Identity & Access Management

Agenda

- What is AWS IAM
- Features
- Identities
- User Types
- User Sign-in to Account
- Switch Role
- Role to EC2 Instance
- Password Policy
- How to Access AWS
- Multi-Factor Authentication (MFA)
- Permissions
- Permission Types
- Policies Structure
- User Based Policies
- Resource Based Policies
- Resource Based Permission
- Policies Types
- Request Flow
- Limitations
- Quiz
- Hands-On

What is IAM

- AWS Identity and Access Management (IAM) is a web service that helps you securely control access to AWS resources for your users.

- You can use IAM to control who can use your AWS resources (authentication) and what resources they can use and in what ways (authorization).

IAM Features

- Shared access to AWS account
- Granular permissions
- Secure access to AWS resources
- Multi-factor authentication (MFA)
- Identity federation
- Identity information for assurance
- Eventually Consistent
- Free to Use

AWS IAM: Identities

User

Group

Roles

AWS IAM: User Types

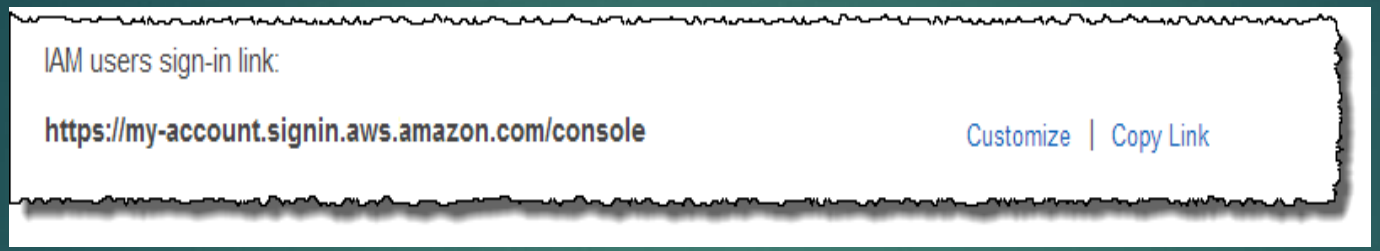
- Root User
  - When you create an AWS account, you create an AWS account root user identity—that is, the email.
  - When you use your root user credentials, you have complete, unrestricted access to all resources in your AWS account.

- IAM User
  - IAM users are not separate accounts; they are users within your account.
  - Each user can have its own password for access to the AWS Management Console.
  - An IAM user doesn't have to represent an actual person; you can create an IAM user in order to generate an access key for an application.

- Federating Existing Users
  - If your users already have a way to be authenticated—for example, by signing in to your corporate network—you can federate those user identities into AWS.

AWS IAM: How IAM User Sign-in to Account

- By default, the sign-in URL for your account includes your account ID.



- You can create a unique sign-in URL for your account so that the URL includes a name instead of an account ID

- The sign-in endpoint follows this pattern:
  - https://alias.signin.aws.amazon.com/console

