

Manage Users, Groups, and Permissions in AEM

Agenda:

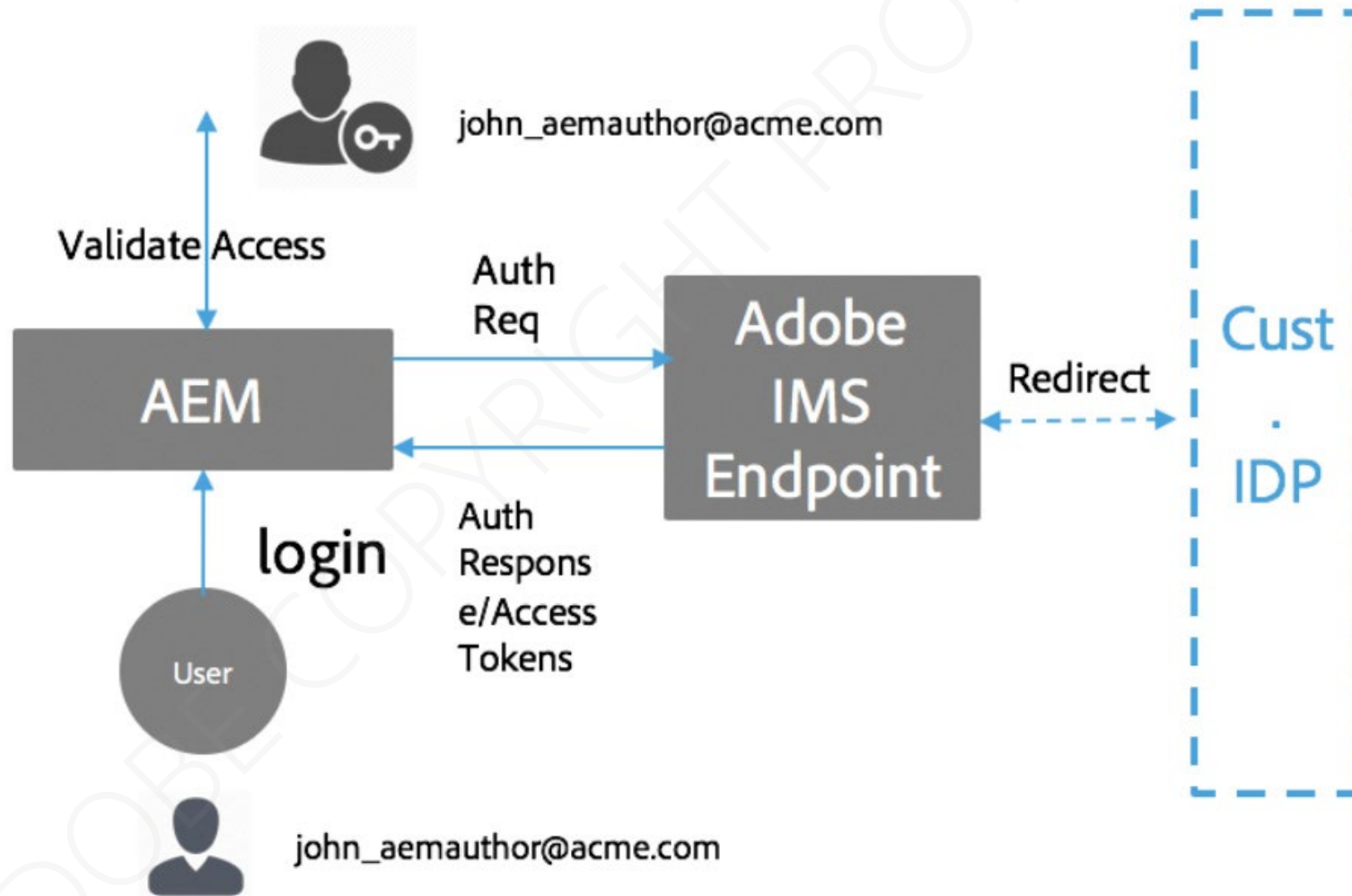
- Identity Management System (IMS)
- Permissions and Access Control Lists (ACLs)
- Creating and configuring users and groups in Adobe Experience Manager (AEM)
- Group permissions
- Synchronization of permission groups to the Maven project



Adobe Identity Management System (IMS)

- An authentication service for all Adobe products
- Can be configured with a customer's
 - SSO
 - LDAP
 - Other Identity Providers (IDP)
- AEM IMS service accounts are tightly integrated with
 - Adobe Launch
 - Adobe Target
 - Adobe Stock
 - Any Adobe I/O connection

IMS architecture



IMS IDs

Three types of identities of accounts to authenticate and authorize users:

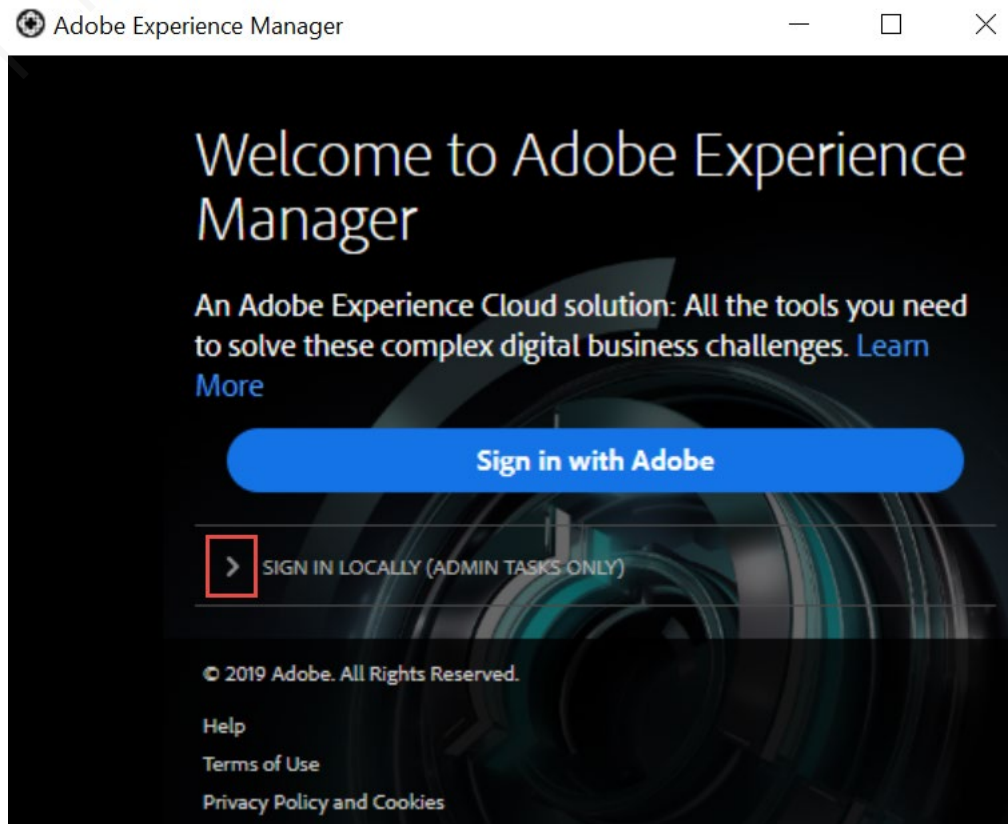
- Federated ID
- Enterprise ID
- Adobe ID

The Enterprise and Federated ID models are an ideal choice for enterprises.

