Metadata Security in SAS® 9.3 – Step-by-Step

Johannes Jørgensen
Cecily Hoffritz

January, 2012

Contents

1 Background

This document is the third version of Metadata Security in SAS® – Step-by-Step and is relevant for SAS® 9.3.

The previous version is found here and it is also very much worth your while to read as the focus is slightly different.

Here is a list of the main differences between the SAS® 9.2 and 9.3 version of Metadata Security in SAS® – Step-by-Step:

1. ***SASUSERS – Denied ACT*** has been renamed ***PUBLIC and SASUSERS – Denied ACT*** and now incorporates the denial of all permissions for both groups. This is more of a pedagogical revision to quench questions during classes on administration.
2. Addition of explicit grant of WMM for Administrators in the Default ACT and SAS Administrator Settings. 9.3 shows an inherited grant.
3. Addition of BI Developers – Server ACT so that BI Developers have WM permission server-side. It is used for metadata objects below a server context, for example, source code repositories when BI developers need to save the physical program behind the stored process when creating them through an application such as SAS Enterprise Guide.
4. System Users – Read Only ACT has been renamed SAS General Servers – Read Only ACT because this adheres more to our rule of calling the ACT the same name as the group it includes.
5. Colour coding of groups so that it is easier to see when a folder structure is tedious to secure.

In Denmark, customers can request that their platform environment is turbocharged with the automatic creation of pre-defined logical and physical folders, libraries, standardized groups and access control templates following best practice standards. Turbocharging the environment is often done by SAS Institute consultants but customers request to be active participants in the turbocharging process and attend the course **Turbocharge your initial environment for data integration (DI3)** in Denmark and abroad.

All groups, access control templates and folder structure described in this document are a part of the Turbocharge Toolkit which contains a comprehensive guide, scripts and SAS import packages and is supplied during the course.

For you to fully understand this best practice, it is important that you have a basic understanding of security in regards to Access Control Templates, Access Control Entries and the rules of inheritance. It is also important that you understand the identity relationship between groups and users.

Operative system security is not addressed here but you need to prioritize this in conjunction with metadata security.

Due to the heavy use of colour coding in tables, it is advised that you print this document in colour!

2 Setting up standard groups in metadata

The standard groups referred to in this security setup are found in the table below. We typically use them when starting a data management/warehouse project at a customer to make a speedy head start.

Table 1: Overview of standard groups and description.

Group	Description
DI Developers	Expert user working in all phases of a Data Warehouse (DW) project, except data modeling. Imports and registers metadata for data sources and targets and creates jobs for deployment. Uses SAS applications dedicated to DW in combination with SAS programming. Typical applications: SAS Data Integration Studio.
BI Developers	Expert user who automates, integrates and distributes reports and analyses for the organization. Uses SAS applications dedicated to BI development in combination with SAS programming. Typical applications: SAS Enterprise Guide, SAS Stored Process Facility, SAS Information Map Studio, SAS Web Report Studio, SAS Add-In, SAS BI Dashboard.
Analysts	Expert user who works with statistical analysis, forecasting and data mining. Uses SAS applications dedicated to analysis in combination with SAS programming. Typical applications: SAS Enterprise Miner, SAS Enterprise Guide, SAS/STAT.
Report Creators	Super user who designs and creates reports for o wn department. Uses SAS applications which are intuitive and user-friendly. Typical applications: SAS Web Report Studio, SAS Add-In.
Report Consumers	End user who uses information/reports, typically from a company intranet site. The use of SAS applications other than viewing reports is limited. Typical applications: SAS Portal, Sharepoint.

We colour code the above groups into three areas which makes it all much easier to understand which folders belong to which groups.

Table 2: Overview of groups and colour codes.