- IAM users in your account have access only to the authorised AWS resources

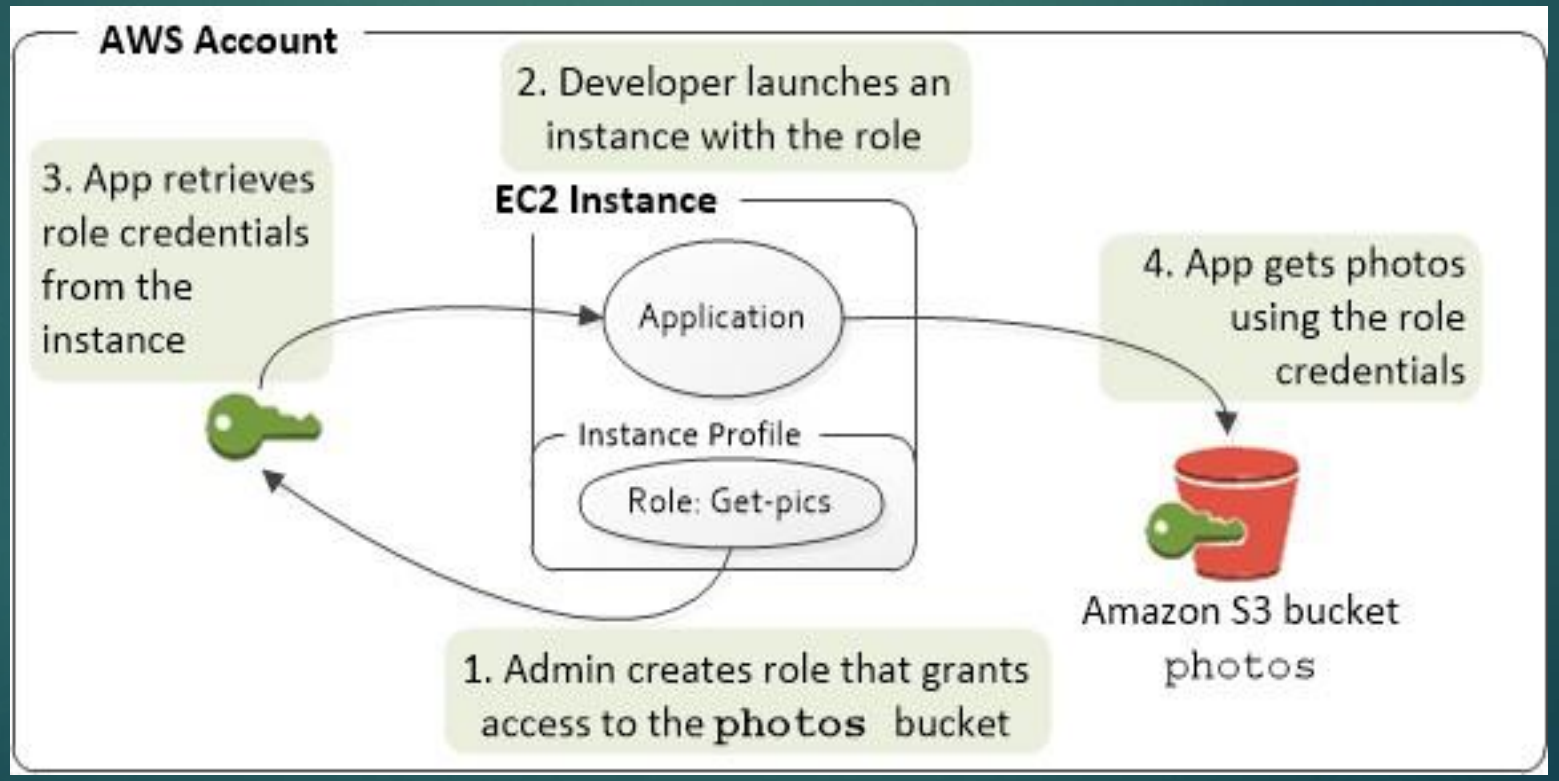
AWS IAM: Switch Role

- To allow users from one AWS account to access resources in another AWS account, create a role.
- If user needs to work with in the Production environment in the AWS Management Console, he can do so by using Switch Role.
- He specifies the account ID or alias and the role name, and his permissions immediately switch to those permitted by the role.
- He can then use the console to work with the production.
- While user is using the role, he also cannot make use of his power-user privileges in the Development account.

