

# AWS PROJECT 3

## Provision EC2 instance with Lambda

### Create an IAM Policy

The screenshot shows the AWS IAM Management Console 'Create policy' page. The 'JSON' tab is selected, displaying the following policy document:

```
1 {
2   "Version": "2012-10-17",
3   "Statement": [
4     {
5       "Effect": "Allow",
6       "Action": [
7         "ec2:Describe*",
8         "ec2:CreateKeyPair",
9         "ec2:CreateSecurityGroup",
10        "ec2:AuthorizeSecurityGroupIngress",
11        "ec2:AuthorizeSecurityGroupEgress",
12        "ec2:CreateTags",
13        "ec2:DescribeTags",
14        "ec2:RunInstances"
15      ],
16      "Resource": "*",
17      "Condition": {
```

Character count: 327 of 6,144.

Buttons: Cancel, Review policy

The screenshot shows the 'Review policy' step of the AWS IAM Management Console 'Create policy' page. The policy name is 'mypolicy'.

**Name\*** mypolicy  
Use alphanumeric and '+', '@', '\_' characters. Maximum 128 characters.

**Description**  
Maximum 1000 characters. Use alphanumeric and '+', '@', '\_' characters.

**Summary**  
This policy defines some actions, resources, or conditions that do not provide permissions. To grant access, policies must have an action that has an applicable resource or condition. For details, choose **Show remaining**. [Learn more](#)

**Filter**

Service	Access level	Resource	Request condition
Allow (1 of 264 services) <a href="#">Show remaining 263</a>			
EC2	Limited: Write, Tagging	All resources	ec2:Region = us-east-1

# Create an IAM Role.

The screenshot shows the 'Create role' page in the AWS IAM Management Console. The browser address bar shows the URL: `console.aws.amazon.com/iam/home?region=us-east-1#/roles$new?step=type`. The page has a progress bar with four steps: 1 (selected), 2, 3, and 4. The main heading is 'Create role'. Below it is the section 'Select type of trusted entity'. There are four options: 'AWS service' (selected), 'Another AWS account', 'Web identity', and 'SAML 2.0 federation'. The 'AWS service' option is highlighted with a blue border and includes the text 'Allows AWS services to perform actions on your behalf. [Learn more](#)'. Below this is the section 'Choose a use case'. Under 'Common use cases', 'EC2' and 'Lambda' are listed. 'Lambda' is selected and highlighted with a blue background. Below this is the section 'Or select a service to view its use cases', which lists various AWS services in a grid: API Gateway, CloudWatch Events, EKS, IoT Things Graph, Redshift, AWS Backup, CodeBuild, EMR, KMS, Rekognition, AWS Chatbot, CodeDeploy, ElastiCache, Kinesis, RoboMaker, AWS Marketplace, CodeGuru, Elastic Beanstalk, Lake Formation, and S3. At the bottom right, there are 'Cancel' and 'Next: Permissions' buttons. The footer shows the AWS logo, 'Feedback', 'English (US)', and copyright information.

1 2 3 4

## Create role

Select type of trusted entity

**AWS service**  
EC2, Lambda and others

**Another AWS account**  
Belonging to you or 3rd party

**Web identity**  
Cognito or any OpenID provider

**SAML 2.0 federation**  
Your corporate directory

Allows AWS services to perform actions on your behalf. [Learn more](#)

Choose a use case

**Common use cases**

**EC2**  
Allows EC2 instances to call AWS services on your behalf.

**Lambda**  
Allows Lambda functions to call AWS services on your behalf.

Or select a service to view its use cases

<a href="#">API Gateway</a>	<a href="#">CloudWatch Events</a>	<a href="#">EKS</a>	<a href="#">IoT Things Graph</a>	<a href="#">Redshift</a>
<a href="#">AWS Backup</a>	<a href="#">CodeBuild</a>	<a href="#">EMR</a>	<a href="#">KMS</a>	<a href="#">Rekognition</a>
<a href="#">AWS Chatbot</a>	<a href="#">CodeDeploy</a>	<a href="#">ElastiCache</a>	<a href="#">Kinesis</a>	<a href="#">RoboMaker</a>
<a href="#">AWS Marketplace</a>	<a href="#">CodeGuru</a>	<a href="#">Elastic Beanstalk</a>	<a href="#">Lake Formation</a>	<a href="#">S3</a>

