AWS IAM
Identity and Access Management

David Krohn, AWS DevOps Engineer

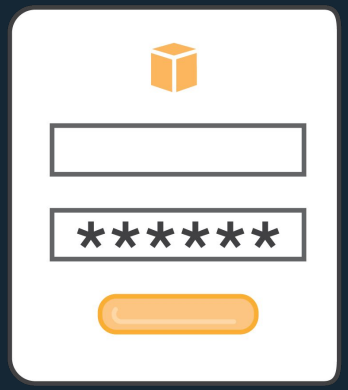
AWS Identity and Access Management

- IAM Essentials
- IAM Root
- IAM Users
  - IAM Groups
- IAM API Keys
- IAM Policies
- IAM Roles
- IAM Temporary Credentials
- IAM Federated Access

Identity and Access Management Means …

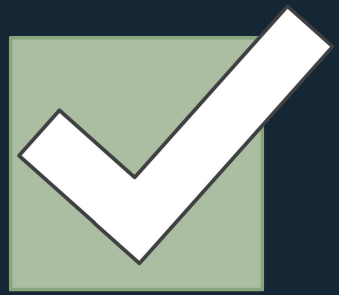
AUTHENTIFICATION

Validate identities securely.



AUTHORIZATION

Manage access using fine-grained policies.



AUDIT / GOVERNANCE

Meet compliance requirements.

IAM Essentials

- IAM provides access and access permissions to AWS resources
- IAM is global to all AWS regions
- Manage:
  - Users, Groups, Roles, Policies, API Keys
- Non-explicit deny rule set:
  - New created IAM Users have no access to any AWS Services

IAM Root

Account Owner ID (Root Account)

- Access to all subscribed services
- Access to billing
- Access to console and APIs
- Access to Customer Support

IAM Root

DO NOT USE ROOT FOR DAILY WORK

TASKS THAT REQUIRE ROOT CREDENTIALS

IAM Users

- Identified by an ARN
- Permissions
  - Attached or inline permissions
- Security credentials
  - Sign-in credentials
  - Access Key
  - SSH Key

IAM Groups

- Allows to assign IAM permission policies
- Easier access management

IAM API Keys

- Required for programmatic calls to AWS
  - AWS CLI
  - Tools for Windows Powershell
  - AWS SDKs
  - Direct HTTP calls using the AWS service APIs
- API Keys are only available during creation
- Roles does not have API credentials
- In the AWS console you can only see the access key

