Policy

A **policy** is a document that formally states one or more permissions.

### JSON Policy Document Structure

* **Version** - Specify the version of the policy language that you want to use (as a best practice, use the latest 2012-10-17 version)
* **Statement** — container which includes information about a single permissions (consists from next elements)
* **Sid** (optional) — id
* **Effect —** Use Allowor Denyto indicate whether the policy allows or denies access
* **Principal** (optional) - Account, user, role, or federated user to which you would like to allow or deny access
* **Action** - Include a list of actions that the policy allows or denies
* **Resource** (optional) - A list of resources to which the actions apply
* **Condition** (optional) - Specify the circumstances under which the policy grants permission





<https://docs.aws.amazon.com/IAM/latest/UserGuide/access_policies.html>

Examples

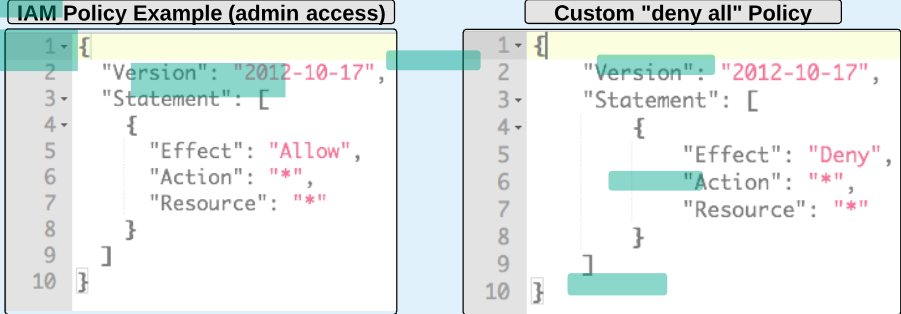
<https://docs.aws.amazon.com/IAM/latest/UserGuide/access_policies_examples.html>

## 

## Allow and Deny

By default, an **explicit deny** always overrides an **explicit allow**.

This allows for the use of a «deny all» policy to quickly restrict all access a user may have through multiple attached policies.



Templates

IAM provides pre-built policy templates to assign to users and groups:

* **Administrator access —** Full access to all AWS resources
* **Power user access** — Like «Administrator access», but it does not allow user/group managment
* **Read-only access —** Only view AWS resources (user can view what is in an S3 bucket)

You can also create custom IAM permission policies using the policy generator or written from scratch.

## Identity-based policies

**Identity-based policies** are permissions policies that you attach to an IAM identity (such as an IAM user, group, or role).

* **Managed policies -** policies that you can attach to multiple users, groups, and roles in your AWS account
* **AWS managed policies** – Managed policies that are created and managed by AWS
* **Customer managed policies** – Managed policies that you create and manage in your AWS account.
* **Inline policies** – Policies that you create and manage and that are embedded directly into a single user, group, or role.

## Resource-based policies

Resource-based policies are permissions policies that you attach to a resource (control what actions a specified principal can perform on that resource and under what conditions).

Policies cannot be directly attached to AWS resources (such as an EC2 instance).

## Policies and Accounts

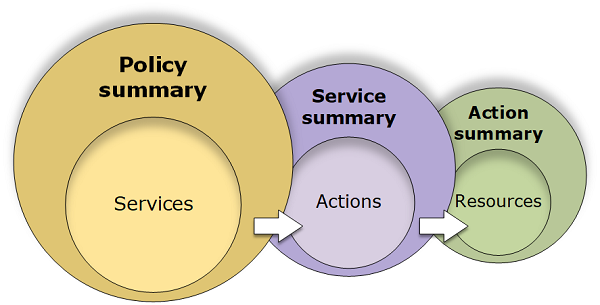
If one manages a single account in AWS, then you define the permissions within that account using policies.

If one manages multiple accounts, it is more difficult to manage permissions for your users. You can use IAM roles, resource-based policies, or access control lists (ACLs) for cross-account permissions. It is recommended to use AWS Organizations.

## Policies and Users

When you create an IAM user, they can't access anything in your account until you give them permission. More than one policy can be attached to a user.

The IAM console includes **summary**tables that describe the access level:

1. **policy summary** - includes a list of services
2. **service summary** - includes a list of the actions and associated permissions for the chosen service
3. **action summary** - includes a list of resources and conditions for the chosen action  
   

<https://docs.aws.amazon.com/IAM/latest/UserGuide/images/policy_summaries-diagram.png>

## Policies and Groups

You can organize IAM users into IAM **groups** and attach a policy to a group. More than one policy can be attached to a group.

Users or groups can have multiple policies attached to them that grant different permissions. In that case, the users' permissions are calculated based on the combination of policies.

## Federated Users and Roles

To assign permissions to federated users, you can create an entity referred to as a role and define permissions for the role. When a federated user signs in to AWS, the user is associated with the role and is granted the permissions that are defined in the role.

<https://docs.aws.amazon.com/IAM/latest/UserGuide/introduction_access-management.html>