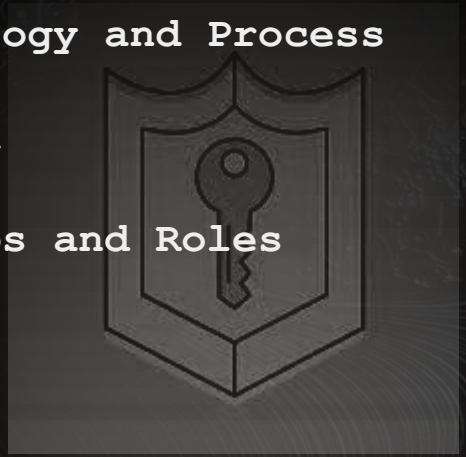
AWS Security Series

Part 2.1: Introduction to AWS Identity & Access Management with Security Best Practices

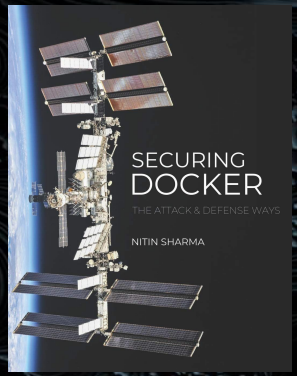
Nitin Sharma
CyberSecurity and DevSecOps Engineer
LinkedIn: linkedin.com/in/nitins87
Quora: quora.com/profile/NitinS-1
Blog: 4hathacker.in

Contents



$ whoami

- Cybersecurity and DevSecOps professional experienced in Cloud Security, Container Security and DevOps Research
- Certifications:



- Published author for "Securing Docker - The Attack & Defense Ways" book under CyberSecrets Publication
- Half Marathon runner, Cyclist and Fitness Enthusiast
- Helping out beginners in Cloud, DevOps and CyberSec at Quora

$ whoami (in the past…)

$ Identity and Access Management (IAM)

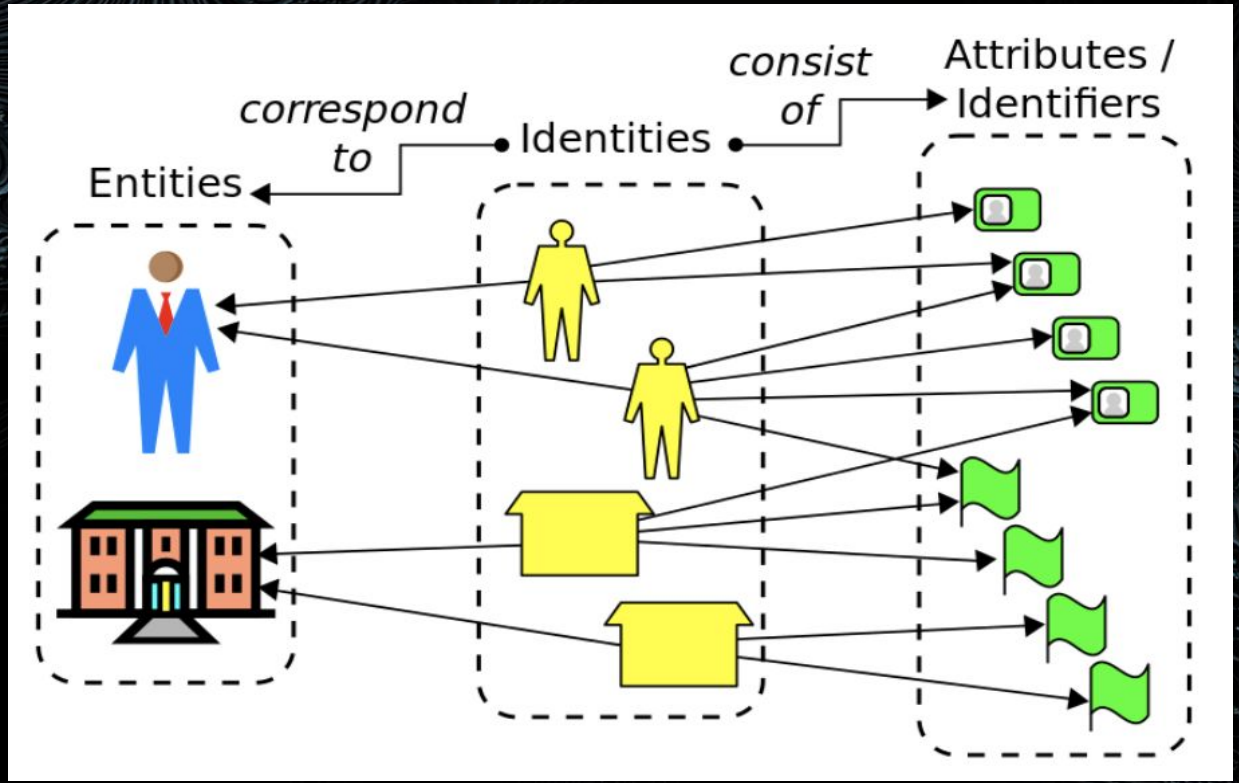
- Gartner defines IAM as,

  "The security discipline that enables the right individuals to access the right resources at the right times for the right reasons"

- IAM is a very broad area of practice which requires efficient domain specialists.

- IAM and IdM/IdAM are used interchangeably.

$ IAM Terminology



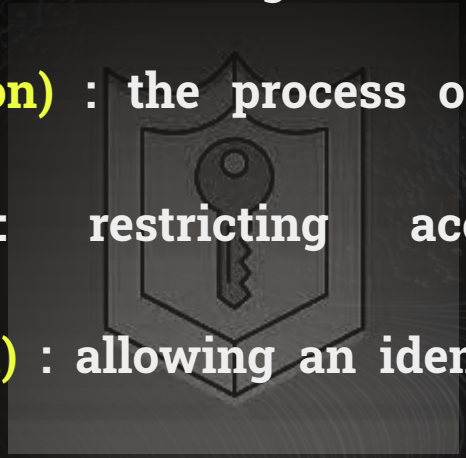
(Source: Wikipedia)

$ IAM Terminology

- ***Entity*** : the person or “thing” that will have an identity.
- ***Identity*** **(or digital identity)** : the unique expression of an entity within a given namespace.
- ***Identifier*** : the means by which an identity can be asserted. (e.g. Passport)
- ***Persona*** **(Groups)** : the expression of an identity with attributes that indicates context
- ***Attributes*** : facets of an identity. (e.g. IP Address)

$ IAM Terminology

- ***Role*** : identities can have multiple roles which indicate context. A way of delegation of trust.
- ***Authn*** (Authentication) : the process of confirming an identity.
- ***Access Control*** : restricting access to a resource.
- ***Authz*** (Authorization) : allowing an identity access to something.
- ***Entitlement*** : mapping an identity (including roles, personas, and attributes) to an auth(z).

