

A Shared Goal, Secure Metadata

Minh Duong, University of Houston TIMES

ABSTRACT

SAS Metadata is a critical component of the SAS Intelligence Platform. SAS provides access to the metadata from a variety of environments. This interoperability provides access to metadata from SAS applications as well as in-house applications. This flexibility reduces the need for multiple metadata servers but also increases the need to secure your metadata. Protecting metadata from unauthorized access, use, destruction, or modification is key to any security policy. In this paper we examine authentication, authorization decisions, and securing metadata objects.

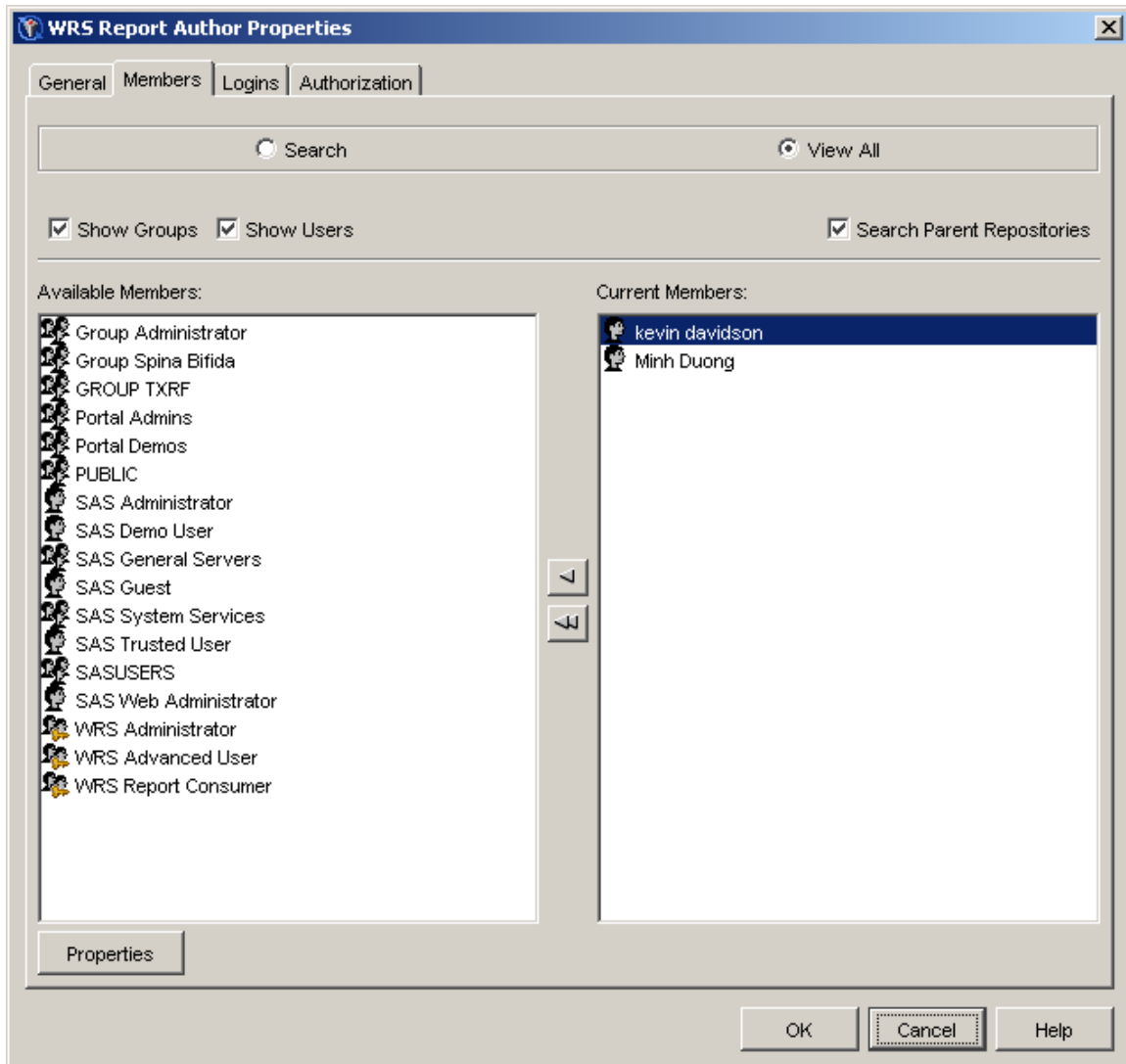
INTRODUCTION

The basic tenets of security are identification, authentication, authorization, and auditing. Identification is the process of identifying yourself. This can be thought of as giving a business card or username. No verification occurs during the identification process. The authentication process verifies the identity. This can be thought of as presenting an official document or entering a password. Authorization is the concept of determining what resources can be used by resource consumers. An example of this ability can be thought of as opening a document on a file server or access to certain floor in a building. Auditing is recording and analyzing what has happened. This can be reviewing access logs or watching recordings from a security camera.

