



AWS IAM





Table of Contents

- ▶ Introduction to IAM
- ▶ IAM - Users
- ▶ IAM - Policies
- ▶ IAM - User Groups
- ▶ IAM - Roles



1

Introduction to IAM

What Is IAM?



IAM = Intity & Access Management

Authentication

Prove your identity

- Username + Password + {MFA}
- or
- Access Key + Secret Key
- or
- Access Key + Secret Key + Session Token

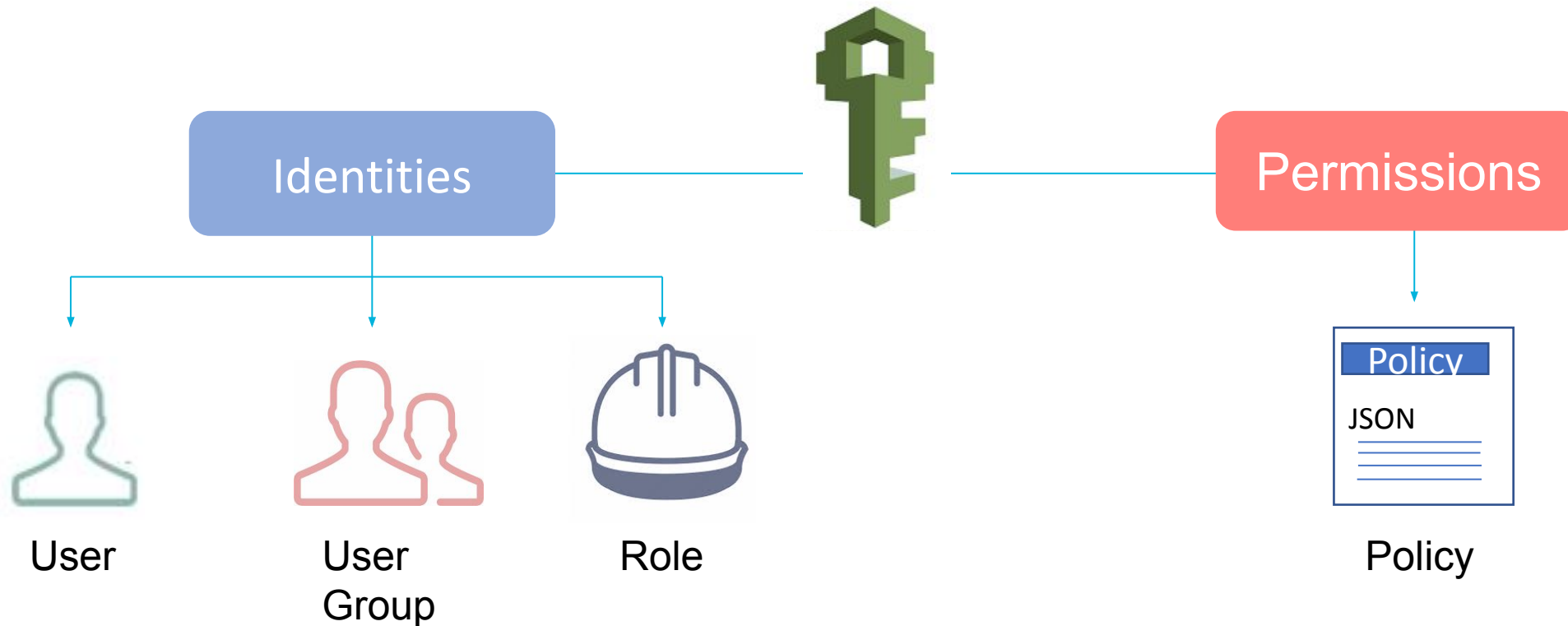
Authorization

Permission to access resources

- IAM Policies
- and/or
- Resource Policies

Introduction to IAM

Categorizing IAM Components



- IAM components can be mainly categorized under two terms; **Identities** and **Permissions**.



2

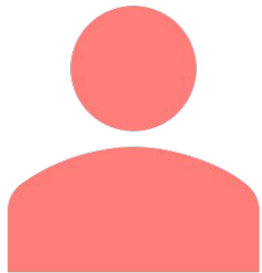
IAM Users

IAM Users



What is IAM User?

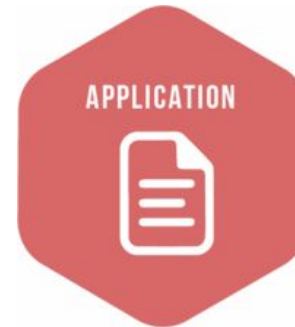
(IAM) user is an entity that you create in AWS to represent the person or application that uses it to interact with AWS



Real person



Software



Web Application

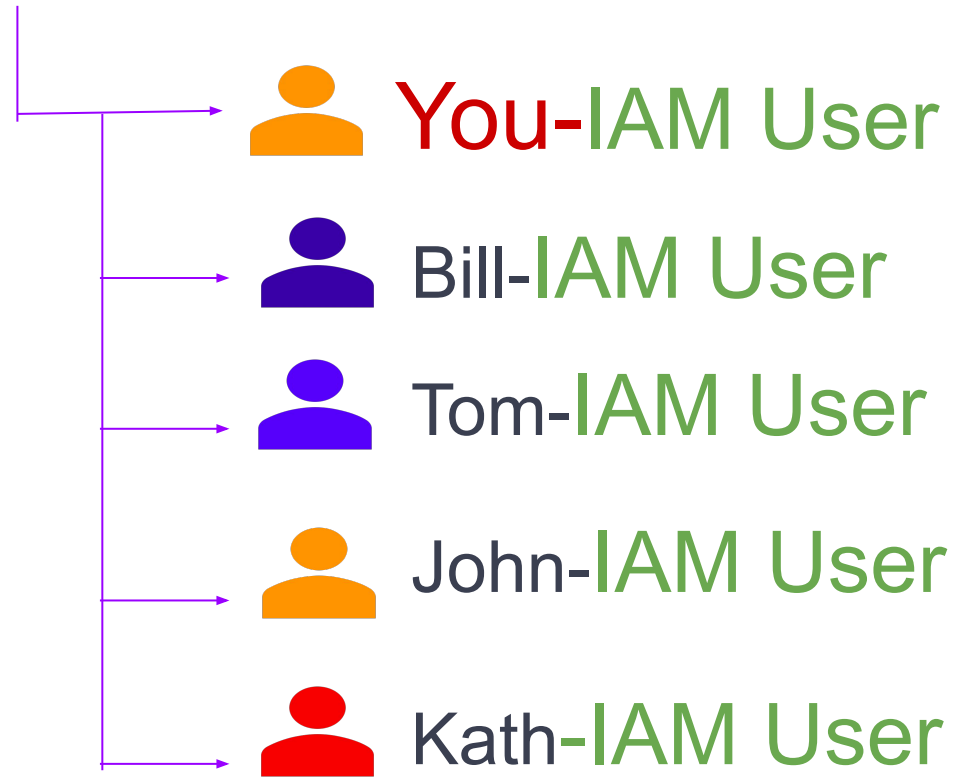


Services Account

IAM Users

What is Root User and IAM User.