\* Required

Cancel Next: Permissions

The screenshot shows the 'Create role' page in the AWS IAM Management Console, Step 2: Attach permissions policies. The browser address bar shows the URL: `console.aws.amazon.com/iam/home?region=us-east-1#/roles$new?step=permissions&commonUseCase=Lambda%2BLambda&selectedUseCase=Lambda`. The page has a progress bar with four steps: 1, 2 (selected), 3, and 4. The main heading is 'Create role'. Below it is the section 'Attach permissions policies'. There is a 'Create policy' button. Below this is a search bar with the text 'Filter policies' and a search input field containing 'mypolicy'. The search results show 'Showing 1 result'. Below this is a table with two columns: 'Policy name' and 'Used as'. The table has one row with the policy name 'mypolicy' and 'Used as' set to 'None'. At the bottom right, there are 'Cancel', 'Previous', and 'Next: Tags' buttons. The footer shows the AWS logo, 'Feedback', 'English (US)', and copyright information.

1 2 3 4

## Create role

Attach permissions policies

Choose one or more policies to attach to your new role.

Create policy

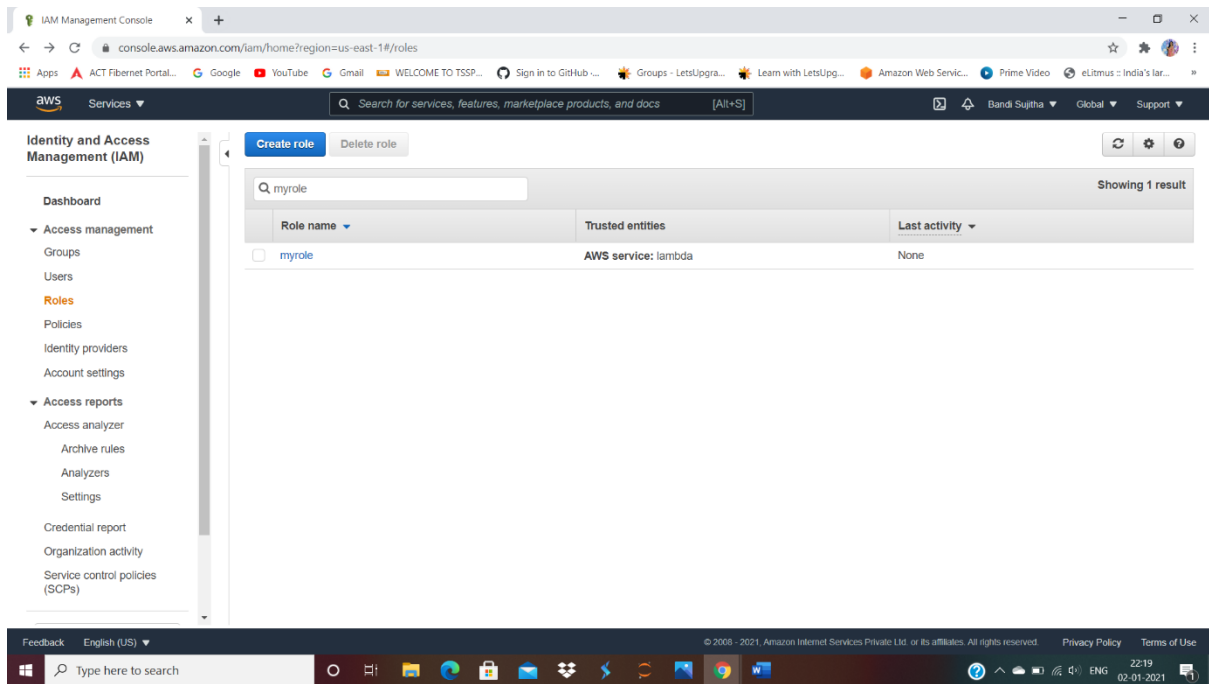
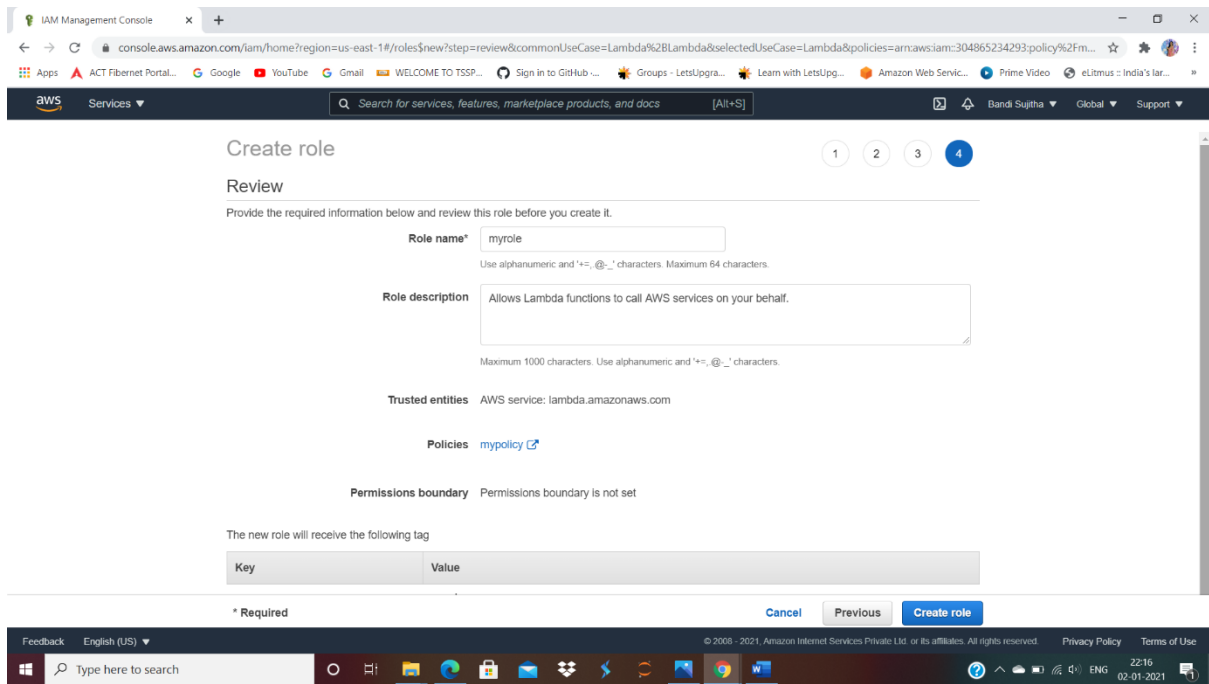
Filter policies  Showing 1 result

Policy name	Used as
<input checked="" type="checkbox"/> mypolicy	None

Set permissions boundary

\* Required

Cancel Previous Next: Tags



# Creation of a Lambda Function

The screenshot shows the 'Create function' page in the AWS console. The page title is 'Create function' with an 'Info' link. Below the title, it says 'Choose one of the following options to create your function.' There are four options: 'Author from scratch' (selected with a blue dot), 'Use a blueprint', 'Container image', and 'Browse serverless app repository'. The 'Author from scratch' option has a subtext: 'Start with a simple Hello World example.' Below these options is the 'Basic information' section. It contains a 'Function name' field with the value 'myEC2LambdaFunction', a 'Runtime' dropdown menu set to 'Python 3.8', and a 'Permissions' section with a link to 'Change default execution role'. The bottom of the page shows the AWS navigation bar and a Windows taskbar.

