Paper 1789-2018

Managing Complex SAS® Metadata Security Using Nested Groups to Organize Logical Roles

Stephen Overton, Overton Technologies

ABSTRACT

SAS® Metadata security can be complicated to setup and cumbersome to maintain over time. Often times organizations need to manage many itemized groups and roles across multiple logical roles, especially with some industry-specific SAS solutions. In some cases, user groups and roles have no standards or framework to follow, which forces SAS administrators to guess to get an initial security model and process in place. To make matters even more complicated, permissions could be mutually inclusive across logical roles. This paper will present a scalable methodology to manage any number of itemized groups and roles within SAS metadata by providing a standardized template and approach using nested groups which roll up into customizable logical roles. This approach can be scaled for larger organizations by layering more nested groups as the number of users and group grow. Once deployed, the SAS administrator will only need to place SAS user accounts in top-level groups based on logical roles versus managing many groups.

INTRODUCTION

Security is critical to every environment as well as the process to maintain over time. In the SAS® Intelligence Platform, the SAS® Metadata Server is used to manage security for the environment. It is composed of many small pieces of security controls to manage every major aspect including data access, metadata folder access, component access, users, groups, roles, and functionality within individual tools of the environment. Having many small pieces to define becomes challenging for business users and IT, even with general communication of capabilities desired versus delivered.

The targeted reader of this paper is the SAS Platform Administrator as well as the Solution Architect looking to effectively manage the complexity of many small security pieces to manage. This paper will describe key concepts that can be leveraged for simple or complex environments. SAS® Management Console is the primary thick-client tool leveraged by the SAS Platform Administrator. The two important deliverables of the methodology described in this paper is a security model to implement within the User Manager of SAS® Management Console and a conceptual process to follow to manage other areas of SAS metadata. The User Manager is shown in Figure 1. Further details on managing the SAS® Intelligence Platform can be found through SAS Support documentation provided in the References section of this paper.

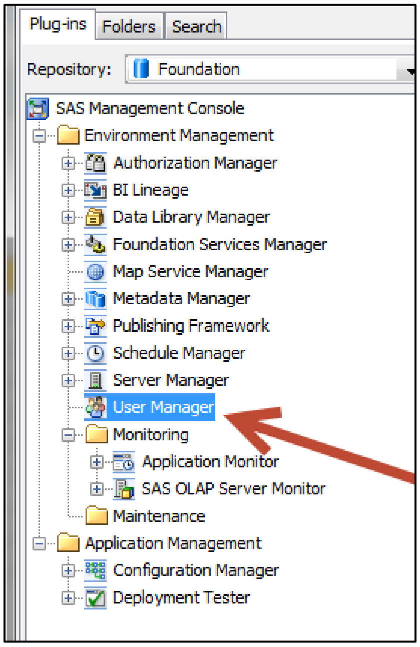


Figure 1: User Manager in SAS® Management Console.

AUTHENTICATION VERSUS AUTHORIZATION

Understanding the conceptual difference between authentication and authorization is important. Authentication is the process which makes the simple binary decision if a user is active or not. Authorization is the process which decides what a user can access and perform in the environment. By default, the SAS® Intelligence Platform relies on the host operating system for user authentication and SAS® Metadata for user authorization. Common systems that are used for host authentication are Microsoft Active Directory, PAM, or UNIX password file structure. Other technologies can be leveraged such as direct LDAP authentication. See SAS® Intelligence Platform Customer documentation for further details on configuring SAS for host authentication.

The following diagram in Figure 2 provides a conceptual view of host authentication within the SAS® Intelligence Platform.