IDENTIFICATION

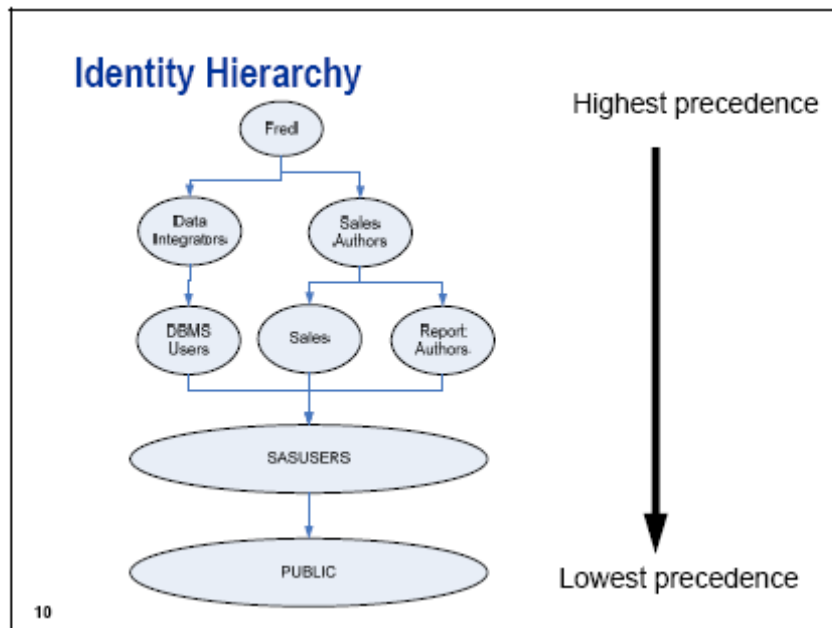
A metadata identity is a collection of information about a person or group. Having a person or individual metadata identity allows a person or group to log on to a certain application, store or retrieve personal logon information, and participate in change management. All person identities must be defined in the foundation repository. They cannot be defined in custom or project repository. Each user can only have one individual metadata identity.

A group is a collection of information about a set of users. Groups should be created to manage permission efficiently. You can assign users with similar access requirements to one group and give that group permissions to the appropriate resource. Groups also enable multiple users to use a shared account to access a particular resource, support pooling of servers and control which users are routed to which servers. Also certain applications map groups to roles that give users special abilities within the application.



For example, Web Report Studio uses the WRS Report Author group to allow report creation.

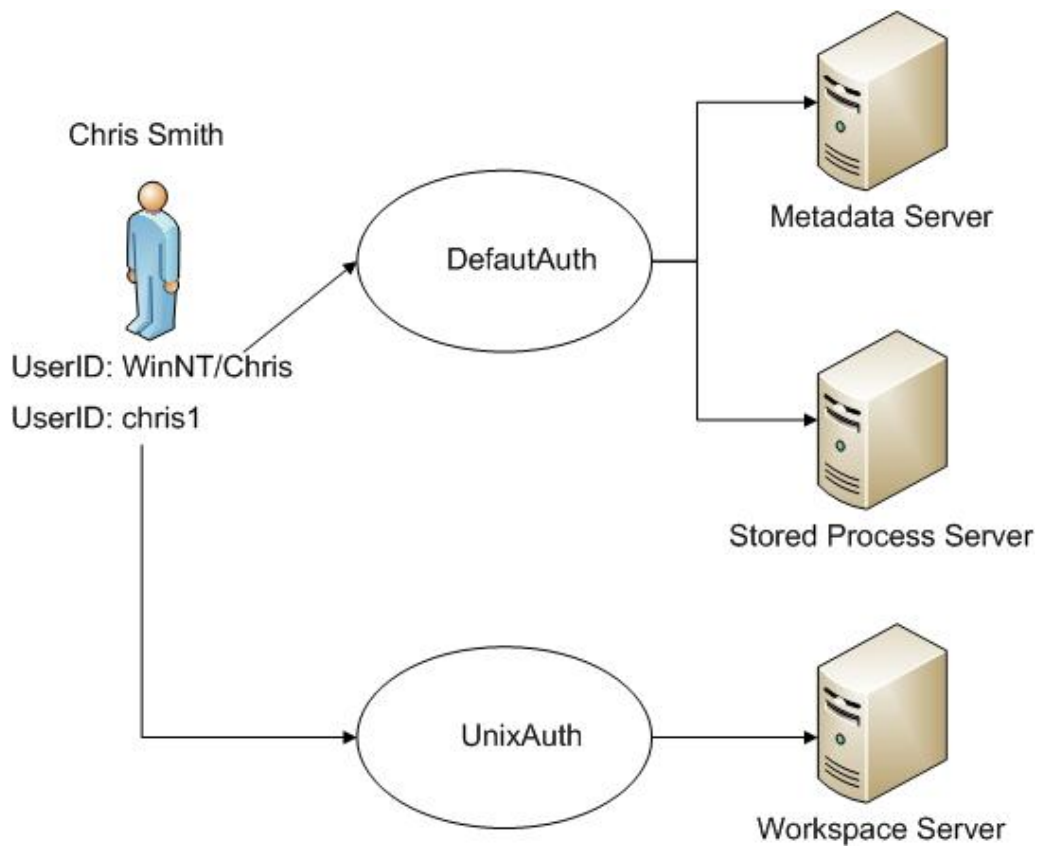
Groups can also be used to represent an organizational structure. Different groups can be used to define different geographic region or divisions within an organization. All groups should be defined in the foundation repository, not in any custom or project repository.



