AWS USER GROUPS MUMBAI

presents

SHETALKS

TECH TALKS

WORKSHOP

FIRECHAT

QUIZ

9TH APRIL,2022

2-5 PM IST

In association with

AWS User Groups Pune

AWS User Groups Nashik

Our Community Partners

pyladies Mumbai

WOMEN IN DATA SCIENCE MUMBAI & KERALA

WOMEN IN BIG DATA INDIA

WOMEN IN BIG DATA BANGLADESH

Speaker

Aishwarya Gupta

Sr. Cloud Engineer - AWS at Quantiphi

Agenda

AWS IAM

AWS Secrets Manager

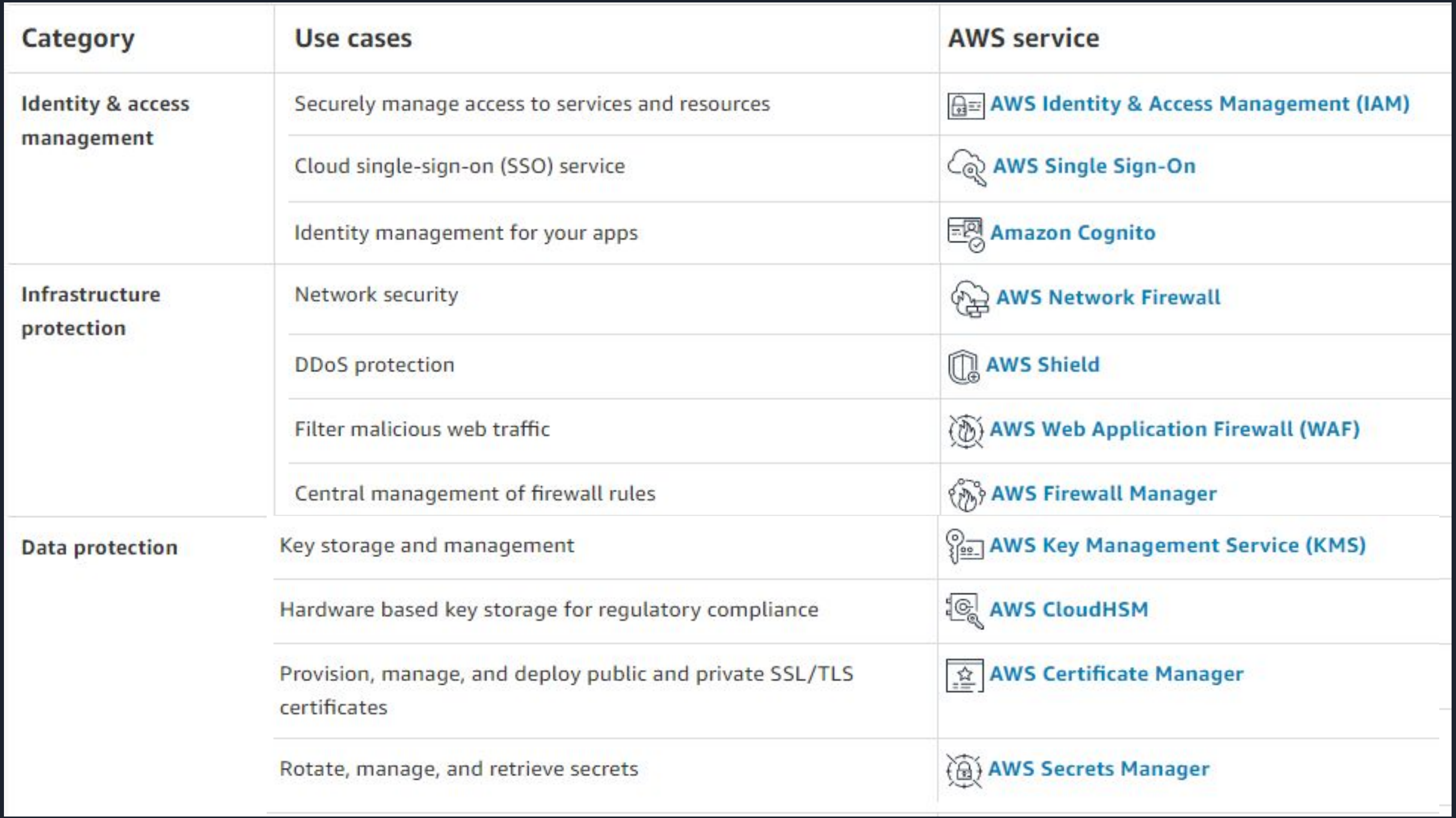
AWS KMS

AWS Certificate Manager

AWS WAF

AWS Shield

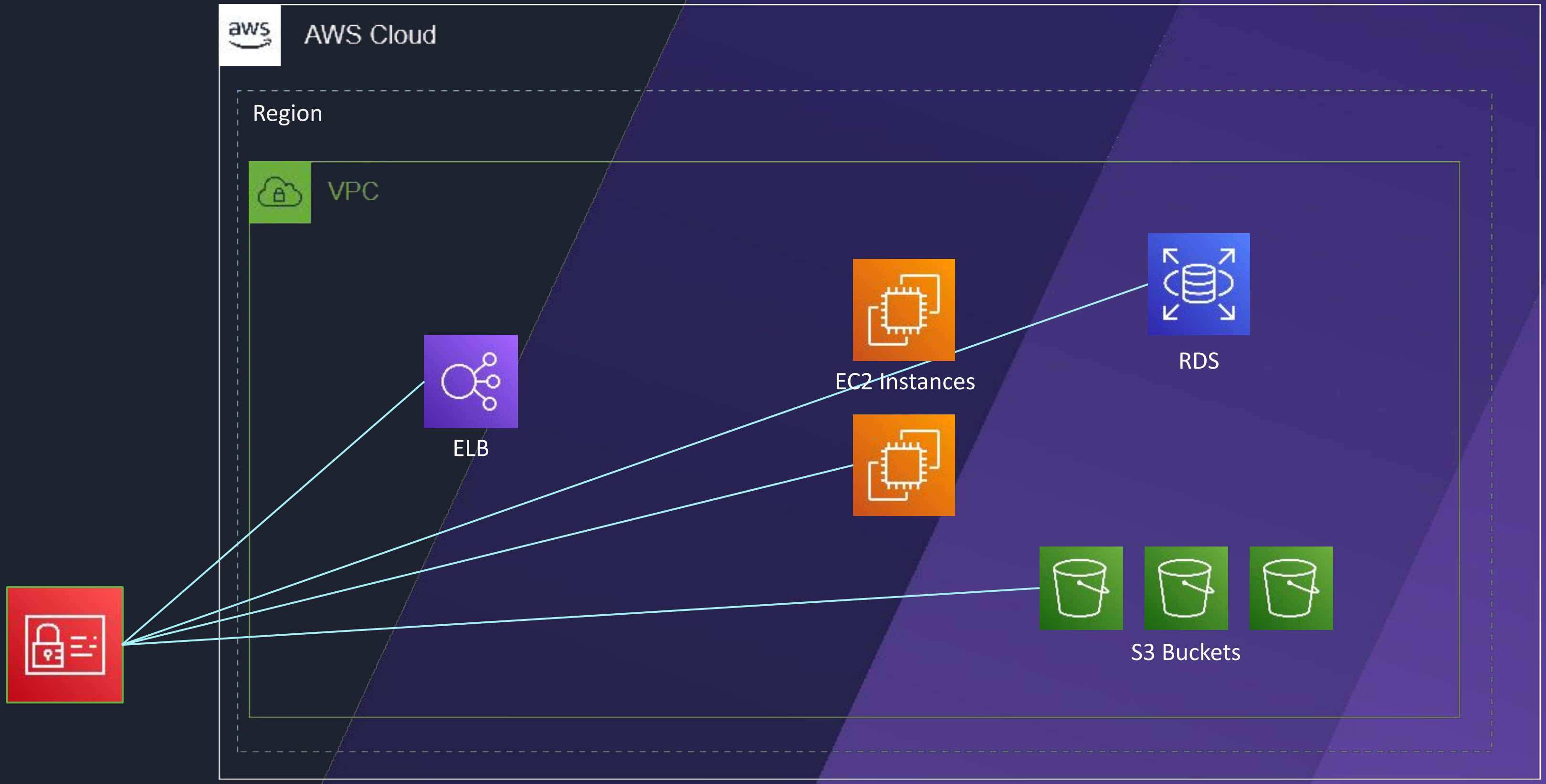
Security on AWS



Category	Use cases	AWS service
Identity & access management	Securely manage access to services and resources	AWS Identity & Access Management (IAM)
	Cloud single-sign-on (SSO) service	AWS Single Sign-On
	Identity management for your apps	Amazon Cognito
Infrastructure protection	Network security	AWS Network Firewall
	DDoS protection	AWS Shield
	Filter malicious web traffic	AWS Web Application Firewall (WAF)