AWS Account Owner - Root User (You)



- Root User is a special user
- Username is **email** used to create account
- Generally, **cannot limit permissions** of Root User
- **Cannot delete** Root User
- Best practices:
 - **Enable MFA** for Root User
 - Don't user Root User for **day-to-day work**
 - Keep **password** in a secure location

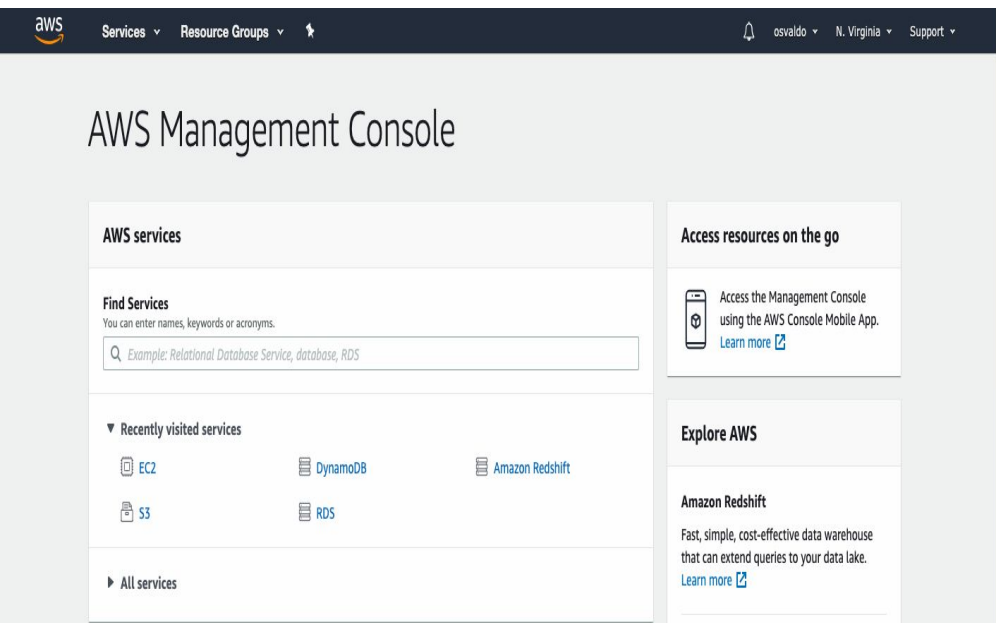
IAM Users

Access Types



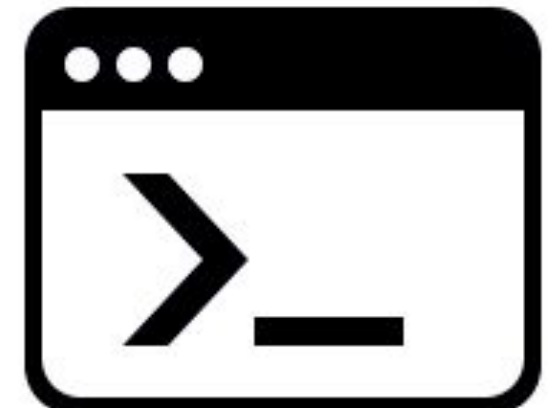
AWS Management Console

Programmatic Access



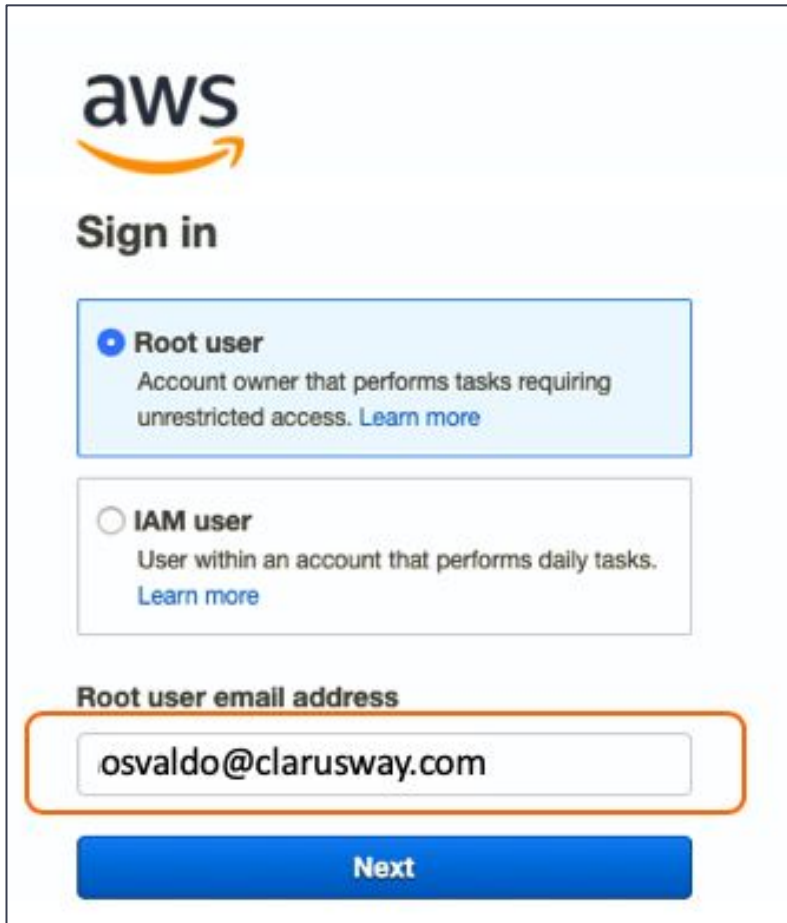
ROOT USER

IAM USER.

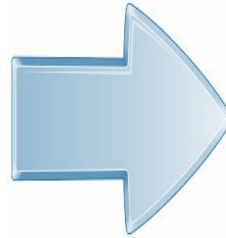


IAM Users

Sign in with Root User- AWS Management Console Access



The screenshot shows the AWS sign-in page. At the top is the AWS logo and the text "Sign in". Below this are two radio button options: "Root user" (selected) and "IAM user". The "Root user" option includes a description: "Account owner that performs tasks requiring unrestricted access. [Learn more](#)". The "IAM user" option includes a description: "User within an account that performs daily tasks. [Learn more](#)". Below these options is a section titled "Root user email address" with a text input field containing "osvaldo@clarusway.com". An orange border highlights this input field. At the bottom is a blue "Next" button.