The identity hierarchy is a ranking based on the user's individual metadata identity and group membership. A user's identity hierarchy includes the individual's metadata identity, groups that have the user's identity as a member, a user-defined group that has another group as a member, sasusers, and public. Membership to the sasusers and public groups are implicit. The sasusers group includes all members of the public group that have a metadata identity. The public group includes everyone who can authenticate to the metadata server.

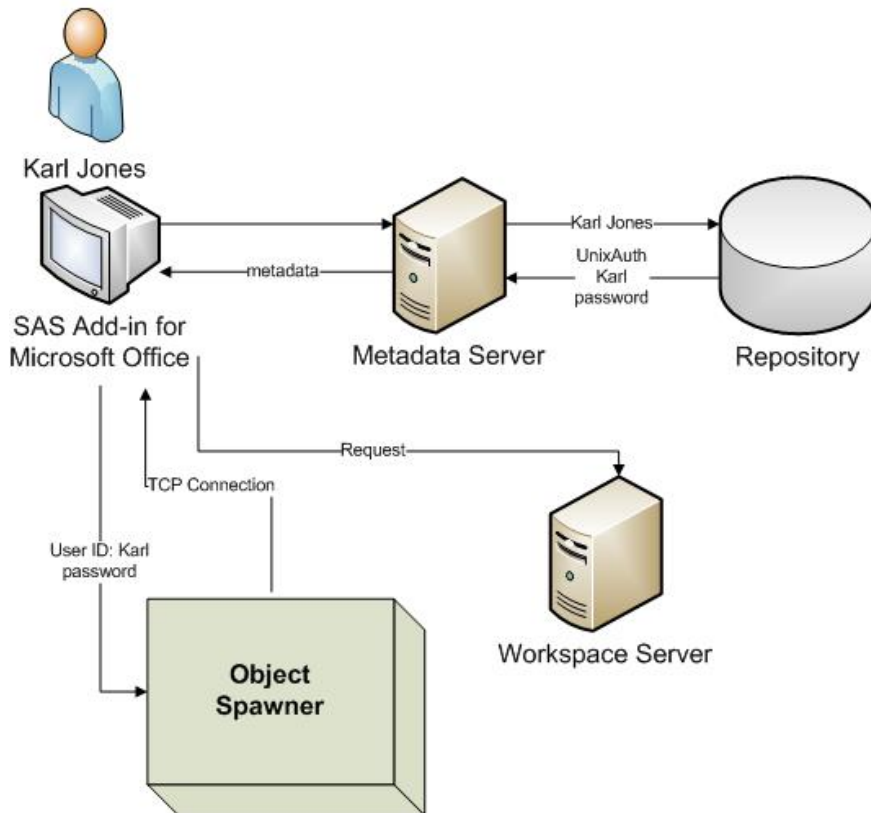
AUTHENTICATION

Before you can determine what a user can see or do, you must first authenticate the user's identity. In the SAS Intelligence Platform, a user's identity is verified when a user first logs on to an application, and when the user requests access to other systems. The user's ID and password can be provided by interactive prompting for the user credentials, retrieval from a file, or retrieval from metadata. Authentication is done by the authentication providers. In most cases, SAS servers rely on host authentication to verify identities. The host computer can be setup to use the operating system, LDAP or Active Directory. After the initial authentication occurs, additional authentication can be used to access other servers. Additional authentication can be used to enable users to access a variety of computing resources without being prompted repeatedly for credentials. SAS allows the applications to obtain credentials by reusing credentials that are cached from an interactive log on, retrieval from the metadata repository, or shared user context. The cached credentials are only valid on one authentication domain and can only be reused for servers using the same authentication domain.



