. Create security groups
4. Create ACTs
5. Apply the security model to folders
6. Create libraries and register tables within folders
7. Test permissions (SASDEMO is often used for testing)
8. Create users
9. Add users to groups
10. Review and verify security with users

In an existing environment the most frequent tasks are adding new users and modify existing users; however, as the business changes administrators also need to update folders, create new groups, register new libraries, etc. as well.

User Management

Adding and Modifying Users

The first action in user management is setting up the users. All users must have unique identity set up in SAS. For simplicity, user names in SAS can be created to exactly match the user names in the host authentication provider (i.e. Active Directory); however, this is not required. The user's identity in SAS is paired with the user's identity in Active Directory on the **Accounts** tab of the **User Properties** dialog box in SAS Management Console.

To add a new user:

1. On the **Plug-ins** tab, select **User Manager**.
2. Right-click and select **New → User**.
3. On the **General** tab, enter the SAS user name and Display Name. Add additional details if desired.
4. On the **Accounts** tab, click **New**. In the New Login dialog box, select the **DefaultAuth** authentication domain and enter the user's domain account ID. It is not necessary to include a password in this login.
5. Click **OK** to save the new login. Click **OK** again to save the new user.
6. Use the **Groups and Roles** tab to add the user to the appropriate groups.

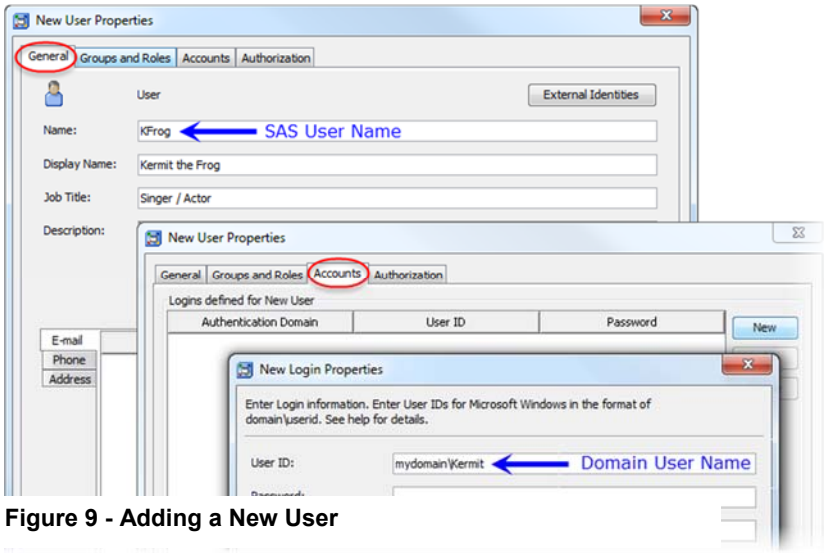


Figure 9 - Adding a New User

While the users exist and must be managed in both realms; the enforcement of corporate password policy resides solely with the authentication provider. SAS pairs the SAS user name with the domain Username / Password credentials provided when the user logs in to the system.

Note: Users are automatically added to PUBLIC and SASUSERS when they are created. The administrator does not need to explicitly add users to those groups.

To modify an existing user:

1. In **User Manager**, select the user, group, or role that you want to modify.
2. Right-click and select **Properties**.
3. Select the desired tab to change the user's details, or add additional accounts, groups, or roles.

Adding and Modifying Groups

While it is possible to assign permissions to individual users, it is much more efficient to manage access controls using groups. Once a group is created and either added to an ACT or used to assign permissions directly to an object (ACE), the administrator can simply move users in and out of the appropriate groups as needed to manage their level of access. In addition, groups can also be used to manage roles or to make shared credential available to multiple users.

To add a new group:

1. On the **Plug-ins** tab, select **User Manager**.
2. Right-click and select **New → Group**.
3. On the **General** tab, enter the group name.
4. On the **Members** tab, add the users who are to be members of this group. Note: You can also add groups as members of other groups.
5. On the **Groups and Roles** tab, add the group as a member of another groups (optional)
6. On the **Accounts** tab, add a shared login if you are using this group to share credentials. (optional)

To modify an existing group:

1. In **User Manager**, select the group that you want to modify.
2. Right-click and select **Properties**.
3. Select the desired tab to change the group's details, or add additional accounts, users, groups, or roles.