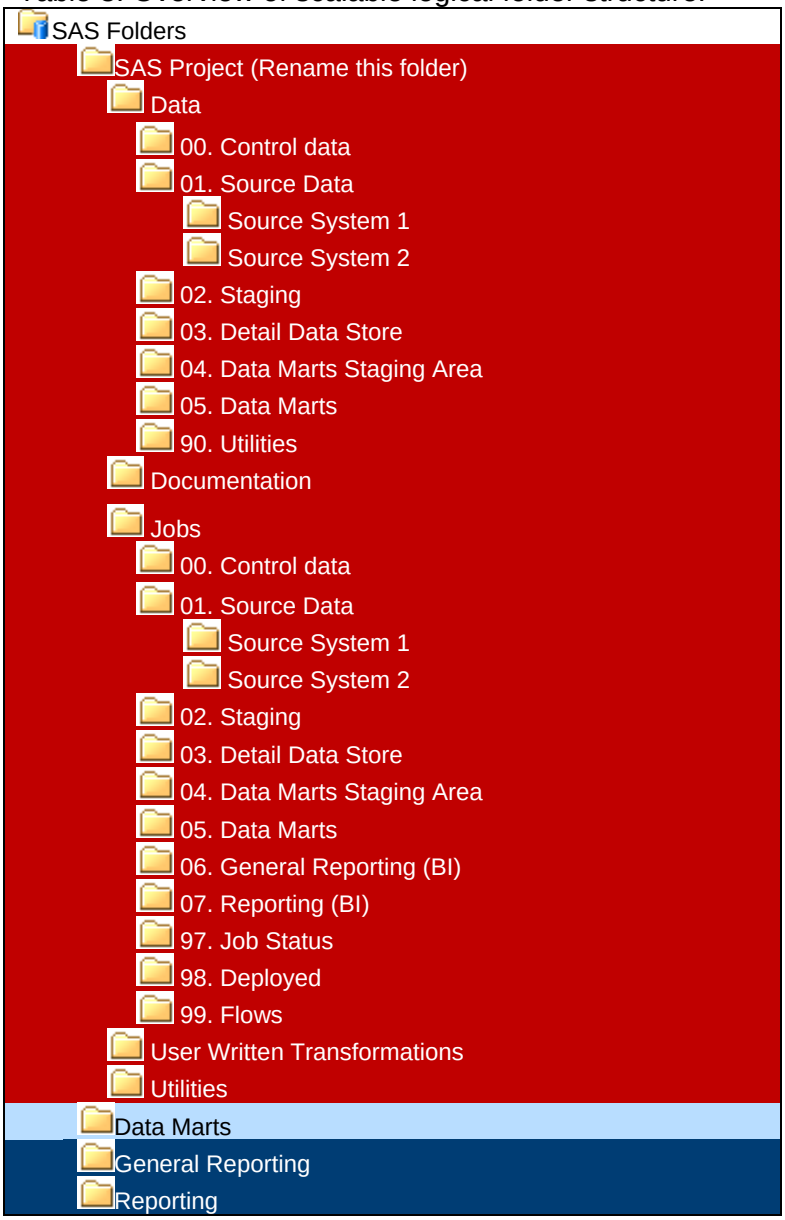
DI Developers	Users working with all tasks relevant for data management/data integration without unrestricted status.	Data Management folders
BI Developers	Users who are specialists working with business/analytical intelligence.	Specialist folders
Analysts	Users who are specialists working with statistical analysis.	Specialist folders
Report Creators	Super users who work with reporting.	Business User folders
Report Consumers	End users.	Business User folders

3 Setting up folders in metadata

The folder structure portrayed below is a best practice standard used in Denmark. It is currently under revision by the committee for best practices in Denmark so that it may be broadly used across diversified business areas. Look out for an announcement for updates to this structure as it continuously will be updated based on project experiences.

It is a flexible and scalable model used when there is a need for multiple projects on the same metadata server. Notice, how the coloured areas are separated from each other and top folders are indented at the same position below “SAS Folders” making it very easy to secure folders at top level and making the click path very short for users.

Table 3: Overview of scalable logical folder structure.



You need to have a holistic approach regarding the folder structure for all three coloured areas. The red and light blue coloured areas are standardized and are provided during the initial phase of a data management/warehouse project where SAS consultants are involved. They are rarely translated into local language since only highly specialized users will be seeing them.