The Adobe ID is best suited for individuals.

IMS for AEM authentication

- Customer onboarding to the Admin console is a prerequisite
- Must have an IMS organization provisioned by Adobe
- Joint effort between Adobe and customers
- Supports Local and IMS authentication

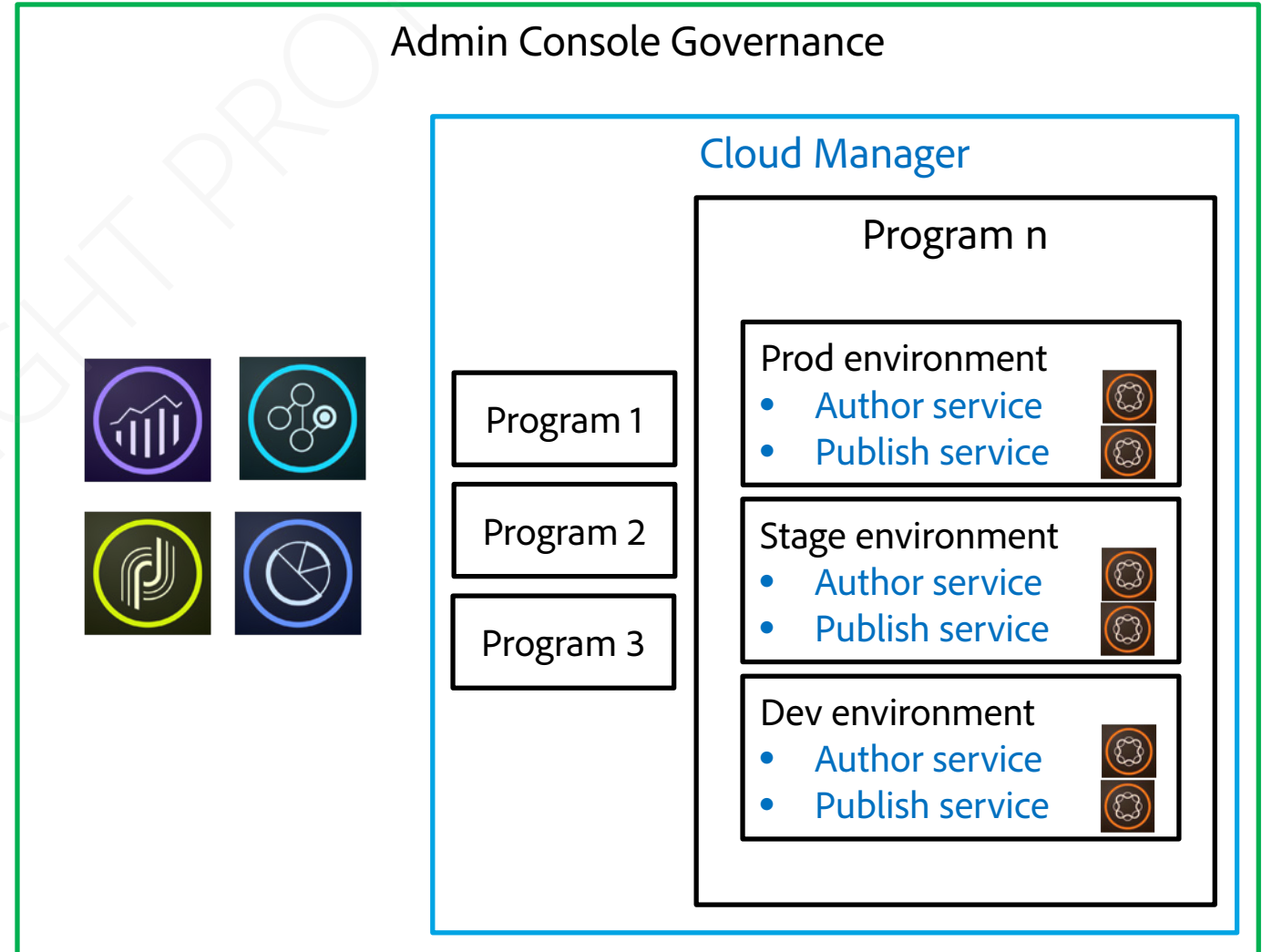


Admin console

- Admin Console manages all users and groups in the Adobe Experience Cloud
- Users can be added product profiles in the Admin Console
- Product profiles give access to Cloud Manager, AEMaaCS and other Adobe products
- Administrators use the Adobe Admin Console to
 - Create, import, and manage user accounts
 - Assign users to Adobe products
 - Add users to product profiles (groups) to assign permissions
- AEM as a Cloud Service users:
 - Log in to AEM with IMS credentials
 - Product profiles the user is apart of are imported as AEM groups

