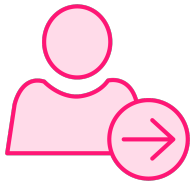
AWS Identity and Access Management (IAM): IAM Users and Groups

Andru Estes
Principal Author
andru-estes

AWS IAM Users and Groups

IAM Users and Groups



IAM User: Entity within IAM meant to represent a human or a dedicated service account.

IAM Group: Collections of IAM users to simplify permission management. Users can belong to multiple IAM Groups at one time. No nesting groups!

Long-term Credentials: IAM Users authenticate via Username and Password, or by using static IAM Access Keys

You assign permissions to IAM Users via Permissions Policies!

We will take a look at those next...

IAM Policies

Policy

A policy is an object in AWS that, when associated with an identity or resource, defines their permissions.

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

Policy Types

Identity-based	Resource-based	Permissions Boundaries
Organization Service Control Policies	Access Control Lists	Session Policies

Policy Types

We are going to focus on these for now!

Identity-based

Resource-based

Identity-based Policies

Attached to IAM identities to grant permissions

Written and stored as JSON documents

By default, permissions are implicitly denied!

Two forms: Managed and Inline

Managed Policies

Attachable, standlone policies that are reusable

Given a resource ARN after creation

AWS Managed Policies

Created and managed by AWS. Usable by everyone.

Customer Managed Policies

You create, manage, and reuse however you want.

Inline Policies

Added directly to a single IAM Identity for specific use cases

One-to-one relationship

Not reusable

Deleting the IAM Identity results in deletion of the Inline Policy as well

Resource-based Policies

JSON policy documents that get attached to AWS resources

Examples: Bucket Policy, KMS Key Policy

Grant permissions to a specified IAM principal for the resource

All resource-based policies are Inline policies

Grant cross-account access by specifying another account as the principal*

** Only half the process/trust relationship. The other account must also have an identity-based policy granting permissions in place.*

Exploring an IAM Policy

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AllowListAndGetObjects",
            "Effect": "Allow",
            "Principal": {
                "AWS": “arn:aws:iam::111111111111:user/cool_user”
            },
            "Action": [
                "s3:ListBucket",
                "s3:GetObject"
            ],
            "Resource": [
                "arn:aws:s3:::your-bucket-name",
                "arn:aws:s3:::your-bucket-name/*"
            ],
            "Condition": {
                "Bool": {
                    "aws:MultiFactorAuthPresent": "true"
                }
            }
        }
    ]
}
```

```
{
   "Version": "2012-10-17",
   "Statement": [
      {
         "Sid": "AllowListAndGetObjects",
         "Effect": "Allow",
         "Principal": {
            "AWS": “arn:aws:iam::111111111111:user/cool_user”
         },
         "Action": [
            "s3:ListBucket",
            "s3:GetObject"
         ],
         "Resource": [
            "arn:aws:s3:::your-bucket-name",
            "arn:aws:s3:::your-bucket-name/*"
         ],
         "Condition": {
            "Bool": {
               "aws:MultiFactorAuthPresent": "true"
            }
         }
      }
   ]
}
```

Version

REQUIRED

The version of the policy language

Currently always “2012-10-17”

```
{
   "Version": "2012-10-17",
   "Statement": [
      {
         "Sid": "AllowListAndGetObjects",
         "Effect": "Allow",
         "Principal": {
            "AWS": “arn:aws:iam::111111111111:user/cool_user”
         },
         "Action": [
            "s3:ListBucket",
            "s3:GetObject"
         ],
         "Resource": [
            "arn:aws:s3:::your-bucket-name",
            "arn:aws:s3:::your-bucket-name/*"
         ],
         "Condition": {
            "Bool": {
               "aws:MultiFactorAuthPresent": "true"
            }
         }
      }
   ]
}
```

Statement

REQUIRED

List containing permission statements

```
{
   "Version": "2012-10-17",
   "Statement": [
      {
         "Sid": "AllowListAndGetObjects",
         "Effect": "Allow",
         "Principal": {
            "AWS": “arn:aws:iam::111111111111:user/cool_user”
         },
         "Action": [
            "s3:ListBucket",
            "s3:GetObject"
         ],
         "Resource": [
            "arn:aws:s3:::your-bucket-name",
            "arn:aws:s3:::your-bucket-name/*"
         ],
         "Condition": {
            "Bool": {
               "aws:MultiFactorAuthPresent": "true"
            }
         }
      }
   ]
}
```

Sid

OPTIONAL

An identifier for your statements in the policy