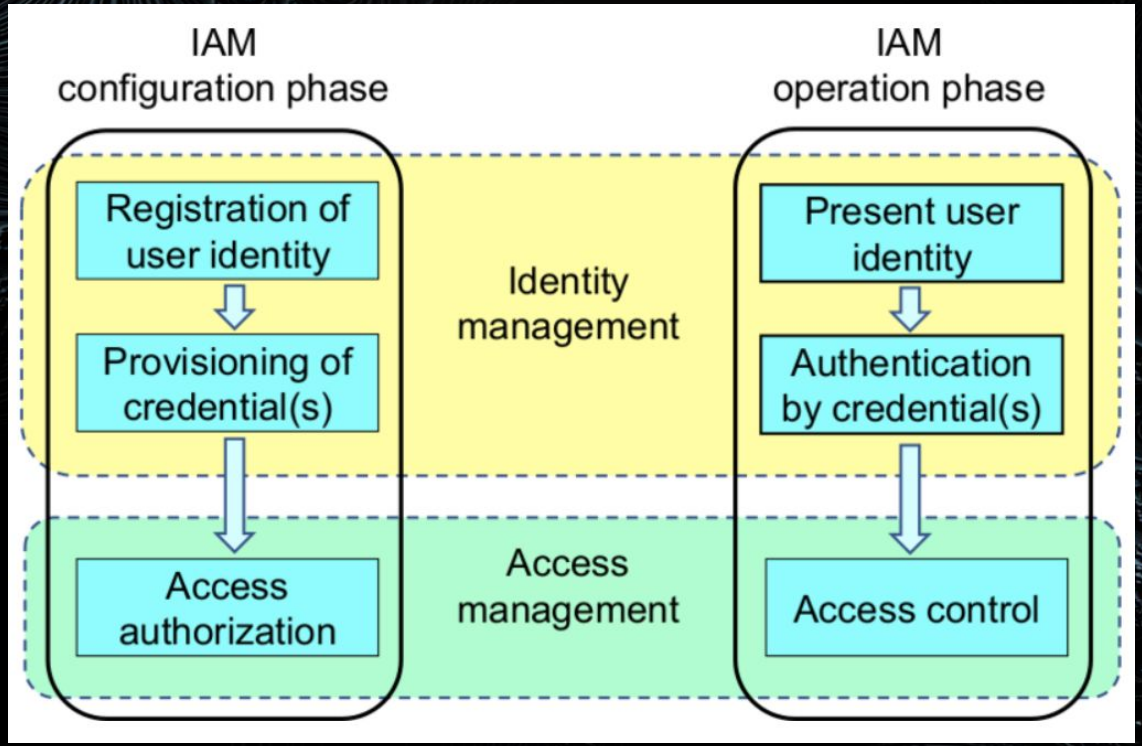


$ IAM Process (Entitlement)

Entitlement	Super-Admin	Service-1 Admin	Service-2 Admin	Dev	Security-Audit	Security-Admin
Service 1 List	X	X		X	X	X
Service 2 List	X		X	X	X	X
Service 1 Modify Network	X	X		X		X
Service 2 Modify Security Rule	X	X				X
Read Audit Logs	X				X	X

(Source: CSA Guidance v4 - Domain 12)

$ IAM Process (Overall)



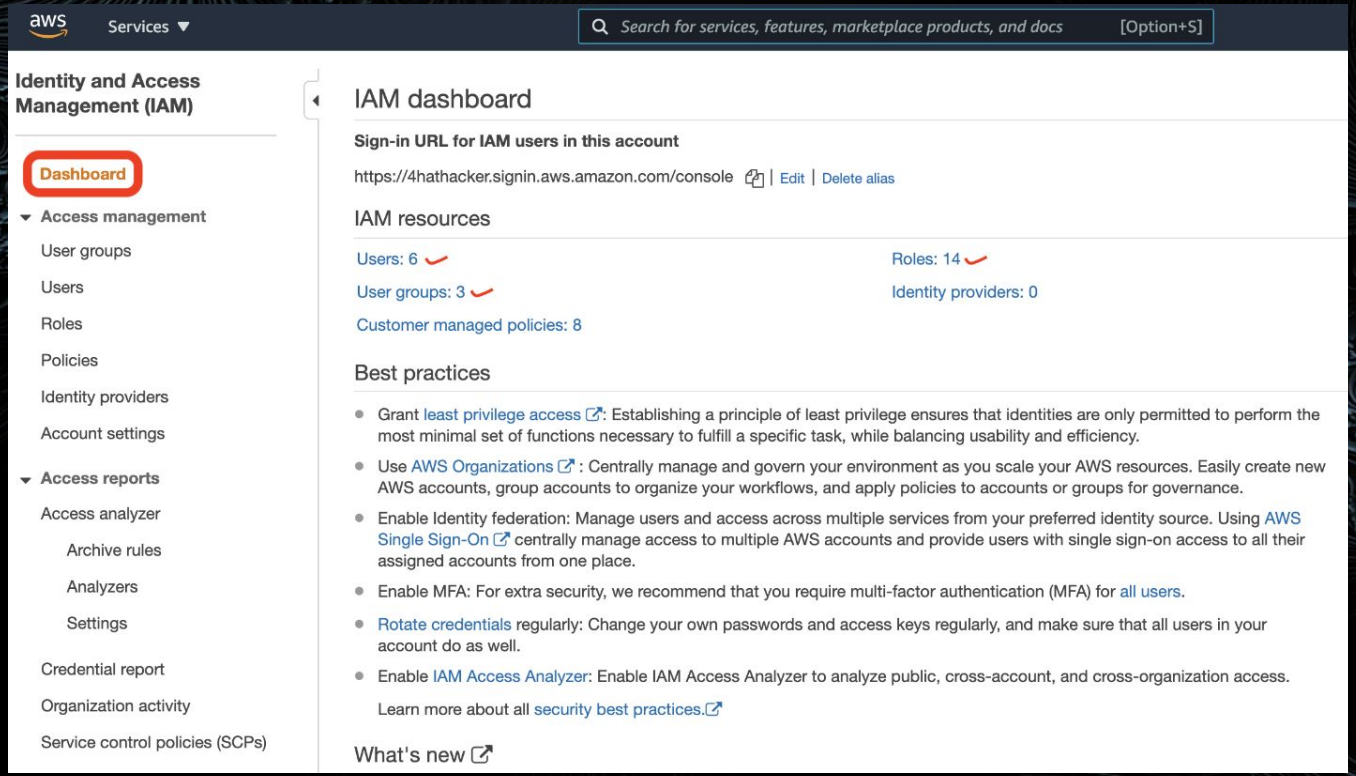
(Source: Wikipedia)