IAM Policies

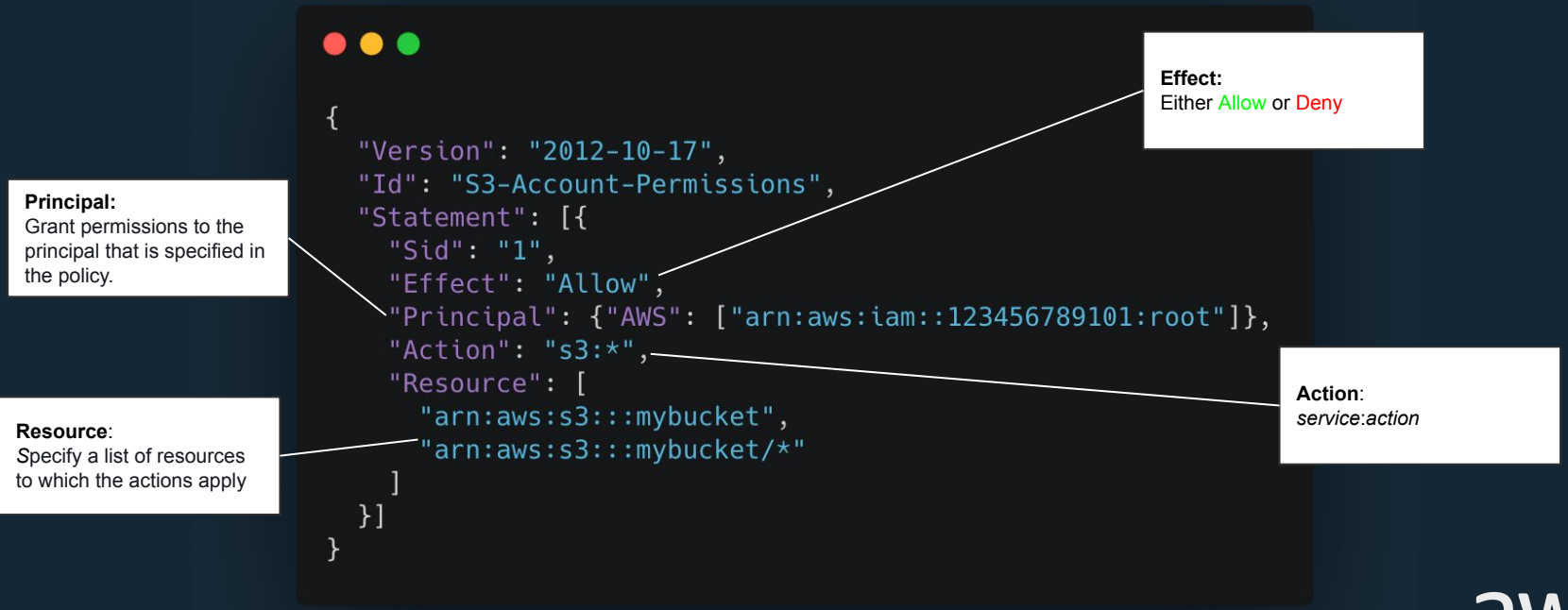
- A policy is a document that formally states one or more permissions
- There are two kinds of permissions:
  - Explicit deny: Explicit denies will always override explicit allows
  - Explicit allows: Detailed access rights to AWS services
- IAM policies take effect immediately
- IAM provides prebuilt policy templates

https://policysim.aws.amazon.com/home/index.jsp

IAM Policies

- Can be attached to Identities or resources
- Identities are
  - User
  - Groups
  - Roles
- Resources eg.
  - S3 Buckets
  - KMS Key
  - SQS Key

IAM Policies - What does an policy look like?