	Central management of firewall rules	AWS Firewall Manager
Data protection	Key storage and management	AWS Key Management Service (KMS)
	Hardware based key storage for regulatory compliance	AWS CloudHSM
	Provision, manage, and deploy public and private SSL/TLS certificates	AWS Certificate Manager
	Rotate, manage, and retrieve secrets	AWS Secrets Manager

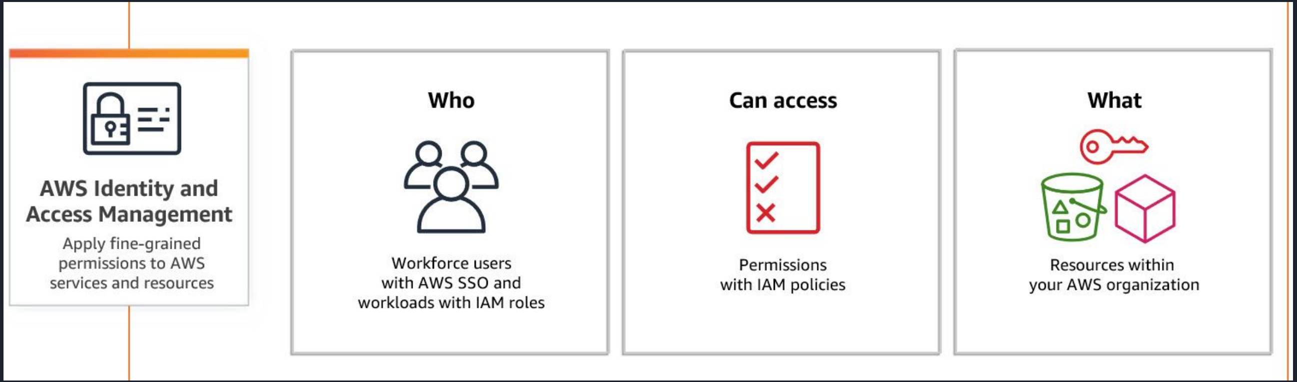
AWS Identity and Access Management (IAM)

AWS
User Groups


AWS Cloud
Region
VPC
ELB
EC2 Instances
RDS
S3 Buckets

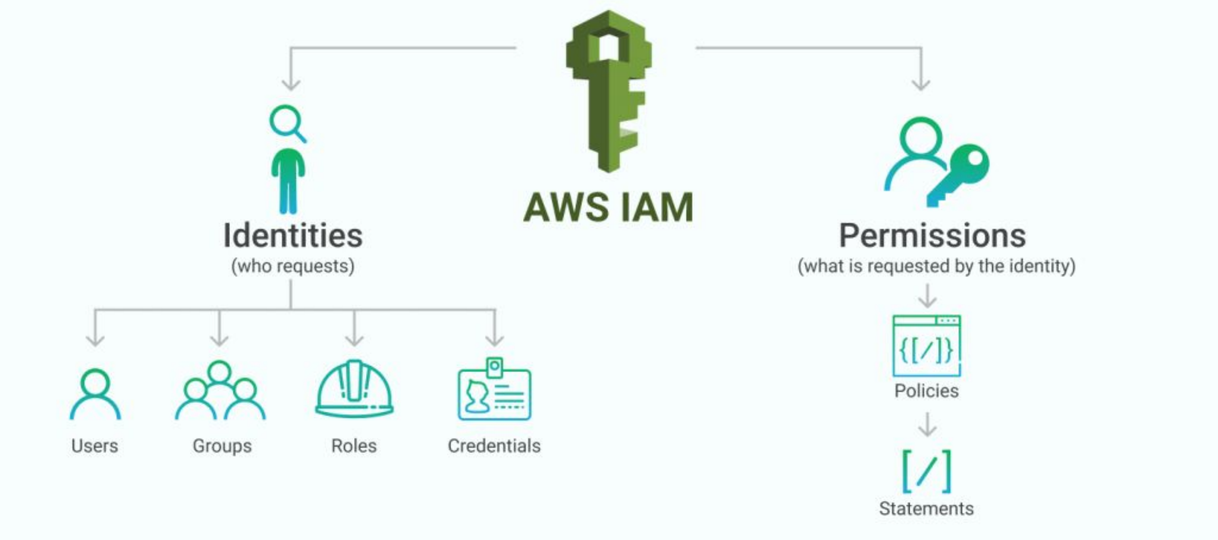
AWS IAM Overview

- AWS Identity and Access Management (IAM) provides fine-grained access control across all of AWS.
- With IAM,
  - you can specify who can access which services and resources, and under which conditions.
  - you manage permissions to ensure least-privilege permissions.
- IAM is a **Global** feature of your AWS account and is offered at **no additional charge**.