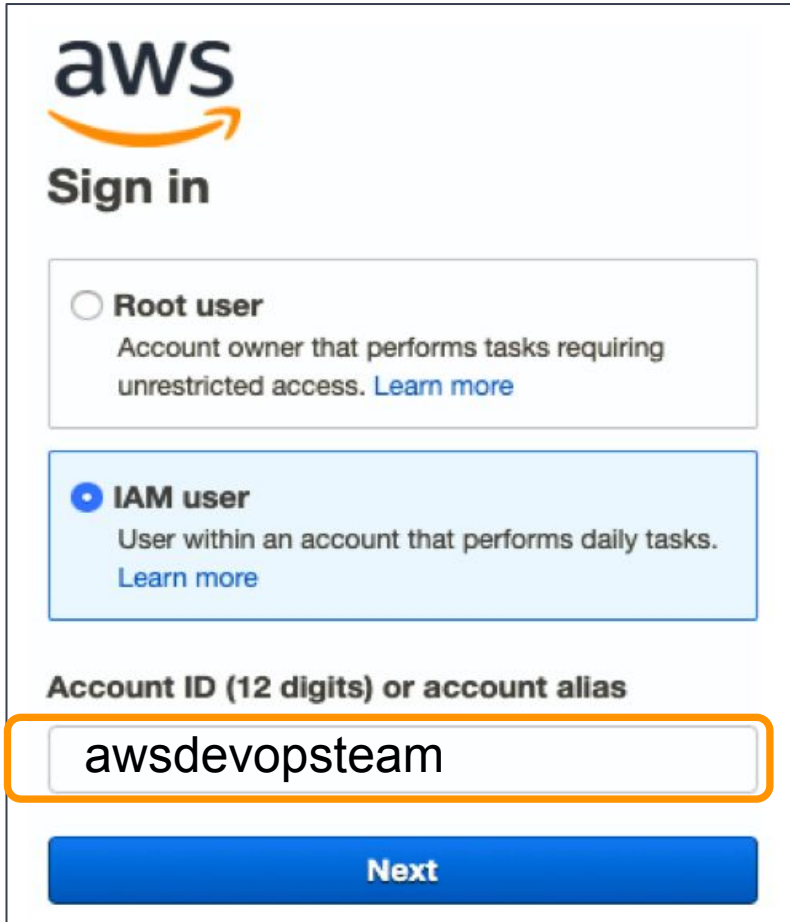
E-mail
Password



The screenshot shows the AWS "Root user sign in" page. At the top is the AWS logo and the text "Root user sign in ⓘ". Below this is the "Email" field with the value "osvaldo@clarusway.com". The "Password" field is shown with masked characters "....." and a "Forgot password?" link. A blue "Sign in" button is below the password field. At the bottom are two links: "Sign in to a different account" and "Create a new AWS account".

IAM Users

Sign in with IAM User- AWS Management Console Access



aws
Sign in

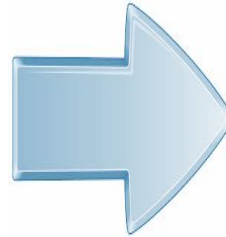
☐ **Root user**
Account owner that performs tasks requiring unrestricted access. [Learn more](#)

☒ **IAM user**
User within an account that performs daily tasks. [Learn more](#)

Account ID (12 digits) or account alias

awsdevopsteam

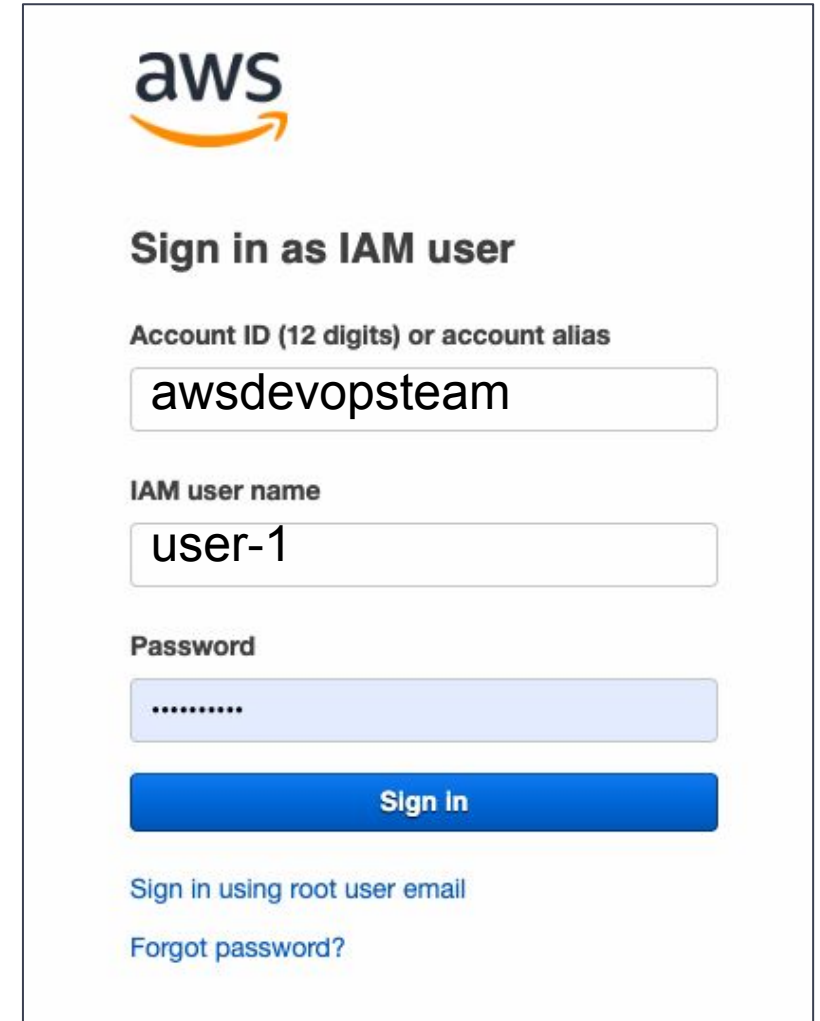
Next



Account ID/Alias

User name

Password



aws

Sign in as IAM user

Account ID (12 digits) or account alias

awsdevopsteam

IAM user name

user-1

Password

.....