The dark blue folders are suggestions for top-level folders for business users and can be translated into local language. It is important to design the folder structure below these top folders as early as possible and preferably in parallel with the data management/warehouse project. Designing them should be a collaborative effort between SAS consultants and customer representatives who work with business intelligence.

Depending on requirements data is typically replicated from the red data mart area to a light blue mart area to keep blue users out of the red area. This entails the need for a separate physical location for each data mart. An alternative method is to switch the library from red to light blue area to avoid replication indicating a switch between *the data preparation area* and the *end user data area*".

Below the folder "Reporting" you might have folders designed according to organization or subject areas. Below "General Reporting" you might have folders which contain non-sensitive reports and data which can be seen by all employees. Observe that these folders are not numbered because if some of them were to be hidden then you will get a sequence with missing numbers.

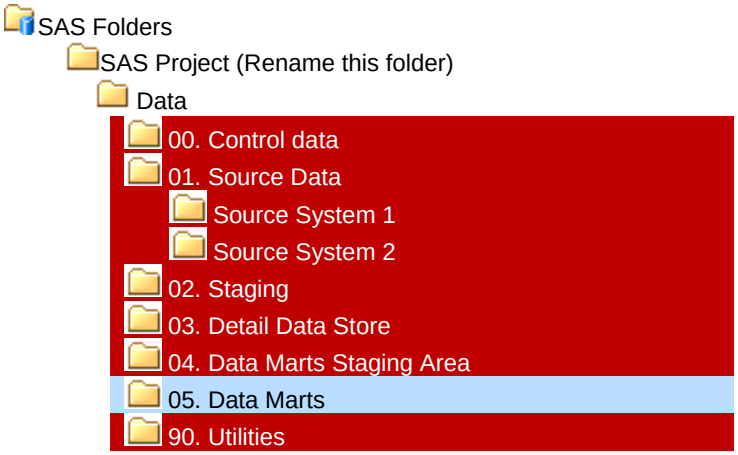
The physical folder structure mirrors the logical one:

D:\SAS_Folders\SAS Project\ …
D:\SAS_Folders\Data Marts\...
D:\SAS_Folders\General Reporting\...
D:\SAS_Folders\Reporting\...

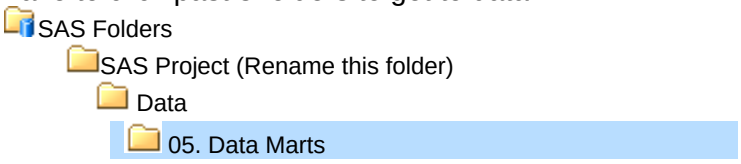
4 Looking at examples of folder structures that may cause administration issues

The folder structure (partial view) below is tedious to manage security wise. Observe that the light blue “Data Marts” folder is mixed in with the red folders. Red folders are always hidden from all blue users and security will have to be set on nearly each and every red sub-folder to keep the blue users out, augmenting administration of security considerably. If you design your own folder structure, try colour coding it so you can see whether it poses tedious administration.

Table 4: Partial view of folder structure with mixed colours, resulting in tedious administration.

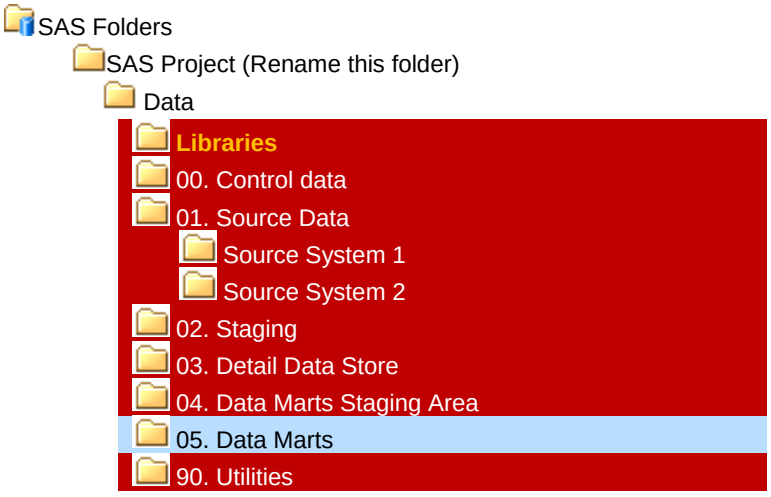


A BI Developer would see these folders and will wonder why folders 01. – 04. are missing and will have to click past 3 folders to get to data.