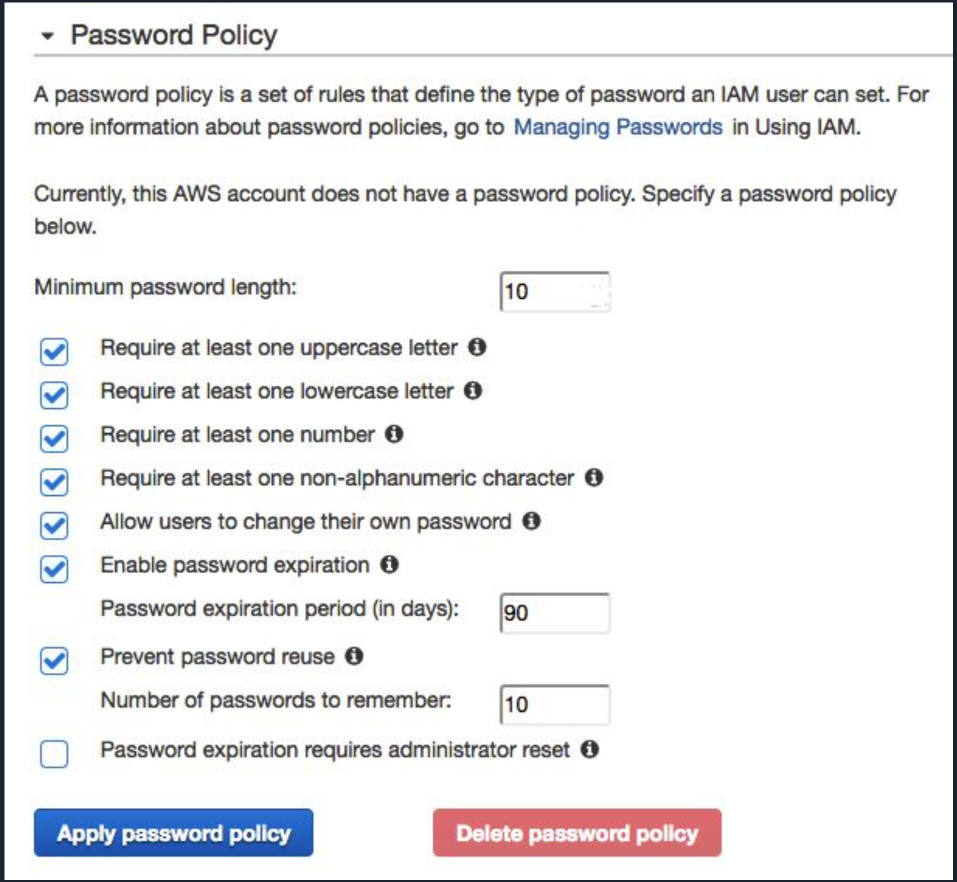

AWS Identity and Access Management
Apply fine-grained permissions to AWS services and resources
Who
Workforce users with AWS SSO and workloads with IAM roles
Can access
Permissions with IAM policies
What
Resources within your AWS organization

AWS IAM - Users and Groups

- Root account created by default, shouldn't be used or shared as it as Administrative privileges.
- An **IAM user** is an entity that you create in AWS to represent the person or application that uses it to interact with AWS. A user in AWS consists of a name and credentials.
- Users performing same tasks with the same resources can be added to a Group.



IAM – User Password and MFA



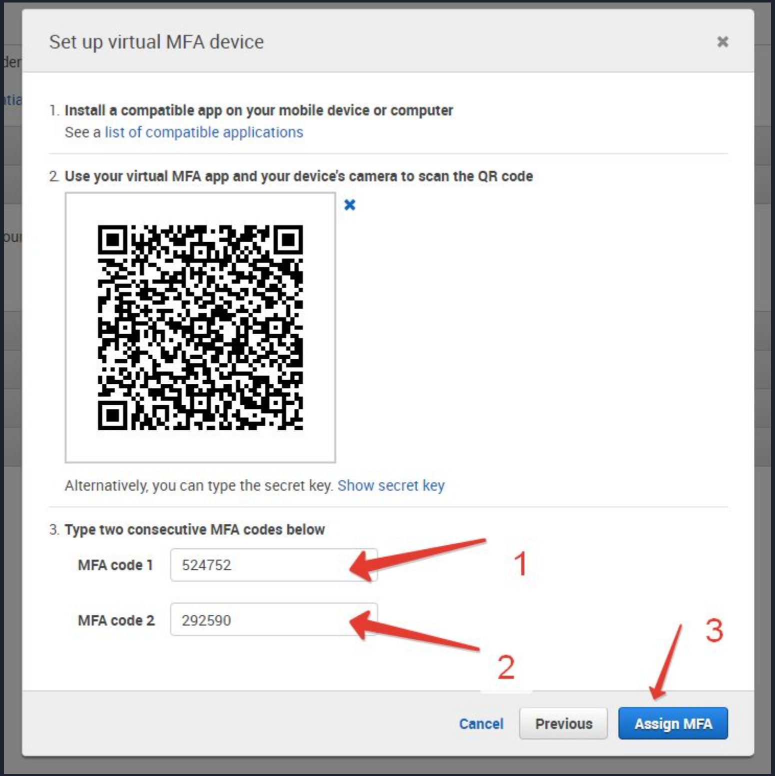
Password Policy

A password policy is a set of rules that define the type of password an IAM user can set. For more information about password policies, go to Managing Passwords in Using IAM.

Currently, this AWS account does not have a password policy. Specify a password policy below.

Minimum password length: 10

- ☑ Require at least one uppercase letter
- ☑ Require at least one lowercase letter
- ☑ Require at least one number
- ☑ Require at least one non-alphanumeric character
- ☑ Allow users to change their own password
- ☑ Enable password expiration
  - Password expiration period (in days): 90
- ☑ Prevent password reuse
  - Number of passwords to remember: 10
- □ Password expiration requires administrator reset

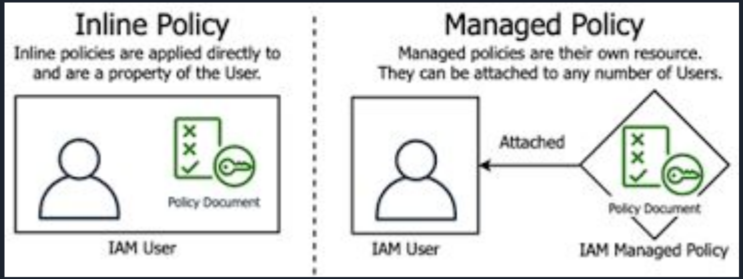


Set up virtual MFA device

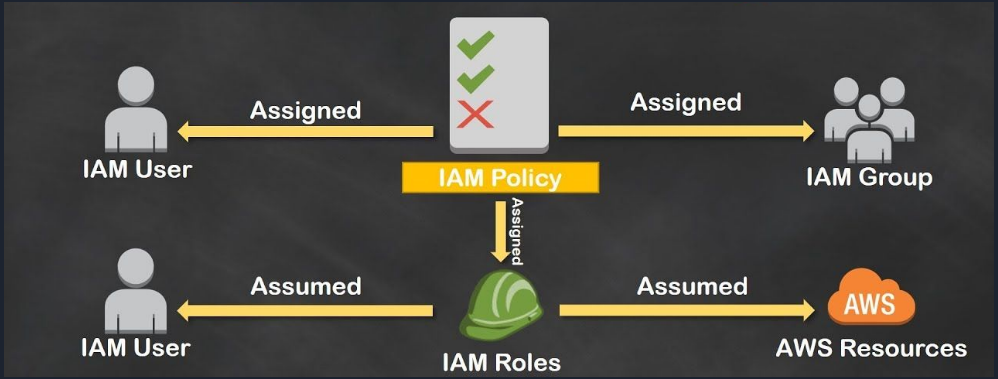
1. Install a compatible app on your mobile device or computer
   See a list of compatible applications
2. Use your virtual MFA app and your device's camera to scan the QR code
   Alternatively, you can type the secret key. Show secret key
3. Type two consecutive MFA codes below
   MFA code 1: 524752
   MFA code 2: 292590

AWS IAM - Policy

- You manage access in AWS by creating JSON based policies and attaching them to IAM identities (users, groups of users, or roles) or AWS resources.
- In AWS you apply the least privilege principle.
- **AWS managed policies** – Managed policies that are created and managed by AWS.
- **Customer managed policies** – Managed policies that you create and manage in your account.
- **Inline policies** – Policies that you add directly to a single user, group, or role.



