Access Control for Organizations using IAM

Google Cloud offers Cloud Identity and Access Management (IAM), which lets you give more granular access to specific Google Cloud 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 organization level, and how to create and manage IAM policies for organizations using the Resource Manager API. For a detailed description of Cloud Identity and Access Management, 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).

Permissions and roles

With Cloud Identity and Access Management, every Google Cloud method requires that the account making the API request has appropriate permissions to access the resource. Permissions allow users to perform specific actions on Cloud resources. For example, the `resourcemanager.organizations.list` permission allows a user to list the organizations they own, while `resourcemanager.organizations.update` allows a user to update an organization's metadata.

The following table lists the permissions that the caller must have to call an organization method:

Method	Required permissions
`cloudresourcemanager.organizations.get()`	`resourcemanage`

(https://cloud.google.com/resource-manager/reference/rest/v1/organizations/get)	**`organizations.get`**.
`cloudresourcemanager.organizations.search()` (https://cloud.google.com/resource-manager/reference/rest/v1/organizations/search)	Returns all Organizations for which the user has the **`resourcemanage organizations.get`** permission.
`cloudresourcemanager.organizations.getIamPolicy()` (https://cloud.google.com/resource-manager/reference/rest/v1/organizations/getIamPolicy)	**`resourcemanage organizations.getIamPolicy`**
`cloudresourcemanager.organizations.setIamPolicy()` (https://cloud.google.com/resource-manager/reference/rest/v1/organizations/setIamPolicy)	**`resourcemanage organizations.setIamPolicy`**
`cloudresourcemanager.organizations.testIamPermissions()` (https://cloud.google.com/resource-manager/reference/rest/v1/organizations/testIamPermissions)	Does not require an permission.

You don't directly give users permissions; instead, you grant them *roles*, which have one or more permissions bundled within them.

You can grant one or more roles on the same resource.

Using predefined roles

The following table lists the roles that you can grant to access an organization's properties, the description of what the role does, and the permissions bundled within that role.

Note: In addition to the roles listed in the table below, other Cloud Platform services offer IAM roles that can be set at the organization level. For a list of all the roles that can be granted on the organization level, see Understanding Roles (https://cloud.google.com/iam/docs/understanding-roles).

Role	Description	Permissions
`roles/resourcemanager.`	Access to administer all resources belonging to the	• **`orgpolicy.policy.get`**

`organizationAdmin`	organization. By default, this role does not include privileges for billing or organization role administration.	• `resourcemanager.folders.get` • `resourcemanager.folders.getIamPolicy` • `resourcemanager.folders.list` • `resourcemanager.folders.setIamPolicy` • `resourcemanager.organizations.get` • `resourcemanager.organizations.getIamPolicy` • `resourcemanager.organizations.setIamPolicy` • `resourcemanager.organizations.update` • `resourcemanager.projectInvites.get` • `resourcemanager.projects.get` • `resourcemanager.projects.getIamPolicy` • `resourcemanager.projects.list` • `resourcemanager.projects.setIamPolicy`
`roles/resourcemanager.organizationViewer`	Access to view the organization's display name. Granting this role to a user will allow that user to see the organization in the Cloud Console without having access to view all resources in the organization.	• `resourcemanager.organizations.get`
`roles/orgpolicy.policyAdmin`	Provides access to define what restrictions an organization wants to place on the configuration of cloud resources by setting Organization Policies.	• `orgpolicy.*`
`roles/browser`	Access to browse resources in an organization	• `resourcemanager.folders.get` • `resourcemanager.folders.list` • `resourcemanager.organizations.get` • `resourcemanager.projectInvites.get` • `resourcemanager.projects.get` • `resourcemanager.projects.getIamPolicy`

- `resourcemanager.projects.list`

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.

Viewing existing access for an organization

You can view what roles a user is granted for an organization to by getting the organization-level IAM policy. You can view a policy of an organization using the Cloud Platform Console, the `gcloud` command-line tool, or the `getIamPolicy()` (https://cloud.google.com/resource-manager/reference/rest/v1beta1/organizations/getIamPolicy) method.

CONSOLE GCLOUD API PYTHON

To view granted roles at the organization level using the Google Cloud Console:

1. Go to the **Manage resources** page in the Cloud Console:

   OPEN THE MANAGE RESOURCES PAGE (HTTPS://CONSOLE.CLOUD.GOOGLE.COM/CLOUD-RESOUR

2. On the **Organization** drop-down list, select your organization.
3. Select the check box for the Organization resource.
4. On the right side **Info Panel**, under **Permissions**, click to expand a role and display all members who have that role.

Granting access to an organization

Organization Admins can grant IAM roles to team members so that they can access an organization's resources and APIs. You can grant roles to a Google Account email, a Google Group, a service account, or a G Suite domain. You can use the Google Cloud Console, the `gcloud` tool, or the `setIamPolicy()` (https://cloud.google.com/resource-manager/reference/rest/v1beta1/organizations/setIamPolicy) method to grant roles.

CONSOLE GCLOUD API PYTHON

To set access control at the organization level using the Google Cloud Console:

1. Go to the **Manage resources** page in the Cloud Console:

   OPEN THE MANAGE RESOURCES PAGE (HTTPS://CONSOLE.CLOUD.GOOGLE.COM/CLOUD-RESOUR

2. On the **Organization** drop-down list, select your organization.
3. Select the check box for the Organization resource. If you do not have a Folder resource, the Organization resource will not be visible. To continue, see the instructions for granting roles through the Cloud IAM (https://cloud.google.com/iam/docs/granting-changing-revoking-access#grant_access) page.
4. If the **Info Panel** pane on the right is hidden, click **Show Info Panel** in the top right corner.
5. In the **Info Panel** pane, in the **Permissions** tab, click **Add Member**.
6. In the **New members** field, enter the team members you want to add. You can designate a Google Account email, a Google Group, a service account, or a G Suite domain.
7. In the **Select a role** drop-down list, select the role you want to grant to the team members.
8. Click **Add**.

Testing permissions

You can test Cloud IAM permissions on a user for an organization with the `testIamPermissions()` (https://cloud.google.com/resource-manager/reference/rest/v1beta1/organizations/testIamPermissions) method. This method takes the resource URL and the set of permissions you want to test as input parameters, and returns the subset of these permissions that the user has access to.

You typically don't invoke `testIamPermission()` if you're using Cloud Platform Console directly to manage permissions. `testIamPermissions()` is intended for integration with your proprietary software such as a customized graphical user interface. For example, the Cloud Platform Console uses `testIamPermissions()` internally to determine which UI should be available to the logged-in user.

API PYTHON

You can use the `testIamPermissions()` (https://cloud.google.com/resource-manager/reference/rest/v1/organizations/testIamPermissions) method to check which of the given permissions the caller has for the given resource. This method takes a resource name and a set of permissions as parameters, and returns the subset of permissions that the caller has.

Here is some sample code to test permissions for an organization:

```
Request:

POST https://cloudresourcemanager.googleapis.com/v1/organization/12345:testIamPermi

{
    "permissions":  [
        "resourcemanager.organizations.get",
        "resourcemanager.organizations.update"
    ]
}

Response:

{
    "permissions": [
        "resourcemanager.organizations.get"
    ]
}
```

Last updated December 4, 2019.