Service Control Policies (SCPs)

1 Introduction

1. SCPs are rules that control what you can and cannot do in your AWS accounts
2. They work at the level of your whole organization or specific groups within it
3. SCPs can't give you new permissions. Instead, they can only limit what's allowed.
4. You set these rules from one central place, and they apply to all users and roles in the accounts they cover.
5. A user without any IAM permission policies has no access, even if the applicable SCPs allow all services and all actions.
6. Exclusivity to AWS Organizations
   1. SCPs are exclusive to AWS Organizations and can only be used when you've enabled an organization in your AWS account
   2. They are part of the service control features offered by AWS Organizations.

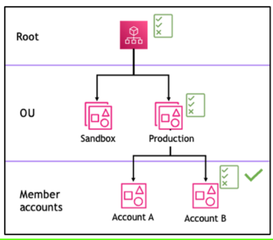
2 Key Points

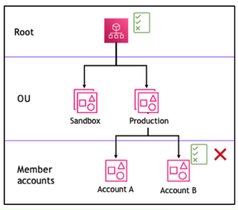
1. Layered Architecture
   1. SCPs can be applied at different levels within an organization
      1. The root level
      2. Organizational units (OUs)
      3. Individual accounts
   2. Policies applied at higher levels affect all entities within their scope.
2. Scope of Application
   1. An SCP restricts permissions for IAM users and roles in member accounts, including the member account's root user.
   2. SCPs affect only member accounts in the organization. They have no effect on users or roles in the management account.
3. Deny by Default
   1. SCPs are deny-based. Pimary function is to deny or restrict permissions across accounts in an organization.
   2. If a permission is blocked at any level above the account, either implicitly or explicitly a user or role in the affected account can't use that permission
4. Evaluation Logic
   1. When determining whether an action is allowed, AWS evaluates SCPs in conjunction with IAM policies.
   2. An action must be allowed by IAM policies and not explicitly denied by SCPs to proceed.

3 Default SCP

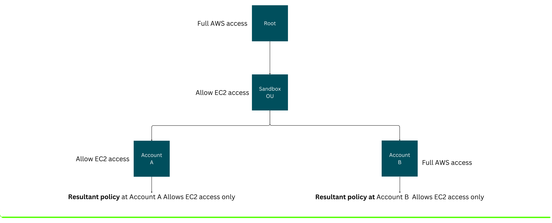
1. Automatically attached to all entities in an AWS Organization, allowing unrestricted access to AWS services.
2. Permits all actions (*) on all resources (*), ensuring no SCP-imposed restrictions at the outset.
3. Serves as an upper limit on permissions; actual access is defined by IAM policies.

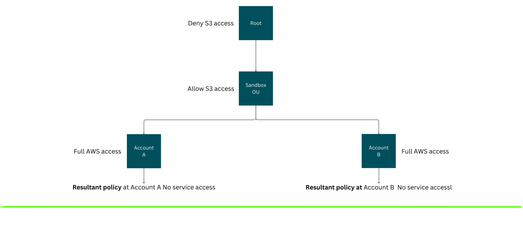
4 Examples

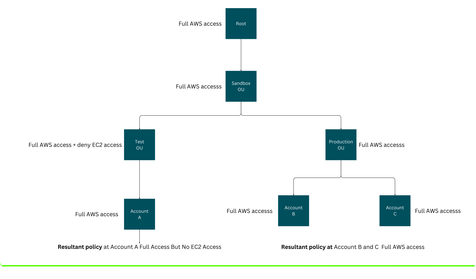
1. How SCPs work with Allow
   1. 

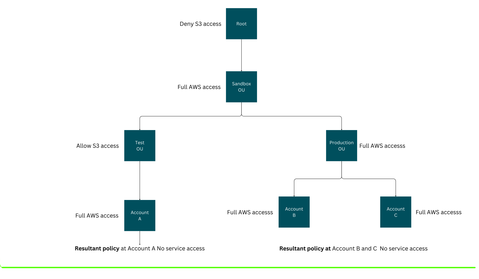
   2. 

2. How SCPs work with Deny
   1. 

   2. 

   3. 

   4. 

   5. 

   6. 

5 Example Of SCP