$ AWS IAM - Introduction

- AWS Identity and Access Management (IAM) is a web service that helps you securely control access to AWS resources. You use IAM to control who is **authenticated (signed in)** and **authorized (has permissions)** to use resources.

- When you first create an AWS account, you begin with a single sign-in identity that has complete access to all AWS services and resources in the account. This identity is called the **AWS account root user** and **is accessed by signing in with the email address and password that you used to create the account**.

$ AWS IAM (Dashboard)- Introduction



$ AWS IAM - User

- **unique identity** recognized by AWS services and applications.
- Similar to a login user in an operating system like Windows or UNIX, a user has a unique name and can identify itself using familiar security credentials such as a password or access key.
- A user can be an individual, system, or application requiring access to AWS services. IAM supports users (referred to as "***IAM users***") managed in AWS's Identity Management system, and it also enables you to grant access to AWS resources for users managed outside of AWS in your corporate directory (referred to as "federated users")

$ AWS IAM - User

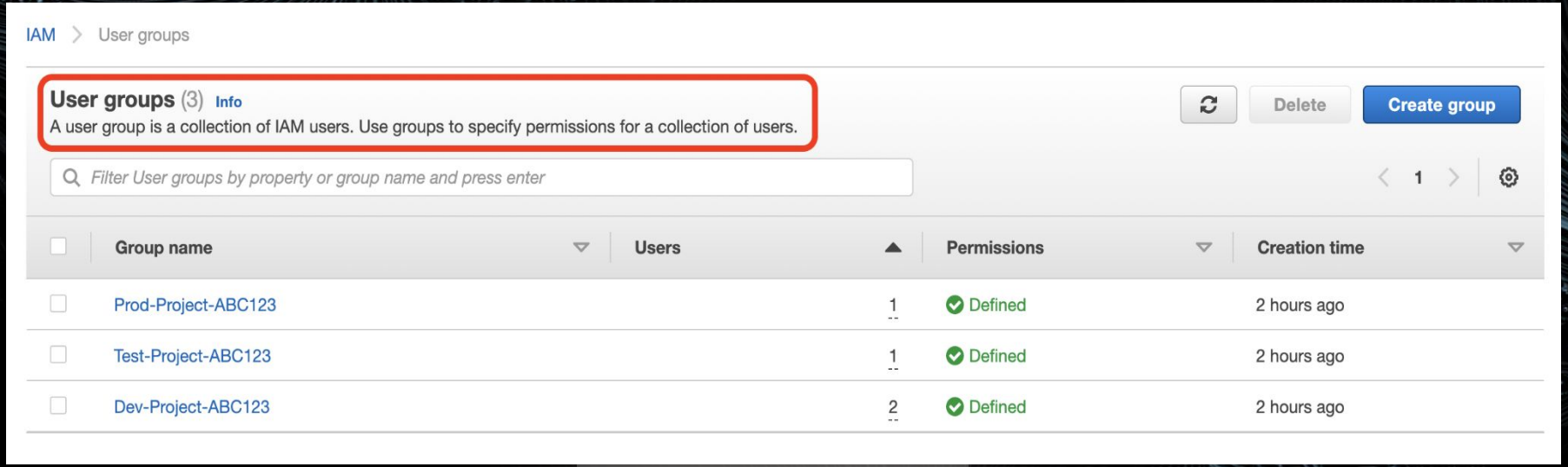
User name	Groups	Access key age	Password age	Last activity	MFA
4hathacker	None	9 days	119 days	8 days	Virtual
INT-Auditor	None	9 days	None	8 days	Not enabled
UserDev1	Dev-Project-ABC123	None	Today	None	Not enabled
UserDev2	Dev-Project-ABC123	None	Today	None	Not enabled
UserProd1	Prod-Project-ABC123	None	Today	None	Not enabled
UserTest1	Test-Project-ABC123	None	Today	None	Not enabled

Important Note:
IAM User = Permanent Long term Creds for direct interaction

$ AWS IAM - Group/User Group

- A user group is a collection of IAM users.
- User groups simplify permissions management by letting you grant, change, and remove permissions for multiple users at once.
- For example, you can create a user group named "Admins" and give that group administrative permissions. Any user in that group automatically has the permissions that are assigned to the group.
- A User can belong to multiple groups however Groups cannot belong to other groups.
- Groups do not have security credentials, and cannot access web services directly.