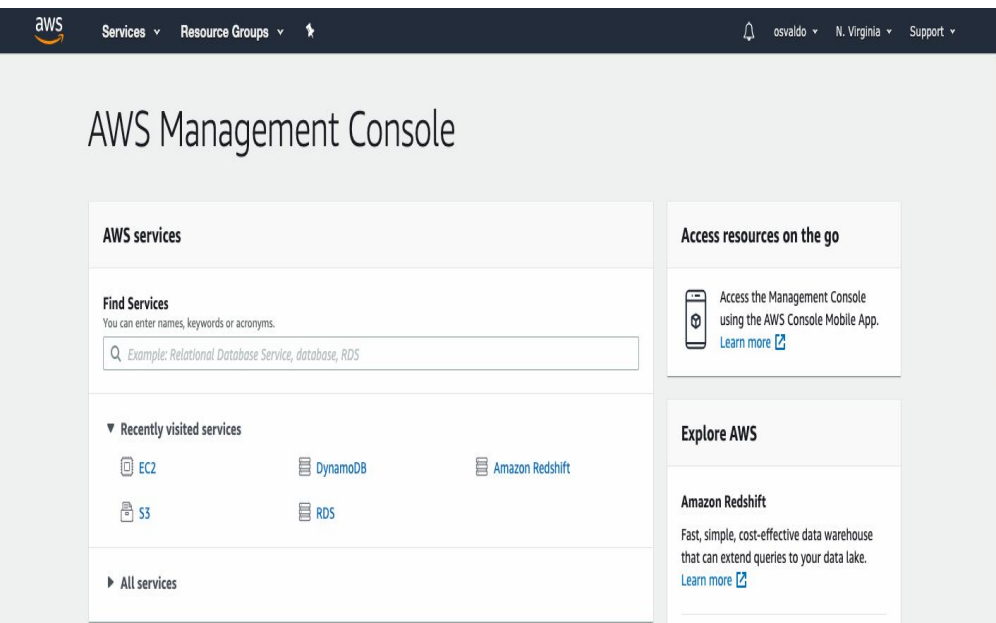
Sign in

[Sign in using root user email](#)

[Forgot password?](#)



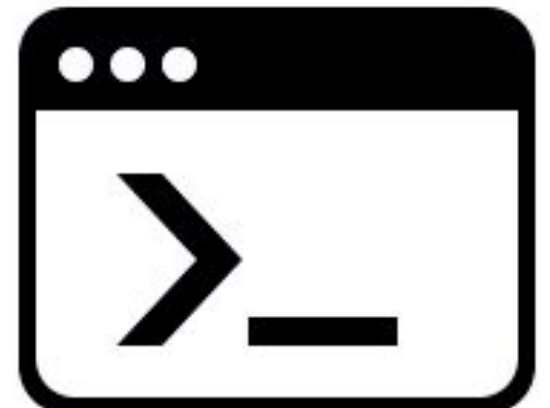
AWS Management Console



ROOT USER

IAM USER.

Programmatic Access



IAM Users

Sign in with IAM User- Programmatic Access

SDKs



Android



iOS



Java



JavaScript



.NET



Node.js



PHP



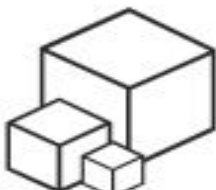
Python (boto)



Ruby



Xamarin



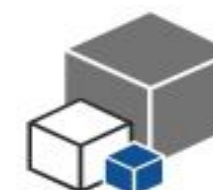
AWS CLI



AWS Toolkit
for Eclipse



AWS Toolkit
for Visual Studio



AWS Tools
for Windows
PowerShell

IAM Users

Sign in with IAM User- Programmatic Access



```
Last login: Fri Apr 24 22:44:56 on ttys000
Clarusway-MacBook-Air:~ user$
```

Access Key ID !!!!!

Secret Access Key !!!!!

****ROOT USER**

IAM USER.



