

# AWS ORGANIZATION LAB

## Prerequisite:

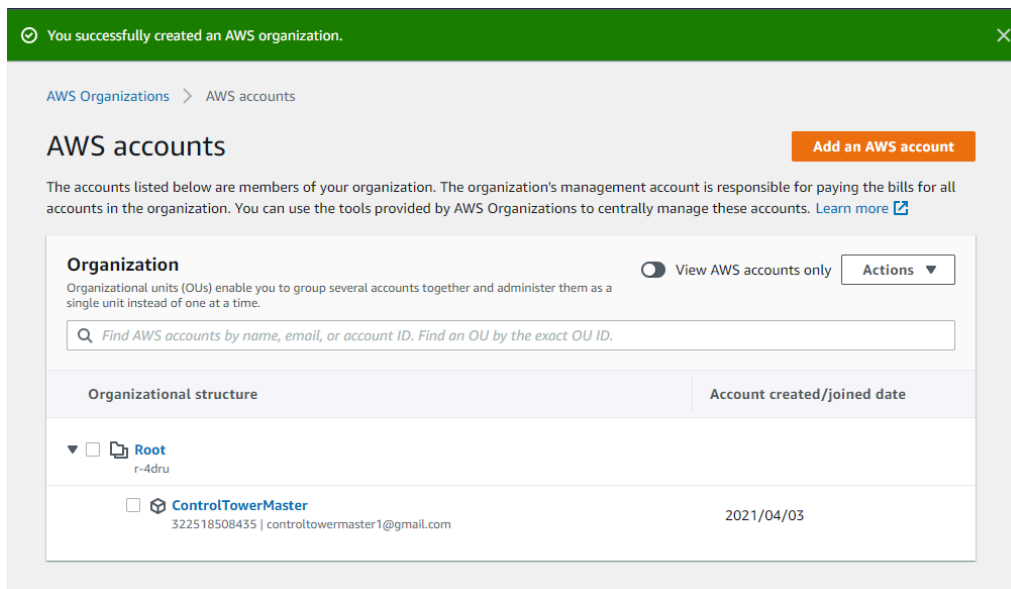
1. Have an AWS account which will be the management account
2. Have an email address not tied to any AWS account. This email will be used to create an AWS account from within the organization

## Tasks:

1. Create an AWS organization with the current account as the management account
2. Create another account from within your organization that will be automatically added to your organization.
3. Use switch role to move from one account to another
4. Use SCPs to establish guardrails/restrictions of what is allowed and denied
5. Test restrictions