$ AWS IAM - Group

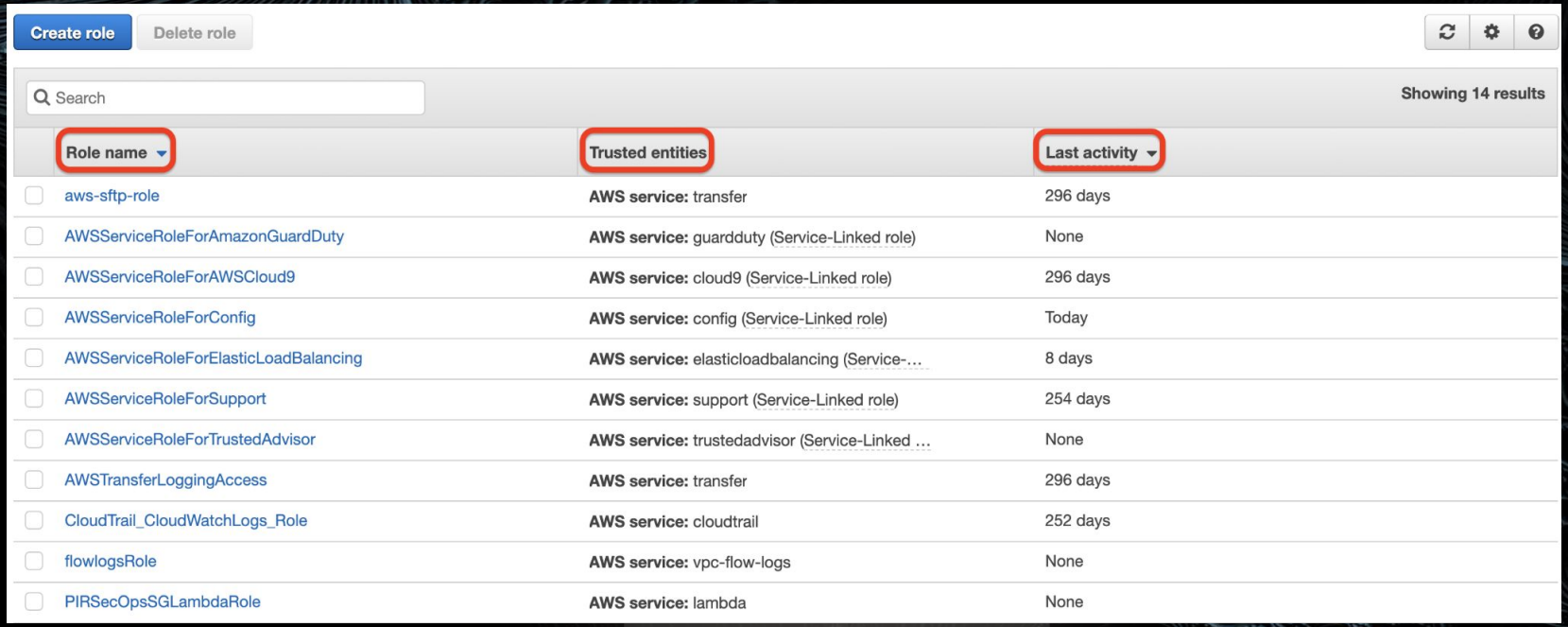


Important Note:
IAM Group = Management Convenience for a team

$ AWS IAM - Role

- defines a set of permissions for making AWS service requests.
- IAM roles are not associated with a specific user or group.
- IAM roles cannot make direct requests to AWS services.
- There is no limit to the number of IAM roles one can assume, but one can only act as single IAM role when making requests to AWS services.
- A role that links to an AWS service (also known as a **linked service/service linked role**) such that only the linked service can assume the role (Delegating permissions to AWS Services on your behalf)