For example, Chris Smith uses the WinNT/Chris login to access the Metadata server. Since the metadata server and stored process server are using the same authentication domain, the same credentials can be reused to access the stored process server as well. Those credentials will not be able to access the workspace server because the workspace server is associated with a different authentication domain.

In some situations when cached credentials cannot be used, retrieval of credentials from the metadata can be used. If the user has access to a login for the appropriate authentication domain, the login can be retrieved from the metadata. Users have access to logins associated with their individual metadata identity and logins associated with groups in their hierarchy.

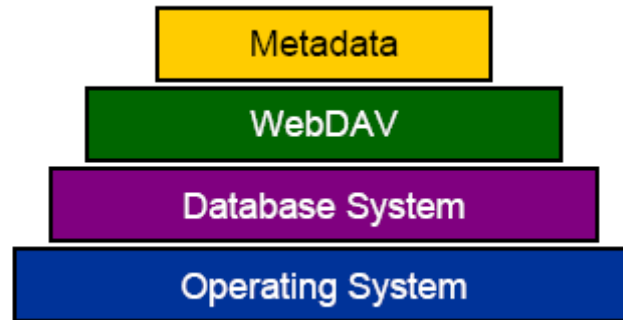


For Example, Karl Jones is using SAS add-in for Microsoft office to run some reports. The metadata server recognizes the handle belonging to Karl and searches the repository for the workspace server. The metadata server retrieves the name, connection information, and authentication domain of the workspace server. Since the cached credentials cannot be used, the metadata server searches the repository for credentials. The requested metadata, the host, the port and Karl's login associated with the workspace server's authentication domain is returned back the application. The application then uses the host and port number to send the request to object spawner. After the communication is established, the application passes the user's credentials to the object spawner. Once the login is determined to be valid then the object spawner provides the application with a TCP connection that the application should use to communicate directly with the workspace server. Then application sends its request to execute the task directly to the workspace server.

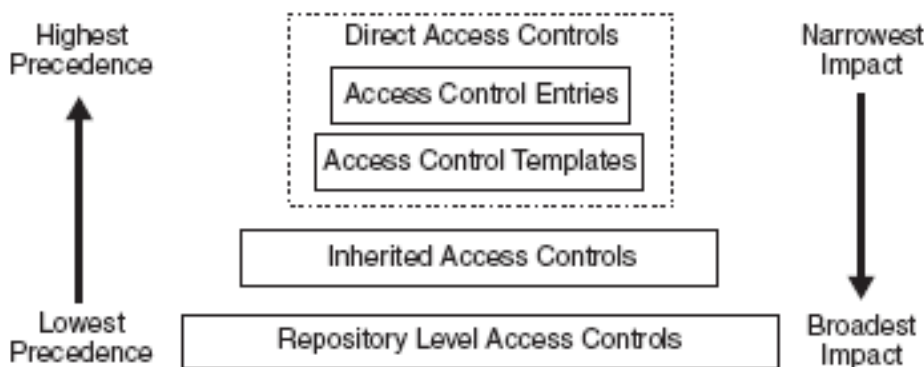
Shared context is used when single sign-on is implemented for web applications. Web applications can use shared context and session information to launch one web application within another web application. For example, a user can access the SAS Web Report Viewer from the portal application without logging in again. This is possible because the SAS Report Viewer shares the session and user context that was initiated when the user logged on to the SAS Information Delivery Portal.