Inline Policy
Inline policies are applied directly to and are a property of the User.
Policy Document
IAM User
Managed Policy
Managed policies are their own resource. They can be attached to any number of Users.
Attached
Policy Document
IAM User
IAM Managed Policy



AWS IAM Roles for Services

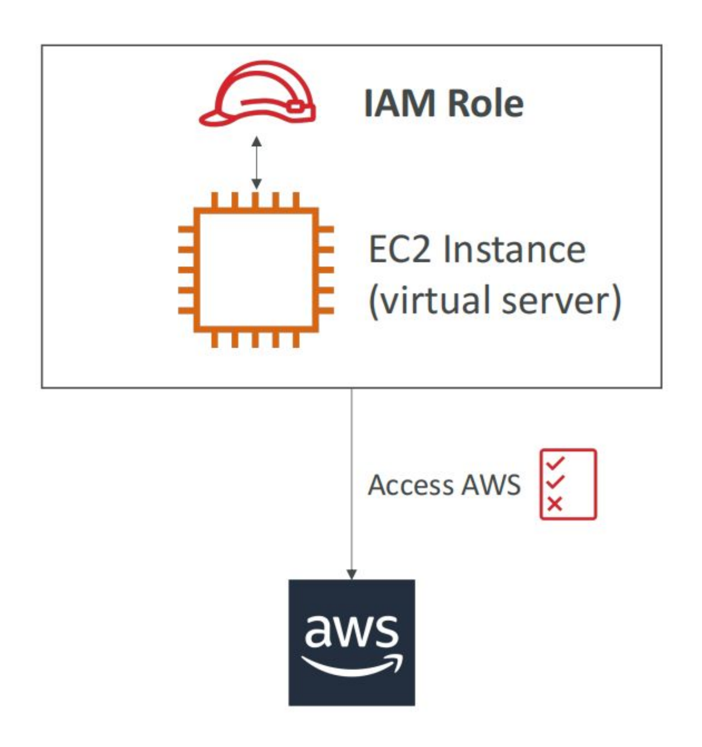
Some AWS service will need to perform actions on your behalf

To do so, we will assign permissions to AWS services with IAM Roles

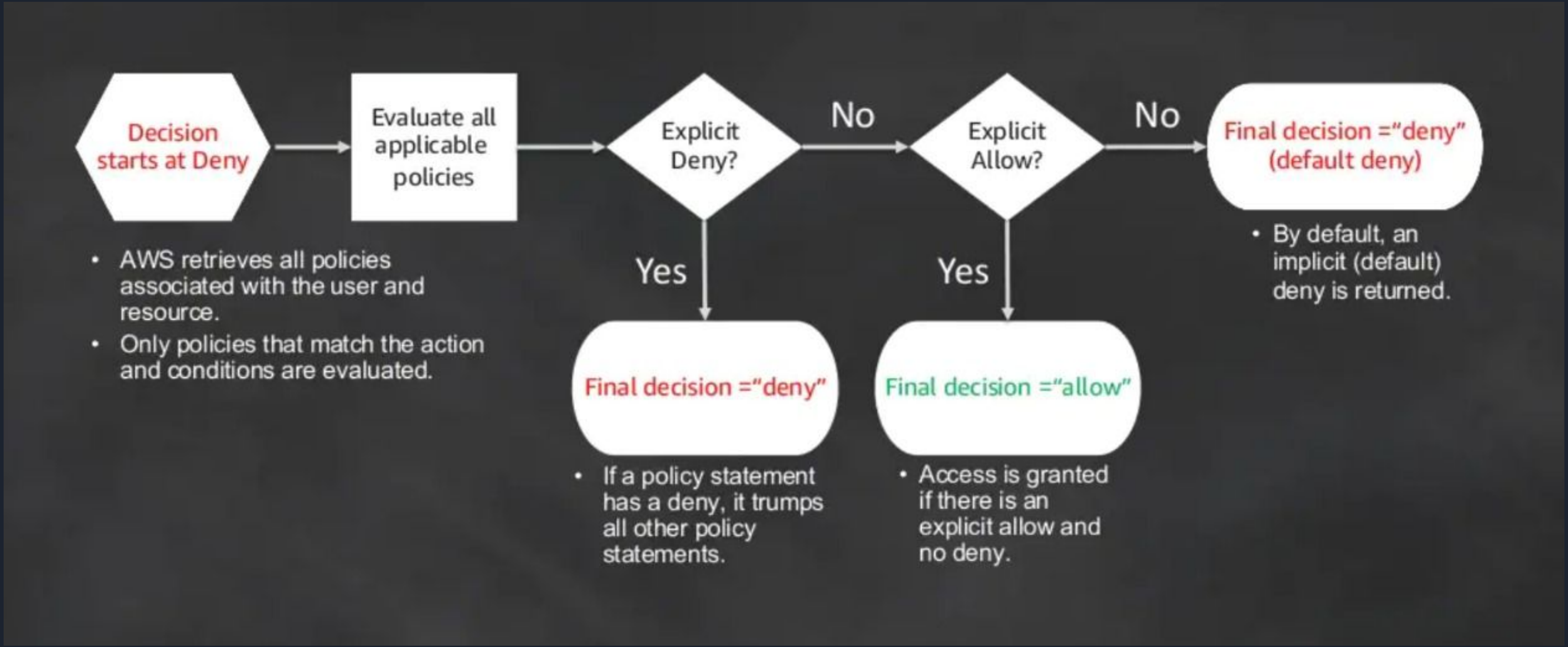
Some Common roles:

- EC2 Instance Roles
- Lambda Function Roles
- Roles for CloudFormation

DEMO



IAM policy Evaluation Logic



Decision starts at Deny

Evaluate all applicable policies

Explicit Deny?

No

Explicit Allow?

No

Final decision ="deny" (default deny)

Yes

Yes

Final decision ="deny"

Final decision ="allow"

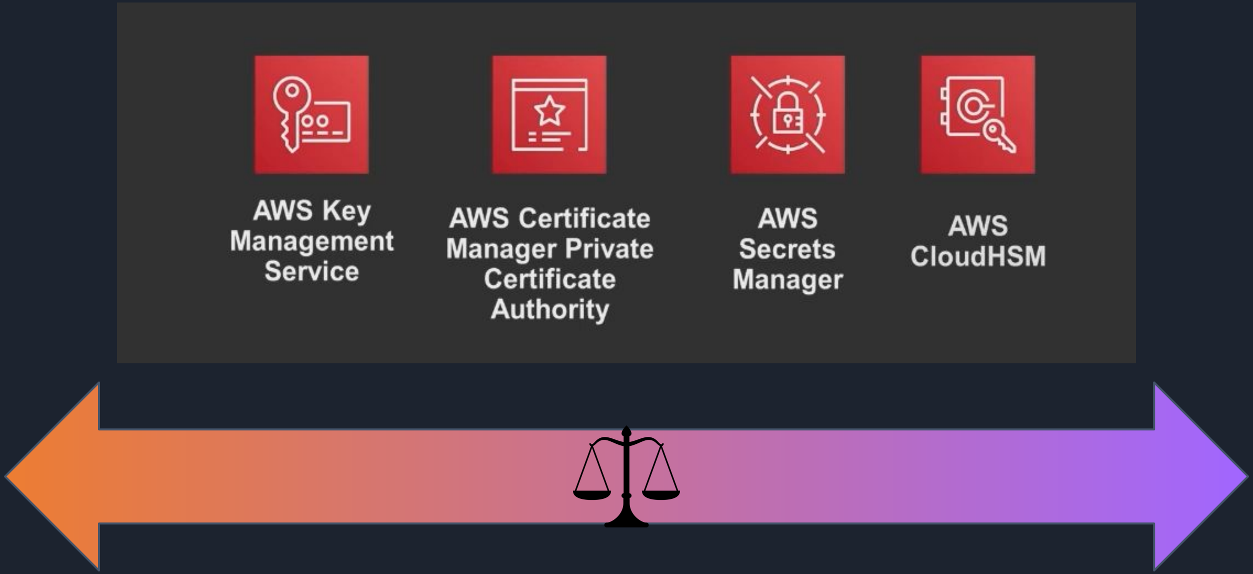
- AWS retrieves all policies associated with the user and resource.
- Only policies that match the action and conditions are evaluated.