```
{
   "Version": "2012-10-17",
   "Statement": [
      {
         "Sid": "AllowListAndGetObjects",
         "Effect": "Allow",
         "Principal": {
            "AWS": “arn:aws:iam::111111111111:user/cool_user”
         },
         "Action": [
            "s3:ListBucket",
            "s3:GetObject"
         ],
         "Resource": [
            "arn:aws:s3:::your-bucket-name",
            "arn:aws:s3:::your-bucket-name/*"
         ],
         "Condition": {
            "Bool": {
               "aws:MultiFactorAuthPresent": "true"
            }
         }
      }
   ]
}
```

Effect

REQUIRED

Are you Allowing or Denying access

```
{
   "Version": "2012-10-17",
   "Statement": [
      {
         "Sid": "AllowListAndGetObjects",
         "Effect": "Allow",
         "Principal": {
            "AWS": “arn:aws:iam::111111111111:user/cool_user”
         },
         "Action": [
            "s3:ListBucket",
            "s3:GetObject"
         ],
         "Resource": [
            "arn:aws:s3:::your-bucket-name",
            "arn:aws:s3:::your-bucket-name/*"
         ],
         "Condition": {
            "Bool": {
               "aws:MultiFactorAuthPresent": "true"
            }
         }
      }
   ]
}
```

Principal

DEPENDENT ON POLICY TYPE

The IAM Identity that you are apply this policy to (*granting or denying permissions*)

Uses the ARN

```
{
   "Version": "2012-10-17",
   "Statement": [
      {
         "Sid": "AllowListAndGetObjects",
         "Effect": "Allow",
         "Principal": {
            "AWS": “arn:aws:iam::111111111111:user/cool_user”
         },
         "Action": [
            "s3:ListBucket",
            "s3:GetObject"
         ],
         "Resource": [
            "arn:aws:s3:::your-bucket-name",
            "arn:aws:s3:::your-bucket-name/*"
         ],
         "Condition": {
            "Bool": {
               "aws:MultiFactorAuthPresent": "true"
            }
         }
      }
   ]
}
```

Action

DEPENDENT ON POLICY TYPE

Which API calls (*actions*) are you allowing or denying

Dependent on your Resources

```
{
   "Version": "2012-10-17",
   "Statement": [
      {
         "Sid": "AllowListAndGetObjects",
         "Effect": "Allow",
         "Principal": {
            "AWS": “arn:aws:iam::111111111111:user/cool_user”
         },
         "Action": [
            "s3:ListBucket",
            "s3:GetObject"
         ],
         "Resource": [
            "arn:aws:s3:::your-bucket-name",
            "arn:aws:s3:::your-bucket-name/*"
         ],
         "Condition": {
            "Bool": {
               "aws:MultiFactorAuthPresent": "true"
            }
         }
      }
   ]
}
```

Resource

DEPENDENT ON POLICY TYPE

The list of AWS resources that you are applying the respective actions to

Directly impacts your Actions list