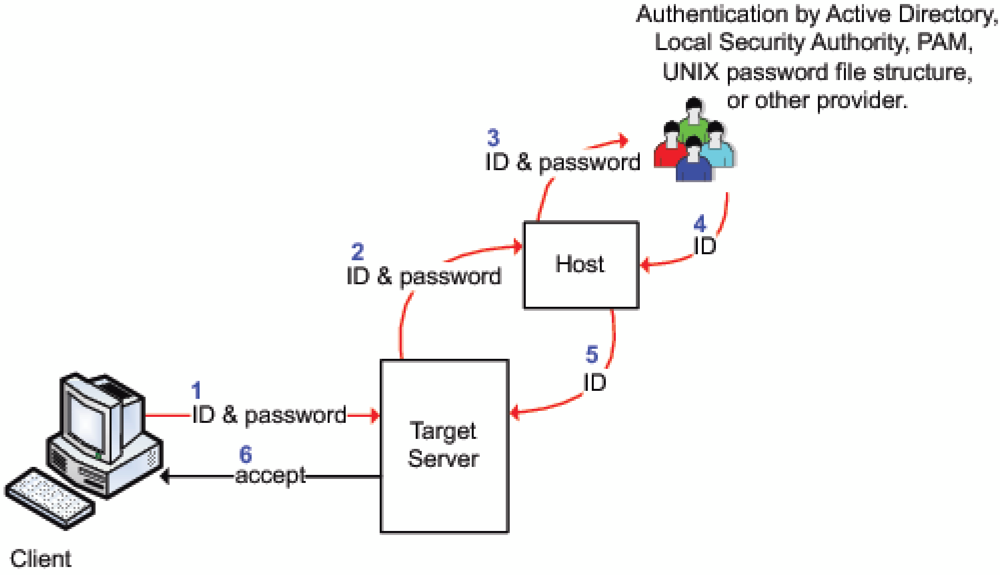


Figure 2: Conceptual host authentication process.

The following diagram in Figure 3 provides a conceptual architecture diagram to depict where host authentication and authorization occurs in a typical tiered environment.

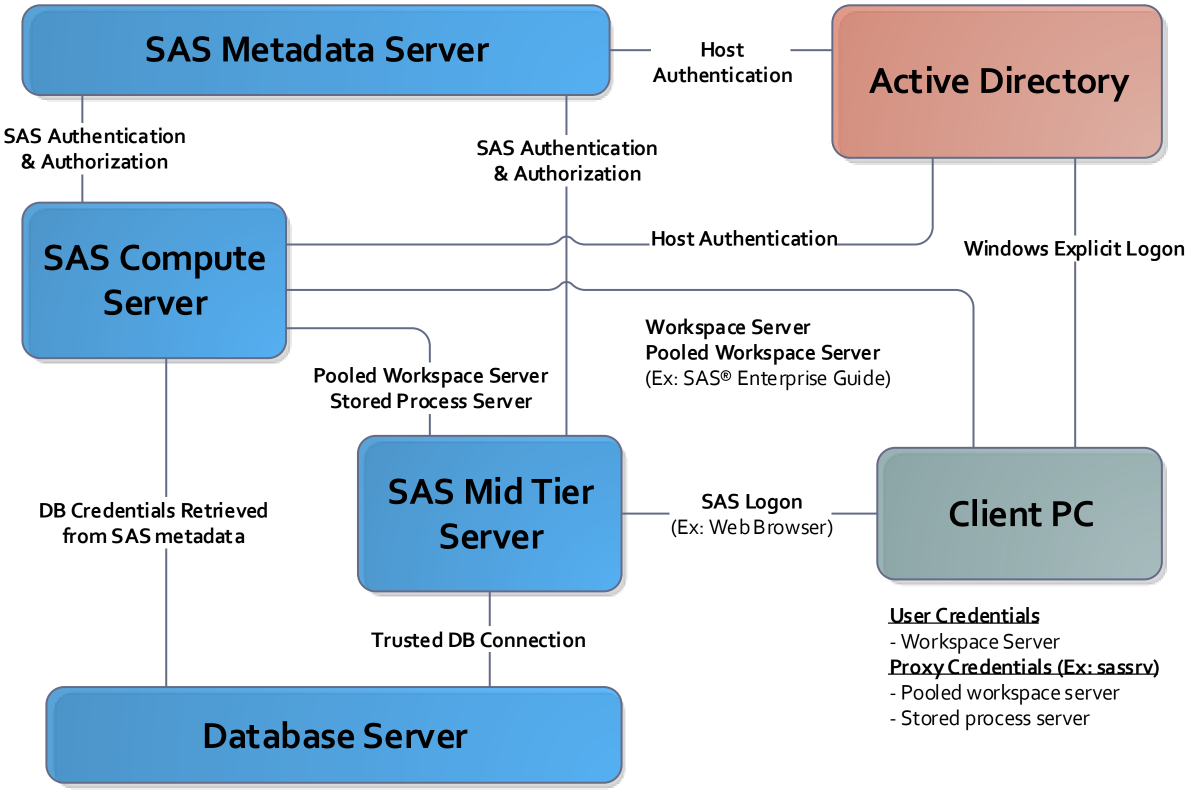


Figure 3: Conceptual authentication and authorization architecture.