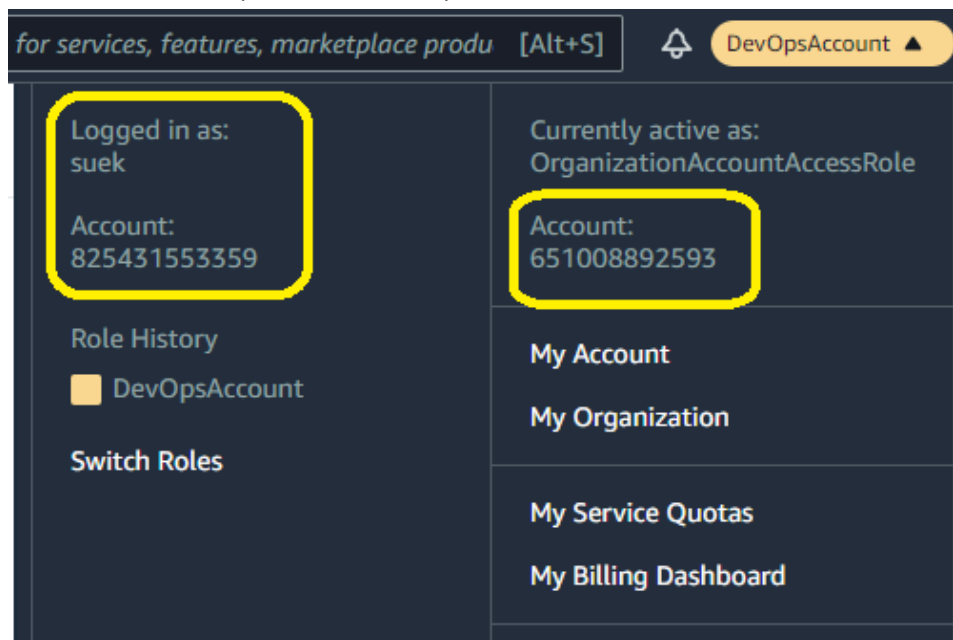
## LAB1: CREATE AN AWS ORGANIZATION

1. Log into the AWS account which will be considered the master account as an IAM user with Admin privileges.
2. Click on 'create an organization' to create an AWS organization in the master account. Once the organization is created you will see the root container with the master account in it



3. After the Organization is created, you should receive an email in the management account to verify your email address before you can invite existing AWS accounts to join your organization.

4. Click on [Add an AWS account](#). *You can either invite an existing account or Create a new account which will automatically be added to the organization.*
  - a. Enter the parameters to create a new account.
    - i. For Account Name use **JJTECH-DevAccount**
    - ii. You can leave the IAM role name blank to automatically use the default role name from AWS (**OrganizationAccountAccessRole**) or you can add your own role name
  - b. Choose [Create AWS account](#) This account will be created for you and automatically added to your organization
5. You can access the newly created account by either using the IAM switch role that was created above or using the root user credentials that was created by the organization.
  - a. **To access with switch role:**
    - i. Open the AWS Management Console using IAM user credentials.
    - ii. Choose your account name at the top of the page, and then select Switch Role.  
Important: If you are signed in with root user credentials, you can't switch roles. You must be signed in as an IAM user or role. For more information, see [Switching to a role \(Console\)](#).
    - iii. Enter the account number and role name for the member account.
    - iv. (Optional) You can also enter a custom display name (maximum 64 characters) and a display color for the member account.
    - v. Choose Switch Role and you will be taken to the new AWS account while logged into the master account. (See screen below).

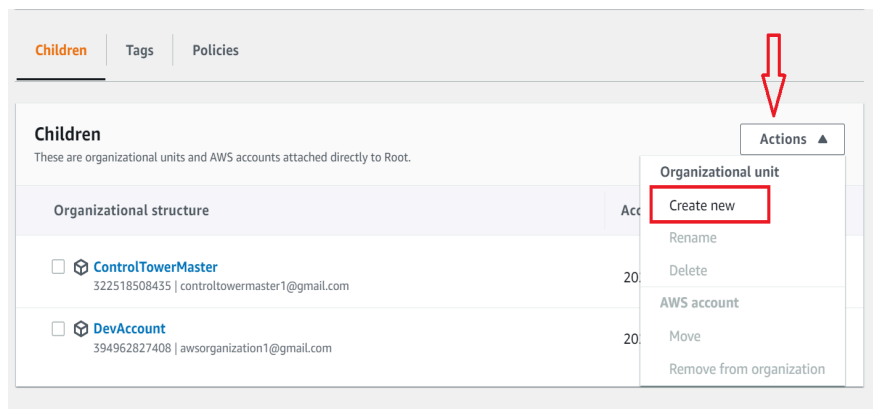


- b. **To retrieve the root user credentials:**

- i. Go to the sign in page of the AWS console and choose [Sign in using root account credentials](#)
- ii. Choose [Forgot your password?](#) and enter the information that is required to reset the password to a new one.
- iii. Check your email and choose the reset password link.