3 IAM Polices

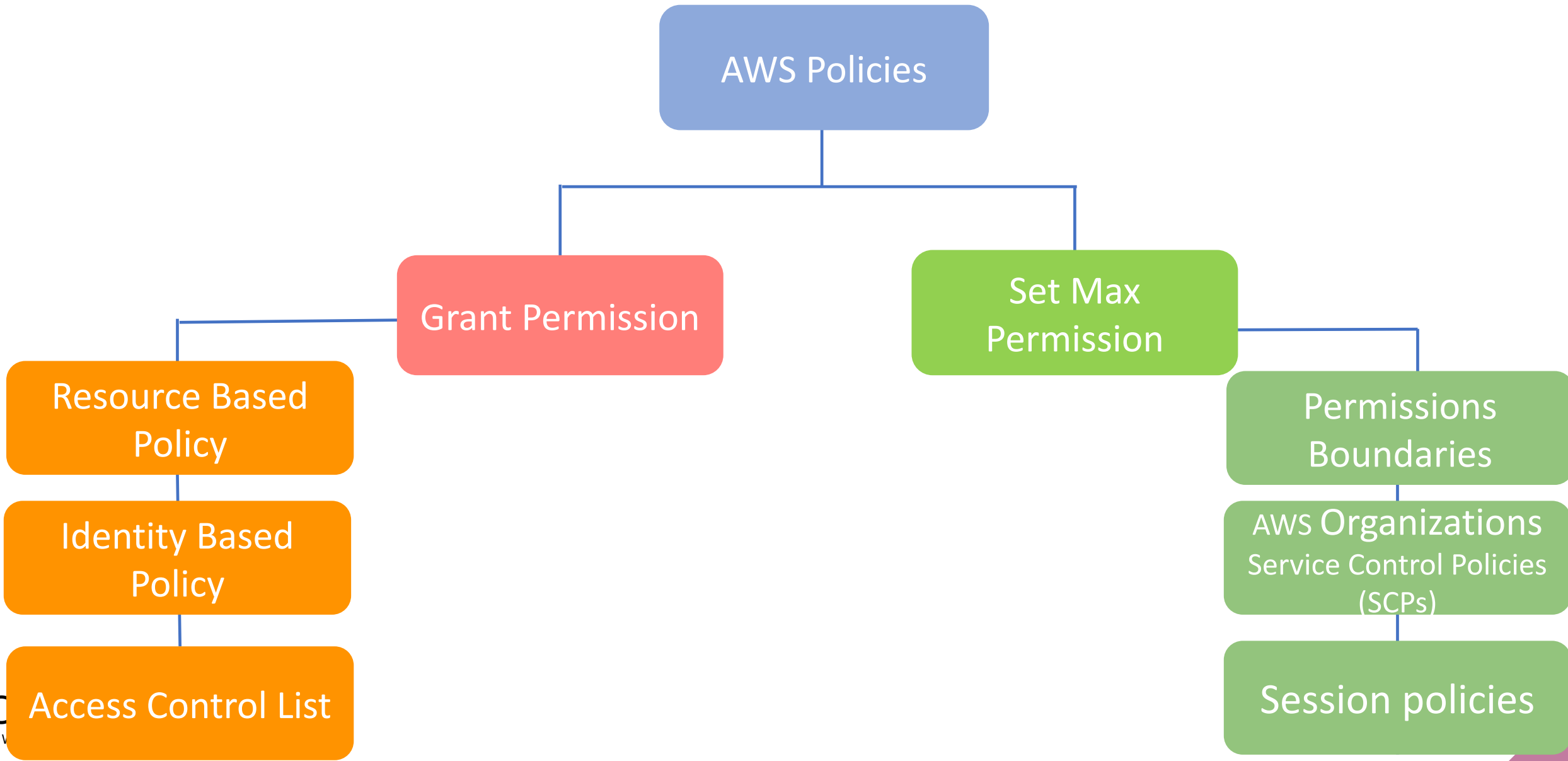
IAM Policies

What is a Policy?



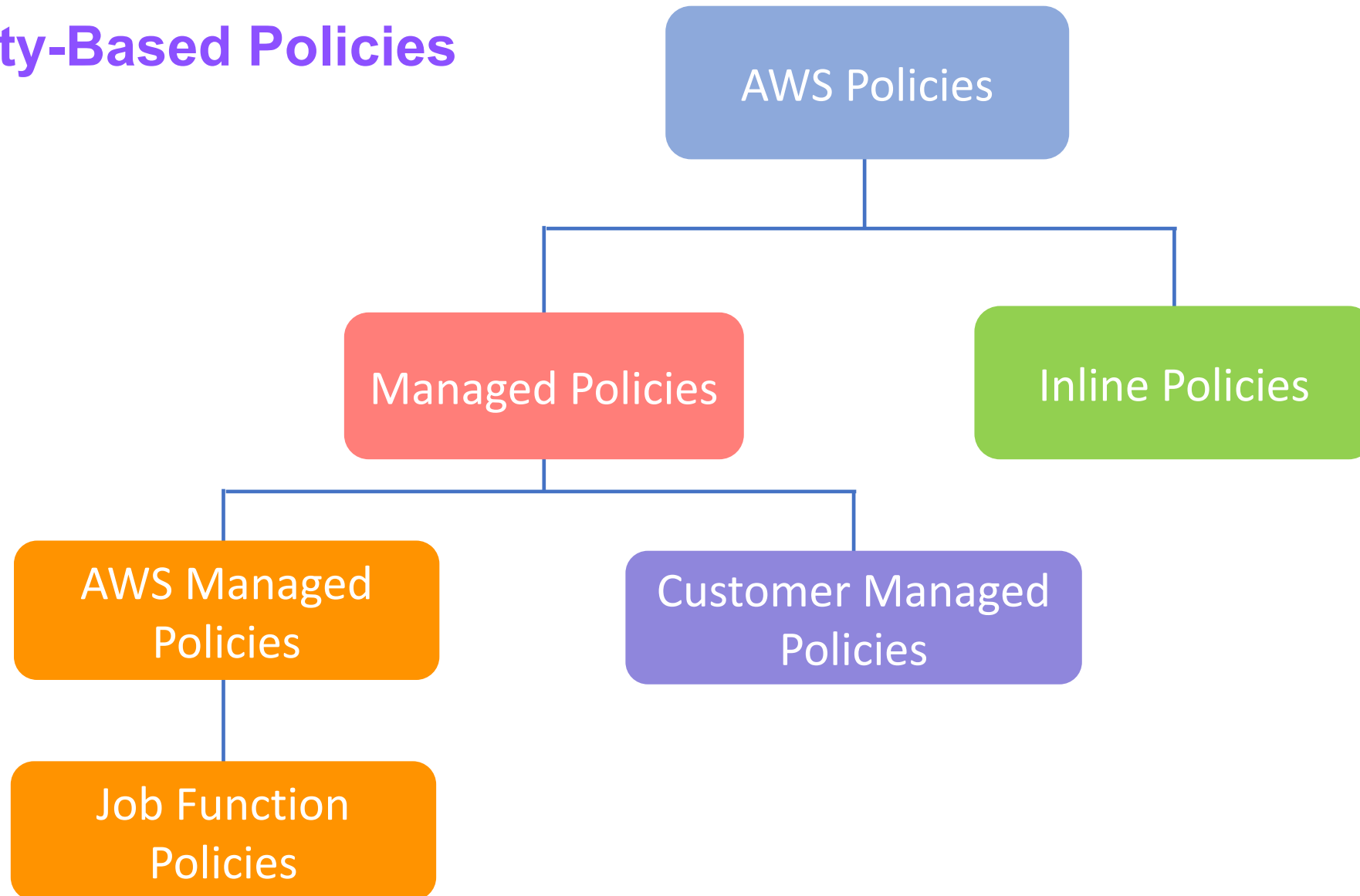
- A policy is an object used to define the **permissions** of an identity or resource in AWS
- Permissions in the policies determine whether the request is **allowed** or **denied**.
- Policies are stored in AWS as **JSON** documents.

Policy Types



IAM Policies

Identity-Based Policies



IAM Policies

Policies - JSON Identifiers

```
1 {  
2   "Version": "2012-10-17",  
3   "Statement": [  
4     {  
5       "Effect": "Allow",  
6       "Action": "*",  
7       "Resource": "*"   
8     }  
9   ]  
10 }
```

Version: Specifies the version of the policy document.

Statement: The basic part of a policy where you define permissions

Effect: It determines what the statement actually does. Can contain only the **Allow** or **Deny** values.