COMMON ACCESS CONTROL LAYERS IN SAS METADATA

The most common components to manage in SAS metadata include access to data, what functionality a user has based on the solutions and tools available in the environment, and what metadata folders can be access on a read or read-write basis. The following section will provide conceptual definition around these common components.

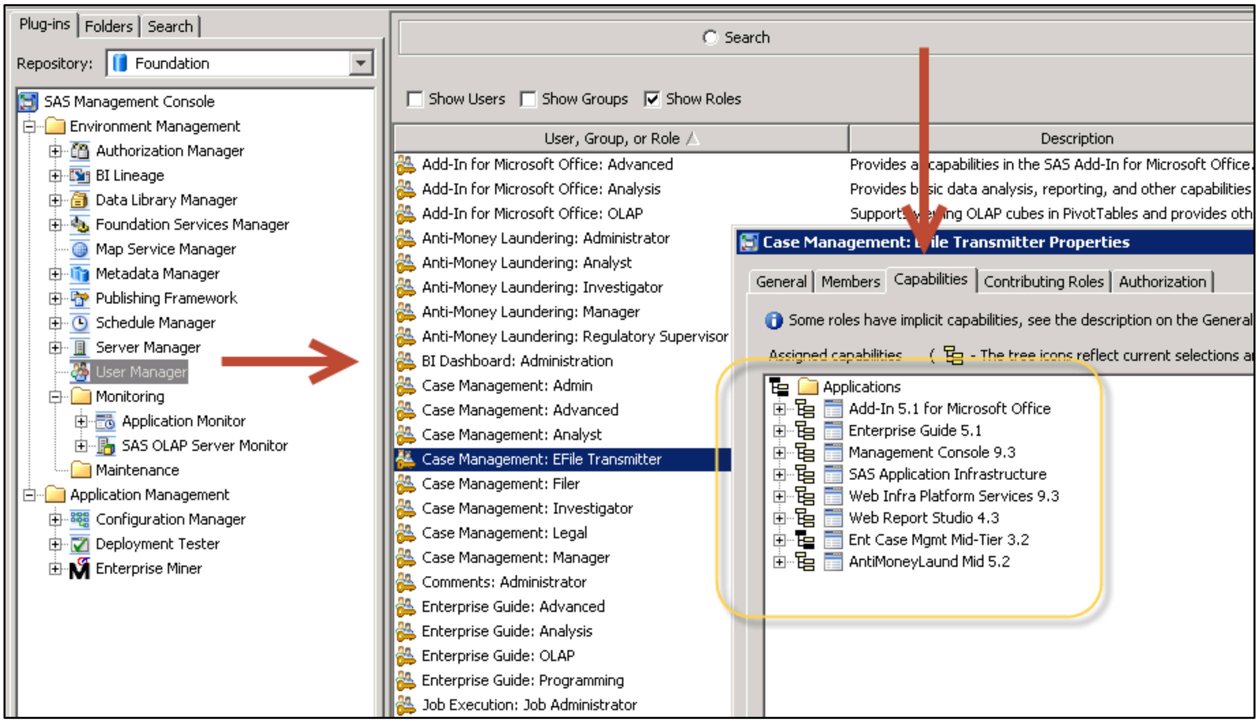
DATA ACCESS

Data is a very critical component of a SAS environment since one of SAS's primary purposes is to analyze and present information about data so business users can make informed decisions. Data can reside in many locations. The only data that can be managed by a security model in SAS metadata is data which can be accessed by the server components of the environment which are managed by the SAS® Metadata Server.

Traditional relational databases are very common ways to store data, as well as file system mounts on the SAS compute tier, and other third-party systems such as Hadoop. The SAS server can act as a proxy to systems which live outside of the local file system of the SAS compute tier. Often service accounts are used by SAS to access these systems or the end user's credentials can be passed through to the source system if desired. All critical pieces to connect to any supported database can be managed within SAS Metadata. Regardless of what is defined in SAS metadata, the source data storage technology or database will almost always require some form of authentication and authorization.

FUNCTIONAL CAPABILITIES

It is important to separate "what" a user can do in terms of function versus what information a user can access in terms of data. Data and functionality are best managed as two separate but equally important layers of an environment. Managing data and functionality separately gives the most flexibility and scalability. Functionality for a SAS environment can be found in the "Roles and Capabilities" available through SAS® Management Console's User Manager as shown in Figure 4 below.