A worse scenario is the one below. Observe that there is a “Libraries” folder containing all the libraries for data sources and targets used in the data management/warehouse project. In the previous examples, libraries resided locally in each data folder. With a library separated from the “Data Marts” folder and residing in a red hidden folder no light blue users will be able to see data in the “Data Marts” folder. In order to make blue data visible to users it is possible to tweak security directly on a library object but this increases administration considerably and you really have to know your way round security to figure this out.

Table 5: Partial overview of “bad” folder structure.



5 Understanding the golden rules of the security best practice

The following six generic Golden Rules are the heart and soul of our metadata security best practice. Abiding by these rules still allows you a great deal of creativity when you design your own security setup. Some of them are more meaningful later on when we present to you an example security implementation.

A prerequisite for understanding our rules is that you are well versed in the metadata security model and understand the meaning behind the white, green and grey background colour codes.

Rule #1: Apply Access Control Templates (ACTs) on resources.
It is absolutely strictly forbidden for you to use Access Control Entries - ACEs (tick marks with white background on resources). Our recommendation for your securing resources is a combination of inheritance (tick marks with grey background) and ACTs (tick marks with green background). If you only apply ACTs and not ACEs your life as an administrator will be easier because you can maintain all security changes centrally in the Authorization Manager Plugin of SAS Management Console.

Note, that in a standard software installation and configuration you might already see ACEs applied which we advise you not to change and the security design behind row level security on maps and cubes forces you to apply ACEs.

Rule #2: Add only groups in ACTs.
From a maintenance point of view, it is much easier working on a group level than on a person level. Once you design your ACTs containing groups, all you need to do is synchronize users into these groups. We suggest that you only add one group per ACT whose name is synonymous with the group.

Rule #3: ACTs with explicit groups (not PUBLIC or SASUSERS) are only allowed to grant access, never deny.
This is the most important rule of them all. It ensures a **100 % guarantee** that you no longer will experience security conflicts. Whether you are a member of more than one group or your groups are members of other groups is not an issue.

It is also the hardest rule for you to comply with because it is so easy to tick mark a denial for a permission to compensate for too much access on a specific resource. If you breach this rule you will really topple the apple cart and your compensational permission denials for explicit groups will cause havoc to this best practice and in the end they give you a bad headache!

Rule #4: Apply, whenever needed, ACTs with explicit group(s) granting access in combination with ACT denying RM for PUBLIC/SASUSERS.
This rule is for situations where you want to allow selected groups to view certain metadata objects while hiding them from others. The process of showing metadata folders to some and hiding them from others is a good example of this. This rule is not in conflict with rule #3 because we are denying access for implicit groups, not explicit groups. It allows us to rely heavily on resolving security conflicts via the identity hierarchy.

Here is an example of how this rule works. Remember, if you are registered in metadata you are always a member of the implicit group SASUSERS and most probably belong to at least one explicit group. You are of course also an implicit member of PUBLIC.

Table 6: How rule #4 works.

Scenario	Access Control Template 1 is applied on folder A granting Read Metadata to group ABC.		Access Control Template 2 is applied on the same folder A denying Read Metadata to group SASUSERS/PUBLIC.	The group ABC is allowed to see the folder A while all others not belonging to ABC aren't.
Step 1	ACT	**=**	ACT	1. Check **type** of Access Control (not the permission). Are they equal? Here they are because the folder A is secured by 2 ACTs (rule #1 is applied here). If they are equal then the conflict cannot be resolved via access control **type** and the identity hierarchy must be checked as well.
Step 2	Explicit group ✓	**<>**	Implicit group	2. Check identity hierarchy. Are they equal? Here they are <u>not</u> because the explicit group ABC wins over the implicit group SASUSERS/PUBLIC. Your membership of the explicit group ABC will give you the grant. If you are not a member of the explicit group ABC then you are still implicitly member of SASUSERS/PUBLIC and you will be denied access instead.

