SECURITY MODEL DESIGN PRINCIPLES

SAS ADMINISTRATOR USER GROUP SESSION 2 2015

SAS PLATFORM SECURITY

SESSION OUTLINE

1. What a good security model should cover
2. Whistle-stop tour of features in SAS which support a security model
3. Designing a maintainable model for authorisation
4. Questions & where to learn more

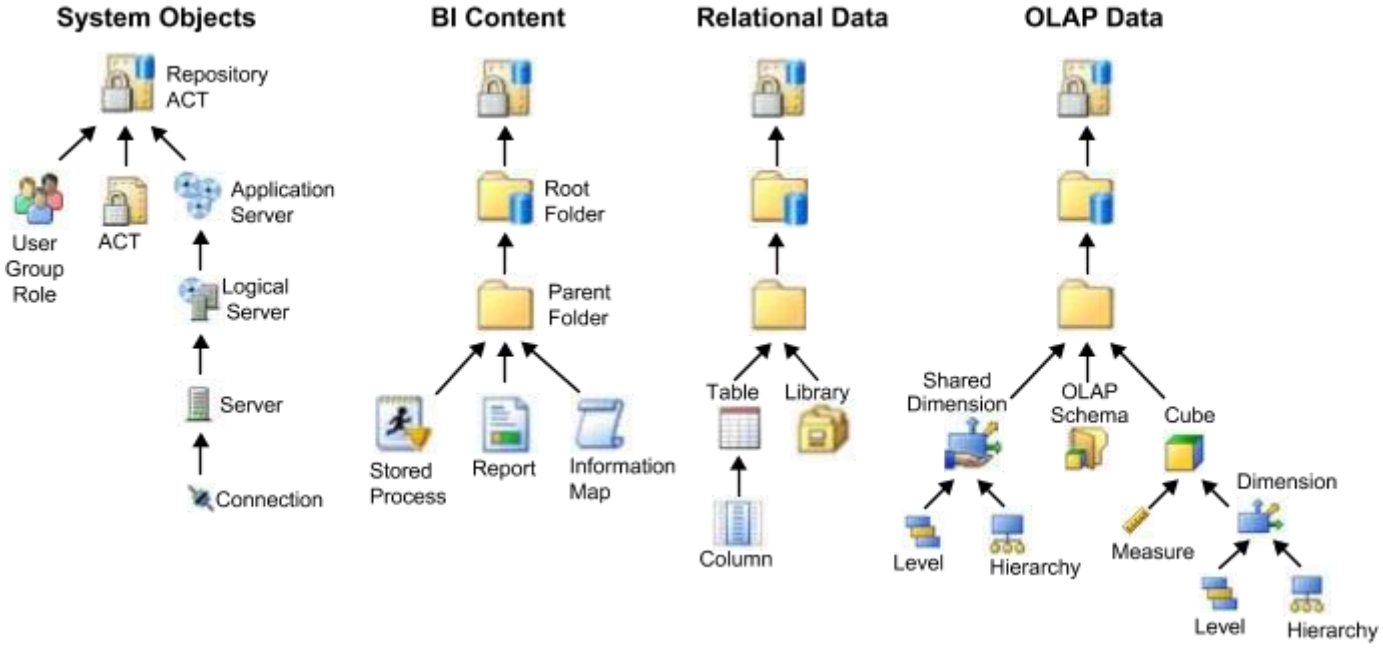
1. WHAT A GOOD SECURITY MODEL SHOULD COVER

- Authentication
- Access rights (authorisation) for SAS, Database and OS assets belonging to the SAS application
- Protection of system integrity
- Audit

2. WHISTLE-STOP TOUR OF FEATURES IN SAS WHICH SUPPORT A SECURITY MODEL

1. Supported authentication mechanisms
2. Metadata security: Users, Groups, Access Control Entries (ACEs) and Access Control Templates (ACTs)
3. SAS's rules for how metadata permissions are inherited
4. Database security
5. OS and file-level security
6. SAS/Secure for credentials, over-the-wire encryption and encrypting data at rest
7. Compliance with FIPS 140-2
8. Synchronising authorisation in SAS with external databases
9. Resilience/clustering
10. Backup and restore tools
11. Security Macros/Environment Manager/APM/Other tools