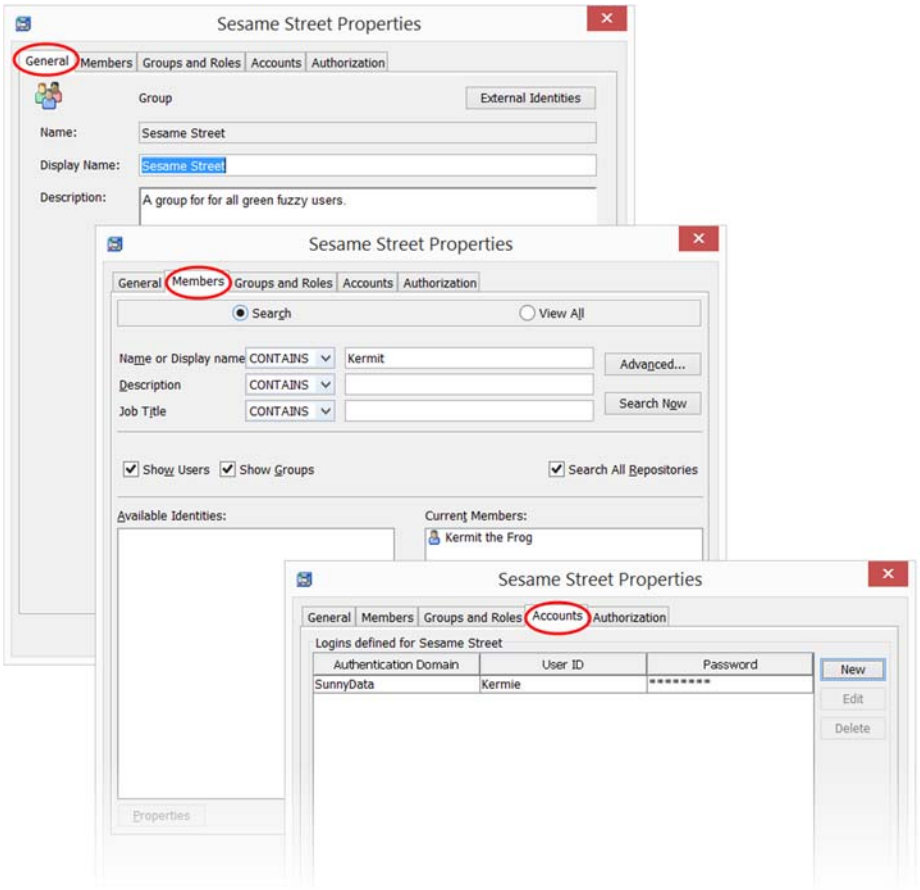
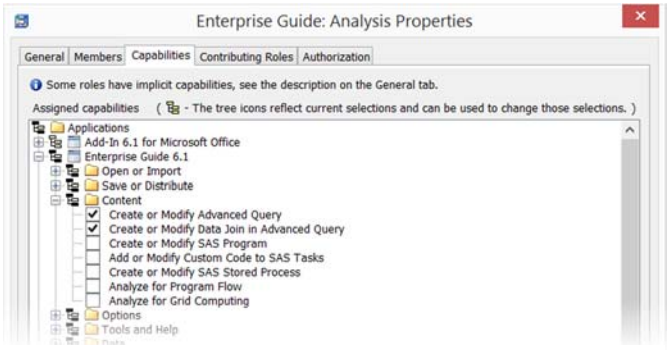


Figure 10 - Adding a New Group

For additional details about adding new users or group, please consult the *SAS® 9.4 Management Console: Guide to Users and Permissions.*

Roles

Roles control which application features (such as buttons, tabs, and menu items) are visible to the user. These features are referred to as capabilities. For example, in SAS Enterprise Guide, a role can control whether the user has the option to create a new program *in that specific interface*. Capabilities for SAS programs are governed individually and users may have many roles.



Capabilities are additive; therefore, in the SAS Enterprise Guide example, the user may not have the ability to create a new program in SAS Enterprise Guide in one role, but may be retain the ability to create a new program if that capability has been granted in any other role that has been assigned.