Lambda

console.aws.amazon.com/lambda/home?region=us-east-1#/create/function

Services Search for services, features, marketplace products, and docs [Alt+S]

Bandi Sujitha N. Virginia Support

## Create function [Info](#)

Choose one of the following options to create your function.

**Author from scratch** [Info](#)

Start with a simple Hello World example.

**Use a blueprint**

Build a Lambda application from sample code and configuration presets for common use cases.

**Container image**

Select a container image to deploy for your function.

**Browse serverless app repository**

Deploy a sample Lambda application from the AWS Serverless Application Repository.

### Basic information

**Function name**  
Enter a name that describes the purpose of your function.

myEC2LambdaFunction

Use only letters, numbers, hyphens, or underscores with no spaces.

**Runtime** [Info](#)  
Choose the language to use to write your function.

Python 3.8

**Permissions** [Info](#)  
By default, Lambda will create an execution role with permissions to upload logs to Amazon CloudWatch Logs. You can customize this default role later when adding triggers.

[Change default execution role](#)

Feedback English (US) © 2008 - 2021, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. Privacy Policy Terms of Use

Type here to search

This screenshot shows the 'Create function' page with the 'Change default execution role' section expanded. The 'Function name' field still contains 'myEC2LambdaFunction' and the 'Runtime' is 'Python 3.8'. The 'Permissions' section now shows three radio button options: 'Create a new role with basic Lambda permissions', 'Use an existing role' (selected), and 'Create a new role from AWS policy templates'. Below these, the 'Existing role' section shows a dropdown menu with 'myrole' selected and a 'View the myrole role on the IAM console.' link. At the bottom right, there are 'Cancel' and 'Create function' buttons. The rest of the page layout, including the navigation bar and taskbar, remains the same.

Lambda

console.aws.amazon.com/lambda/home?region=us-east-1#/create/function

Services Search for services, features, marketplace products, and docs [Alt+S]

Bandi Sujitha N. Virginia Support

## Create function [Info](#)

Choose one of the following options to create your function.

**Author from scratch** [Info](#)

Start with a simple Hello World example.

**Use a blueprint**

Build a Lambda application from sample code and configuration presets for common use cases.

**Container image**

Select a container image to deploy for your function.

**Browse serverless app repository**

Deploy a sample Lambda application from the AWS Serverless Application Repository.

### Basic information

**Function name**  
Enter a name that describes the purpose of your function.

myEC2LambdaFunction

Use only letters, numbers, hyphens, or underscores with no spaces.

**Runtime** [Info](#)  
Choose the language to use to write your function.

Python 3.8

**Permissions** [Info](#)  
By default, Lambda will create an execution role with permissions to upload logs to Amazon CloudWatch Logs. You can customize this default role later when adding triggers.

**Change default execution role**

**Execution role**  
Choose a role that defines the permissions of your function. To create a custom role, go to the [IAM console](#).

☐ Create a new role with basic Lambda permissions

☒ Use an existing role

☐ Create a new role from AWS policy templates

**Existing role**  
Choose an existing role that you've created to be used with this Lambda function. The role must have permission to upload logs to Amazon CloudWatch Logs.