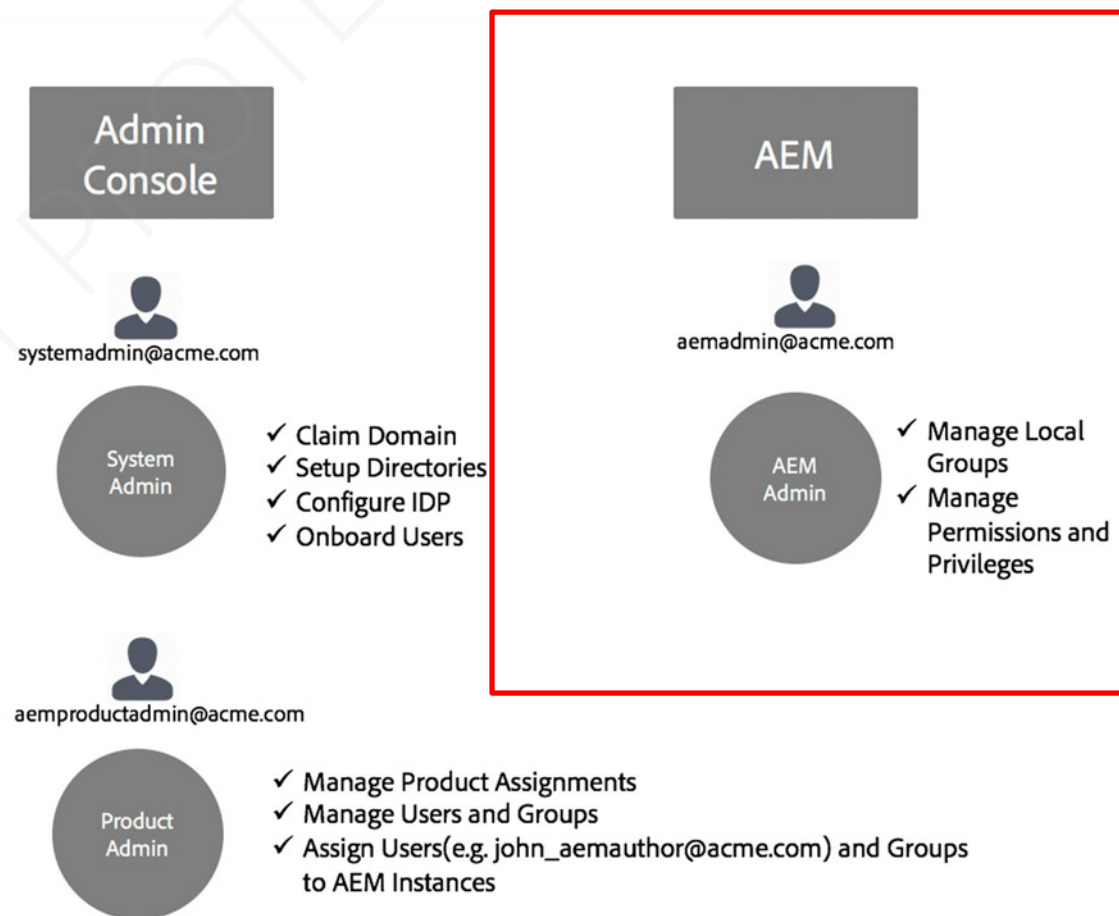
AEM product profiles in Admin Console

- Controls Cloud Manager access for
 - Program management
 - Git Access
 - Environment Deployments
 - Pipeline Execution
- Controls AEM Service Authentication
 - Separate access per service
- Syncs Groups for Authorization
 - Permissions for groups occurs within the AEM service



Understanding Admin roles in AEM

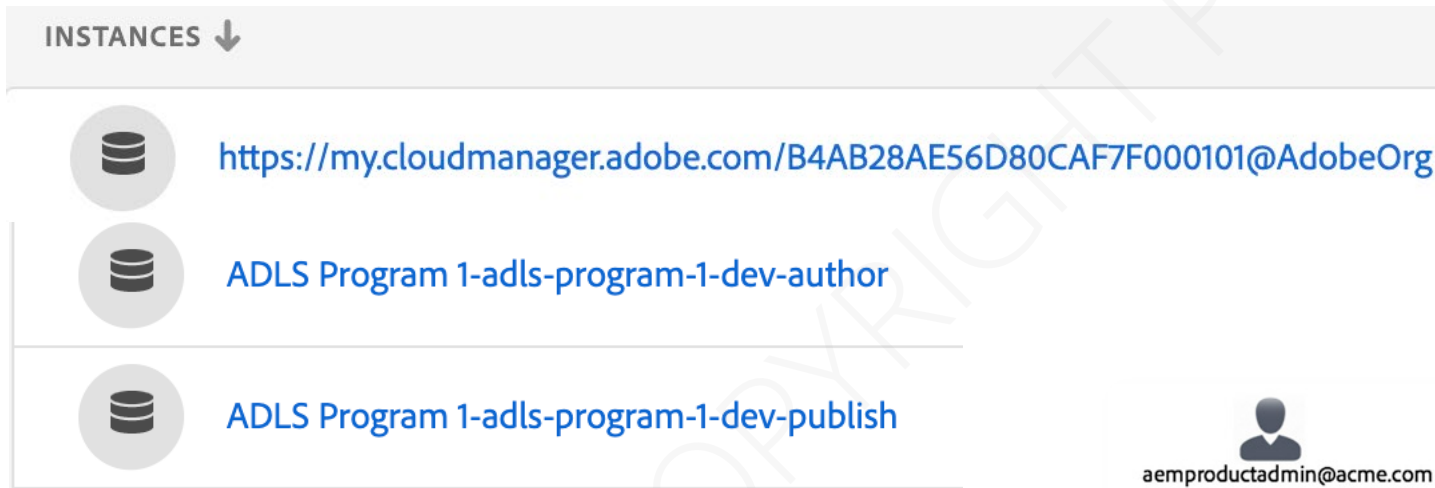
- System Admin
 - Federated ID
 - Enterprise ID
 - Adobe ID
- AEM Product Admin
 - Add users to Cloud Manager
 - Add users to AEM Services
 - Create new product profiles for group membership
- AEM Admin
 - Manages AEM group permissions (within AEM)



AEM Admin will be the focus of this Module

Terminologies in the Admin console

- Target, Analytics, Audience Manager (Products)
- Adobe Experience Manager Cloud Service (Product)




aemproductadmin@acme.com



- ✓ Manage Product Assignments
- ✓ Manage Users and Groups
- ✓ Assign Users(e.g. john_aemauthor@acme.com) and Groups to AEM Instances

instances = AEM Services

cloudmanager instance = Governs Cloud Manager




Instance naming

Adobe Experience Manager Cloud Service (Product)

- <https://my.cloudmanager.adobe.com/<UID>>
- <programName>-<envName>-[author | publish] (instance)
- <programName>-<envName>-[author | publish] (instance)
 - AEM Users-XXXX (profile)
 - AEM Administrators-XXXX (profile)

instances = AEM Services

cloudmanager instance = Governs Cloud Manager

INSTANCES ↓	
	https://my.cloudmanager.adobe.com/B4AB28AE56D80CAF7F000101@AdobeOrg
	ADLS Program 1-adls-program-1-dev-author
	ADLS Program 1-adls-program-1-dev-publish

AEM service profiles

AEMaaCS (Product)

