Access Control for Folders using IAM

Google Cloud Platform offers Identity and Access Management (IAM), which lets you give more granular access to specific Google Cloud Platform resources and prevents unwanted access to other resources. IAM lets you adopt the security principle of least privilege (https://en.wikipedia.org/wiki/Principle_of_least_privilege), so you grant only the necessary access to your resources.

IAM lets you control **who (users)** has **what access (roles)** to **which resources** by setting IAM policies. IAM policies grant specific role(s) to a user giving the user certain permissions.

This page explains the Identity and Access Management (IAM) (https://cloud.google.com/iam/) roles that are available at the Folders level, and how to create and manage IAM policies for folders using the Resource Manager API. For a detailed description of Cloud IAM, read the IAM documentation (https://cloud.google.com/iam/docs/overview). In particular, see Granting, Changing, and Revoking Access (https://cloud.google.com/iam/docs/granting-changing-revoking-access).

Overview of IAM roles for Folders

To help you configure your IAM roles, the following table lists:

- The type of actions you want to enable
- The roles required to perform those actions
- The resource level on which you need to apply those roles

Note: IAM policies are additive and inherited down the hierarchy. This means that moving a resource into a folder could grant additional permissions to that resource. Conversely, moving a resource out of a folder might mean losing permissions to that resource.

Type of actions	Roles required	Resource level

Type of actions	Roles required	Resource level
Administer folders across the Organization	Folder Admin	Organization
Administer a folder and all projects and folders it contains	Folder Admin	Specific folder
Access and administer a folder's IAM policies	Folder IAM Admin	Specific folder
Create new folders	Folder Creator	Parent resource for the location of the new folders
Move folders and projects	Folder Mover	Parent resource for both the original folder location and the new folder location
Move a project to a new folder	Project Editor **or** Project Owner	Parent resource for both the original project location and new project location
Delete a folder	Folder Editor or Folder Admin	Specific folder

Best practices for using IAM roles and permissions with Folders

When assigning IAM roles and permissions for use with Folders, keep the following in mind:

- Use groups whenever possible to manage members.
- Minimize usage of primitive roles, such as owner, editor, and viewer. Instead, try to use the predefined roles for principle of least privilege.
- For folder-wide management, assign permissions at folder level and have projects inherit them automatically. For example, you could assign a department administrator group the **Folder Admin** role on the folder. Network administrators that need to have department-wide permissions can have the **Network Admin** role for the folder.
- Carefully consider what permissions might change before moving a resource out of a folder. Otherwise, you could risk breaking existing apps or workflows that require those permissions on that resource.
- Plan and test your resource hierarchy carefully before moving production projects under folders. One way to do this is to create a test folder under your Organization resource and creating a prototype of your intended hierarchy ahead of time.

Understanding folder roles and permissions

Default roles

When you create a folder, you are granted the **Folder Admin** and **Folder Editor** roles for the folder to provide you full control as the creator. See below for the permissions that these roles provide. These default roles can be changed as normal in an Cloud IAM policy.

Folder Admin role

The **Folder Admin** role has all available folder permissions.

Grants Permissions:

`orgpolicy.policy.get`	permission to get organization policies on a resource
`resourcemanager.folders.get`	permission to get a folder or descendant folders
`resourcemanager.folders.create`	permission to create a folder
`resourcemanager.folders.list`	permission to list folders below a resource
`resourcemanager.folders.move`	permission to move folders out of or into a resource
`resourcemanager.folders.update`	permission to update a folder's name
`resourcemanager.folders.delete`	permission to delete a folder
`resourcemanager.folders.undelete`	permission to undelete a folder
`resourcemanager.folders.getIamPolicy`	permission to get a folder's IAM policy
`resourcemanager.folders.setIamPolicy`	permission to set a folder's IAM policy
`resourcemanager.projects.get`	permission to get a project
`resourcemanager.projects.list`	permission to list projects below a resource
`resourcemanager.projects.move`	permission to move projects out of or into a resource
`resourcemanager.projects.getIamPolicy`	permission to get a project's IAM policy
`resourcemanager.projects.setIamPolicy`	permission to set a projects's IAM policy

Folder IAM Admin role

The **Folder IAM Admin** role allows users to administer IAM policies on folders.

Grants Permissions:

`resourcemanager.folders.get`	permission to get a folder or descendant folders
`resourcemanager.folders.getIamPolicy`	permission to get a folder's IAM policy
`resourcemanager.folders.setIamPolicy`	permission to set a folder's IAM policy

Folder Creator role

The **Folder Creator** role grants permissions needed to browse the hierarchy and create folders.