Figure 11 - Enterprise Guide Analysis Role (partial)

Not all applications have roles and not all features are governed using roles. In addition, a feature that is not managed using roles cannot be converted to a capability and will remain available to the user; however, any custom tasks (SAS Enterprise Guide) or custom plug-ins (SAS Management Console) can be registered as capabilities.

Lastly, SAS provides a number of out-of-the-box roles. It is strongly recommended that administrators wishing to modify the capabilities that are granted to users should create new roles to do so. This preserves the original SAS provided roles as templates that can be used to restore capabilities in the event of an undesirable outcome when changes are applied.

Differences between Roles and Groups

While they may seem similar at first, there are several key distinctions between roles and groups. The *SAS® 9.4 Intelligence Platform: Security Administration Guide, Second Edition* lists the following five differences administrators should be aware of:

- Roles and groups serve distinct purposes. You cannot assign permissions to a role or capabilities to a group.
- The identity hierarchy is relevant for groups, but not for roles. If you are a member of a role, you have all of that role's capabilities, regardless of whether you are a direct member of that role and what your other memberships are.
- You can deny a permission to a group, but you cannot deny a capability to a role. Each role either provides or does not provide each capability. No role takes capabilities away from its members.
- A group's permissions are not displayed as part of a group definition, but a role's capabilities are displayed as part of a role definition.
- A group can be a member of another group, but a role cannot be a member of another role. Instead, one role can contribute its capabilities to another role.

Adding and Modifying Access Control Templates (ACTs)

SAS provides several predefined ACTs such as the Default ACT and the SAS Administrator ACT; however, an administrator may want to create a custom ACT to centrally manage permission patterns that are used repeatedly. The most frequent use case for this is to manage the visibility of content to the SAS users.

For example, an ACT can be created that prevents all users, except administrators, from seeing any SAS content by denying ReadMetadata to the PUBLIC group and applying to where the desired default permission is to deny read access. This can be at the repository level, department level, etc., depending upon how the security framework is established.

Due to the order of precedence in the Identity Hierarchy, this denial will be overridden by a grant of ReadMetadata on any object to SASUSERS, a specific group, or a specific user. This common approach to security is referred to as “Deny-Grant Back” due to the blanket denial of the general user and the granting back of permissions only to authorized users.

To create a custom ACT:

1. On the Plug-ins tab, expand Authorization Manager
2. Right-click Access Control Templates, and select New Access Control Template.
3. On the General tab, enter the ACT name.
4. On the Permission Pattern tab, click Add and in the Add Users and Groups dialog box, select the desired identities and move them to the Selected Identities list box. Click OK.
5. On the Permission Pattern tab, define explicit settings for each identity.
6. Click OK to close the dialog box and save the ACT.

Folder Management

As discussed previously, folders are often where permissions are assigned to groups, either through an ACE or through an ACT so this is a good point to review what permissions are available.

Permissions Definitions and Planning

The following charts describe what each permission abbreviation means and what actions are controlled by the permission. Permissions are broken down into two categories: General-Purpose Permissions and Specialized Permissions.

General Purpose Permissions

Abbreviation	Permission	Actions
RM	ReadMetadata	View an object.
WM	WriteMetadata	Edit, delete, or set permissions for an object.
WMM	WriteMemberMetadata	Add an object to a folder or delete an object from a folder.

Specialized Permissions

Abbreviation	Permission	Actions
A	Administer	Operate certain SAS servers and spawners.
C	Create	Add data through the metadata LIBNAME engine.
R	Read	Read data through the metadata LBNAME engine.
W	Write	Update data through the metadata LIBNAME engine.
D	Delete	Delete data through the metadata LIBNAME engine.
MMM	ManageMemberMetadata	Change the membership of the Group and Role.
MCM	ManageCredentialsMetadata	Manage accounts and trusted logins of User and Group.

Source: SAS® 9.4 Intelligence Platform: Security Administration Guide, Second Edition.

WriteMetadata and WriteMemberMetadata are special permissions designed to protect the integrity of the metadata folder structure. WriteMemberMetadata allows users to contribute data to a folder, but disallows them from deleting, removing, or renaming the folder, while WriteMetadata allows those actions. A key point to note is that when a user is able to interact with an object, they are also granted the ability to update the settings on that object. Similarly, if a user is able to write to a folder, they will be able to the create subfolders underneath that folder.