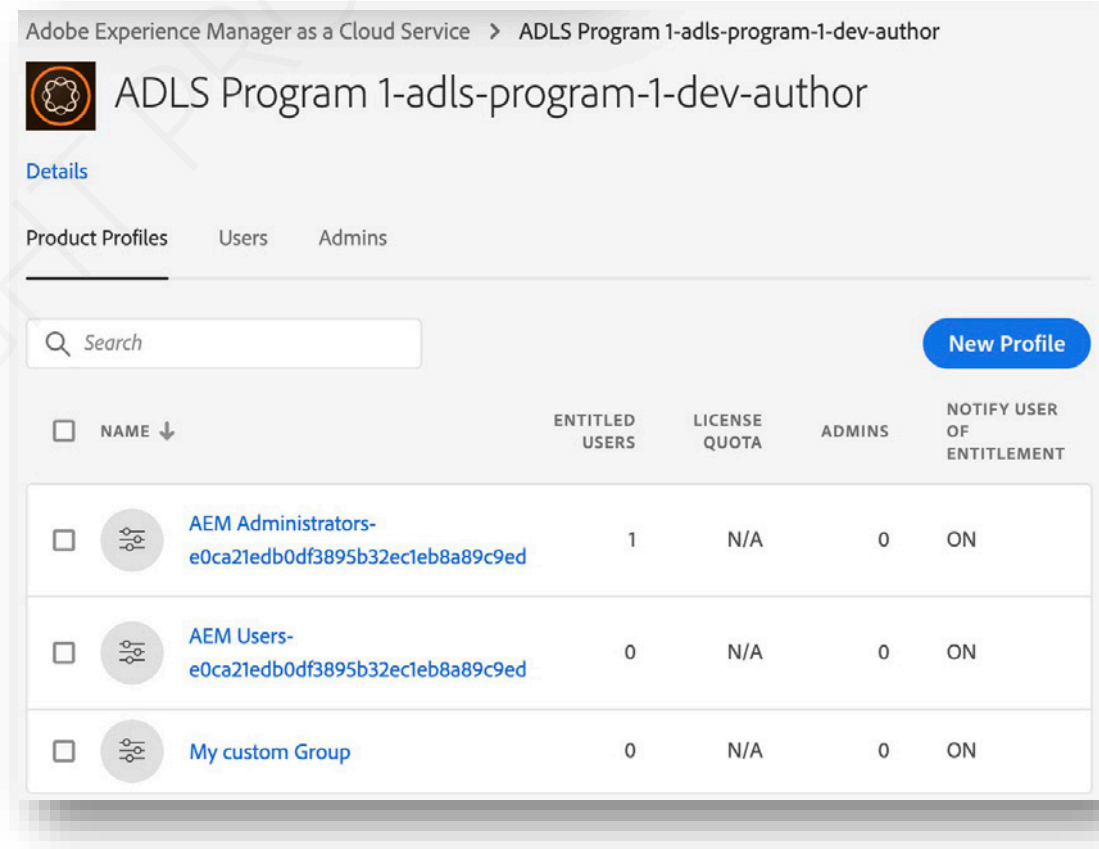
- AEM Service (Instance)
 - AEM Users-<UID> (profile)
 - AEM Administrators-<UID> (profile)
 - <custom profiles> (profile)

Profiles sync as AEM groups




AEM Groups are permissioned within the AEM Service

instances = AEM Services

cloudmanager instance = Governs Cloud Manager



The screenshot shows the 'ADLS Program 1-adls-program-1-dev-author' page in the Adobe Experience Manager console. It displays a table of product profiles under the 'Product Profiles' tab. The table has columns for 'NAME', 'ENTITLED USERS', 'LICENSE QUOTA', 'ADMINS', and 'NOTIFY USER OF ENTITLEMENT'. There are three profiles listed: 'AEM Administrators', 'AEM Users', and 'My custom Group'. Each profile has a checkbox on the left and a gear icon for configuration. The 'AEM Administrators' profile has 1 entitled user, while the others have 0. All profiles have a license quota of 'N/A' and 0 admins. The 'NOTIFY USER OF ENTITLEMENT' column is set to 'ON' for all profiles. A 'New Profile' button is visible in the top right corner of the table area.

	NAME ↓	ENTITLED USERS	LICENSE QUOTA	ADMINS	NOTIFY USER OF ENTITLEMENT
<input type="checkbox"/>	 AEM Administrators- e0ca21edb0df3895b32ec1eb8a89c9ed	1	N/A	0	ON
<input type="checkbox"/>	 AEM Users- e0ca21edb0df3895b32ec1eb8a89c9ed	0	N/A	0	ON
<input type="checkbox"/>	 My custom Group	0	N/A	0	ON

Cloud manager profiles

AEMaaCS (Product)

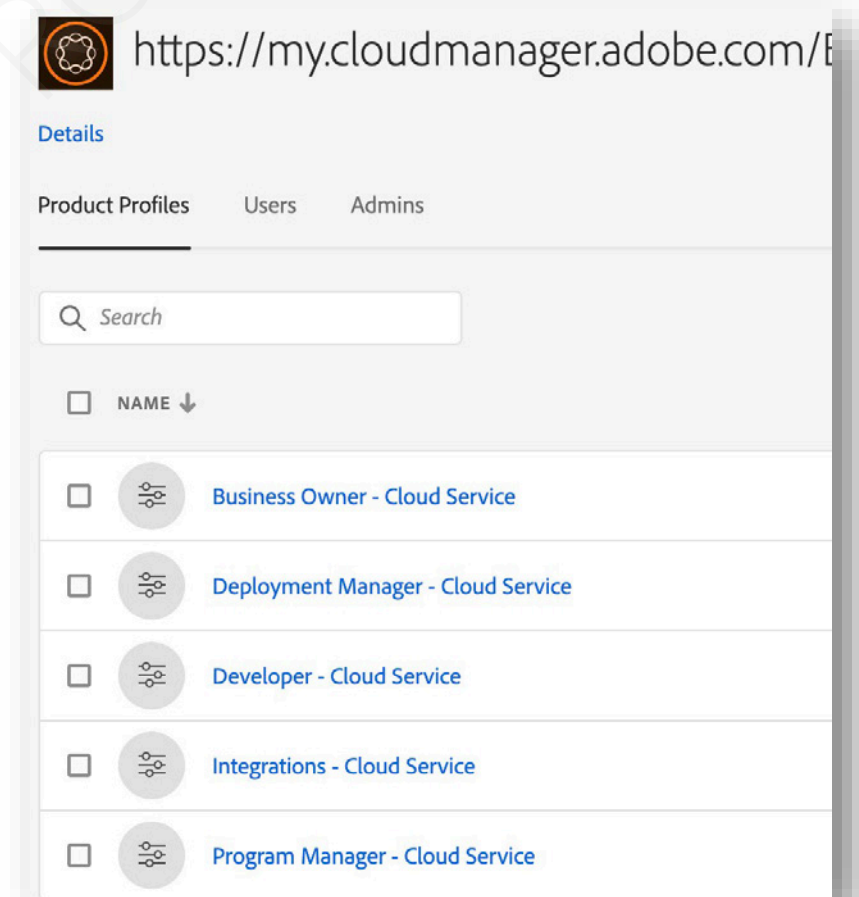
- <https://my.cloudmanager.adobe.com/<UID>>