User, Group, or Role	Description
Add-In for Microsoft Office: Advanced	Provides a capabilities in the SAS Add-In for Microsoft Office
Add-In for Microsoft Office: Analysis	Provides basic data analysis, reporting, and other capabilities
Add-In for Microsoft Office: OLAP	Supports viewing OLAP cubes in PivotTables and provides oth
Anti-Money Laundering: Administrator	
Anti-Money Laundering: Analyst	
Anti-Money Laundering: Investigator	
Anti-Money Laundering: Manager	
Anti-Money Laundering: Regulatory Supervisor	
BI Dashboard: Administration	
Case Management: Admin	
Case Management: Advanced	
Case Management: Analyst	
Case Management: EFile Transmitter	
Case Management: Filer	
Case Management: Investigator	
Case Management: Legal	
Case Management: Manager	
Comments: Administrator	
Enterprise Guide: Advanced	
Enterprise Guide: Analysis	
Enterprise Guide: OLAP	
Enterprise Guide: Programming	
Job Execution: Job Administrator	

Case Management: EFile Transmitter Properties
General
Members
Capabilities
Contributing Roles
Authorization
Some roles have implicit capabilities, see the description on the General
Assigned capabilities
- The tree icons reflect current selections a
Applications
Add-In 5.1 for Microsoft Office
Enterprise Guide 5.1
Management Console 9.3
SAS Application Infrastructure
Web Infra Platform Services 9.3
Web Report Studio 4.3
Ent Case Mgmt Mid-Tier 3.2
AntiMoneyLaund Mid 5.2

Figure 4: Roles and Capabilities in SAS® Management Console.

METADATA FOLDER ACCESS

Many components of the SAS® Intelligence Platform are contained within folders of the SAS metadata repository. Folders are managed through the "Folders" tab of SAS Management Console. Having a well-defined model to manage both the structure and access of these folders is critical for the security and scalability of the environment.

File System Folder Access

Managing local file system folder or directory access is outside the scope of this paper. Like third party databases, it is still important to note that the local file system or network mounted storage of the SAS compute tier is always managed by the respective operating system of the environment. SAS libraries can be defined to access locally available filesystems within the context of Data Access, as described by the previous section, but the ultimate authentication and authorization is still managed by the source operating system.

SECURITY MODEL FRAMEWORK

The security model can be thought of as a framework that expands with the growth of the environment. Specific security model controls within each access control layer can grow the most frequent, while overall layers that organize specific controls will probably remain static over time. As shown in Figure 5 below, the highest level access control is the logical role which contains the necessary access controls to support the job function of the logical role in the SAS environment.

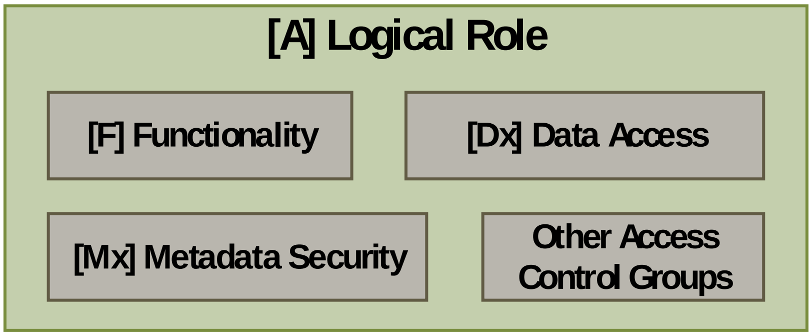


Figure 5: Logical Roles contain many access control layers managed by groups in SAS metadata.

DEFINING AND MANAGING LOGICAL ROLES

Logical roles are organized typically by the job functions within the business user base of the SAS environment. Logical roles can have mutually inclusive capabilities that can be easily managed with a nested group structure described later in this section. Defining logical roles for a SAS environment will be an extensive process initially, but become more fluid as the environment grows and adoption of this security model framework occurs.

Examples of logical roles are defined using the following naming scheme:

- [A] SAS Platform Administrator
- [A] AML Manager
- [A] AML Supervisor
- [A] AML Analyst
- [A] Developer

The important naming standard to retain is the "[A]" prefix. The remainder of the group name can be the actual logical role required for the respective SAS environment.

MANAGING ACCESS CONTROL LAYERS THROUGH NESTED GROUPS

The next level beneath logical roles provides access control layers to specific areas of the security model. This portion of the framework is organized using groups within the SAS security model. Groups can contain other groups if needed for larger organizations. Ultimately these access control groups contain atomic-level groups which regulate very specific pieces of the environment.