```
{
   "Version": "2012-10-17",
   "Statement": [
      {
         "Sid": "AllowListAndGetObjects",
         "Effect": "Allow",
         "Principal": {
            "AWS": “arn:aws:iam::111111111111:user/cool_user”
         },
         "Action": [
            "s3:ListBucket",
            "s3:GetObject"
         ],
         "Resource": [
            "arn:aws:s3:::your-bucket-name",
            "arn:aws:s3:::your-bucket-name/*"
         ],
         "Condition": {
            "Bool": {
               "aws:MultiFactorAuthPresent": "true"
            }
         }
      }
   ]
}
```

Condition

OPTIONAL

Special conditions that you can set to further customize policies

Common examples: Require MFA, External ID, Source IP Ranges

Understanding AWS IAM Access Keys

Access Keys Overview

Long-term credentials for IAM users and the Root user account

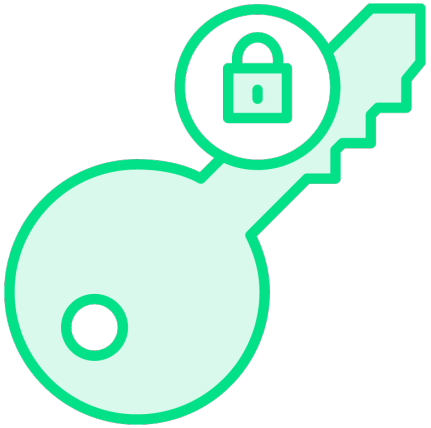
Used to sign programmatic requests via CLI or an AWS SDK

Composed of Access Key ID and the Secret Access Key

Secret Access Key is only viewable at time of creation

Access Key Pairs

Access Key ID
Similar to a username



Secret Access Key
Like a password for your Access Key ID

You must use both the Access Key ID and the Secret Access Key together to make any authenticated calls to AWS.

Protect these like you would any other set of credentials!

AWS Access Keys Example

Access Key ID: AKIAEXAMPLE12345

Secret Access Key: wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY

AWS IAM Credential Reports

Report that you generate and download that lists:

- All users in the account
- Status of passwords
- Status of access keys
- MFA status

Useful for auditing and compliance

Can generate a new one every four hours

Module Summary and Exam Tips

It is generally best practice to leverage IAM Groups to pass permissions to multiple users

IAM Policy Types

IAM policies are JSON documents that dictate the permissions allowed

Remember the difference between identity-based and resource-based policies

Know use cases for both managed policies and inline policies

```
{
   "Version": "2012-10-17",
   "Statement": [
      {
         "Sid": "AllowListAndGetObjects",
         "Effect": "Allow",
         "Principal": {
            "AWS": “arn:aws:iam::111111111111:user/cool_user”
         },
         "Action": [
            "s3:ListBucket",
            "s3:GetObject"
         ],
         "Resource": [
            "arn:aws:s3:::your-bucket-name",
            "arn:aws:s3:::your-bucket-name/*"
         ],
         "Condition": {
            "Bool": {
               "aws:MultiFactorAuthPresent": "true"
            }
         }
      }
   ]
}
```

Understand how to read a policy document!

```
{
   "Version": "2012-10-17",
   "Statement": [
      {
         "Sid": "AllowListAndGetObjects",
         "Effect": "Allow",
         "Principal": {
            "AWS": “arn:aws:iam::111111111111:user/cool_user”
         },
         "Action": [
            "s3:ListBucket",
            "s3:GetObject"
         ],
         "Resource": [
            "arn:aws:s3:::your-bucket-name",
            "arn:aws:s3:::your-bucket-name/*"
         ],
         "Condition": {
            "Bool": {
               "aws:MultiFactorAuthPresent": "true"
            }
         }
      }
   ]
}
```

Know what each sections does!

Effect: Are we allowing or denying?

Principal: Who/what are we defining permissions for?

Action: What API calls are we allowing or denying?

Resource: What AWS resource ARNS are we allowing or denying actions on?

Condition: What kind of special conditions do we want to be met for this policy’s actions to go into effect?

IAM Users do not have any permissions when they are first created.

This is known as having an implicit deny for any actions!

Remember This Order

Explicit Deny

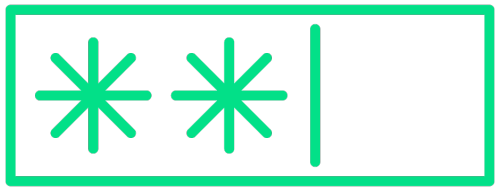
Explicit Allow

Implicit Deny (*default*)

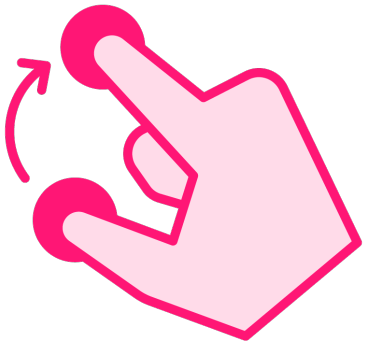
In case of overlapping permissions, an explicit deny ALWAYS wins!

Permanent IAM Credentials

Access Key pairs can be used for programmatic access (*no console access*)



Username and password authentication can be used for console access



Rotate key pairs and passwords based on your best practice guidelines

IAM Credential Reports can generate reports with new data every 4 hours regarding IAM users, access keys, and MFA status

Any reports generated within 4 hours of each will only contain the existing data from before