Each profile is outlined on helpx. In general:

- Business Owner - full control
- Deployment Manager – manages pipelines
- Developer – access to git/pipelines
- Integrations – for services
- Program Manager – Mainly read and some execution permissions

instances = AEM Services

cloudmanager instance = Governs Cloud Manager



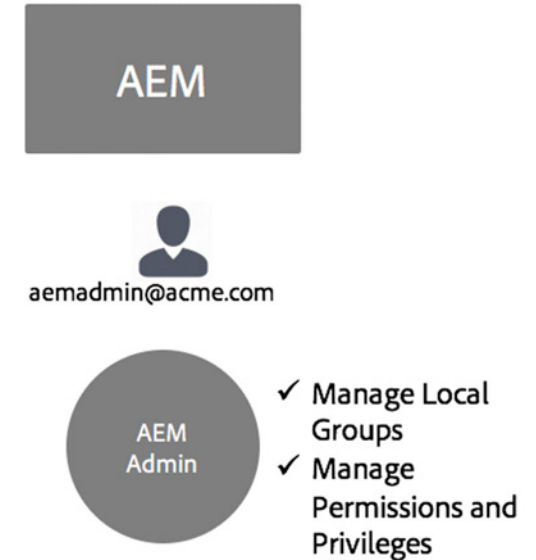
Access control and authorization for profiles (groups)

Access Control

- Restricting access to a resource
- Stored as a list of permissions granted or denied

Authorization

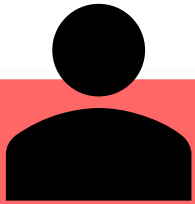
- Determining the action based on user access in specific areas within the system
- Comparing the user's identity against a list of permissions



Occurs within an AEM Service

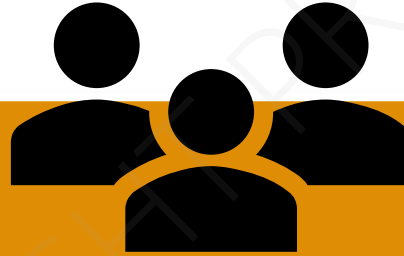
Principals, Users, and Groups

A principal in AEM can be a user or a group.



Users:

- Are humans or an external system
- Have accounts to store the details required to access AEM



Groups:

- Are collections of users and other groups
- Are used to simplify the AEM permissions



Service Users:

- Are special users associated with OSGi services

Default users for AEM

admin:

- System administration account
- Available **only** on Local AEM
 - admin/admin

Admin rights can be given by

- Adding users to the **administrators** group within the AEM Service
- Adding users to the **AEM Administrators-<UID>** profile in the Admin Console

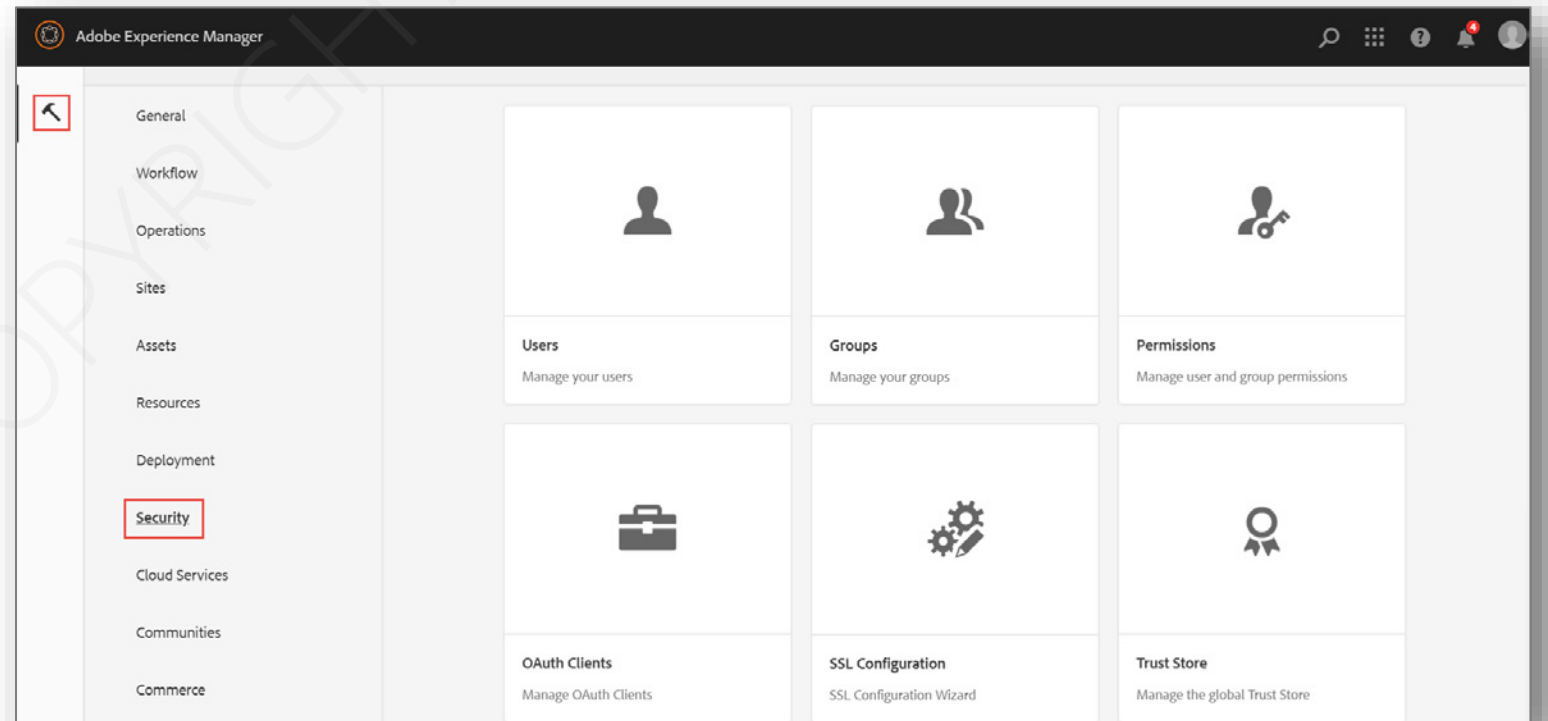
anonymous:

- Has the default rights for unauthenticated access to an AEM service
- Has minimum read access rights
- Most often used on AEM Publish Service

AEM service Security console

Within AEM, the Security console can

- View synced users/groups
- Create local users/groups (will not sync to the Admin Console)
- Set granular permissions