The following outline provides a list of common access control layers to manage within SAS metadata as well as the naming prefix that will be expanded upon later in this paper:

- [F] – Functional groups manage what actions and abilities users can perform using “Roles & Capabilities” defined in metadata.
- [Dx] – Data access groups manage what data users have access to.
  - [DR] – read only connection
  - [DW] – read and write connection
- [Mx] – Manage layers of metadata folder access
  - [MR] – read only permissions
  - [MW] – read and write permissions.
- [P] – Portal pages within SAS Information Delivery Portal (if applicable).
- [SNA] – SAS Social Network Analysis alert series access (if applicable).
- [ECM-WF] – SAS Enterprise Case Management workflow access (if applicable).

Specific examples will be provided in the next section of this paper for each of the conceptual access control layers listed above along with matching logical roles.

NAMING STANDARDS

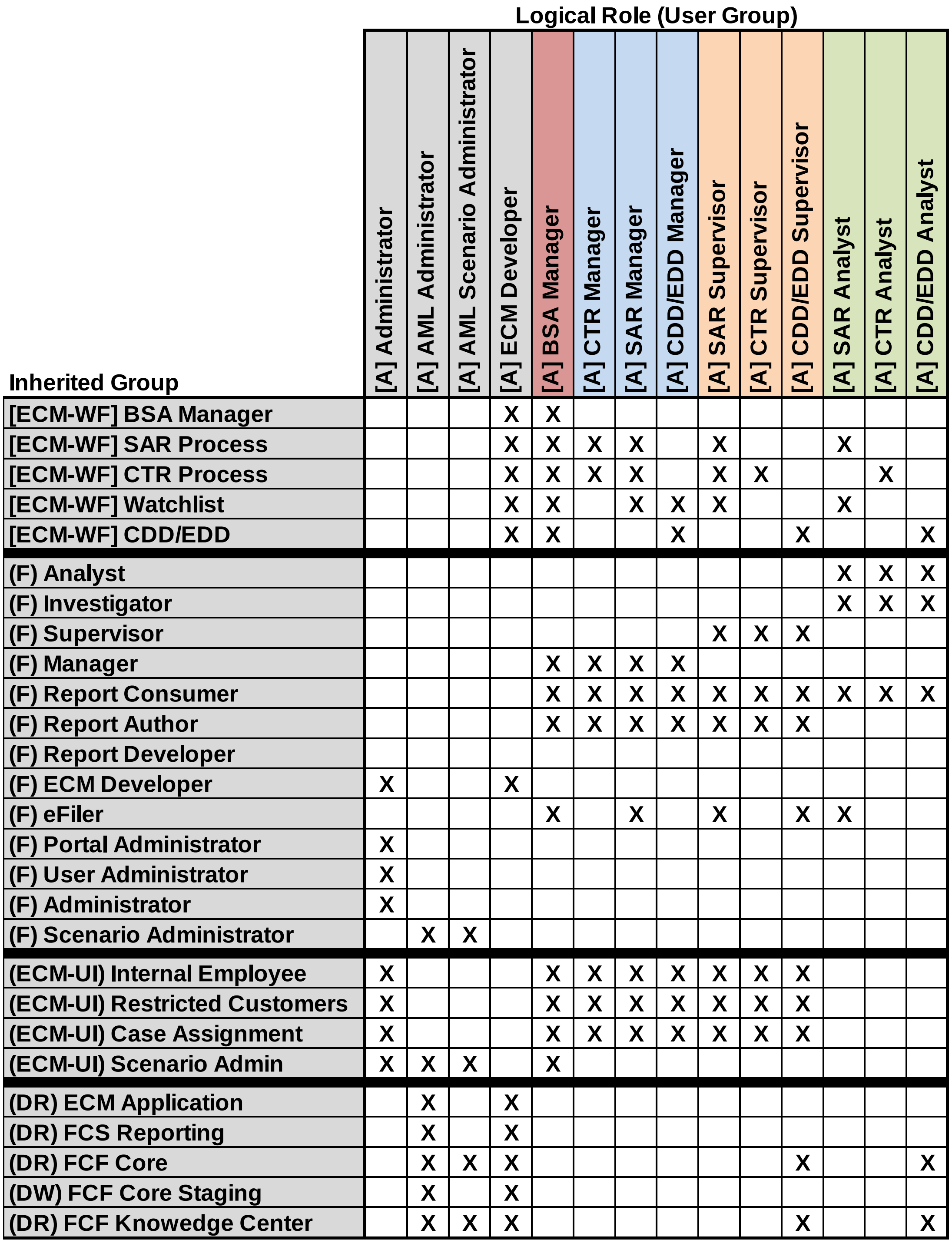
As shown above, the use of square brackets is very prevalent. One advantage of using special characters is for the simple ability to sort the User Manager within SAS Management Console and have the security model groups appear at the top of the list. The critical naming standards defined above include the “[A]”, “[Dx]”, “[F]”, and “[Mx]” groups. Other groups may or may not be applicable. There may be other access control layers to define depending on what functionality and products are available. As always, leverage the description fields within SAS metadata to self-document as much as possible.