IAM Policies - Identity Policy

User

attached to

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ec2:AttachVolume",
                "ec2:DetachVolume"
            ],
            "Resource": [
                "arn:aws:ec2:*:*:volume/*",
                "arn:aws:ec2:*:*:instance/*"
            ]
        }
    ]
}
```

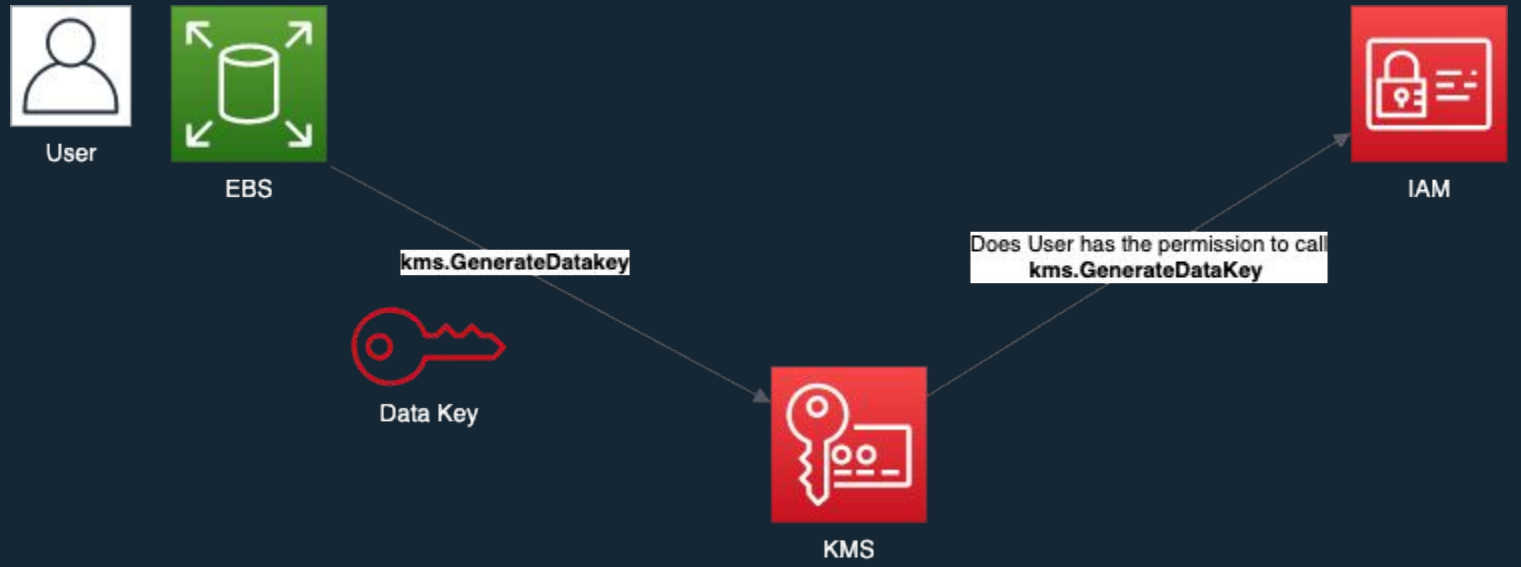
IAM Policies - Resource Policy

KMS Key

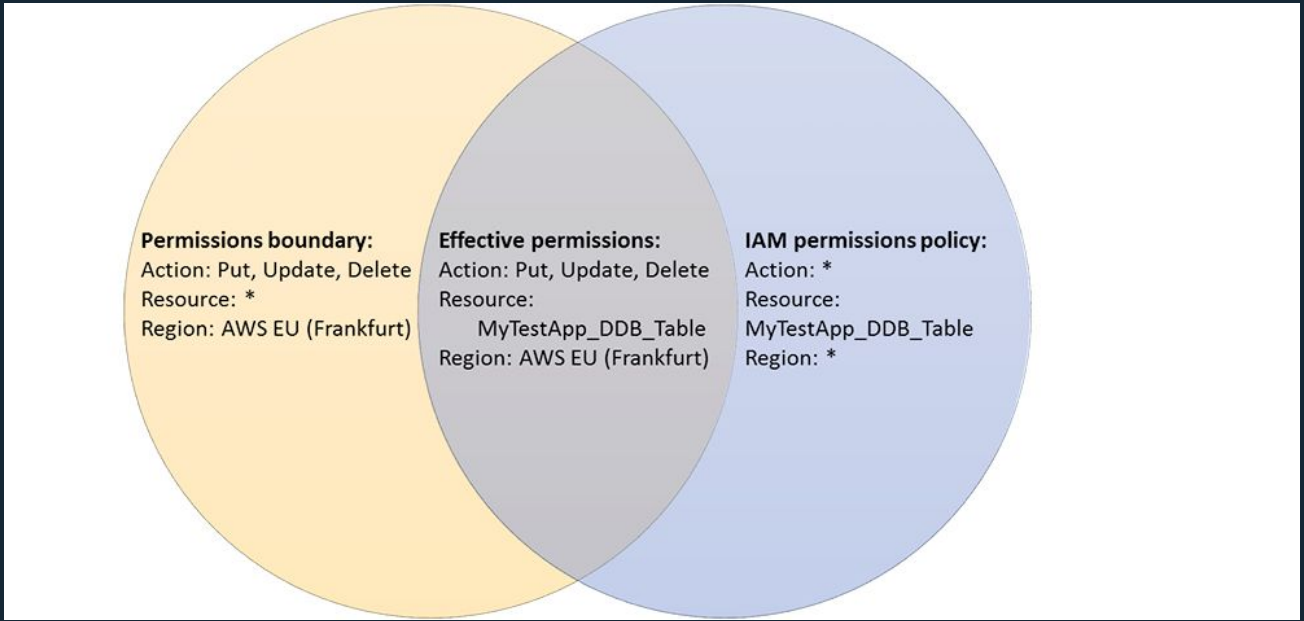
attached to