AWS IAM: IAM Role to EC2 Instance

- Use an IAM role to manage *temporary credentials* for applications that run on an EC2 instance

- The role supplies temporary permissions that applications can use when they make calls to other AWS resources



AWS IAM: Password Policy

- Minimum password length
- Require at least one uppercase letter
- Require at least one lowercase letter
- Require at least one nonalphanumeric character
- Allow users to change their own password
- Enable password expiration
- Prevent password reuse
- Password expiration requires administrator reset

AWS IAM: How to Access AWS

Console Password

Access Keys

AWS IAM: Multi-Factor Authentication (MFA)

Security Token based

MFA device (hardware or virtual)

Six-digit numeric code

SMS Text Message based

Mobile device

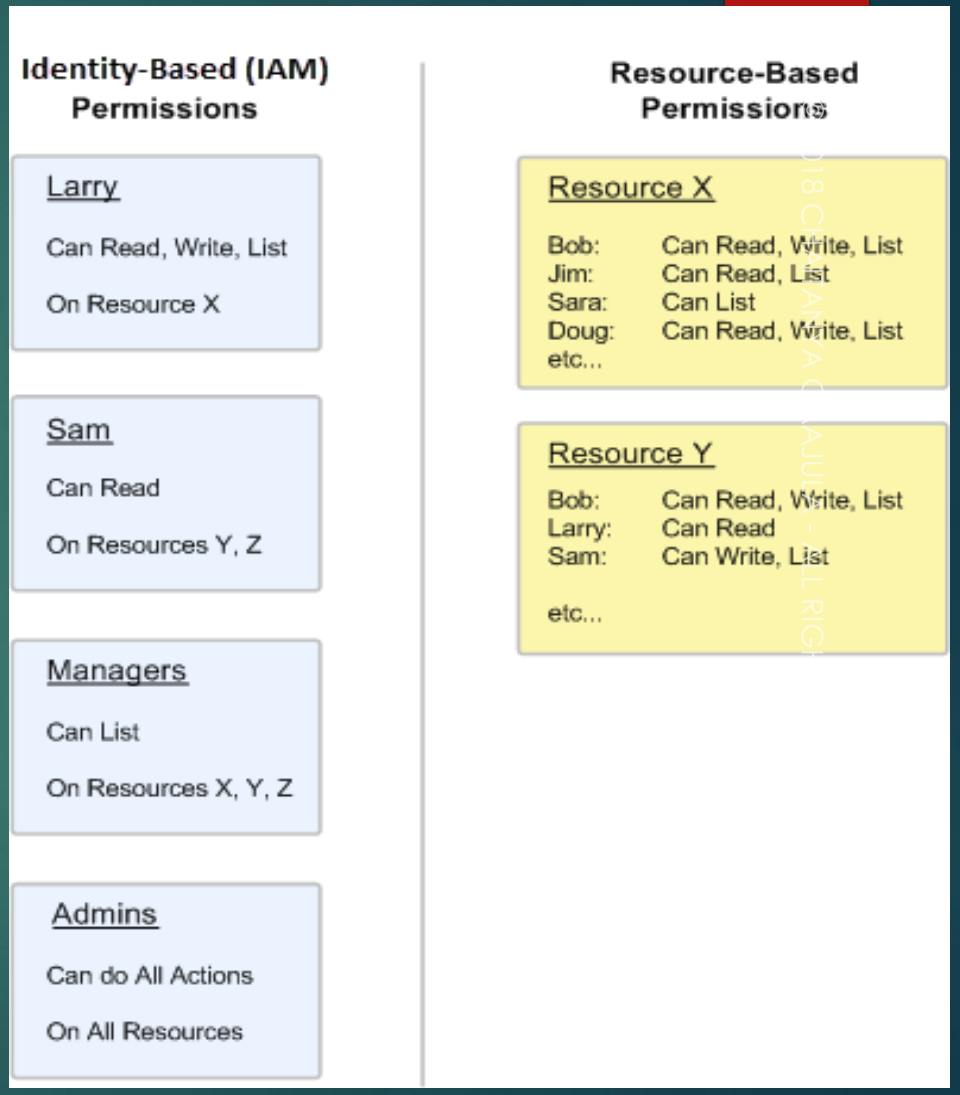
Six-digit numeric code

AWS IAM: Permissions

- Permissions let you specify who has access to AWS resources, and what actions they can perform on those resources.
- Permissions are granted through policies that are created and then attached to users, groups, or roles.
- By default, IAM users can't access anything in your account. You grant permissions to a user by creating a policy.
- Users in your account have multiple policies that together represent the permissions for that user.
- When you give permissions to a group, all users in that group get those permissions.
- To assign permissions to federated users, you can create an entity referred to as a role and define permissions for the role.