Rule #5: Apply always the ACT for Administrators when SASUSERS/PUBLIC have been denied access.
You must always apply this rule in conjunction with rule #4. Restricted administrators are subject to access control like anybody else and are affected by an ACT denying SASUSERS/PUBLIC access.

Rule #6: Design & document first and implement early.
Design your setup and document it on paper first before you implement it. 90% of the work is the design and documentation and once you have that in place, implementation is easy as pie. Remember that a standard initial 9.2/9.3 security setup is a closed one to users other than administrators so your job is to implement security as early as possible to open up access for them.

6 Designing Access Control Templates

We use these permissions in our Access Control Templates:

RM = Read Metadata: Ability to see a metadata object
WM = Write Metadata: Ability to add, modify and delete metadata.
WMM = Write Member Metadata: Ability to add, modify and delete metadata objects in folders.
CM = Check in Metadata: Ability to check metadata into primary repository from project repository.
R = Read: Ability to read data.
W = Write: Ability to modify existing data.
C = Create: Ability to add new data.
D = Delete: Ability to delete data.
A = Administer: Ability to administer SAS OLAP

This standard set of Access Control Templates is loaded at the beginning of a consultancy project primarily to protect the work of DI Developers.

Table 7: Overview of Access Control Templates. **G**: Grant, **D**: Deny, All: permissions.

Default ACT (Repository ACT) (we have applied explicit grant of WMM for administrators)	PUBLIC **D:** *ALL* SAS Administrators **G:** *RM WM WMM CM A* SAS System Services **G:** *RM WM* SASUSERS: **G:** *RM WM CM*
SAS Administrator Settings (we have applied explicit grant of WMM for administrators)	SAS Administrators **G:** *RM WM WMM CM A* SAS System Services **G:** *RM*
DI Developers ACT	DI Developers **G:** *RM WM CM WMM R W C D A*
BI Developers ACT	BI Developers **G:** *RM WMM R W C D*
BI Developers - Server ACT	BI Developers **G:** *RM WM*
Analysts ACT	Analysts **G:** *RM WMM R W C D*
Report Creators ACT	Report Creators **G:** *RM WMM R*
Report Consumers ACT	Report Consumers ACT **G:** *RM R*
PUBLIC and SASUSERS - Denied ACT	PUBLIC **D:** *ALL* SASUSERS **D:** *ALL*
SASUSERS – Read Only ACT	SASUSERS **G:** *RM R* **D:** WM WMM CM *W C D A*
SAS General Servers ACT *1)	SAS General Servers **G:** RM R

*1)SAS General Servers ACT allows the shared account SASSRV to read rows of data when stored processes reference certain library pre-assignments.

7 Applying Access Control Templates to logical folders

Securing folders can be easy especially if you have logical folders which have been designed to minimize time spent on securing them. Bear in mind, that the more differentiated your groups need access to different folders the more complex your security setup will be. In the simple but effective setup in the table below SAS Consultants/DI Developers are up and running very quickly without being hampered by security problems. Security for light blue users is taken care of as well so that they can test data and reports before their final deployment to the general public.

Table 8: Applying security on logical folders.

SAS Folders	
SAS Project (Rename this folder)	PUBLIC and SASUSERS – Denied SAS Admininstrator Settings DI Developers ACT
Data	Inherited settings
00. Control data	Inherited settings
01. Source Data	Inherited settings
Source System 1	Inherited settings
Source System 2	Inherited settings
02. Staging	Inherited settings
03. Detail Data Store	Inherited settings
04. Data Marts Staging Area	Inherited settings
05. Data Marts	Inherited settings
90. Utilities	Inherited settings
Documentation	Inherited settings
Jobs	Inherited settings
00. Control data	Inherited settings
01. Source Data	Inherited settings
Source System 1	Inherited settings
Source System 2	Inherited settings
02. Staging	Inherited settings
03. Detail Data Store	Inherited settings
04. Data Marts Staging Area	Inherited settings
05. Data Marts	Inherited settings
06. General Reporting (BI)	Inherited settings
07. Reporting (BI)	Inherited settings
97. Job Status	Inherited settings
98. Deployed	Inherited settings
99. Flows	Inherited settings
User Written Transformations	Inherited settings
Utilities	Inherited settings
Data Marts	SAS Administrator Settings DI Developers ACT BI Developers ACT Analysts ACT PUBLIC and SASUSERS - Denied ACT
General Reporting	SAS Administrator Settings DI Developers ACT BI Developers ACT