## **LAB2: CREATE ORGANIZATIONAL UNITS**

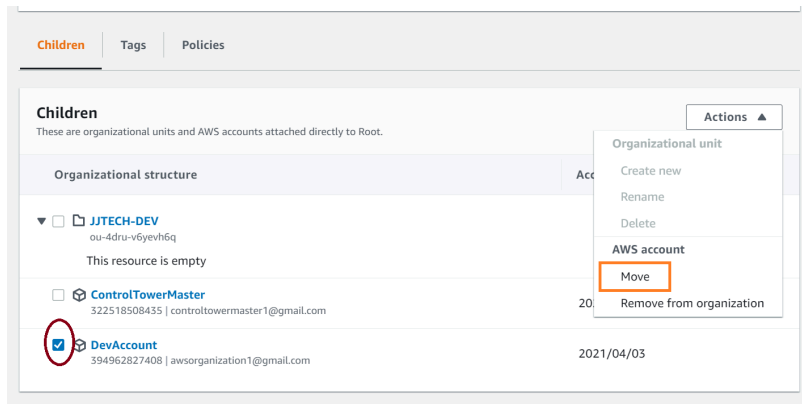
1. Go to the AWS organizations console and navigate to the AWS account page
2. click on the root container, choose actions and then under Organizational unit, choose [create new](#).



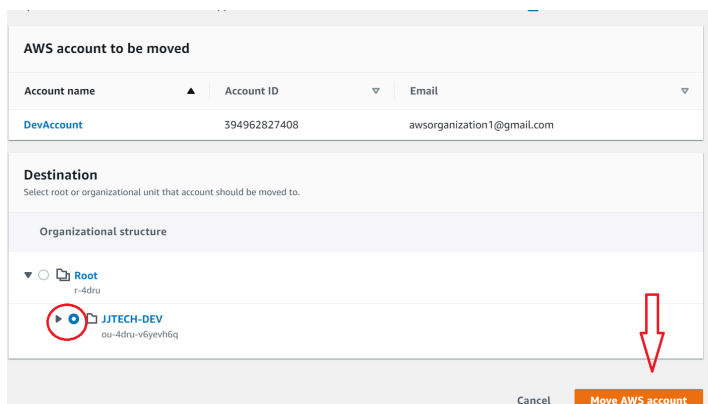
3. Enter the organizational unit name *JJTECH-DEV* and select [Create organizational unit](#).

A screenshot of the 'Create organizational unit in Root' form. The 'Organizational unit name' field is filled with 'JJTECH-DEV' and highlighted with a red box. Below this, there is a 'Tags' section with a red arrow pointing to the 'Create organizational unit' button at the bottom right. The form includes a 'Details' section with a description of organizational units and a 'Tags' section with a description of tags.

4. Select the member you just created from within the AWS organization, choose [Actions](#), and then under AWS account, choose [Move](#).



5. On the *Move AWS account 'DevAccount'* page, choose **JJTECH-DEV** OU and then choose **Move AWS account**.



### **LAB3: ENABLE AND CREATE SERVICE CONTROL POLICIES**

1. Navigate to the policies page, and then choose **Service Control Policies**. On the Service control policies page, choose **Enable service control policies**.

*A green banner appears to inform you that you can now create SCPs in your organization.*

2. Choose create policy to create a policy. Enter **RequiredEC2Tags** as the policy name and description.

Policies > Service control policies > Create policy

permissions. [Learn more](#)

**Policy name \***

RequiredEC2Tags

The policy name can have up to 128 characters.

**Description**

RequiredEC2Tags

The description can have up to 512 characters.

### 3. Next Enter the parameters for your policy statement details

- a. Choose service to add action:
  - i. For services, search **EC2** in the filter
  - ii. For action, search **RunInstances** in the filter
- b. Select Resources within the service:
  - i. Click on **Add resource** and there the below parameters to create an EC2 instance resource. AWS Service = EC2, Resource Type = Instance, Resource ARN = Replace all curly brackets parameters with an asterisk (\*) and it should look like this *arn:aws:ec2:\*:\*:instance/\**
  - ii. Click Add resource

aws Services Search for services, features, marketplace products, and docs [Alt+S] susanne @ 3225-1850-8435

AWS Organizations

Accounts Organize accounts

The new AWS Organizations console experience. We've redesigned the console to make it easier to manage your AWS accounts. [Try out the new console.](#)

Policies > Service control policies

**Add resource**

Specify the resource type and ARN to add for the selected service

AWS Service EC2

Resource type instance

Resource ARN arn:aws:ec2:\*:\*:instance/\*

Replace { placeholders } with your own information.