Fortunately, once a folder structure is in place, object inheritance minimizes the maintenance required on subdirectories that are created by users; however, changing business needs can require the administrator to add, delete, or make changes to the existing folders in the SAS environment.

In addition to thinking through the appropriate security when planning changes, administrators should consider the following folder management tips from the *SAS® 9.4 Intelligence Platform: System Administration Guide, Fourth Edition*:

- It is recommended that you not delete or rename the **User Folders** folder, even if you have permission to do so.
- As a best practice, do not rename an active user's home folder or personal folder. If you do so, a new (empty) personal folder will be created the next time the user refreshes or logs on to an application that requires the folder. In addition, the contents of the renamed folder will not be visible to the user.
- If you delete an active user's home folder or personal folder, the user will lose any existing personal content, and a new (empty) personal folder will be created the next time the user refreshes or logs on to an application that requires the folder.
- Do not delete or rename the **Products** or **System** folders or their subfolders, even if you have permission to do so.

Lastly, in order for a user to navigate to a folder, there must be a ReadMetadata permission path within the folder structure the user can follow. Users are not able to browse past folders unless they are granted ReadMetadata access to them.

Permissions Application

To add direct permissions (ACE) on a folder for a group:

1. On the **Folders** tab, right-click the folder and select **Properties**.
2. On the test folder's **Authorization** tab, click **Add**. Move the desired group o the **Selected Identities** list box and click **OK**.
3. On the **Authorization** tab, the group will have an explicit grant of ReadMetadata permission, designated by the white check box, automatically given. Any permissions that have been inherited by a parent folder will be indicated by a gray check box, and permissions derived from an ACT will be indicated by a green check box.
4. Update the desired permissions by selecting either the grant or deny opposing check box for the permissions you wish to manage and clicking through the following options until the desired option is selected: explicitly clear, explicitly select, inherit.
5. Click **OK** to close the **Properties** dialog, and apply the changes.

To apply an access control template (ACT) to a folder:

1. On the **Folders** tab, right-click the folder and select **Properties**.
2. Click **Access Control Templates**. In the Add and Remove Access Control Templates dialog box, expand the **Foundation** node in the Available list box and select the desired ACT.
3. Click **Properties** to verify the settings that this ACT provides by examining the **Permission Pattern** tab.
4. Click **Cancel** to return to the ACT list.
5. In the Add and Remove Access Control Templates dialog box, move the desired ACT to the **Currently Using** list box.
6. Click **OK** to return to the **Authorization** tab.
7. Click **OK** to close the **Properties** dialog box and apply the changes.

Library Management

The final step in this example is to add access to data in the newly secured folders by creating a SAS library and registering tables to that library. In order to understand how SAS grants authorization to this data, however, it is important to understand what a SAS library is. By definition a SAS library is a group of SAS files that are stored in the same directory and are accessed by the same engine (a SAS component that can read from or write to a file). Authorization to SAS datasets is determined by both the SAS metadata permissions and the user permissions on the host server where the data resides.

While the SAS metadata permissions secure any activity that user performs through metadata, a SAS user that is grated a role in which they are able to submit SAS programs can also write a direct LIBNAME statement to access the data without using the metadata engine. One simple method to ensure consistency between the SAS metadata folders and the host server is to create a folder hierarchy in SAS that replicates what is on the sever, as previously illustrated in **Figure 2**. The administrator can then create groups that can be used to apply security in a manner consistent with the host server.

Similarly, in order for users to access data that is located in an external database, the user must have the appropriate permissions on the database. When interacting with the database, SAS has no ability to override the database administrator's security settings. It is the database administrator who controls who can access, update, create and delete objects and records in the database.

To effectively manage access to an external database, the administrator can create a unique folder and group for the database. The Authentication Domain (if used) can be added to the group, which is then granted permissions on the database folder. This method ensures that the users are granted both SAS metadata permissions as well as database permissions once they added to the group.

Creating Libraries

A SAS library can be configured to access one of several different data source types. While the details of how to configure each type of library varies based upon the data type selected, the general process is the same for all data types.