AUTHORIZATION

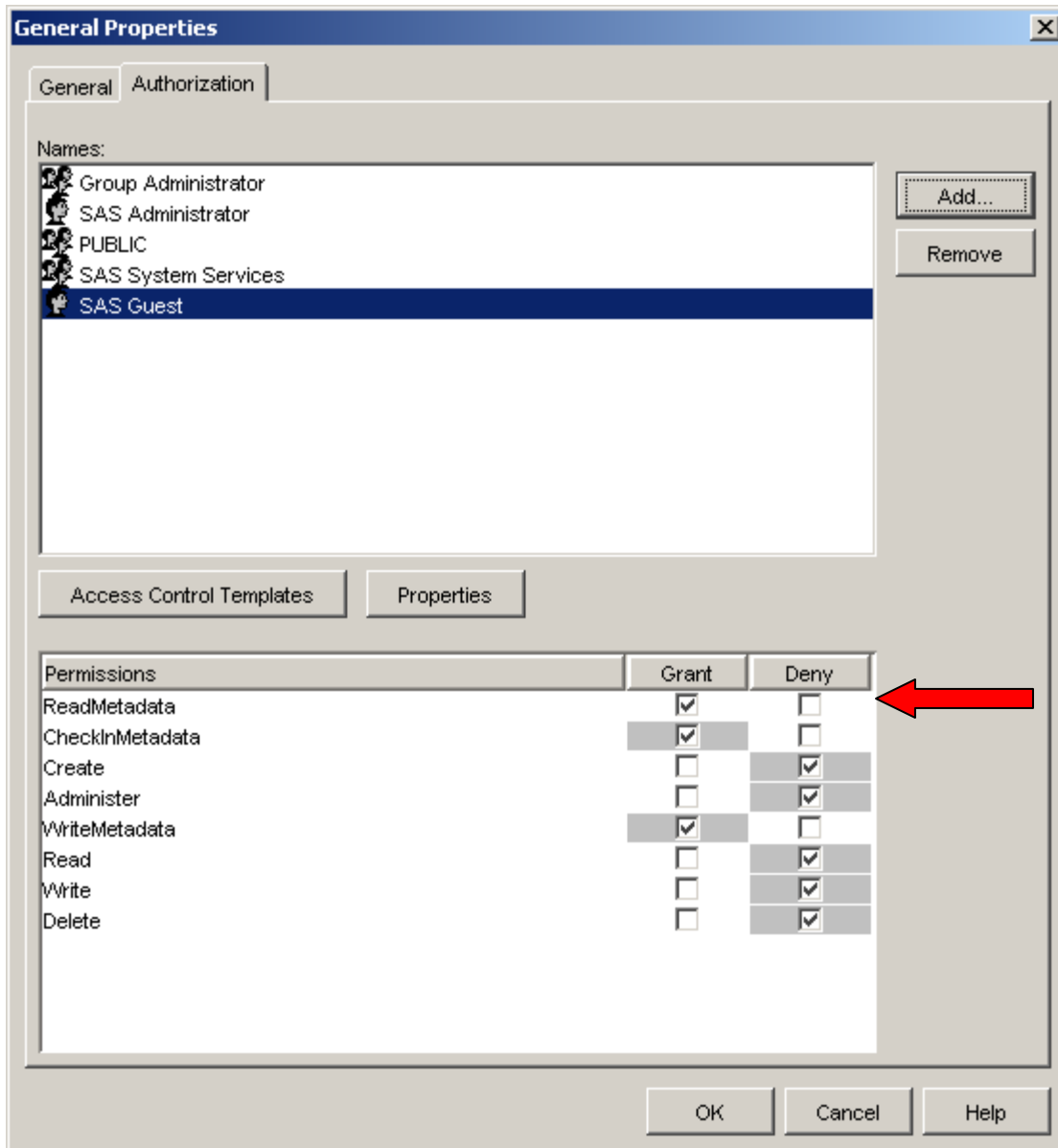
The outcome of the authorization process is a grant or deny for a specific action. Management of access controls are defined, stored, and evaluated by the authorization facility. A user's access can be affected by any part of the authorization layers.