Add resource

- iii. Repeat step d above for a **volume** resource type. This means we will be enforcing this policy on both instance and volumes.  
(*arn:aws:ec2:\*:\*:volume/\**)
- c. Add condition:
  - i. Click on **Add condition** and select the conditionkey **aws:RequestTag**
  - ii. Tag key = CostCenter, Qualifier = Default, Operator = StringNotEquals, Value = 111,222,333,444
  - iii. Click on Add condition

### Add condition

Specify condition information to add for the selected service. [Learn more](#)

**Condition key\*** aws:RequestTag

**Tag key\*** CostCenter

**Qualifier** Default

**Operator\*** StringNotEquals ☐ If exists

**Value** 111,222,333,444

Add multiple values separated with comma(,)

\* Required

➔ Add condition

4. Click on create policy

AWS Organizations
Accounts   Organize accounts   **Policies**   Invitations   Settings

**1. Choose service to add actions**

for.

EC2

EC2 Auto Scaling

EC2 Image Builder

EC2 Instance Connect

EC2 Messages

2. Add resource

3. Add condition

**Create policy**

```

5      "Sid": "Statement1",
6      "Effect": "Deny",
7      "Action": [
8        "ec2:RunInstances"
9      ],
10     "Resource": [
11       "arn:aws:ec2:*:*:instance/*",
12       "arn:aws:ec2:*:*:volume/*"
13     ],
14     "Condition": {
15       "StringNotEquals": {
16         "aws:RequestTag/CostCenter": [
17           "111",
18           "222",
19           "333",
20           "444"
21         ]
22       }
23     }
24   }
25 }
26
          
```

Add statement
Remove statement

\* Required fields

Cancel
Create policy

➡

5. Select policy - Go to Actions and click on Attach policy

**AWS Organizations** ✕

AWS accounts  
Services  
**Policies**  
Settings

Use the old console

Organization ID  
o-m03074qefa

We've redesigned the AWS Organizations console to make it easier to use. Continue to use the new console and [tell us what you think](#). Or you can [use the old console](#).

AWS Organizations > Policies > Service control policies

## Service control policies

[Disable service control policies](#)

Service control policies (SCPs) enable central administration over the permissions available within the accounts in your organization. This helps ensure that your accounts stay within your organization's access control guidelines. [Learn more](#)

**The service control policy editor is currently available in only the original version of the AWS Organizations console.**  
When you complete your edits, you will automatically return to the new version of the console.

**Available policies**

	Name	Kind	Description	Actions
<input type="checkbox"/>	FullAWSAccess	AWS managed policy	Allows access to every operation	Attach policy Delete policy
<input checked="" type="checkbox"/>	RequiredEC2Tags	Customer managed policy	RequiredEC2Tags	

[Create policy](#)

6. Attach policy to the JJTECH-Dev OU
7. Test policy by logging in as a root user into a member account in the JJTECH-Dev OU, Go to any region and test the following
  - a. Create an EC2 instance with no tag (this will fail)
  - b. Create an EC2 instance with a tag key and no value (this will fail)
  - c. Create an EC2 instance with tag key cost-center and value 11(this will fail)
  - d. Create an EC2 instance with tag key CostCenter and value 111 ( this should work)

### Create other SCPs with other policies below and test the effectiveness

1. Prevent users from deleting VPC flow logs and Cloudwatch log groups

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Deny",
      "Action": [
        "ec2:DeleteFlowLogs",
        "logs:DeleteLogGroup",
        "logs:DeleteLogStream"
      ],
      "Resource": "*"
    }
  ]
}
```

```
    }
  ]
}
```

## 2. Prevent any VPC that doest already have internet access from getting it

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Deny",
      "Action": [
        "ec2:AttachInternetGateway",
        "ec2:CreateInternetGateway",
        "ec2:CreateEgressOnlyInternetGateway",
        "ec2:CreateVpcPeeringConnection",
        "ec2:AcceptVpcPeeringConnection",
        "globalaccelerator:Create*",
        "globalaccelerator:Update*"
      ],
      "Resource": "*"
    }
  ]
}
```