GCP Identity and Access Management (IAM) Roles

**How GCP IAM Works**

IAM is a framework of policies and technologies to ensure that the right users have the appropriate access to technology resources. GCP IAM uses the principle of least privilege to grant an account access to the minimal set of permissions required to complete a given task—ensuring users only have access to the resources they really need.  
  
The goal of an IAM policy is to grant access for a resource to a specific authenticated identity or principal. This can come in the form of individual users or applications that need a resource to perform a task. When an authenticated identity attempts to access a resource, IAM references the attached resource policy to determine whether permission should be granted. IAM policies keep valuable digital resources safe and secure so that only specific users can access tools that control cloud functions.

**Key terms**

Defining key terms will be helpful when proceeding through IAM in greater detail in terms of roles and role assignments.

* **Principal:** Google account end user or service account application or identity that needs permission to access a Google Cloud resource.
* **Role:** A collection of permissions that determine which operations can be used on a specific resource.
* **Policy:** Role constraints that bind a principal or identity to a specific action for a resource. [Policies](http://cloud.google.com/iam/docs/policies) define which identities have what kind of access to an attached, specified resource.

**GCP IAM roles explained**

Since permissions cannot be applied directly to users, cloud network administrators must confer roles with specific policy-based permissions to each user, group, or application. [GCP roles include](http://cloud.google.com/iam/docs/understanding-roles):

* **Basic**: IAM basic roles are the most limited form of GCP roles and include owners, editors, and viewers.
* **Predefined**: Predefined roles provide finer-grain access to specific services in the Google Cloud.
* **Custom**: Custom roles provide finer-grain access to an organization-specific list of permissions to meet specific needs.

When you grant a role to one principal, you grant them all of the permissions that a role contains. GCP IAM is scalable in that multiple users in one group can all take on the permissions granted from a single role.

**Basic (primitive) roles**

Originally known as “primitive roles”, basic roles are defined as either owner, editor, or viewer. The levels within basic roles are concentric: owner roles include the permissions of editor roles, and editor roles include the permissions of viewer roles. Viewers only have permissions for read-only access and cannot modify existing resources and data; editors can both view resources as well as take actions to modify existing resources as well as create and delete resources. Owners can manage viewers and editors, permissions and resources for projects, and set up billing processes.

**Limitations**

Basic roles predate GCP IAM and do not follow the *principle of least privilege.* As such, these roles are subject to increased security risks because they include thousands of permissions across all Google Cloud services. For example, granting a user the basic editor role gives them the permission to create and delete resources for most Google Cloud services across the entire project or organization. As such, basic roles should only be granted as a last resort.  
  
Additionally, owners have stipulations depending on the infrastructural level the user is operating at. For example, owners at the project level do not have the same permissions as owners at the organization level. Furthermore, owners at the organization level cannot modify the metadata (role ID and permissions) within a role.

**Basic Roles Pros and Cons**

**Pros**

* Easiest role to set up because it grants access to all datasets.
* Owners can set up billing for projects.
* Cloud Console can grant ownership on projects to a user that is not part of the organization.

**Cons**

* Subject to high security risks.
* An owner at the organization-level cannot update role metadata.
* Basic roles do not contain permissions to perform actions across all services.

**Predefined Roles**

Predefined roles give users granular access to specific resources while enhancing security by maintaining the principle of least privilege, where users are only granted the resources necessary to complete their tasks. In contrast to basic roles, predefined roles are role-binding: lower-level resources inherit the policies and role bindings of higher-level resources. This core concentricity presents a secure approach to organization-owned resources in the cloud, where only authorized users are granted access to mission-critical datasets and information.   
  
Predefined roles are both created and maintained by Google, which means you get the benefits of automatic updates when new permissions, features, or services are added to the Google Cloud. GCP IAM provides many different types of predefined roles for services, including Access Approval roles, Access Context Manager roles, Actions roles, and many others.   
  
Before delegating roles, administrators should identify which permissions need to be granted, find and select the roles that contain those permissions, and decide on which users or applications to grant those roles.

**Limitations**

Delegating predefined roles can be challenging due to the sheer volume of roles available and the possibility of a user being assigned hundreds of roles. Additionally, if predefined roles are not defined specifically enough, they may end up violating the principle of least privilege by including unnecessary permissions—thereby eliminating the major security advantage of predefined roles. To get around this, teams can turn to an access tool that *implements role-based access control (RBAC)* to manage appropriate access to each team for the ideal GCP IAM roles hierarchy.   
  
In addition, while the principle of least privilege is important for security, determining the most appropriate role for a user can be tricky when there are multiple users and multiple predefined roles that need to be assigned correctly.

**Predefined Roles Pros and Cons**

**Pros**

* Principle of least privilege: resources are secure from unauthorized users.
* Provides users with granular access to specific GCP services.
* Automatically updated by Google with new permissions, features, and services.

**Cons**

* Difficult to delegate permissions with a large volume of roles.
* Complex roles that include unrelated and unnecessary permissions.
* Custom roles need to be made when there is no predefined role for a task.

**Custom Roles**

Sometimes users need to be granted more complex permissions that might not have a predefined role already in place—particularly useful when predefined roles are granting overly permissive permissions for your users’ needs. In these cases, role administrators can turn to custom roles to bundle multiple supported permissions to meet a specific need.   
  
A custom role must be created and granted within a specific organization or project. In support of the principle of least privilege, a custom role will only have permissions that are granted at the level at which it was created. For example, a custom role created at the project level will not have the same permissions that are only available at the organizational level, and vice versa.

**Limitations**

As custom roles are created and maintained by users, they do not offer the same automatically updating features as Google-maintained predefined roles. Additionally, custom roles cannot be applied to resources in other projects and organizations—they must be granted within the project or organization it was created in. This can create barriers for administrators when they have users who are working between projects. For example, when permissions that are only supported for organization-level custom roles are copied into a project-level custom role, they will display either a warning or error (e.g., not applicable for project-level custom roles, INVALID\_ARGUMENT, or an HTTP 400 error code).

**Custom Roles Pros and Cons**

**Pros**

* Present a solution to organization-specific role needs.
* Maintains security with the principle of least privilege.

**Cons**

* Can be complicated to set up and result in errors.
* Limited to resources within the project or organization the role was created in.
* Does not automatically update.

**Using the right GCP IAM roles to keep your infrastructure secure**

Basic, predefined, and custom GCP IAM roles can help organizations delegate permissions and secure their data. Basic roles are fast and easy to set up, but offer less security than other role types. Predefined roles are built on the principle of least privilege, and therefore offer more security than basic roles. Custom roles further fine-tune the permissions granted with predefined roles, while maintaining the security of a role-type built on the principle of least privilege.

**Conclusion**

Delegating GCP IAM roles across an organization provides network administrators and organizers with a framework for creating an efficient and secure developer infrastructure. Breaking roles out into basic vs. predefined vs. custom and defining the limitations of each can be a powerful way of setting up a secure hierarchy across projects and organizations.