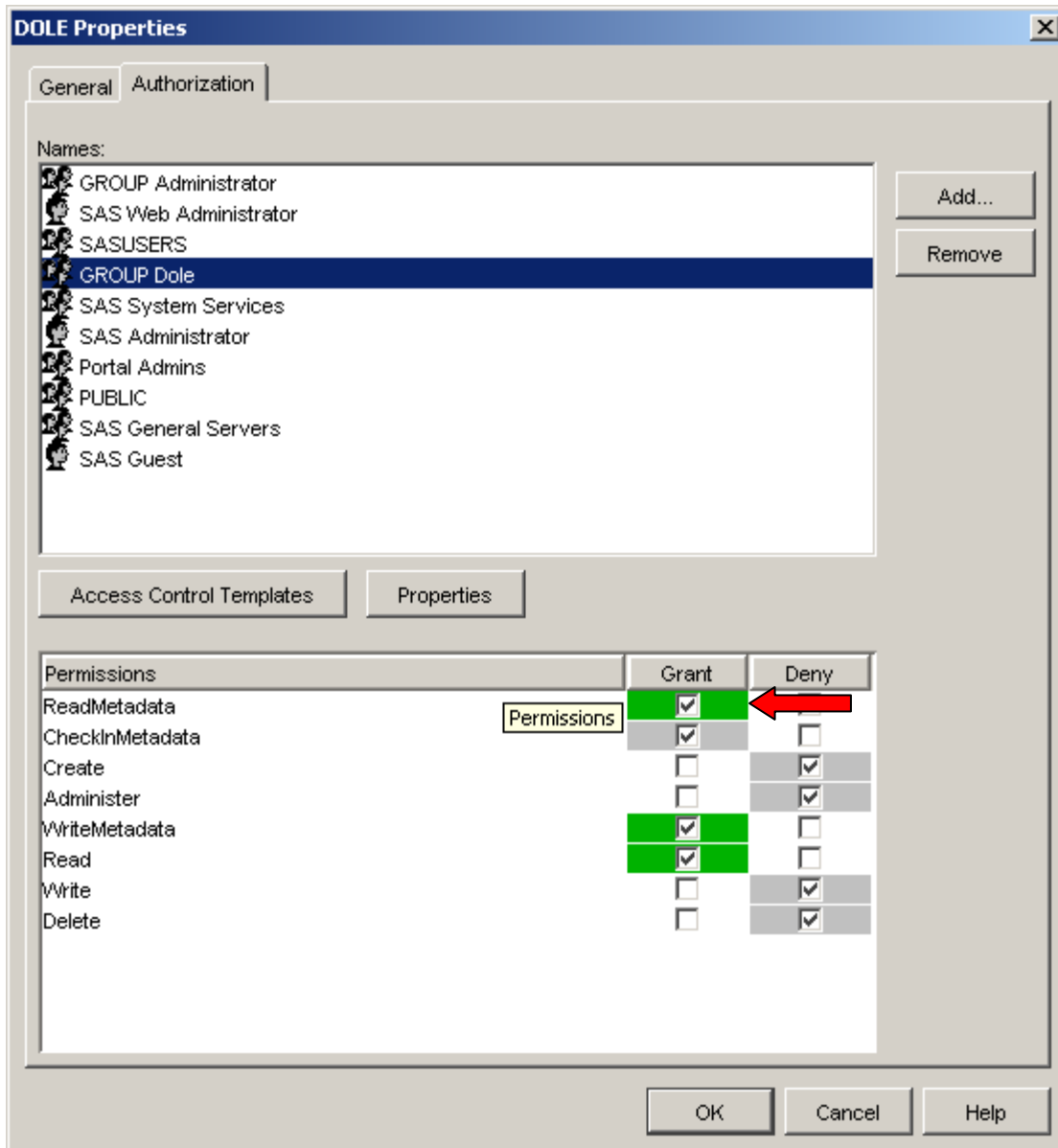
The metadata layer is part of the SAS intelligence platform and is controlled with the Access Manager. The WebDAV layer consists of third party Web server access controls on report content objects. The database system layer consists of permissions to relational database objects, passwords for SAS data sets, and other access control specific to data sources. The operating system authorization layer consists of the file, directory, and system permissions that you specify on the machine. The metadata authorization layer uses the following diagram to make authorization decisions.