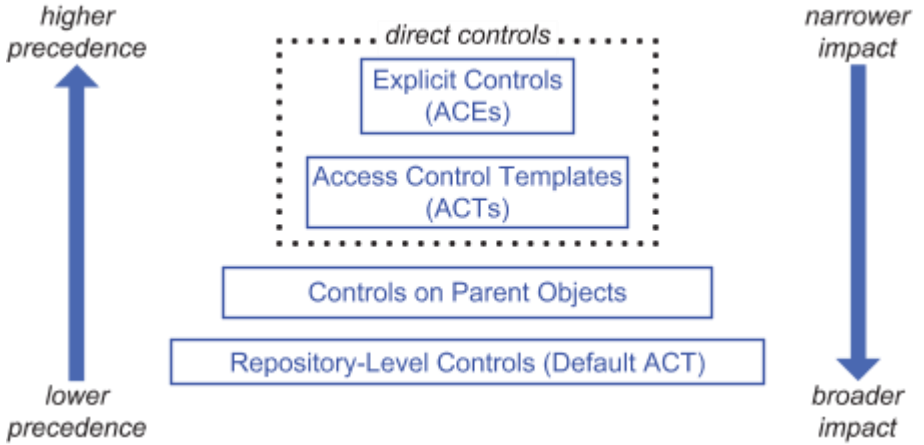
SAS'S RULES FOR HOW METADATA PERMISSIONS ARE INHERITED



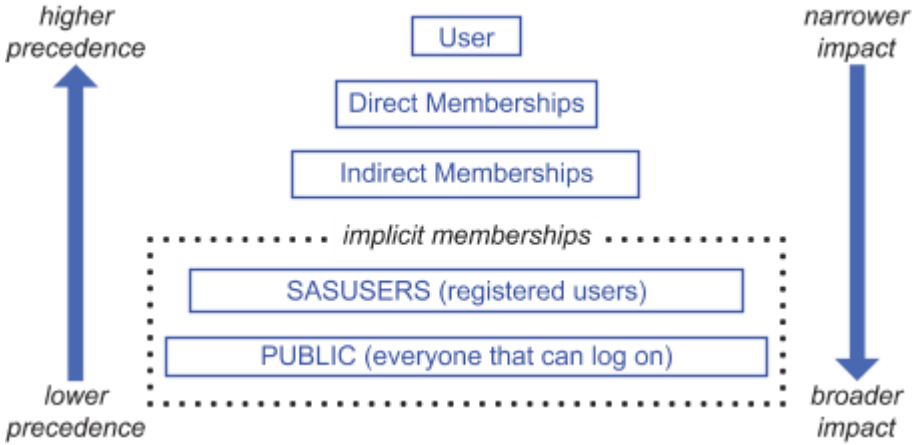
Source: SAS 9.4 Intelligence Platform Security Administration Guide
Authorisation > Metadata Authorization Model > Object Inheritance