```
{
  "Sid": "Allow use of the key",
  "Effect": "Allow",
  "Principal": {"AWS": [
    "arn:aws:iam::123456789012:user/CMKUser",
    "arn:aws:iam::123456789012:role/CMKRole",
    "arn:aws:iam::123456789012:root"
  ]},
  "Action": [
    "kms:Encrypt",
    "kms:Decrypt",
    "kms:ReEncrypt*",
    "kms:GenerateDataKey*",
    "kms:DescribeKey"
  ],
  "Resource": "*"
}
```

IAM Policies - example EBS encrypted Volume



IAM Policies - Permission Boundary



IAM Roles

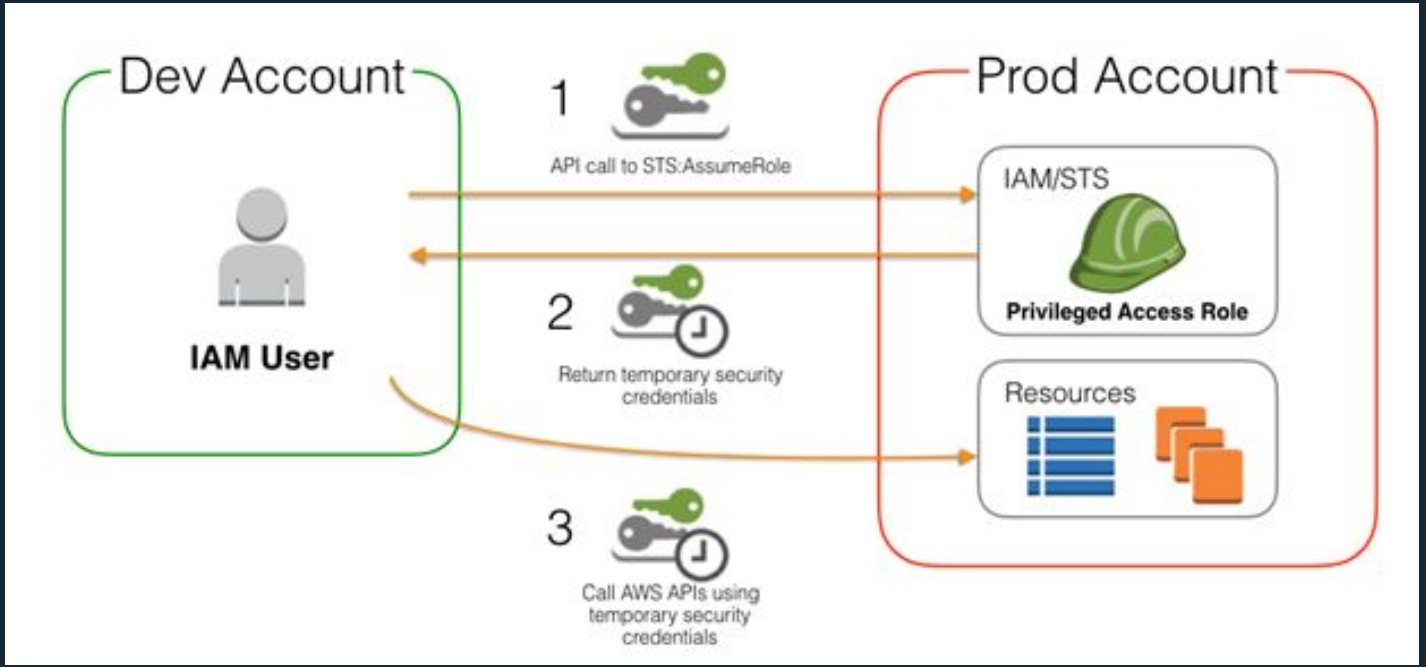
- Roles allow AWS resources, like EC2, to access other AWS services
- Roles can allow access to AWS services for non AWS account holders like Active Directory
- Roles are the only option to assign permission policies to AWS resources
- An EC2 instance can have only one role attached
- Roles can be assumed by other AWS Accounts

IAM Temporary Credentials

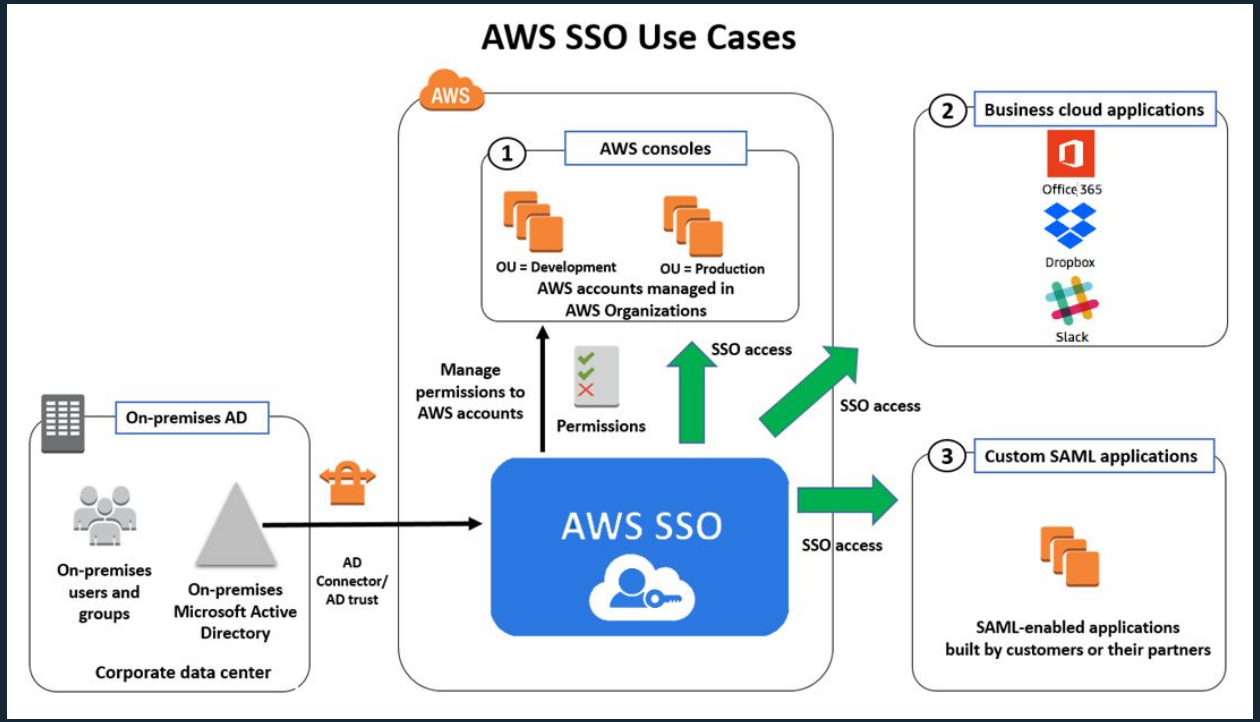
- AWS STS is a service to create and provide temporary security credentials that can control access to your AWS resources

- Advantages:
  - Provide access to resources without need to provide an AWS identity
  - Credentials have a limited lifetime upto 12 hours

IAM Cross Account Roles



IAM Federated Access



IAM Best Practices

- Enable multi-factor authentication (MFA) for privileged users.
- Remove Unnecessary Credentials.
- Use Policy Conditions for Extra Security.
- Avoid hardcoding credentials in source code.
  - You can use **IAM roles** instead
- Try to use temporary credentials
- Use Groups to Assign Permissions to IAM Users
- Least privileges principle
- Automate everything (CloudFormation or TF or CDK)
  - do security validation of your templates eg. cfn-nag

Questions

Labs

Recommended

https://labs.globaldatanet.com/IAMHandsOnLab.pdf

Follow-up

https://identity-round-robin.awsseclabss.com/permission-boundaries/

https://labs.globaldatanet.com/