Users and groups consoles

Adobe Experience Manager

General

Workflow

Operations

Sites

Assets

Resources

Deployment

Security

Users

Manage your users

Groups

Manage your groups

Adobe Experience Manager

Search

Grid

Help

Notifications

User

User Management

Select All

Create

Name	Login Name	Status	Job Title	Published
	account-manager	Enabled	N/A	Not Published
Package-	acs-commons-automatic-package-replicator-service	Enabled	N/A	Not Published
-service	acs-commons-bulk-workflow-service	Enabled	N/A	Not Published

Adobe Experience Manager

Search

Grid

Help

Notifications

User

Group Management

Select All

Create

Description	Members	Published	Modified
	1	Not Published	Not Modified
AEM forms group with permissions to write scripts in Adaptive Forms template	0	Not Published	Not Modified
Analytics Administrators group	11	Not Published	Not Modified

Built-in groups in AEM

Integration/Solution Specific

- Adobe Analytics
- Adobe Target
- AEM Forms
- AEM Assets
- AEM Sites Authoring

AEM Feature Specific

- Projects
- Page templates
- Workflows
- Experience Fragments
- Tagging

Platform Specific

- Administrators
- User Admins
- Contributors (Read only)

Example:

myUser is a member of **myGroup**

myGroup is a member of **content-authors**

myGroup is a member of **workflow-users**

This example gives **myUser** full access to build pages and run workflows

Use built-in groups as much as possible

Access consoles for content

Access is given based on the closest applicable permission up the content tree if there is no local permission

Console	Permissions
Sites > myPage > Properties Experience Fragments > myXF > Properties	Browse, Edit, Delete, Publication, Create sub-page
Assets > myFolder > Properties Collections > myCollection > Properties Projects > myProject > Team	Owner, Editor, Viewer
Configuration Browser > myConfig > Properties (editable templates, Content Fragment Models, contexthub, and cloud services)	Browse, Modify, Delete

Assigning content permissions

+

Add Permissions

Edit Closed User Group

Effective Permissions

English

Closed User Group

Browse page

There is no item.

User or group

Browse page

Edit content

Delete page

Publish/unpublish page

Authors

Add Permissions

Select user or group

Authors

☒ Browse page

☒ Publish/unpublish page

☒ Edit content

☒ Create sub-pages

☒ Delete page

Cancel

Add

Roughly equivalent to
CRUD permissions

Effective Permissions

Principal	Path	Browse	Modify	Delete	Replicate
Authors	/content/we-retail/language-masters/en	✓	✓	✓	✓
content-reader-service	/content	✓			
content-writer-service	/content	✓	✓		
translation-preview-service	/content	✓	✓	✓	
workflow-process-service	/content	✓	✓	✓	✓
activity-service	/content	✓			

Permissions beyond built-in groups and consoles

Use built-in groups as much as possible for initial permissions

Use the Access Consoles to allow non-technical users to manage access

Beyond the basics:

- Access control is handled on the repository level (JCR)
- Access Control Entries (ACE) give access
- Permissions are stored as nodes
- Users and Groups are generically called authorizables

Access Control Entries (ACE)

Access control entries consists of:

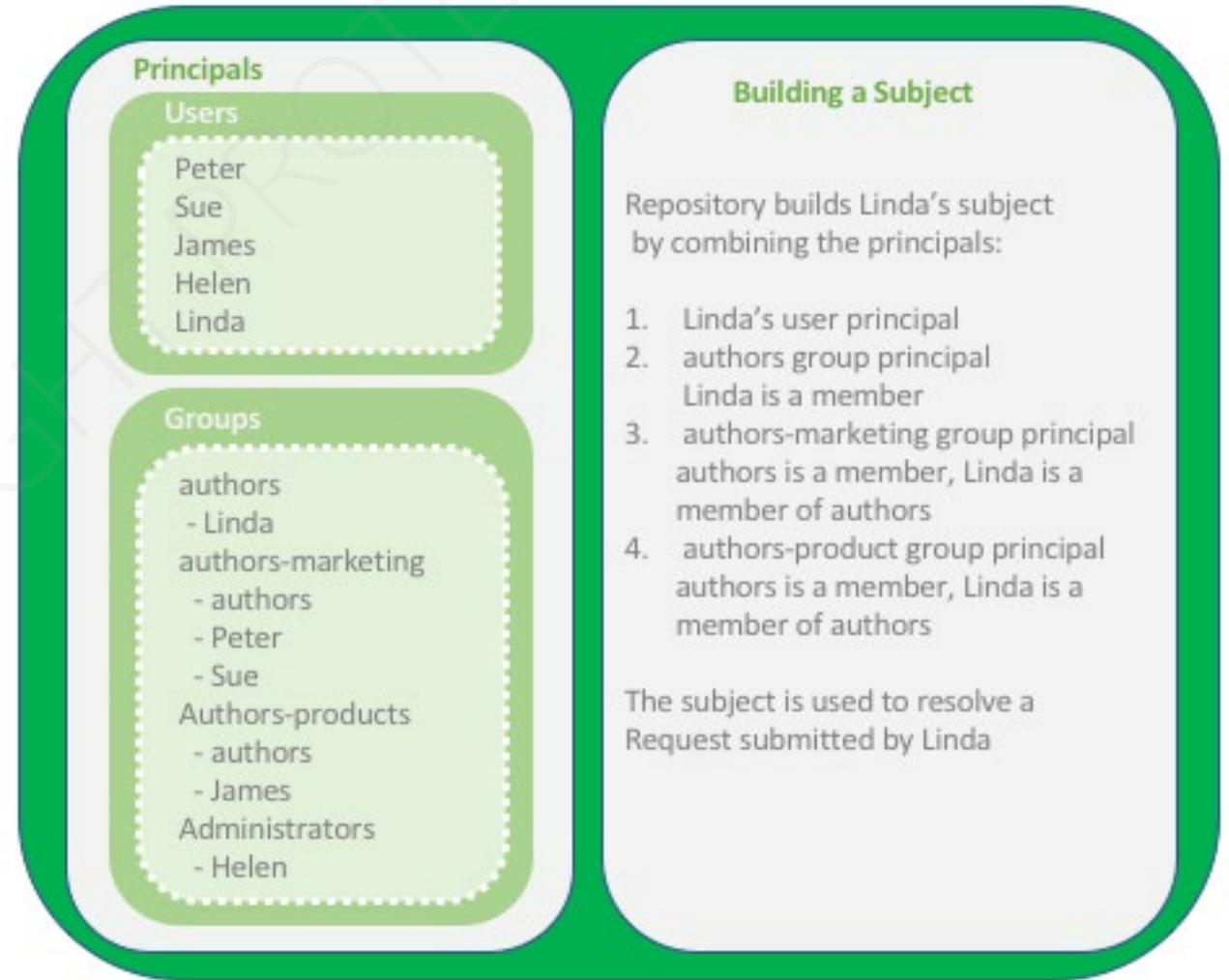
- Principals: A user or a group authorizable
- Subjects: The combined rights assigned to a user
- Permissions/privileges

Access Control Lists (ACL)

- List of ACEs on that specific resource

Example

- A Principal is a user or a group
- A group might contain other principals
- A Subject is a union of all the access rights granted directly to the principal and all the access rights granted through the group membership.