SAS'S RULES FOR HOW METADATA PERMISSIONS ARE INHERITED

Priority and Specificity in Object Inheritance

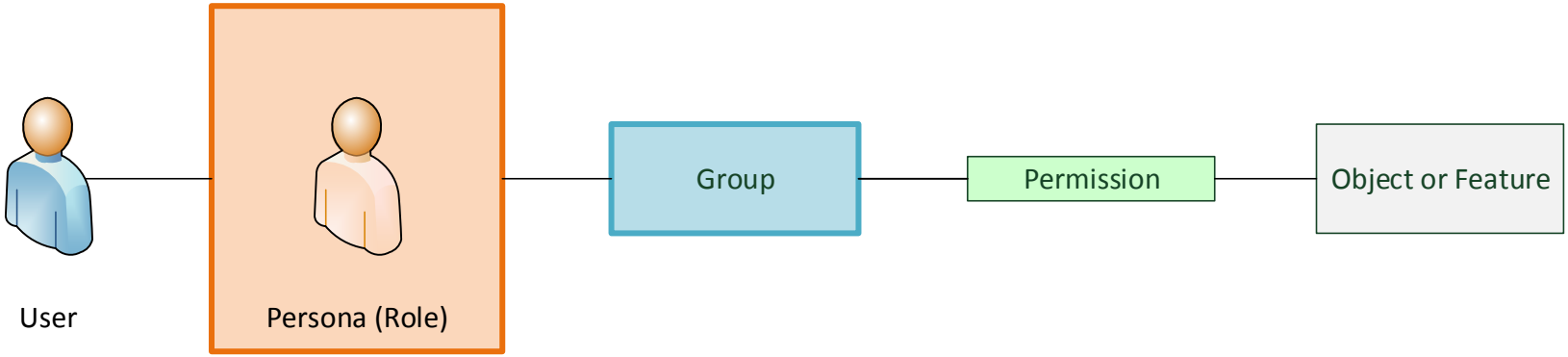


Priority and Specificity in Identity Hierarchy



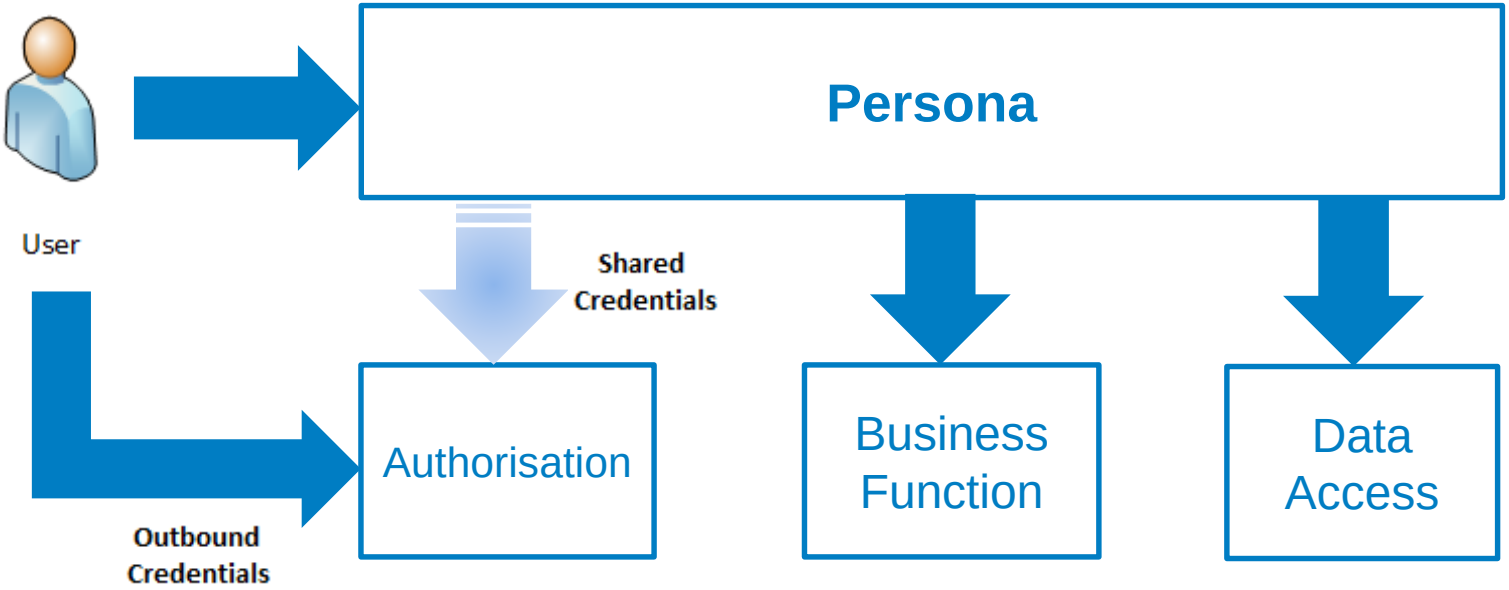
Source: SAS 9.4 Intelligence Platform Security Administration Guide
Authorisation > Metadata Authorization Model > Object Inheritance