$ AWS IAM - Role

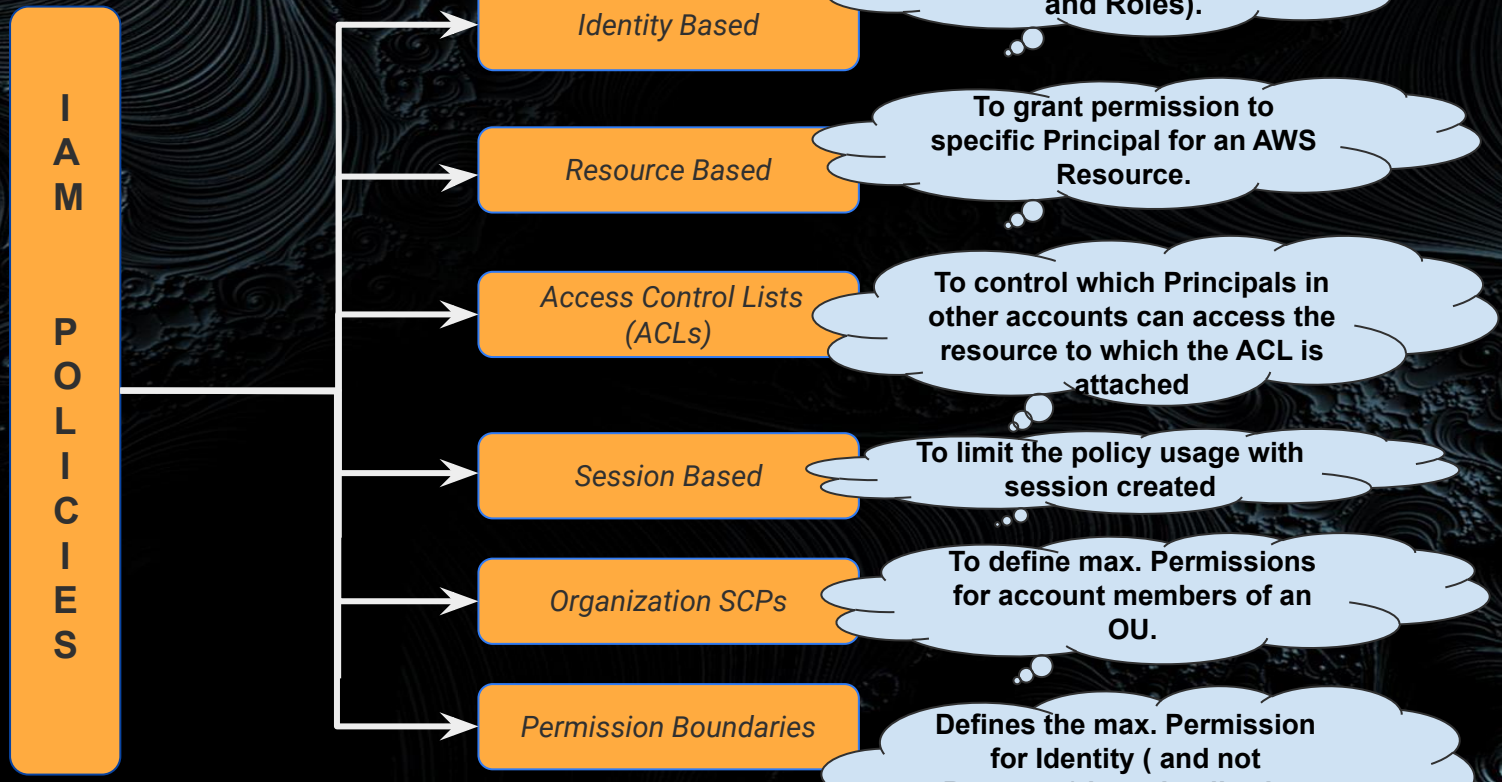


Important Note:
IAM Role = Delegating Trust

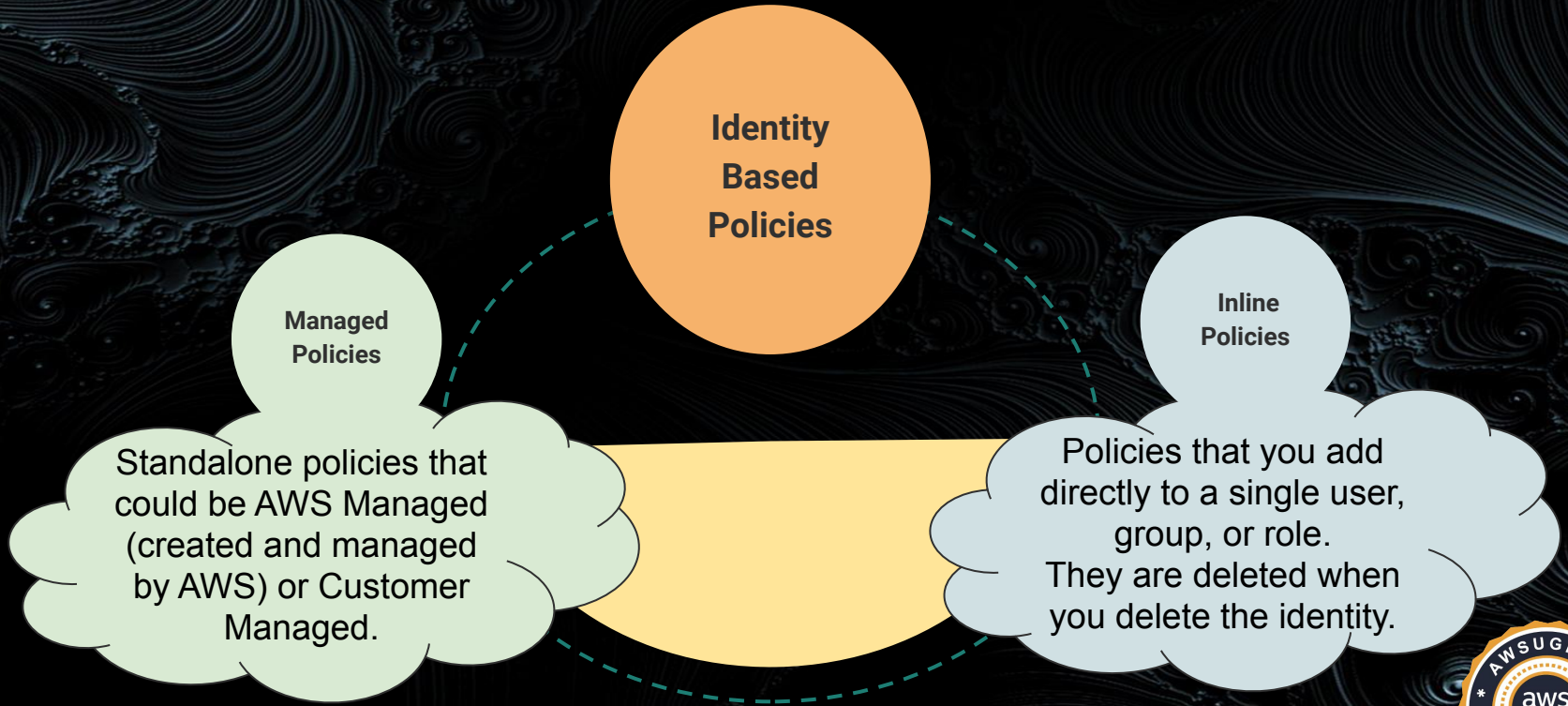
$ AWS IAM - Policies

- A policy is an object in AWS that, when associated with an identity or resource, defines their permissions.
- AWS evaluates these policies when an IAM principal (user or role) makes a request. Permissions in the policies determine whether the request is allowed or denied.
- Define permissions for an action regardless of the method that you use to perform the operation.
- Are JSON formatted documents.
- Attached to a principal (or identity) and resources.