SPEAKING A COMMON LANGUAGE BETWEEN THE BUSINESS AND IT

One big advantage to having a standardized security model is establishing a common language between business users and the IT department. This language consists both of vocabulary to use as well as how to have dialogue for securing the environment and provisioning users. This maximizes productivity over time by reducing internal meetings and making conversations efficient.

COMBINING LOGICAL ROLES WITH LAYERS OF ACCESS CONTROLS

After logical roles and access control layers have been defined and mapped to appropriate components of the SAS environment, the security framework can begin to take shape. The following diagram shown in Figure 6 provides an example matrix which maps top-level logical roles to more detailed access control layers along the left-hand side of the matrix.



Inherited Group	[A] Administrator	[A] AML Administrator	[A] AML Scenario Administrator	[A] ECM Developer	[A] BSA Manager	[A] CTR Manager	[A] SAR Manager	[A] CDD/EDD Manager	[A] SAR Supervisor	[A] CTR Supervisor	[A] CDD/EDD Supervisor	[A] SAR Analyst	[A] CTR Analyst	[A] CDD/EDD Analyst
	Logical Role (User Group)													
[ECM-WF] BSA Manager				X	X									
[ECM-WF] SAR Process				X	X	X	X		X			X		
[ECM-WF] CTR Process				X	X	X	X		X	X			X	
[ECM-WF] Watchlist				X	X		X	X	X			X		
[ECM-WF] CDD/EDD				X	X			X			X			X
(F) Analyst												X	X	X
(F) Investigator												X	X	X
(F) Supervisor									X	X	X			
(F) Manager					X	X	X	X						
(F) Report Consumer					X	X	X	X	X	X	X	X	X	X
(F) Report Author					X	X	X	X	X	X	X			
(F) Report Developer														
(F) ECM Developer	X			X										
(F) eFiler					X		X		X		X	X		
(F) Portal Administrator	X													
(F) User Administrator	X													
(F) Administrator	X													
(F) Scenario Administrator		X	X											
(ECM-UI) Internal Employee	X				X	X	X	X	X	X	X			
(ECM-UI) Restricted Customers	X				X	X	X	X	X	X	X			
(ECM-UI) Case Assignment	X				X	X	X	X	X	X	X			
(ECM-UI) Scenario Admin	X	X	X		X									
(DR) ECM Application		X		X										
(DR) FCS Reporting		X		X										
(DR) FCF Core		X	X	X							X			X
(DW) FCF Core Staging		X		X										
(DR) FCF Knowedge Center		X	X	X							X			X

Figure 6: Security model framework displayed as a matrix crossing logical roles with access control groups.

This security matrix can serve as documentation for audit and compliance as well as operational documentation for the SAS Platform Administrator. This example, and other examples provided in this paper are meant to be starting points to be expanded upon to fit the respective organization.

ONGOING MAINTENANCE OF THE SECURITY MODEL

Regular confirmation and audit of the security model framework should be enforced on a periodic basis. Top-level logical roles should be checked to verify the proper access control groups are contained within the desired memberships. Each nested layer of groups should be followed through to the lowest level group or role. Random sampling of lowest level groups and roles and can be used to verify security from a bottom-up as well.

CONCLUSION

Having a solid security framework to support a SAS environment is a critical component often overlooked and under-estimated. Any size environment benefits from having a well-defined security model with easy-to-use group structures. No matter how complicated and large an organization is, leveraging nested groups in a thoughtful manner as described in this paper will provide a well-defined, solid security model which will answer the demands and rigors of business users and IT compliance.

REFERENCES

"SAS Intelligence Platform Customer Documentation Page." SAS Institute. Accessed March 6, 2018. Available at https://support.sas.com/documentation/onlinedoc/intellplatform/index.html.

CONTACT INFORMATION

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

Stephen Overton
Overton Technologies
(919) 341-9667
soverton@overtontechnologies.com
https://www.stephenoverton.net
https://www.linkedin.com/in/overton/