3 DESIGNING A MAINTAINABLE MODEL FOR AUTHORISATION

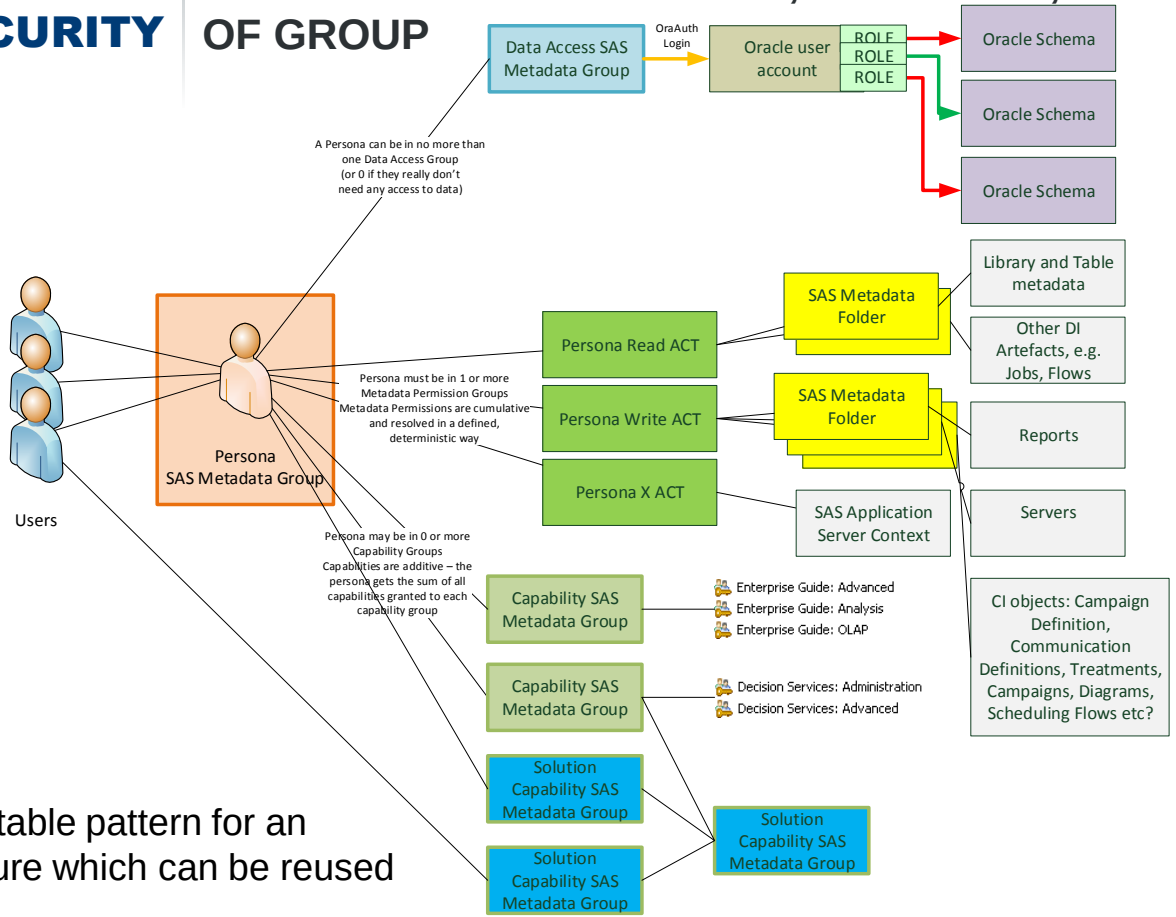


A Persona is just a metadata group, different from other groups only in the way it is used

USERS AND PERSONA DEFINITION



KEEPING IT SIMPLE: USERS, PERSONAS, THREE TYPES OF GROUP



Container: a repeatable pattern for an authorisation structure which can be reused

TIPS FOR KEEPING IT SIMPLE

- **Design a metadata folder hierarchy** for your application, and enforce its use
  - Also design an OS directory structure, which may mirror the metadata folders to a degree
- **Apply permissions to metadata folders**, not to the objects they contain (where possible)
  - All objects in a metadata folder should have the same permissions
- **Don't use Access Control Entries** to grant permissions, only use ACTs, (…and use them on folders)
- **Keep ACTs simple** – each ACT should feature only one Persona, and most need only grant 'read' permissions, 'write' permissions or 'special' permissions. You can therefore name most ACTs e.g. "*Persona* Read ACT"

FIVE KEY TAKEAWAYS TO KEEP IT SIMPLE

1. Ensure user is only in one persona and think of a persona as your container
2. Limit number of roles
3. Re-use the folder structure templates where applicable
4. Avoid use of ACE’s and instead use ACT’s
5. Apply the same security model to ALL environments, instead change the persona that a user belongs to in each environment

QUESTIONS?

REFERENCES

- SAS® 9.3 Intelligence Platform Security Administration Guide
- SAS® 9.4 Intelligence Platform Security Administration Guide, Second Edition
- What's New in Security Administration in SAS 9.4
- FIPS 140-2 Standards Compliance
- Federal Information Processing Standard (FIPS) publication 140-2 on Wikipedia