1. On the **Plug-Ins** tab, expand **Data Library Manager**. Right-click **Libraries**. Then, select the **New Library** option to access the first page of the New Library wizard.
2. Select the library data type from the **SAS Data** list. Click **Next**.
3. Enter an appropriate library name in the **Name** field.
4. Click **Browse** and navigate to the metadata folder where the library and its contents will reside. Click **Next**.
5. Enter the appropriate server information. Click **Next**.
6. Enter the **Libref** for this library.
7. Verify the data engine and enter additional options based upon data type selected. Click **Next**.
8. Examine the final page of the wizard to ensure that the proper values have been entered.
9. Click **Finish** to save the wizard settings.

Selecting the appropriate SAS metadata folder when the library is created ensures that the library and its contents will inherit the permissions from the parent folder immediately upon creation and reduces the risk of data being unintentionally exposed to unauthorized users.

Registering Tables

Once the library has been created, the data associated with the library can be registered in metadata. Just as the location for the library was designated when it was created, the location of the tables is designated when the tables are registered and the tables will inherit their initial permissions from the parent folder.

To register the tables, perform the following steps: (replace with graphic)

1. Select **Data Library Manager** → **Libraries**.
2. Right-click the library that contains the tables that you need to import and select **Register Tables**.
3. Verify that the information displayed is correct. Click **Next**.
4. Click the tables that you need to select.
5. Check the metadata folder path in the Location field. If necessary, click **Browse** and navigate to the metadata folder where the library and its contents will reside. Click **Next**.
6. Verify the selections in the final page of the wizard and click **Finish** to save the wizard settings.

CONCLUSION

Every SAS deployment is different. Different products, different hardware, different user, different business needs… the list goes on. A strong security program starts with assessing the SAS environment from both a technical and business perspective and developing a written execution plan. Understanding basic SAS security terms, the authorization decision process, and how to execute common essential function positions administrators to tackle any security challenge.

By developing a strong understanding of basic SAS security terms such as authentication domain, ACT / ACE, etc. administrators are well prepared to use the extensive resources available through the SAS support site including, but not limited to technical manuals, SAS Tips, blog posts, and white papers. Furthermore, by testing their security plans against the identity hierarchy and object-inheritance path, using the authorization decision rules, administrators can perform a logical walk-though of their plan prior to execution; reducing risk the risk of unintended consequences.

While this paper strives to provide a firm foundation upon which administrators can launch their journey, it meant only to complement existing training programs and documentation. Several of the guides that are relevant to the security administrator were relied upon to build the content of this paper and are listed in the reference section of the paper. Readers are encouraged to use the information they have gained to take a deeper dive into these resources as they apply to their own unique situation.

REFERENCES

- SAS Institute Inc. 2016. *SAS® 9.4 Intelligence Platform: Desktop Application Administration, Sixth Edition*. Cary, NC: SAS Institute Inc.
  Available at: http://support.sas.com/documentation/cdl/en/bidaag/69032/PDF/default/bidaag.pdf
- SAS Institute Inc. 2015. *SAS® 9.4 Intelligence Platform: Data Administration Guide, Fifth Edition*. Cary, NC: SAS Institute Inc.
  Available at: http://support.sas.com/documentation/cdl/en/bidsag/68193/PDF/default/bidsag.pdf
- SAS Institute Inc. 2015. *SAS® 9.4 Intelligence Platform: System Administration Guide, Fourth Edition*. Cary, NC: SAS Institute Inc.
  Available at: http://support.sas.com/documentation/cdl/en/bisag/68240/PDF/default/bisag.pdf
- SAS Institute Inc. 2015. *SAS® 9.4 Management Console: Guide to Users and Permissions*. Cary, NC: SAS Institute Inc.
  Available at: https://support.sas.com/documentation/cdl/en/mcsecug/64770/PDF/default/mcsecug.pdf
- SAS Institute Inc. 2013. *SAS® 9.4 Intelligence Platform: Security Administration Guide, Second Edition*. Cary, NC: SAS Institute Inc.
  Available at: http://support.sas.com/documentation/cdl/en/bisecag/67045/PDF/default/bisecag.pdf
- Faenza, Charyn. 2015. *SAS® Metadata Security 101: A Primer for SAS Administrators and Users Not Familiar with SAS.* SAS Global Forum.
  Available at: http://support.sas.com/resources/papers/proceedings15/3479-2015.pdf

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author at:

Charyn Faenza
F.N.B. Corporation
(724) 983-2474
FaenzaS@fnb-corp.com
www.fnbcorporation.com