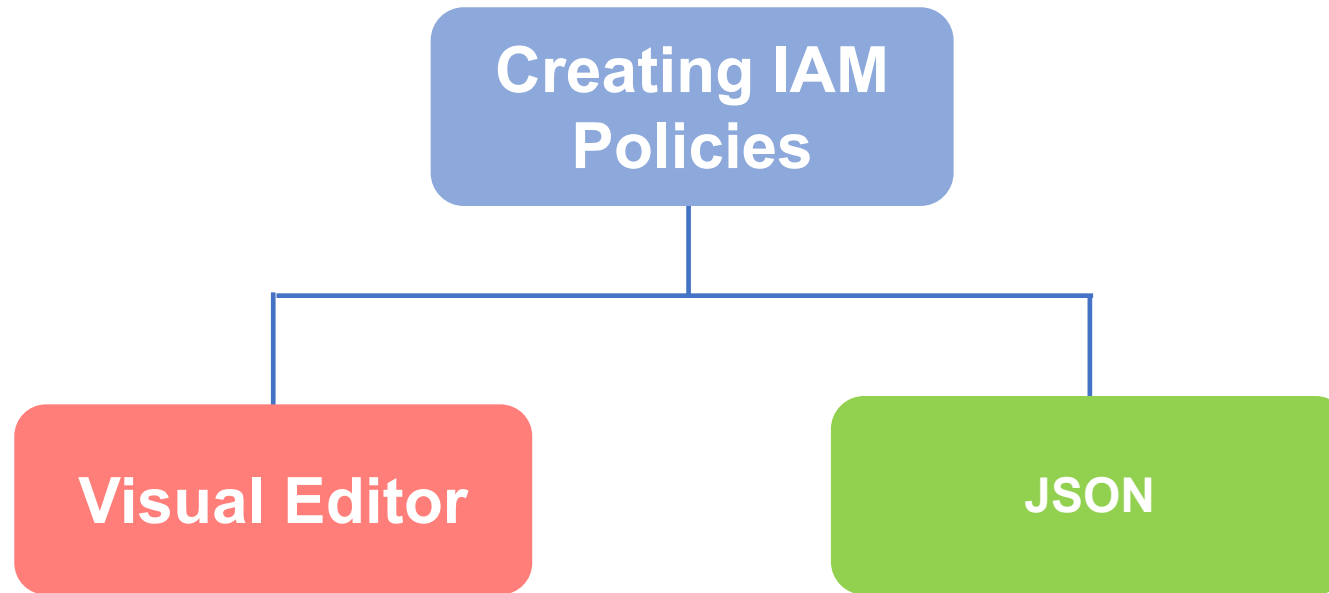
Actions: Determines which actions the identity can perform.

Resource: Explains in which **AWS resources** the statement will perform the operations.