AWS IAM: Permission Types

- Resource-based permissions and resource-level permissions.
  - Resource-based permissions are permissions you can attach directly to a resource
  - Resource-level permissions refers to what actions users can perform, and what they allowed to perform



AWS IAM: Policies Structure

- Each policy is a JSON document.
- A policy is a document that formally states one or more permissions.
- The policy document includes the following elements:
  - Effect
  - Action
  - Resource
  - Condition (Optional)

```
{
"Version": "2012-10-17",
"Statement": [
  {
      "Sid": "FirstStatement",
      "Effect": "Allow",
      "Action": ["iam:ChangePassword"],
      "Resource": "*"
  },
  {
      "Sid": "SecondStatement",
      "Effect": "Allow",
      "Action": "s3:ListAllMyBuckets",
      "Resource": "*"
  },
  {
       "Sid": "ThirdStatement",
       "Effect": "Allow",
       "Action": [
                  "s3:List*",
                  "s3:Get*"
       ],
        "Resource": [
             "arn:aws:s3:::confidential-data",
             "arn:aws:s3:::confidential-data/*"
       ],
       "Condition": {"Bool": {"aws:MultiFactorAuthPresent": "true"}}
  }
 ]
}
```

AWS IAM: User Based Policies

- Attach this policy to an IAM user or group.

- User or group is allowed to perform only this one action (ListBucket) on one Amazon S3 bucket (example_bucket).

```
{
   "Version": "2012-10-17",
     "Statement": {
     "Effect": "Allow",
     "Action": "s3:ListBucket",
     "Resource": "arn:aws:s3:::example_bucket"
   }
}
```

AWS IAM: Resource Based Policy

- A resource-based policy contains slightly different information than a user-based policy.
- In a resource-based policy you specify what actions are permitted and what resource is affected (just like a user-based policy).
- However, you also explicitly list who is allowed access to the resource. (In a user-based policy, the "who" is established by whomever the policy is attached to.)
- Resource-based policies include a Principal element that specifies who is granted the permissions.

```
{
        "Version": "2012-10-17",
        "Statement": {
        "Effect": "Allow",
        "Principal": {"AWS": "arn:aws:iam::777788889999:user/bob"},
        "Action": [
                "s3:PutObject", “
                  s3:PutObjectAcl"
        ],
        "Resource": "arn:aws:s3:::example-bucket/*"
        }
}
```

AWS IAM: Resource Based Permission

- For resource-based permissions, attach a policy to the resource, such as an S3 Bucket.
- Include, who is allowed to access the resource, known as the Principal.
- This Policy attached to an Amazon S3 bucket and grants permission to a specific AWS account to perform any Amazon S3 actions in mybucket.
- This includes both working with the bucket and with the objects in it.

```
{
  "Version": "2012-10-17",
  "Id": "S3-Account-Permissions",
  "Statement": [{
    "Sid": "1",
    "Effect": "Allow",
    "Principal": {"AWS": ["arn:aws:iam::ACCOUNT-ID-WITHOUT-HYPHENS:root"]},
    "Action": "s3:*",
    "Resource": [
    "arn:aws:s3:::mybucket",
    "arn:aws:s3:::mybucket/*"
   ]
  }]
}
```