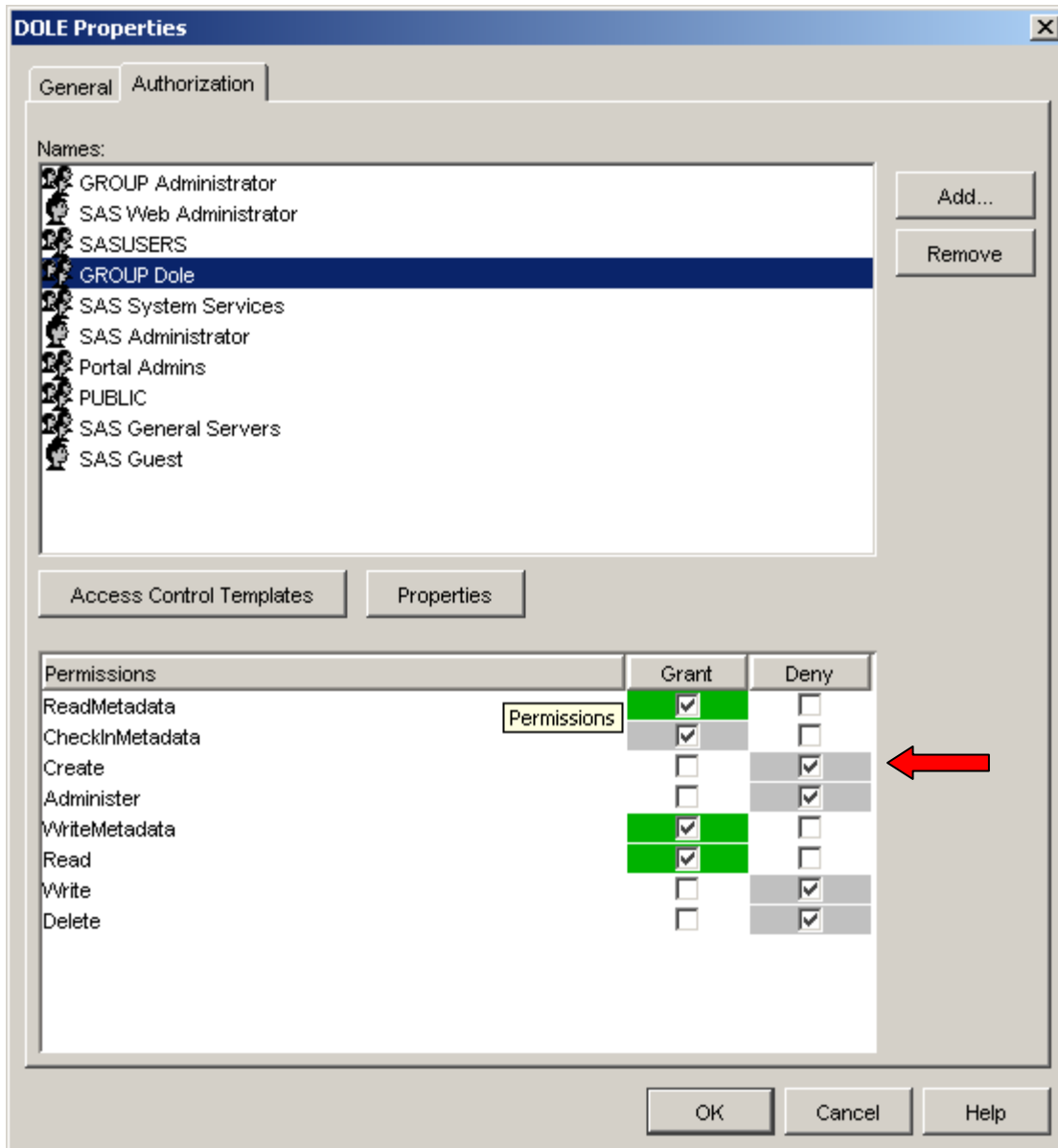
The highest precedence overrides a lower precedence. The narrowest impact will override the broadest impact. Direct Access Controls are Access Control Entries and Access Control Templates. Access Control Entry will be the most granular and will have a higher precedence over Access Control Templates. An ACE will have a white background just like in the picture below.



An Access Control Template is a reusable name pattern of identities and permissions. The ACT can be applied to a metadata object to specify a combination of permissions for a metadata identity or group. The background of the permission settings on the Authorization tab is green when an ACT is applied to a metadata object.



The grey background on the permissions settings mean that the permission is inherited from the parent object or the repository ACT. Inheritance from multiple parents is supported in the SAS metadata environment.



HOW AUTHENTICATION DECISIONS ARE MADE

First the permission on the target resource is examined. Any conflicts that arise from group membership are resolved by the identity hierarchy. For example, permissions assigned to a user overrides a conflict permission that is assigned to a group to which the user belongs. If there is a conflict between an ACE and an ACT at the same level in the identity hierarchy, then the ACE takes precedence. If there is a conflict between two ACE's or two ACT's at the same level 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. If there are no pertinent permissions set on the target resource then the evaluation process continues to the parent object or repository ACT. Inheritance rules are applied to identify all of the target resource's parent object.