- If a policy statement has a deny, it trumps all other policy statements.

- Access is granted if there is an explicit allow and no deny.

- By default, an implicit (default) deny is returned.

AWS Cryptographic Services



AWS Key Management Service

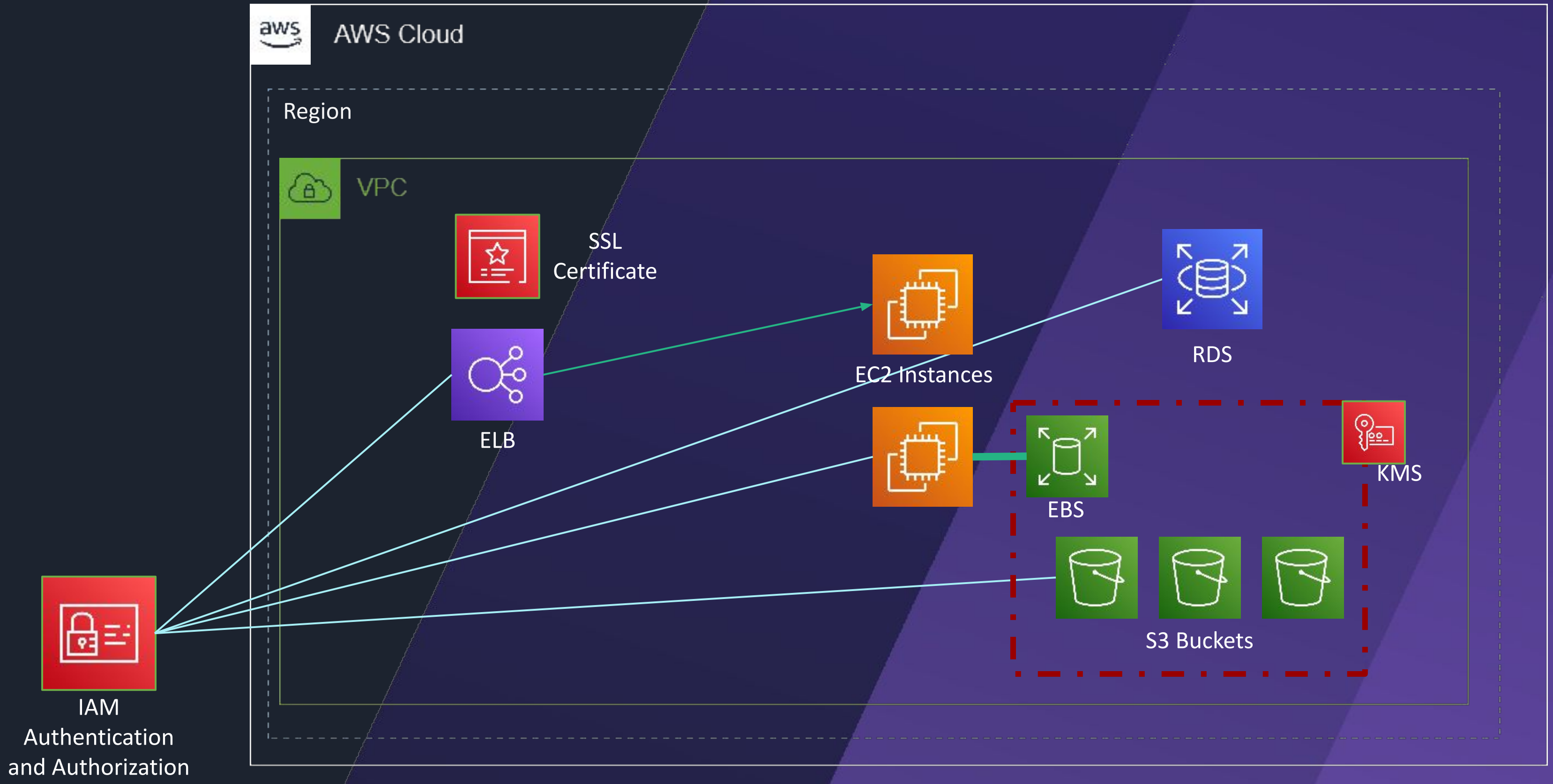
AWS Certificate Manager Private Certificate Authority

AWS Secrets Manager

AWS CloudHSM

Managed by AWS

Managed by you



AWS User Groups

AWS Key Management System (KMS)

AWS
User Groups

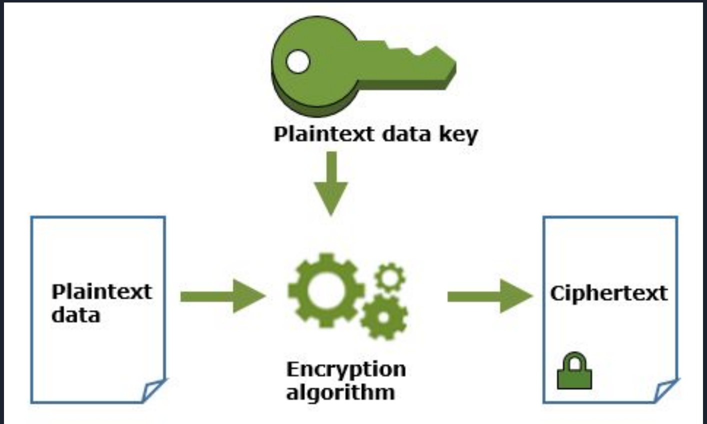
AWS Key Management Service (KMS)

Anytime you hear “encryption” for an AWS service, it’s most likely KMS

- Easy way to control access to your data, AWS manages keys for us
- Fully integrated with IAM for authorization

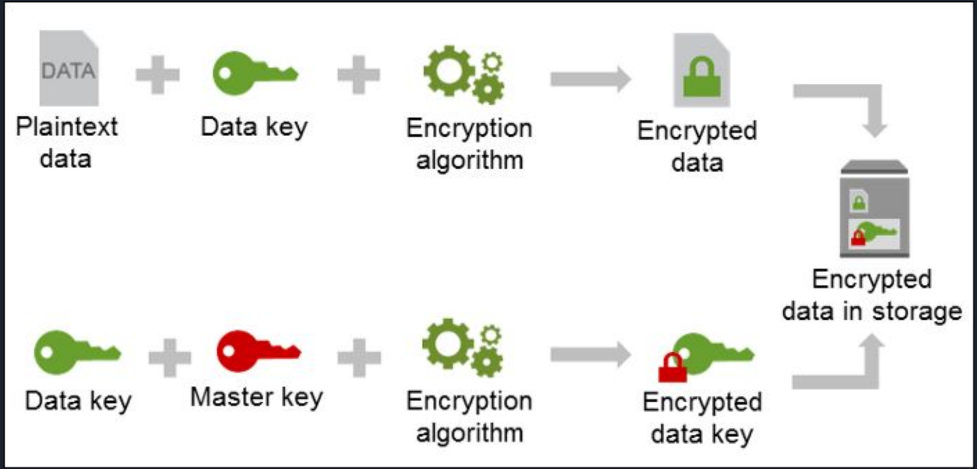
AWS KMS makes it easy for you to create and manage cryptographic keys and control their use across a wide range of AWS services and in your applications - EBS, S3, Redshift, RDS, SSM Parameter store etc