AWS IAM: Policies Types

Managed Policies

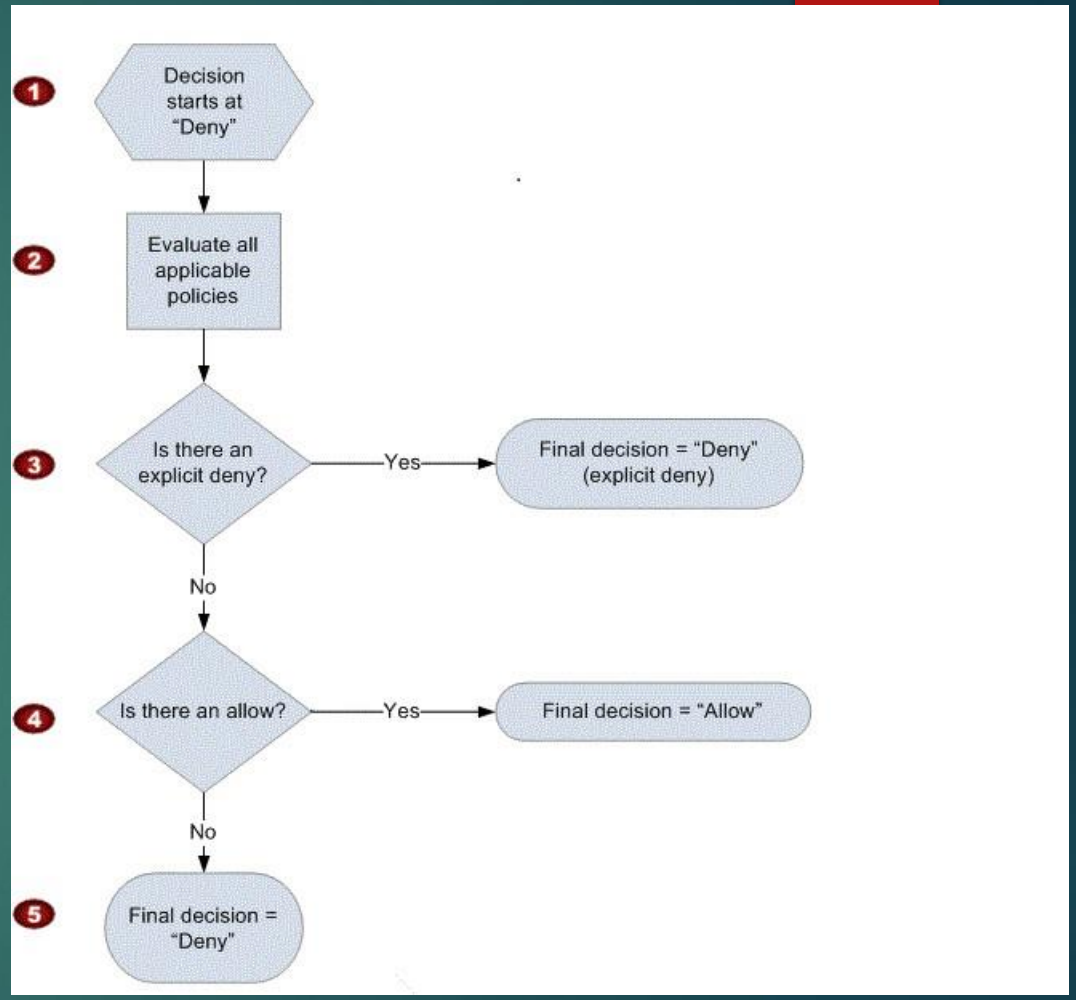
- AWS Managed Policies
- Customer Managed Policies

In-line Policies

AWS IAM: Request Flow

- When a request is made, the AWS service decides whether a given request should be allowed or denied.

- The evaluation logic follows these rules:
  - By default, all requests are denied.
  - An explicit allow overrides this default.
  - An explicit deny overrides any allows.



AWS IAM: Limitation

Resource	Default Limit
Users in an AWS account	5000
Groups in an AWS account	1000
Roles in an AWS account	300
MFA devices in use by an IAM user	1
Access keys assigned to an IAM user	2
Aliases for an AWS account	1
Groups an IAM user can be a member of	10
Managed policies attached to an IAM user	10
Managed policies attached to an IAM group	10
Managed policies attached to an IAM role	10

Hands-On Lab

Thank You