Access rights precedence

- ACEs in the JCR follow these ordered rules:
 1. Check for local access rights on the node.
 2. If there are none, the JCR picks the nearest ancestor whose ACE applies.
 3. If the node has an ACE about the user, it's used and ignores any group statements.
- As a good practice:
 - Do not assign access rights directly to users, only to groups.
 - Use Deny clauses sparingly.
 - For the bulk of ACEs, only use Allow statements.

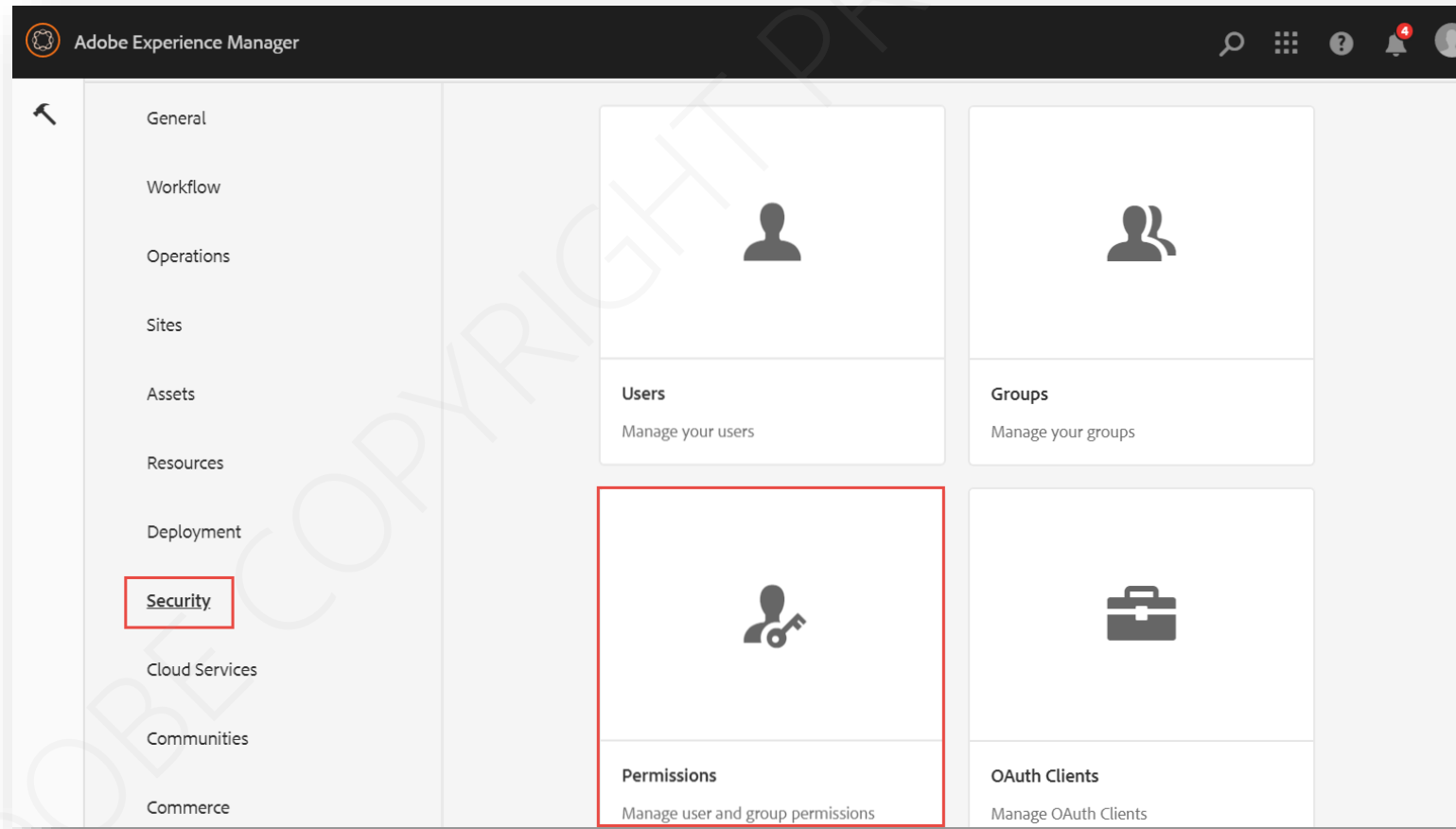
ACE privileges

The JCR has privileges that specify access of the principal (user/group)

- jcr:read
- jcr:modifyProperties
- jcr:addChildNodes & jcr:removeChildNodes.
- jcr:removeNode
- jcr:readAccessControl
- jcr:modifyAccessControl
- crx:replicate
- jcr:write
- And more....