Integrated fully with CloudTrail for auditing function

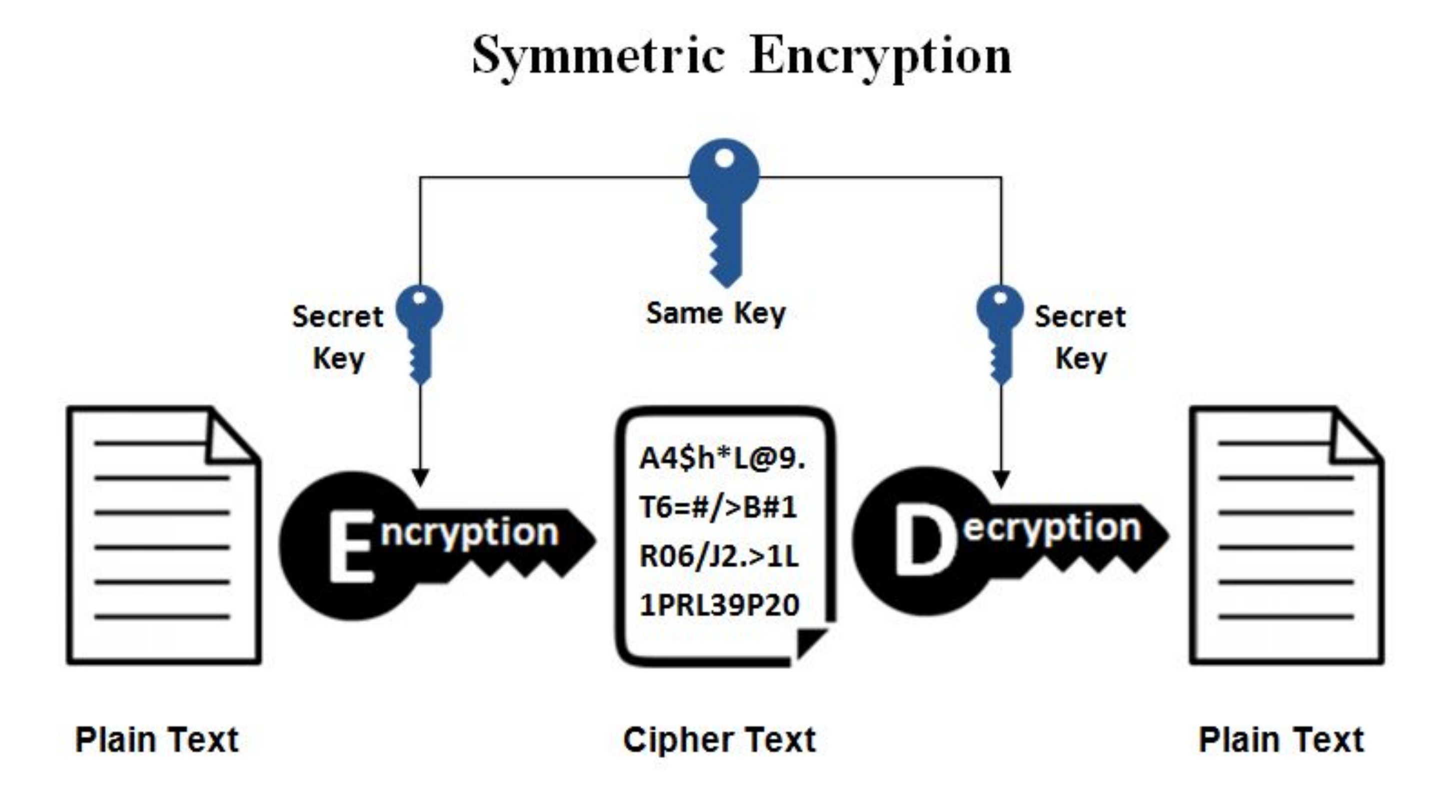


AWS KMS – Customer Master Key (CMK)

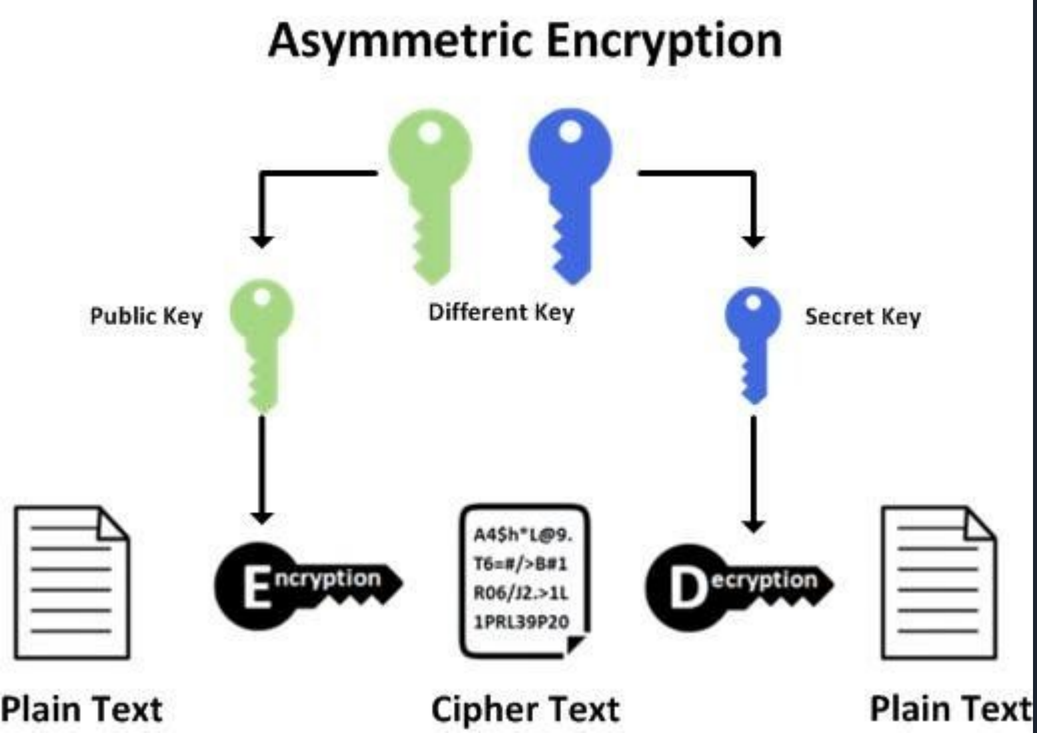
- KMS stores Customer Master Keys(CMK) which is a logical representation of a key.
- Key can be generated by KMS or imported.
- The encrypted data keys are stored with the data
- CMK never leaves KMS and never leaves a region
- CMK can encrypt or decrypt data up to 4KB in size.