	Analysts ACT SASUSERS – Read Only ACT
Reporting	SAS Administrator Settings DI Developers ACT BI Developers ACT Analysts ACT Report Creators ACT Report Consumers ACT PUBLIC and SASUSERS - Denied ACT SAS General Servers ACT

Remember that the dark blue folders are just suggestions of top-level folders and sub-folders need to be designed and implemented. With security set at these top-level folders we ensure that test users can validate data, reports, the environment and the applications users will be working with. Naturally the security setup for the dark blue folders needs to be designed once all the sub-folders are in place.

8 Applying Access Control Templates to SASApp – OLAP Schema

In a standard installation, you can't create cubes because the SASApp - OLAP Schema, which you find in the folder **SASApp – OLAP Schema** below the folder **Shared Data** is tightly locked down with an inherited denial of WM which originates from a system applied ACE denying WM for PUBLIC on SAS Folders.
In our scenario, if you want to allow BI Developers to create cubes, they need the BI Developers ACT applied to the custom folder where they will save the cube metadata object. They also need you to apply the BI Developers ACT to the folder **SASApp – OLAP Schema** as shown below. This ACT only has WMM, not WM but the OLAP schema in the folder will inherit a grant of WM which is necessary for building cubes. Here, DI Developers create cubes so you need to apply the DI Developers ACT as well.
Table 10 – Applying Access Control Templates to create cubes

SAS Folders	Standard protection originating from SAS Administrator Settings applied during installation
My Folder	Standard protection originating from Private User Folder ACT applied during installation
Shared Data	Inherited settings from SAS Folders for Administrators.
SASApp – OLAP Schema	DI Developers ACT
	BI Developers ACT

You need to familiarize yourself with other standard folders and their metadata objects to assess whether they need special consideration for selected groups.

9 Securing server-side metadata objects

In a standard configuration, the group SASUSERS has RM and WM permissions for metadata objects below Server Manager in SAS Management Console which include server contexts and servers. Examples of these are SASApp, SASMeta, SAS Content Server and object spawner. RM and WM permissions allow any registered account to see and modify servers.

Server-side metadata objects are per default not protected and need to be, as shown by the ACT settings below. The SASUSERS – Read Only ACT is a multi-purpose ACT used on folders as well as here. The Read permission is not utilized server-side.

Table 9: Applying Access Control Templates to lock down server-side metadata objects

SAS Management Console Environment Management Server Manager	
SASMeta	SAS Administrator Settings SASUSERS - Read Only ACT
SASApp	SAS Administrator Settings DI Developers ACT SASUSERS - Read Only ACT
SASApp - Logical <all definitions>	SAS Administrator Settings SASUSERS - Read Only ACT
<All other server definitions>	SAS Administrator Settings SASUSERS - Read Only ACT
<All other spawner definitions>	SAS Administrator Settings SASUSERS - Read Only ACT

In the above scenario, DI Developers ACT has been applied to SASApp so that DI Developers have full control of the server context so that they can create/modify data libraries, save stored processes in source code repositories, deploy jobs in deployment directories etc. They can also alter metadata for servers and if you don't want them to do this you need to protect metadata objects at a lower level.

In the example below, DI Developers ACT has been moved from SASApp to Batch Jobs (for job deployment) and SP Source Directory (for saving stored processes). BI Developers are added to SP Source Directory so that they can save stored processes as well.

Table 10: Applying Access Control Templates to a Stored Process Source Directory.

SAS Management Console Environment Management Authorization Manager Resource Management By Location SASApp	
SP Source Directory	BI Developers ACT DI Developers ACT
Batch Jobs	DI Developers ACT