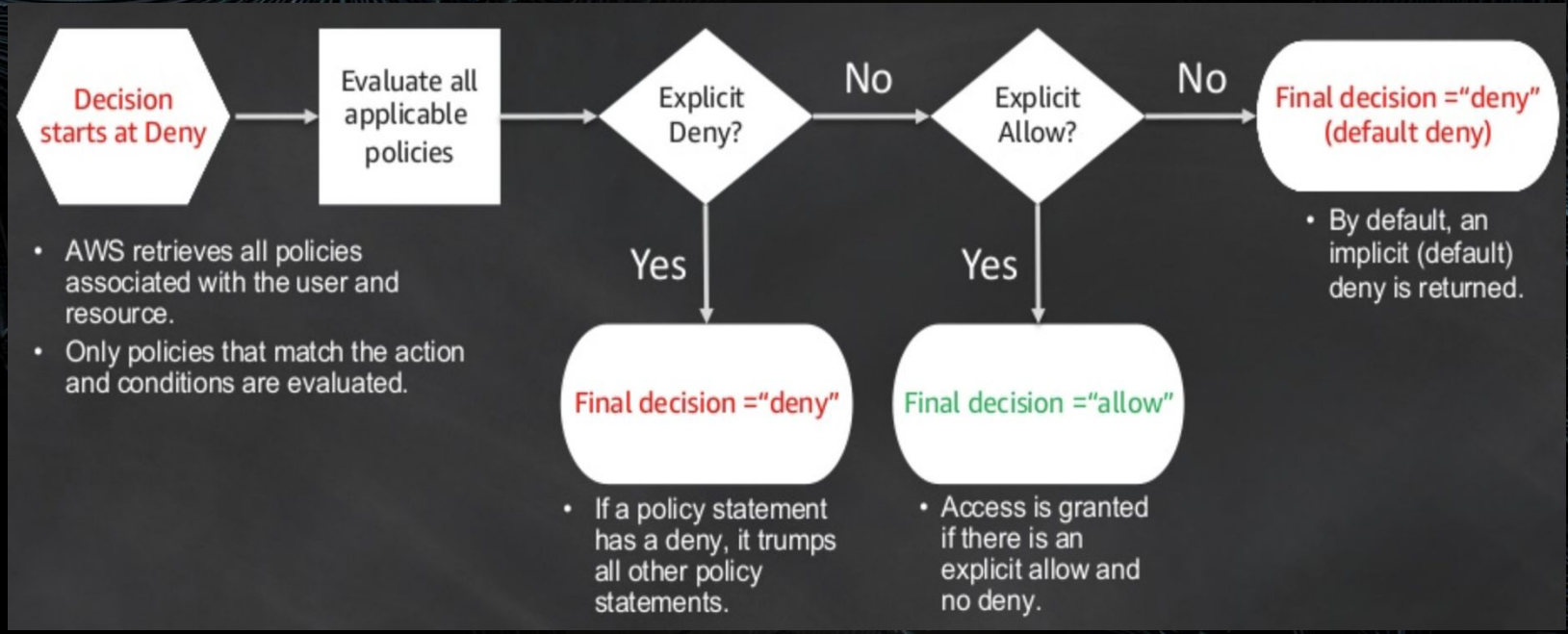
$ AWS IAM - Policies



$ AWS IAM - Identity Based Policies



$ AWS IAM - Policies (Evaluation Logic)



$ AWS IAM - Policies

- IAM Policies follow a PARC Model:

  P - Principal (User, Group or Role. e.g. Prod-ProjectABC123 Group)

  A - Action (e.g. s3:GetObject or ec2:DescribeInstances)

  R - Resource (ARN of EC2 Instance, S3 Bucket. etc.)

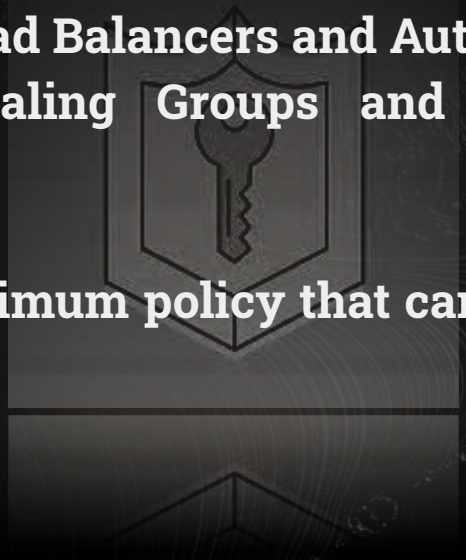
  C - Condition (e.g. ec2:ResourceTag/stack": "dev")

Important Note:
Principal is implicit in Identity Based Policies

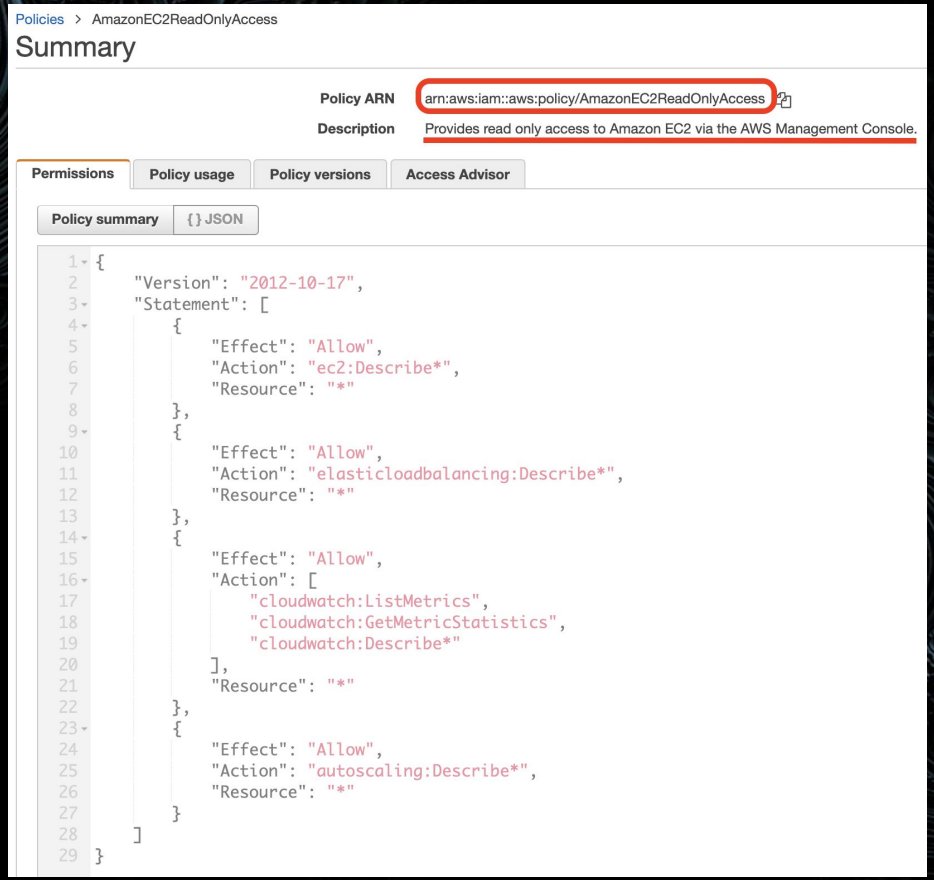
$ AWS IAM - Policies (Scenario #1)

- A group of CloudOps engineers from the ABC company are seeking access to identify and create inventory of AWS EC2 instances attached to all the Load Balancers and AutoScaling Groups to remove unnecessary Autoscaling Groups and unused Load Balancers.