myrole

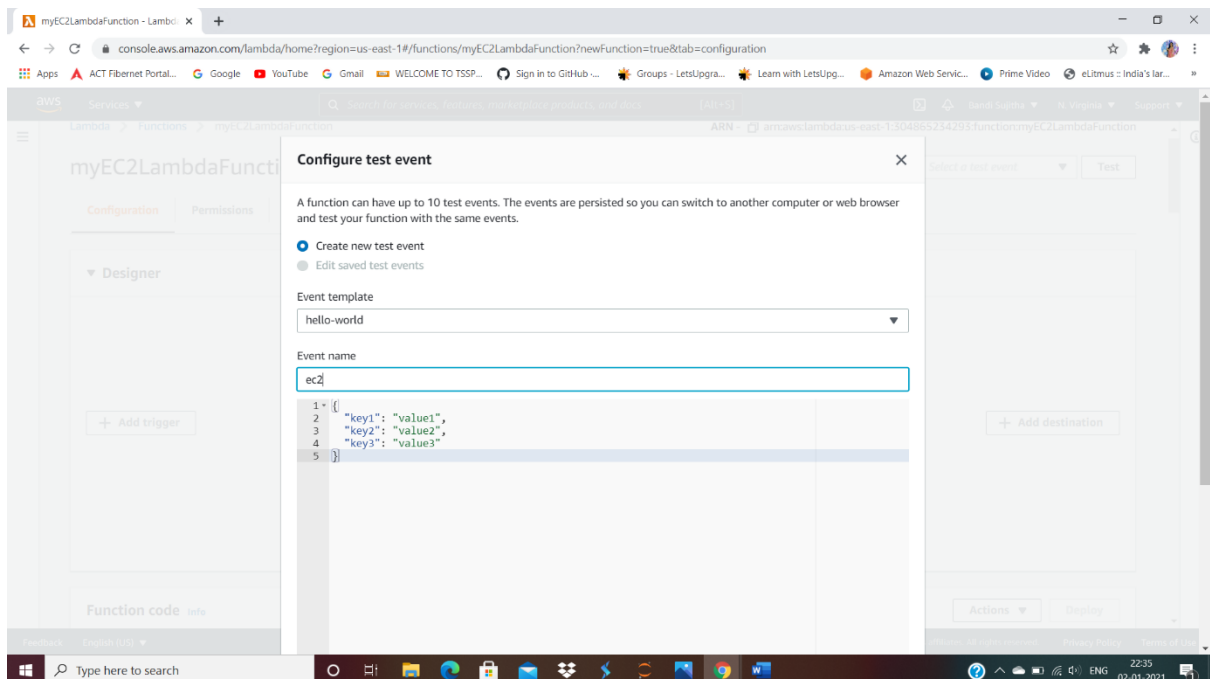
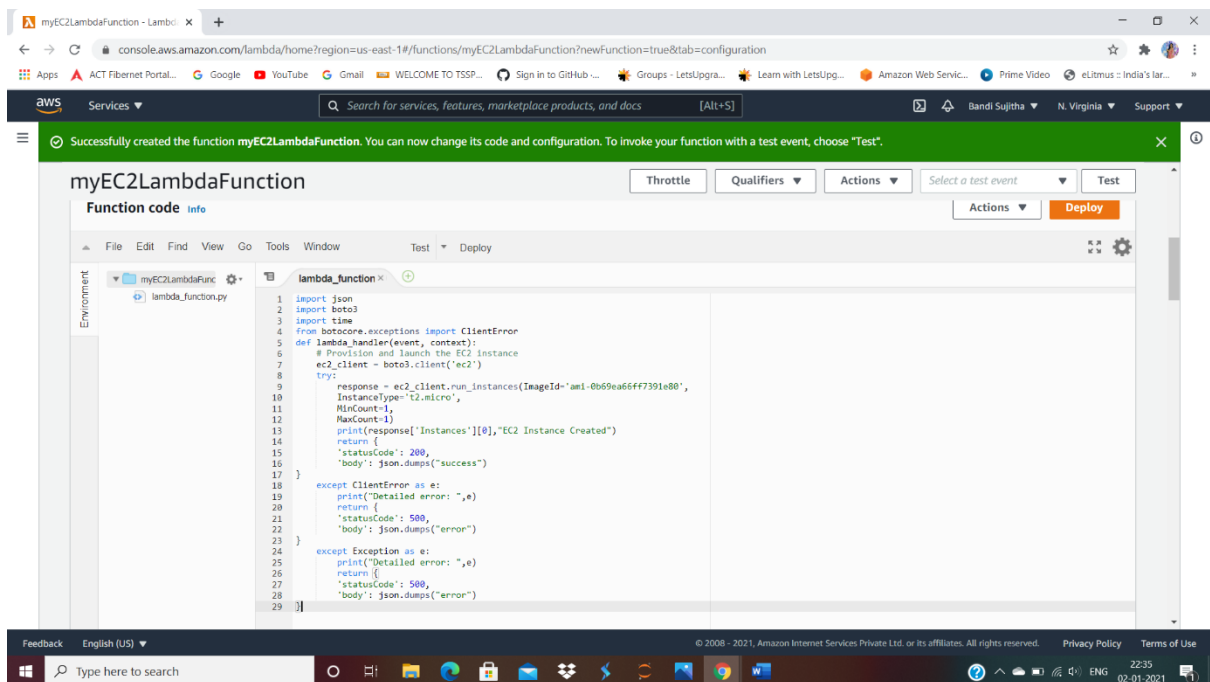
[View the myrole role on the IAM console.](#)