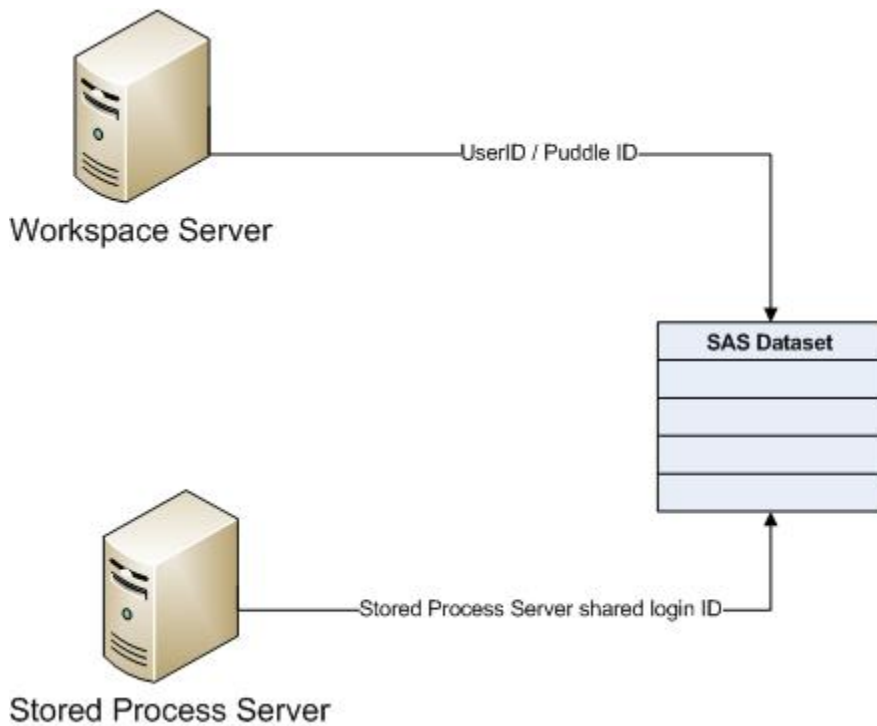
If any of the parent objects convey a grant then access is granted. If all of the parent objects convey denial then access is denied. If there are no parent objects then evaluation continues to the repository ACT. In the repository ACT, the Users and Permissions tab of the repository ACT is examined. If the repository ACT grants or denies the request permission, then that grant or denial is determinative. If the repository ACT neither grant nor denies the permission, then the permission is denied. If no repository ACT is defined then permission is granted.

SECURING METADATA OBJECTS

To secure metadata objects, you will need to secure the repository ACT. It is recommended to create a new ACT in management console and then set the new ACT as the repository ACT. By following the recommended security patterns for the repository ACT in the SAS 9.1.3 Intelligence Platform: Security Administration Guide, you should have the following permissions:

PUBLIC – Deny ReadMetadata, WriteMetadata
SAS Guest – Grant ReadMetadata, WriteMetadata
SAS Demo User – Grant ReadMetadata, WriteMetadata
Administrators group (sasadm) – Grant ReadMetadata, WriteMetadata
SAS System Services group (SAS Web Administrator, SAS Trusted User) – Grant ReadMetadata, WriteMetadata

Creating functional groups based on the tasks users will perform is a good idea. This will allow permissions to be assigned to the group instead of a specific user. Permissions should be assigned to the broadest group in the identity hierarchy. Access control templates should be used to centralize management of identity/permission patterns. This allows permission changes to be made easily in one location. It is important to allow readmetadata at the highest level and allow writemetadata at the lowest level. This is necessary to all users to navigate to the folder through folders to read a metadata object. By only allowing writemetadata only at the lowest level it will prevent users from saving metadata object into places where they should not be. When allowing writemetadata you should also allow readmetadata as well. If writemetadata is granted and readmetadata is denied then it is possible to create duplicate metadata objects. Setting permissions in the metadata layer is only part of securing metadata objects. WEBDAV, relational databases, and physical files will need to be secured as well.



For Example, when securing SAS libraries and tables, you will need to modify the permissions for metadata objects and the underlying physical files. Access for the physical file depends on which type of server is being used to access the library. If a standard workspace server is used then the user's login credentials will need access to the file. If a pooled workspace server is used then puddle login will be used to access the physical file. If a stored process server is used then the shared login will need access to the physical file.

CONCLUSION

Since every organization has different security requirements, developing a security plan should be done first. Securing your SAS BI deployment will involve more than just setting permissions in the access manager. Working with your IT department to secure the operating system layer, database layer, and WEBDAV layer should be done as well.

REFERENCES

SAS® 9.1.3 Intelligence Platform: Security Administration Guide, Second Edition.
<http://support.sas.com/documentation/configuration/bisecag.pdf>

TS-750 -Securing SAS®9 Business Intelligence content Managed in Metadata.
<http://support.sas.com/techsup/technote/ts750.pdf>

CONTACT INFORMATION

Your comments and questions are valued. Contact me at:

Minh Duong
University of Houston TIMES
E-mail: minh.duong@times.uh.edu