What is the bare minimum policy that can help them to achieve this ?



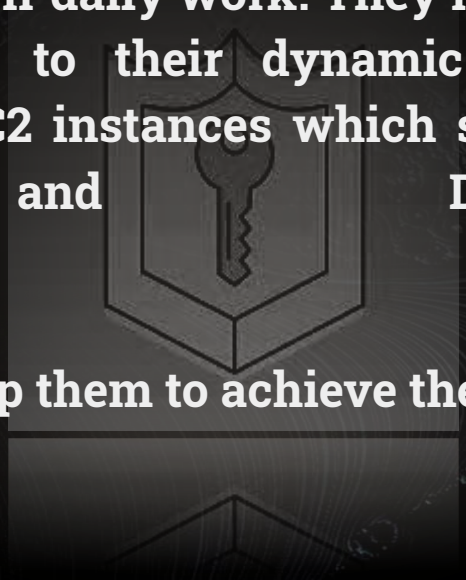
$ AWS IAM - AWS Managed Policy(Scenario #1)



$ AWS IAM - Policies (Scenario #2)

- A team from Data Analytics BU is utilizing on-demand as well as spot instances for their daily work. They might need to start and stop instances according to their dynamic requirements. They are seeking access to EC2 instances which specifically belong to their Project and Department ?

  Which policy can help them to achieve the same ?



$ AWS IAM - Customer Managed Policy

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "StartStopIfTags",
            "Effect": "Allow",
            "Action": [
                "ec2:StartInstances",
                "ec2:StopInstances",
                "ec2:DescribeTags"
            ],
            "Resource": "arn:aws:ec2:region:account-id:instance/*",
            "Condition": {
                "StringEquals": {
                    "ec2:ResourceTag/Project": "DataAnalytics",
                    "aws:PrincipalTag/Department": "Data"
                }
            }
        }
    ]
}
```

$ AWS IAM - Policies (Scenario #3)

- An application running on an AWS EC2 instance needs access to database to perform required tasks. The application must retrieve database credentials from Secrets Manager to achieve the same. The AWS EC2 instance must do it automatically and should only use the current version of the secret.

How this can be achieved and what will be the policy for this task ?

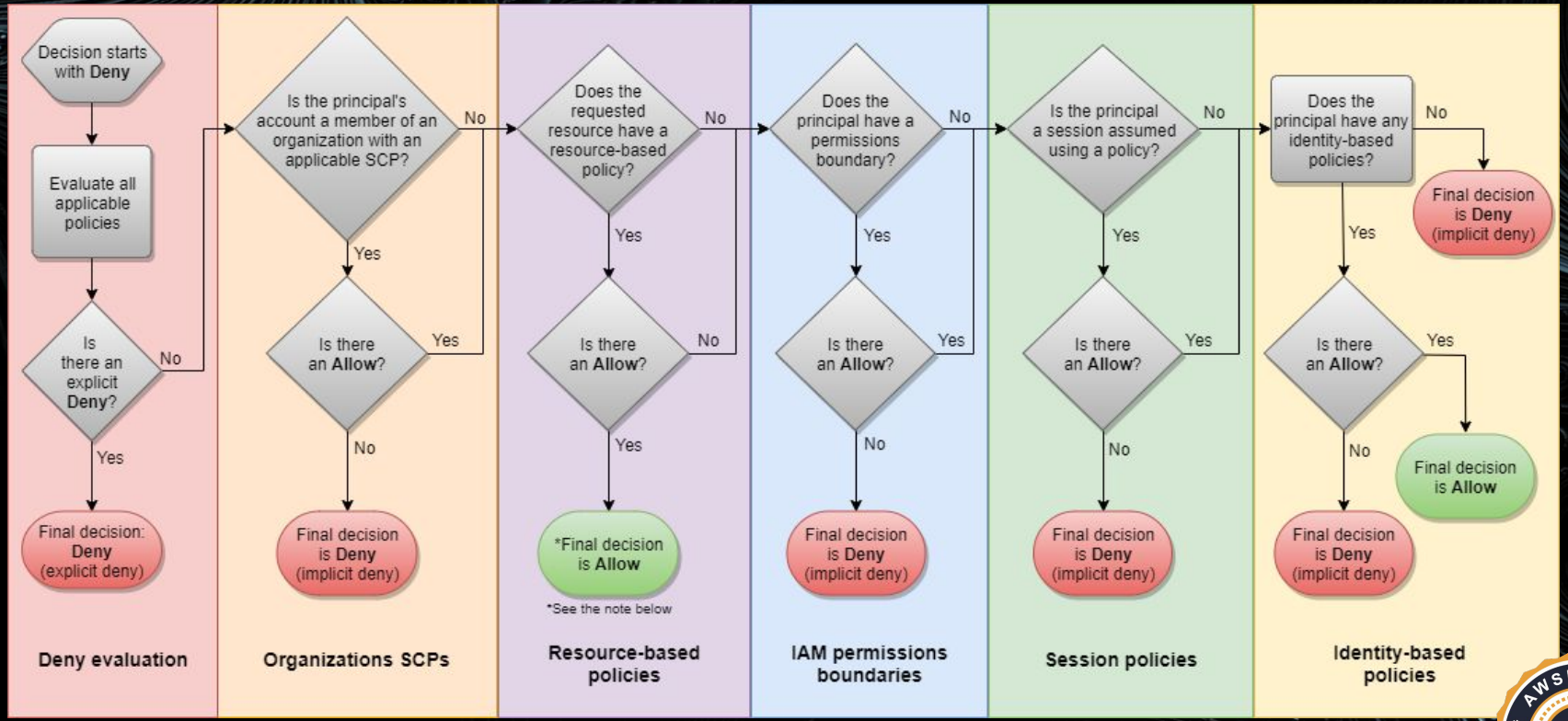
$ AWS IAM - Identity Based Policy(Method#1)

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
           "Sid" : "Stmt1DescribeSecret",
           "Effect": "Allow",
           "Action": [ "secretsmanager:DescribeSecret" ],
           "Resource": "arn:aws:secretsmanager:<region>:<account_id>:secret:TestEnv/*"
        },
        {
            "Sid" : "Stmt2GetSecretValue",
            "Effect": "Allow",
            "Action": [ "secretsmanager:GetSecretValue" ],
            "Resource": "arn:aws:secretsmanager:<region>:<account_id>:secret:TestEnv/*",
            "Condition" : {
                "ForAnyValue:StringLike" : {
                    "secretsmanager:VersionStage" : "AWSCURRENT"
                }
            }
        }
    ]
}
```