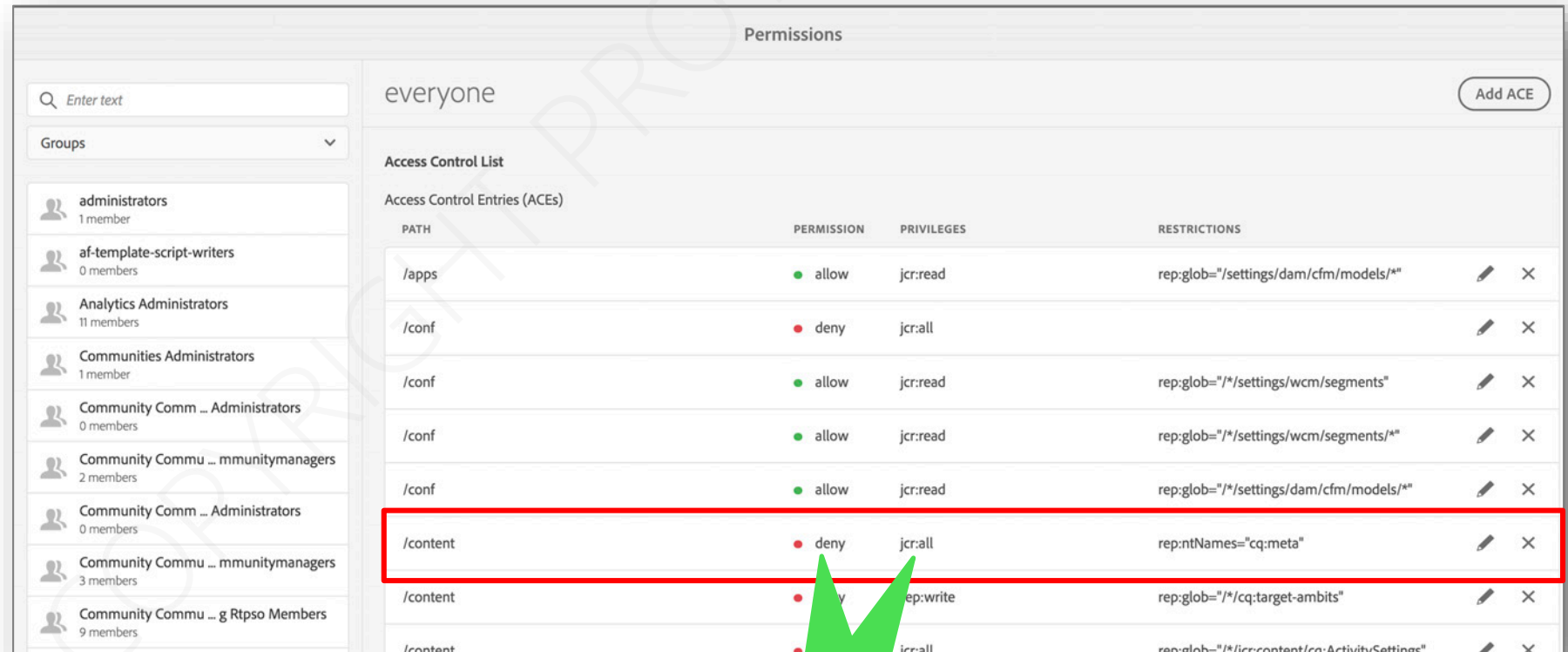
Permissions console

Accessed from **Tools > Security > Permissions**



Permissions console

- The tree in the left pane lists all the users and groups currently available in the system.
- The console includes all the paths in the repository.



Permissions

everyone Add ACE

Access Control List

Access Control Entries (ACEs)

PATH	PERMISSION	PRIVILEGES	RESTRICTIONS		
/apps	allow	jcr:read	rep:glob="/settings/dam/cfm/models/*"		
/conf	deny	jcr:all			
/conf	allow	jcr:read	rep:glob="/*/settings/wcm/segments"		
/conf	allow	jcr:read	rep:glob="/*/settings/wcm/segments/*"		
/conf	allow	jcr:read	rep:glob="/*/settings/dam/cfm/models/*"		
/content	deny	jcr:all	rep:ntNames="cq:meta"		
/content	deny	rep:write	rep:glob="/*cq:target-ambits"		
/content	deny	jcr:all	rep:glob="/*jcr:content/cq:Activity/Settings"		

JCR actions and permissions

Authorizables

Authorizables can be managed from:

1. Adobe IMS

- Manually from the Admin Console
- Externally from an external IDP

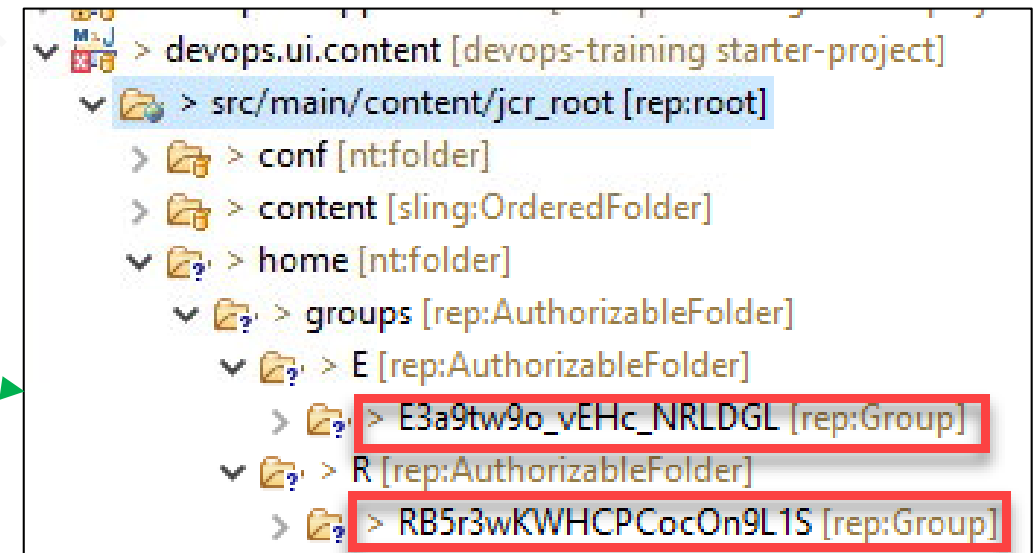
2. Manually in the AEM Service (not synced back to Adobe IMS)

- Via code for repeatability
 - Can be useful for organizational groups and auto-membership of users
- Via UI for a single service
 - Can be useful for testing ACEs and groups

Import permissions into the Maven project

- Update `devops.ui.content > src/main/content/META-INF/vault > filter.xml` with the users, groups, and permissions as needed:

```
<?xml version="1.0" encoding="UTF-8"?>
<workspaceFilter version="1.0">
  <filter root="/conf/training" mode="merge"/>
  <filter root="/content/training" mode="merge"/>
  <filter root="/content/dam/training" mode="merge"/>
  <filter root="/content/experience-fragments/training" mode="merge"/>
  <filter root="/conf/ACME" mode="merge"/>
  <!-- Site Authors group-->
  <filter root="/home/groups/R" mode="merge"/>
  <!-- Site Managers group-->
  <filter root="/home/groups/E" mode="merge"/>
</workspaceFilter>
```



- Right-click `devops.ui.content > src/main/content/jcr_root` and select **Import from server**.

Users, Groups, and Permissions: Best Practices

- Avoid assigning access rights on a user-by-user basis.
- Use Allow statements wherever possible.
- Use the AEM UI to edit permissions and not CRXDE Lite.
- Invest time when configuring users and groups on a new installation.
- Use a test installation to verify and understand the relationships between various users and groups.
- Update the default users and groups immediately after installation to help prevent any security issues



Key takeaways

- Users:
 - The user account stores the user details needed for accessing Adobe Experience Manager. Each user account is unique and stores the basic account details and privileges assigned.
- Groups
 - Collections of users and/or other groups. Their primary purpose is to simplify the maintenance process by reducing the number of entities to be updated.
- ACLs
 - ACLs are made up of individual permissions and are used to determine the order in which these permissions are applied.