AWS KMS - Symmetric Key



AWS KMS - Asymmetric Key



AWS KMS - Key Policies

Control access to KMS keys, “similar” to S3 bucket policies. Difference: you cannot control access without them

Default KMS Key Policy:

- Created if you don’t provide a specific KMS Key Policy
- Complete access to the key to the root user = entire AWS account
- Gives access to the IAM policies to the KMS key

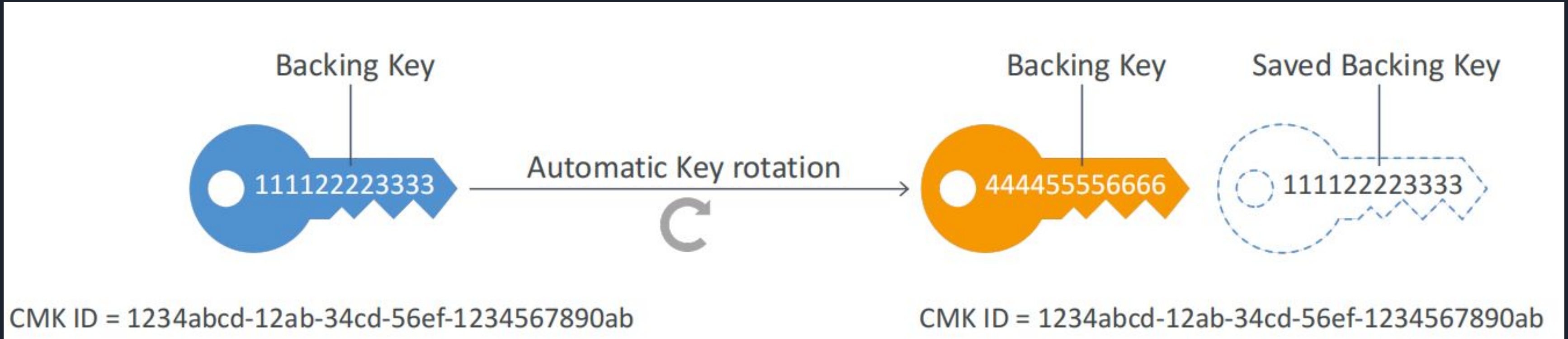
Custom KMS Key Policy:

- Define users, roles that can access the KMS key
- Define who can administer the key
- Useful for cross-account access of your KMS key

AWS KMS Automatic Key Rotation

For Customer-managed CMK (not AWS managed CMK)

- If enabled: automatic key rotation happens every 1 year
- Previous key is kept active so you can decrypt old data
- New Key has the same CMK ID (only the backing key is changed)



AWS Certificate Manager

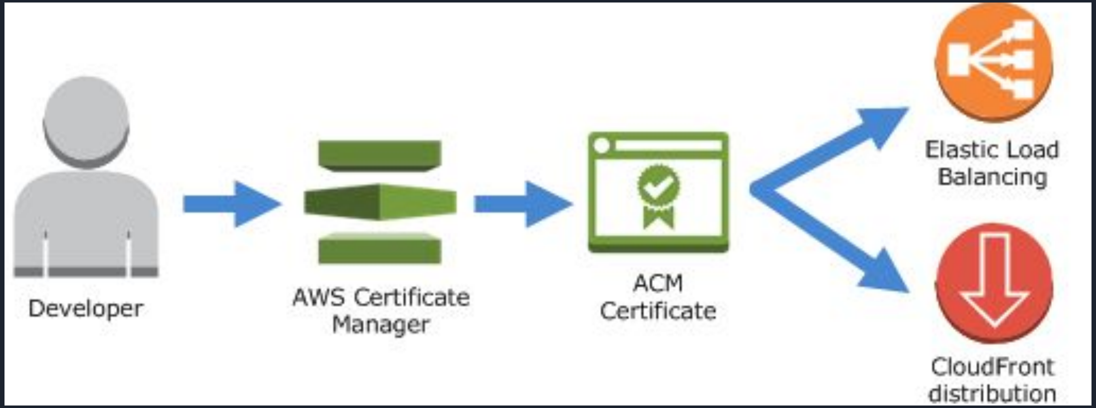
AWS
User Groups

AWS Certificate Manager

- An SSL Certificate allows traffic between your clients and your load balancer to be encrypted in transit (in-flight encryption)

- SSL(**Secure Sockets Layer)** / TLS (**Transport Layer Security**) certificates are used to secure network communications and establish the identity of websites over the Internet as well as resources on private networks.

- With AWS Certificate Manager, you can quickly request a certificate, deploy it on AWS resources, such as Elastic Load Balancers, CloudFront , API Gateway, etc and handle certificate renewals .