Grants Permissions:

`orgpolicy.policy.get`	permission to get organization policies on a resource
`resourcemanager.folders.get`	permission to get a folder
`resourcemanager.folders.list`	permission to list folders below a resource
`resourcemanager.folders.create`	permission to create a folder
`resourcemanager.projects.get`	permission to get a project
`resourcemanager.projects.list`	permission to list projects below a resource

Folder Editor role

The **Folder Editor** role grants permission modify folders as well as to view a folder's IAM policy.

Grants Permissions:

`orgpolicy.policy.get`	permission to get organization policies on a resource
`resourcemanager.folders.get`	permission to get a folder
`resourcemanager.folders.list`	permission to list folders below a resource
`resourcemanager.folders.update`	permission to update a folder's name

`resourcemanager.folders.delete`	permission to delete a folder
`resourcemanager.folders.undelete`	permission to undelete a folder
`resourcemanager.folders.getIamPolicy`	permission to get the IAM policy set on a folder
`resourcemanager.projects.get`	permission to get a project
`resourcemanager.projects.list`	permission to list projects below a resource

Folder Mover role

The **Folder Mover** role grants permission to move projects and folders into and out of a parent Organization or folder.

Grants Permissions:

`resourcemanager.folders.move`	permission to move folders out of or into a resource
`resourcemanager.projects.move`	permission to move projects out of or into a resource

Folder Viewer role

The **Folder Viewer** role grants permission to get a folder and list the folders and projects below a resource.

Grants Permissions:

`orgpolicy.policy.get`	permission to get organization policies on a resource
`resourcemanager.folders.get`	permission to get a folder or descendant folders
`resourcemanager.folders.list`	permission to list folders below a resource
`resourcemanager.projects.get`	permission to get a project
`resourcemanager.projects.list`	permission to list projects below a resource

Creating Custom Roles

In addition to the predefined roles described in this topic, you can also create Custom Roles (https://cloud.google.com/iam/docs/understanding-custom-roles) that are collections of permissions that you tailor to your needs. When creating a Custom Role for use with Resource Manager, be aware of the following points:

- List and get permissions, such as `resourcemanager.projects.get/list`, should always be granted as a pair.
- When your Custom Role includes the `folders.list` and `folders.get` permissions, it should also include `projects.list` and `projects.get`.
- Be aware that the `setIamPolicy` permission for organizations, folders, and projects allows the user to grant all other permissions, and so should be assigned with care.

Granting roles to enable folder browsing

List permissions enable folder browsing. The two types of list permissions that typically need to be granted are `resourcemanager.folders.list`, which allows users to list folders under a resource, and `resourcemanager.projects.list`, which allows users to browse projects under an Organization or folder. The Organization Administrator is initialized with both of these permissions. For users that have not been assigned the **Organization Administrator** role:

- `resourcemanager.folders.list` can be granted via the **Folder Viewer** and **Folder Editor** roles.
- `resourcemanager.projects.list` can be granted via the **Viewer** or **Browser** roles.

For Organization members to browse the entire Organization hierarchy, list permissions should be granted at the Organization level.

Granting roles to enable folder creation

Users that need to create folders must be granted **Folder Creator** role on a resource in the hierarchy above the level at which the folder will be created. It can be helpful to grant browsing permissions along with folder creation permissions so users can effectively navigate to where in the hierarchy the folder will be created. See the section above (#FolderRolesAndPermissions) for more information on browsing permissions.

Folder Creator does not grant a user permission to delete a folder. However, when a person creates a folder, that person is automatically granted the **Folder Editor** role. The **Folder Editor** role enables folder deletion.

Granting roles to enable folder movement

To move a folder from one parent resource to another, users must have the **Folder Mover** role on both old and new parent resources, or on a common ancestor.

Granting roles to enable project movement

To move a project into a folder, users must have the **Project Editor** or **Project Owner** roles on the project and the **Project Mover** on both the source and destination parent resources.

This is slightly different from the requirements for moving a non-org-owned project into the Organization, where users must have the **Project Editor** or **Project Owner** role on the project and the **Project Creator** role on the Organization.

Granting folder-specific roles to enable project creation

To create projects, users must have the **Project Creator** role. However, rather than granting org-wide project creation permission, it can be useful to instead restrict users to viewing and creating projects only within a given folder.

To grant folder-specific permissions:

1. Grant the user the **Organization Viewer** role at the org node level (for example, domain.com).
2. Create a new folder.
3. Add the user to IAM at the folder level and grant them the **Folder Viewer** and **Project Creator** roles.

This allows the user to create projects in their folder without granting them visibility to every project in the greater organization.

Last updated December 4, 2019.