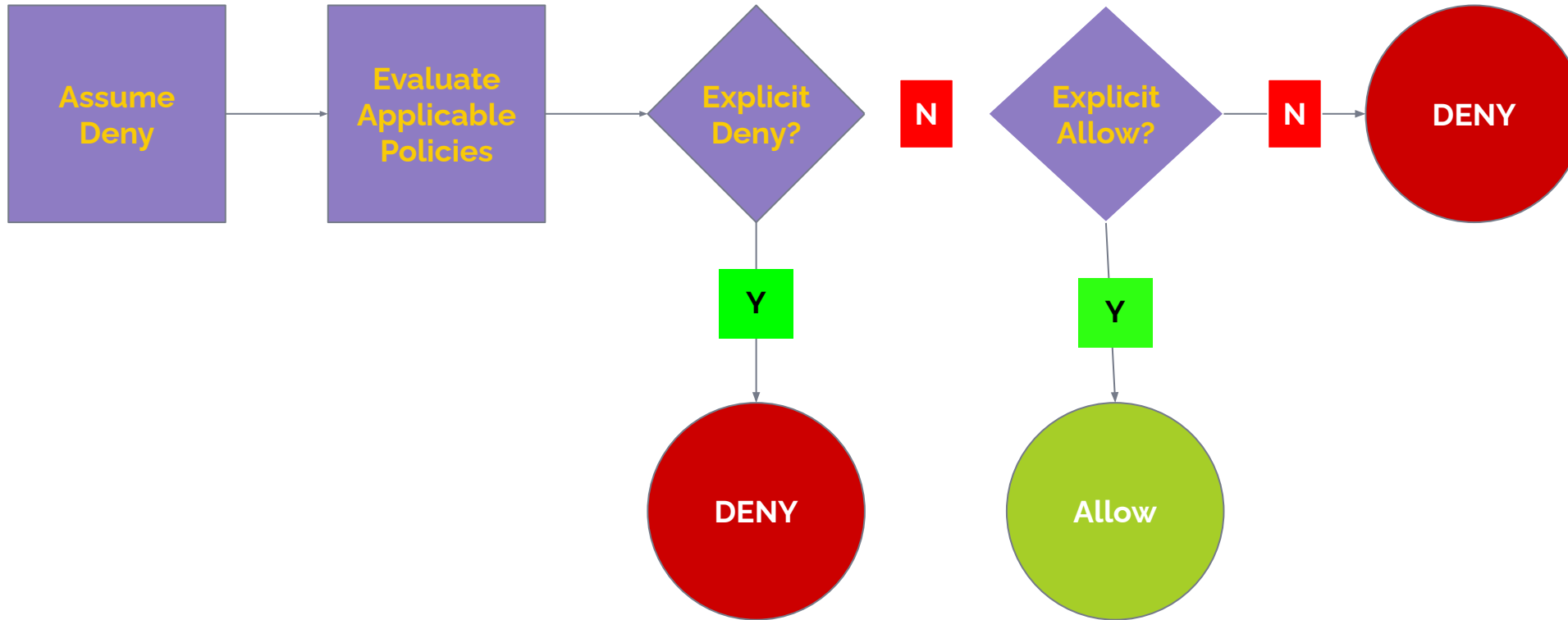
IAM Policies

Creating IAM Policies





Policy Evaluation



- Deny by **default**
- Deny takes **precedence** over allow



4

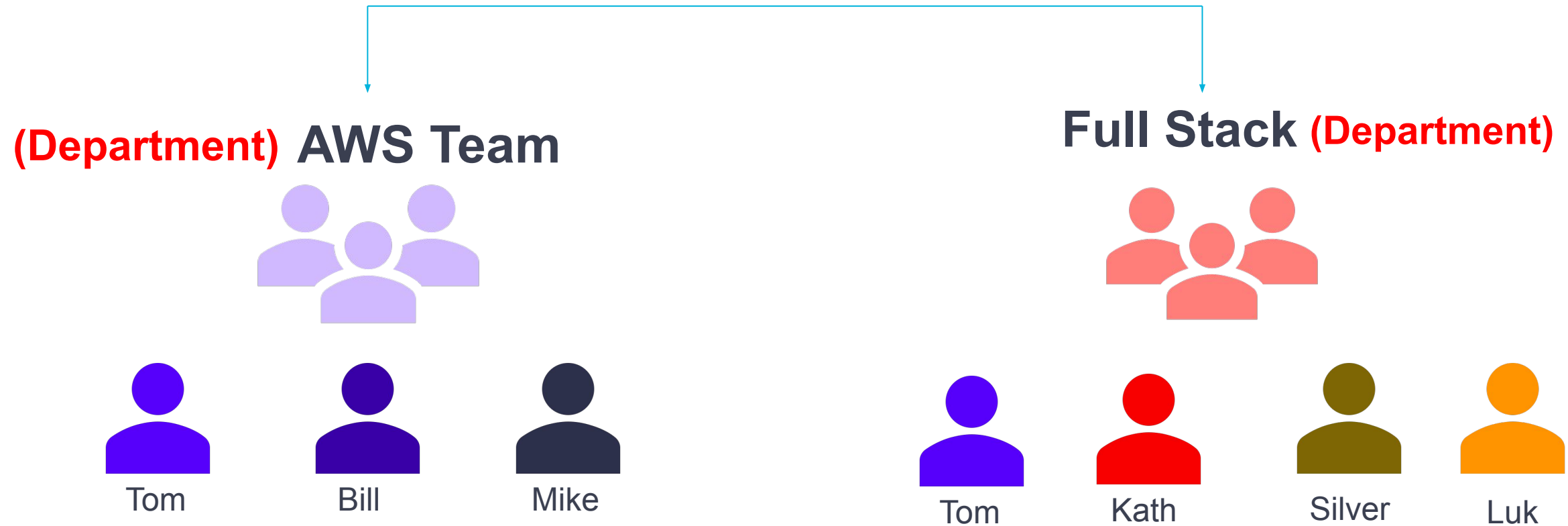
IAM User Groups



IAM User Groups

What is User Group in AWS?

AWS Account (Company)





IAM User Groups

IAM User Group Features

Managed IAM policies can be attached to user groups

Inline IAM policies can be added to user groups

The limit of IAM users in a user group is equal to 5000

User can be a member of 10 different IAM user groups





5

IAM Roles

IAM Roles

What is a Role in AWS?



- The authorization system where we determine how an identity can **access the AWS resources**.
- An IAM role, similar to an IAM user, is an IAM identity that **has specific permissions** that you can create in your account.

IAM Roles

Who can assume an IAM Role?



Another AWS account
Belonging to you or 3rd party



AWS service
EC2, Lambda and others



Web identity
Cognito or any OpenID
provider



SAML 2.0 federation
Your corporate directory



Custom trust policy
Create a custom trust policy to
enable others to perform actions
in this account.



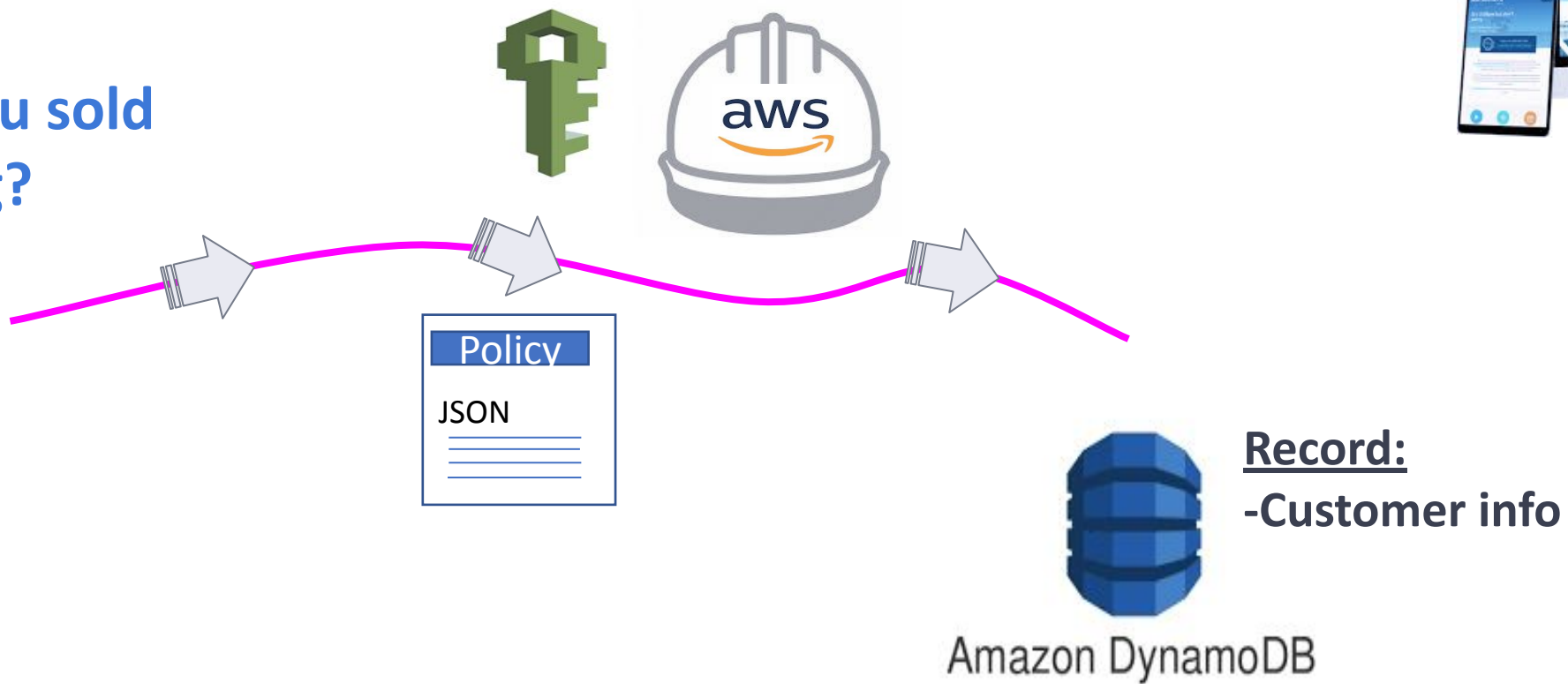
okta

IAM Roles

What does IAM ROLE do ?

www.e-commerce...

What if you sold something?