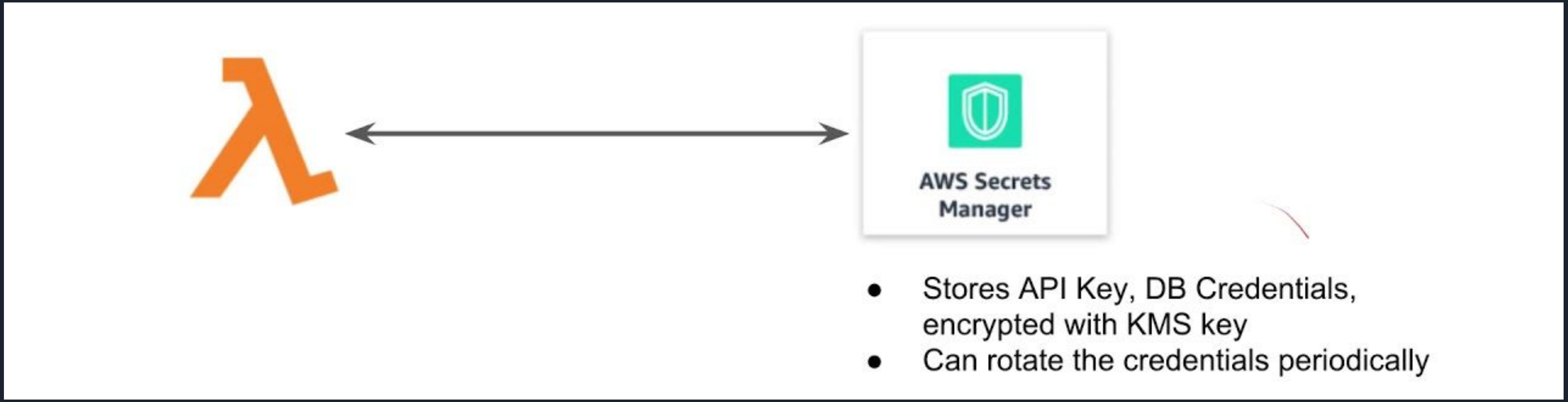


AWS Secrets Manager

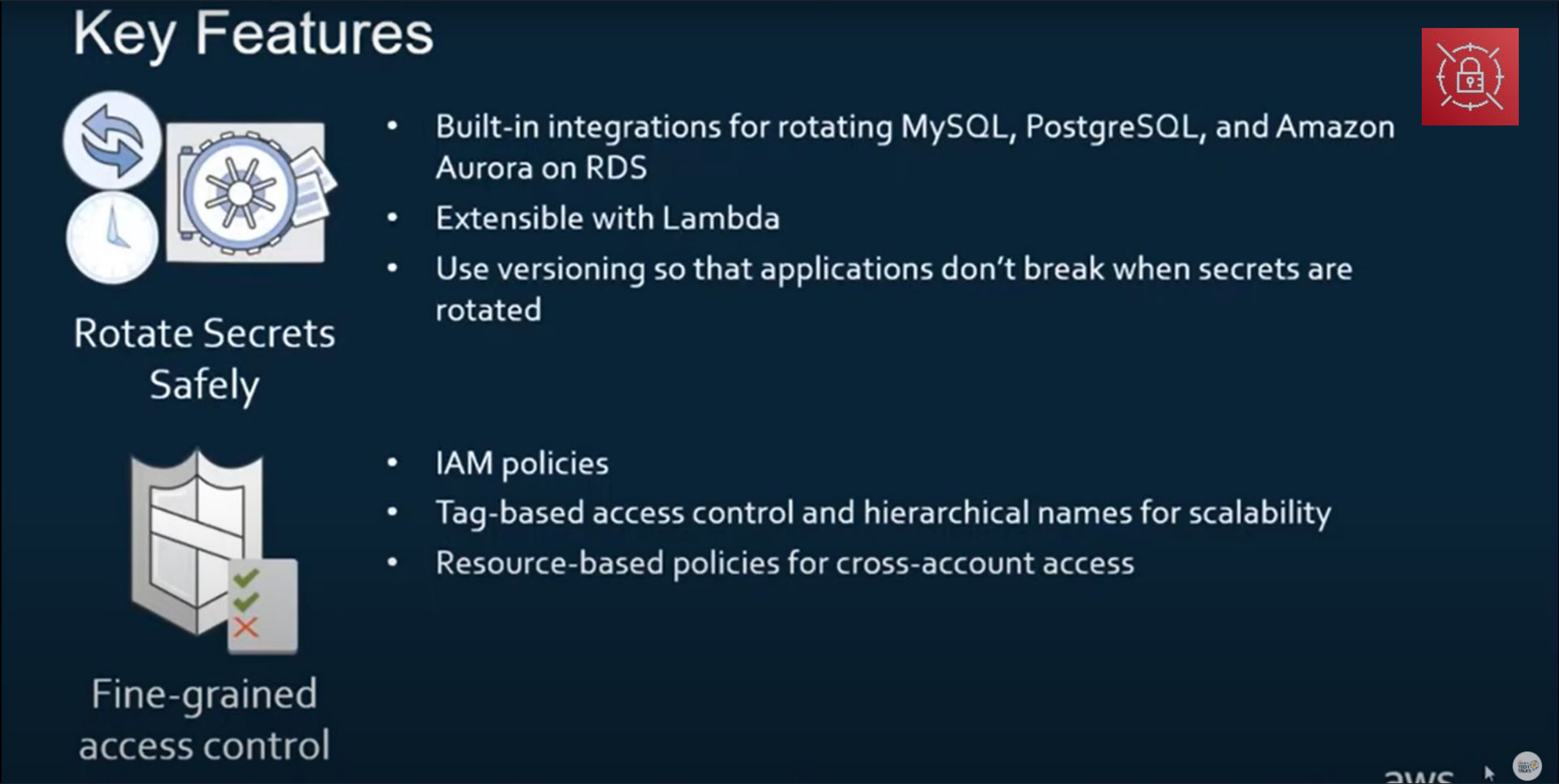
AWS
User Groups

AWS Secrets Manager

- The service enables you to easily rotate, manage, and retrieve database credentials, API keys, and other secrets throughout their lifecycle, provides built-in integration for Amazon RDS, Amazon Redshift, and Amazon DocumentDB.
- Using Secrets Manager, you can help secure secrets by encrypting them with encryption keys that you manage using AWS Key Management Service (KMS).
- Secrets Manager enables you to easily replicate secrets in multiple AWS regions to support your multi-region applications and disaster recovery scenarios.



- Stores API Key, DB Credentials, encrypted with KMS key
- Can rotate the credentials periodically



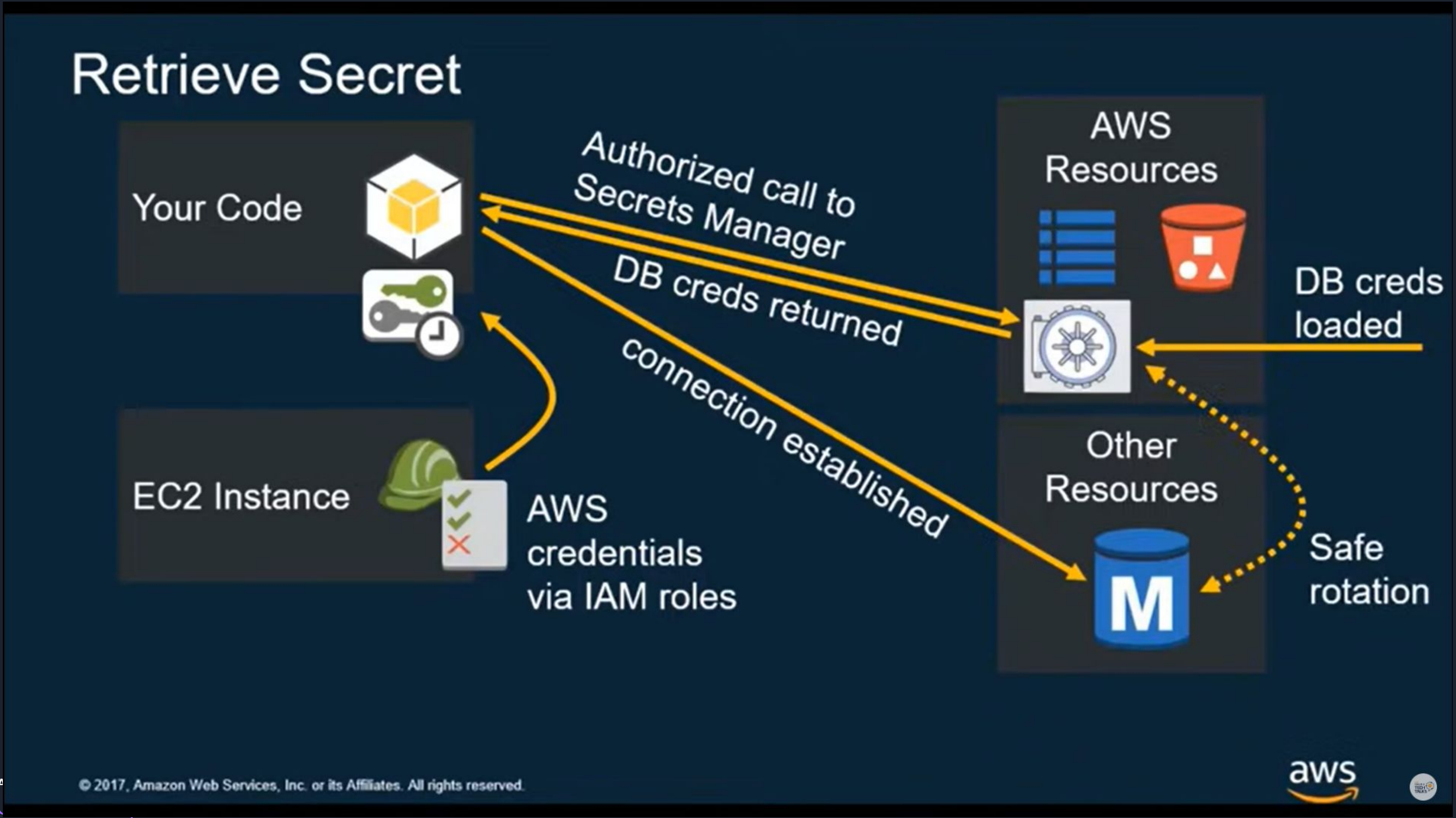
Key Features

Rotate Secrets Safely

- Built-in integrations for rotating MySQL, PostgreSQL, and Amazon Aurora on RDS
- Extensible with Lambda
- Use versioning so that applications don't break when secrets are rotated

Fine-grained access control

- IAM policies
- Tag-based access control and hierarchical names for scalability
- Resource-based policies for cross-account access



Retrieve Secret

Your Code

EC2 Instance

AWS credentials via IAM roles

Authorized call to Secrets Manager

DB creds returned

connection established

AWS Resources

DB creds loaded

Other Resources

Safe rotation

Q&A

Quiz Time !

Quiz

Note: Please use the same name via which you have registered!

Thank you!

AWS
User Groups