$ AWS IAM - Resource Based Policy(Method#2)

```
{
  "Version" : "2012-10-17",
  "Statement" : [
    {
      "Effect": "Allow",
      "Principal": {"AWS": "arn:aws:iam::123456789012:role/EC2RoleToAccessSecrets"},
      "Action": "secretsmanager:GetSecretValue",
      "Resource": "*",
      "Condition": {
        "ForAnyValue:StringEquals": {
          "secretsmanager:VersionStage" : "AWSCURRENT"
        }
      }
    }
  ]
}
```

$ AWS IAM Policies (Full-Evaluation Logic)

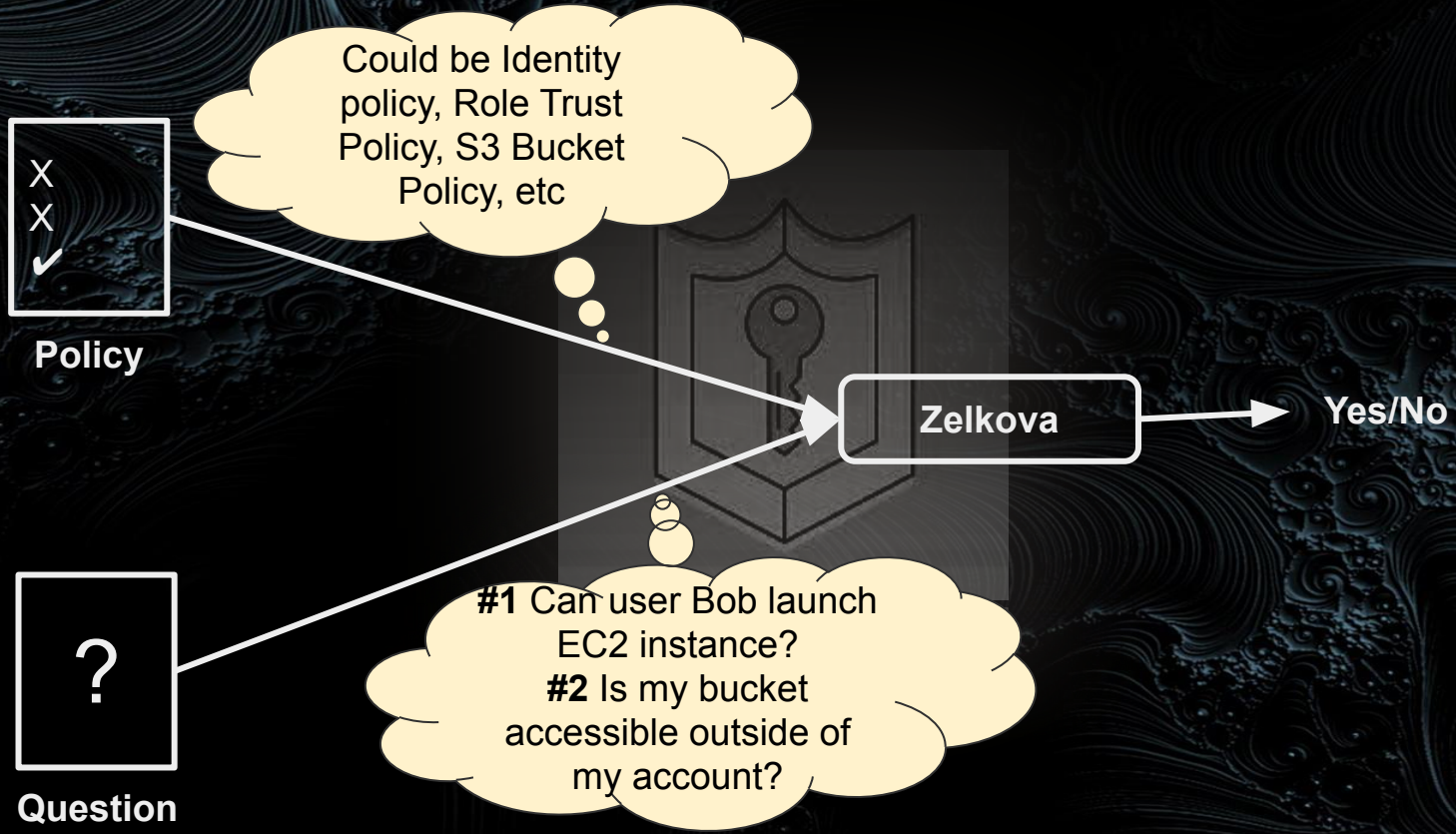


(Source: IAM Documentation)

$ AWS IAM Access Analyzer - Introduction

- Helps to identify the resources in the organization and account (zone of trust), such as S3 Buckets or IAM roles, that are shared with an external entity.
- Any access to resources by principals that are within your zone of trust is considered trusted. Once enabled, Access Analyzer analyzes the policies applied to all of the supported resources in your zone of trust and generates “Findings” with necessary details.
- Onne can review findings to determine whether the access is intended and safe, or the access is unintended and a security risk

$ AWS IAM Access Analyzer - How it works...



(Source: Intro to IAM Access Analyzer, AWS Online Talks)

$ AWS IAM Access Analyzer - Hands On

Demo Time...

- How to enable and setup IAM Access Analyzer ?
- Scenario #1 - Resource Scanning and Findings
- Scenario #2 - Creation Time Policy Evaluation (Latest Feature released in March 2021)

Thank You...

For queries feel free to connect with AWS Delhi User Group at:

1. AWS Delhi User Group (MeetUp) - Learn Together, Grow Together
2. AWS User Group Delhi NCR (Follow us on LinkedIn Page)