IAM Roles

Anatomy of a Role



Trusted Entity

Use case

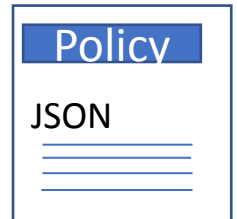
Permission Policy



AWS service
EC2, Lambda and others



EC2



Amazon DynamoDB

Role Credentials



```
aws_access_key_id=ASIA5RBXKVCZWCMV4AFJ
```

```
aws_secret_access_key=23uUyY07I0PKG1URM6iQPV+A8wSsvLEbmHEA37wF
```

```
aws_session_token=IQoJb3JpZ2luX2VjEK//////////wEaCXVzLWVhc3QtMSJHMEUCIGrn7HEV38ejafaba56pEv1UxDIPjFdYLjgLSv0UvpmiA  
iEA4b9Z2Noc0Ah3ru6bogoW+iBRtUrdg05zk7LkM4HQaNsqqgMIFxACGgw5Mjk5NzY0NjE0OTEiDAwgg62YKfWxIzb1TSrvArdvoRgYW4EvWtPAkM9R  
IPk6EpWeHVMbDgVtyk7TGXCRTF6uZpyWSX33QS3Pwvb6d0pwiqomeOFDgG28U82eXrXGoKZnbTmnC+7X0QWgqAUI0Ku2kU/KLLwbLhjpv1Ai/oFpAvG  
0FmZMtVZH+w6/uuyHgZFmPjwgrLTOj0AlnRfA1rjYJm6b2QD6ou5ZMK1JrV/jdW2z0Os7sPVkSA4lH6VPZ2D6vjAnRWDC+0uBV6QUfK1LLeJ1F51bTI  
F3tI2Yu9VnXEV6usAblStCt3NnTpZRnGQTiyUcICLzAiGhJUdZpGQofdLrLEL/MatyglwVA45RpT2MhgH+HPuoIGGT0uISBSt6YQV4/1wf9w2KSIT4U  
dZgaQt8L+TDXiz1/ywn4f11dU0K9vwIINIwp+8s9le7hn1vQPm7HAetLi5mRE30vzXJ6Eoai9RbfgFW7HpxffZLImdOgealQ51w+0Zu7Rx4jGWhWLMo  
WyrJQQw+ZXhgwY6pgESvD6LuI39m2hhJMC3781E8Q4OL+Jn17CysdjNpBH9AjNwGuI9Ad3y3q1u8z1849KzCZCx9GbG/n9YYy3fGnBrrvNY3nrwiA4c  
XKP4KfZU8OIQ3G1LJkK1d24lhhe9UBL3I1ySfMbvDbRoMOXESF6tCpMVLNMa4QaoVY7aThxDvAA6p51pftyPhCK3MJe4qBL4zTC3pXFJe+LPc6uwZ1F  
sL/OTBH
```

Once an entity assumes a role, it receives **temporary credentials** in the form of an **access key**, **secret key** and **session token**.



Note that with an **IAM user**, there is **no session token**, since the credentials are **permanent**



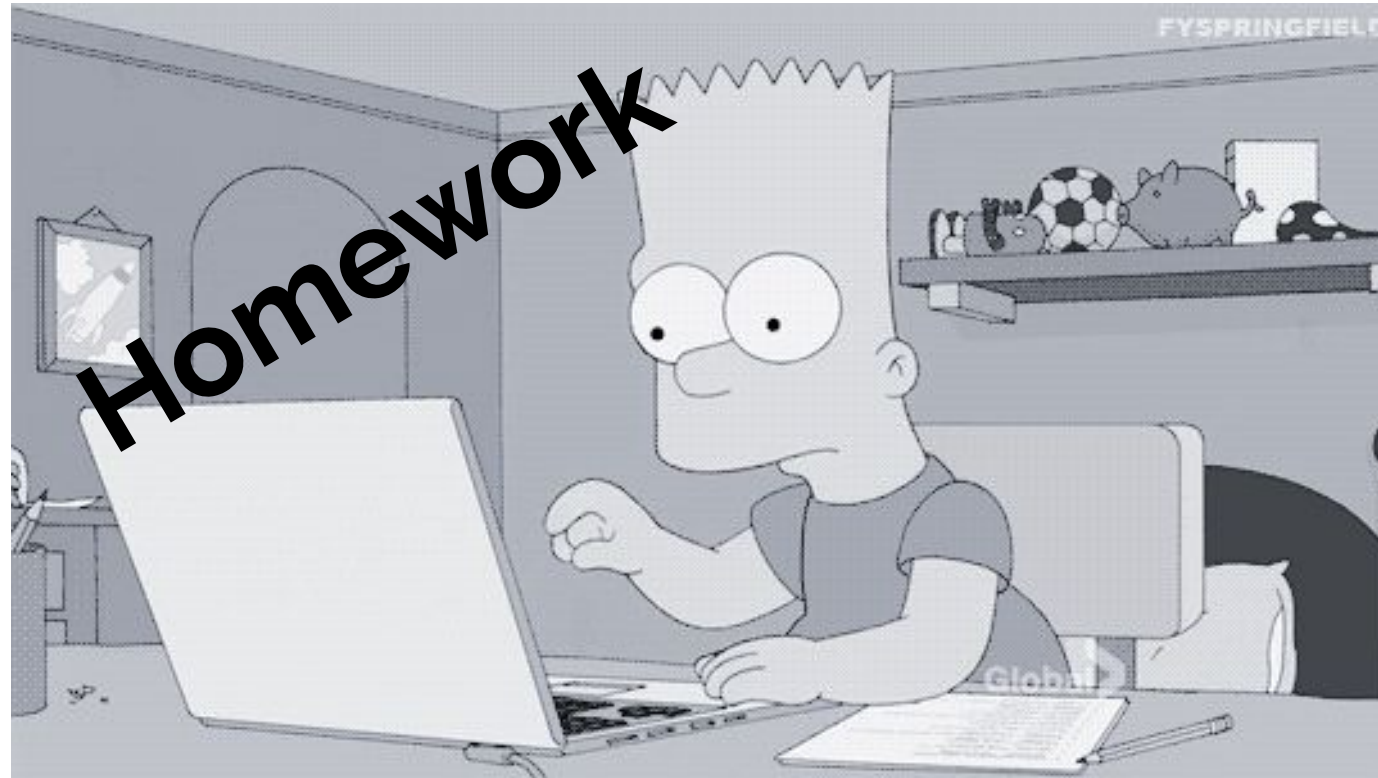
6

Implication of AWS SSO

Implication of AWS SSO- IAM Identity Center



- ▶ Today, most organizations use **AWS SSO to authenticate** users into AWS
- ▶ Users are given permanent credentials to log in to SSO
- ▶ Those permanent credentials (e.g. email & password) enable the **SSO user to assume an IAM** role by way of short term AWS credentials
- ▶ Bottom line:
 - Most organizations today **discourage the use of IAM users**
 - Instead, **roles are used** which map to SSO users
- ▶ **AWS IAM Identity Center** is the updated console for the features of AWS Single Sign-On (AWS SSO)



Video on Multi-Factor Authentication (MFA)

<https://lms.clarusway.com/mod/lesson/view.php?id=7626&pageid=7570>

Let's get our hands dirty!



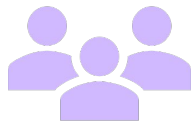
AWS Account owner - Root User (you)



Administrator (you)-Newly Created IAM user

(Department-1)

Database
(RDS FullAccess)



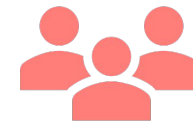
Bill



Tom

(Department-2)

AWS & DevOps
(Power User)



Kath



Tom*



THANKS!

Any questions?

You can find me at:

- ▶ @osvaldo
- ▶ osvaldo@clarusway.com