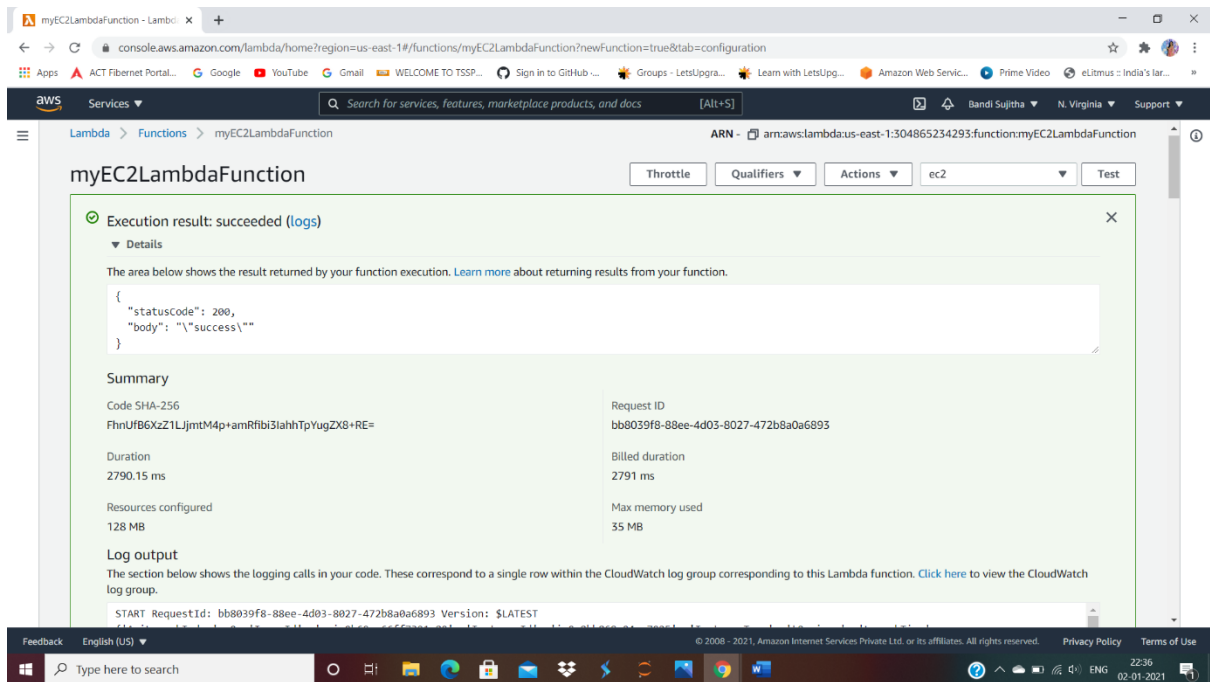
[Advanced settings](#)

Cancel Create function

Feedback English (US) © 2008 - 2021, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. Privacy Policy Terms of Use

Type here to search





## Details of instance that is created while testing Lamda function

