Assignment-5
API Gayeway

Task 1 - Set Up AWS SAM Project and Define Infrastructure
AWS SAM simplifies the deployment of serverless applications. In this task, you will initialize a new AWS SAM project and define the infrastructure needed for the lab.
Initialize the AWS SAM Project

- Open a terminal and run the following command to create a new AWS SAM project:

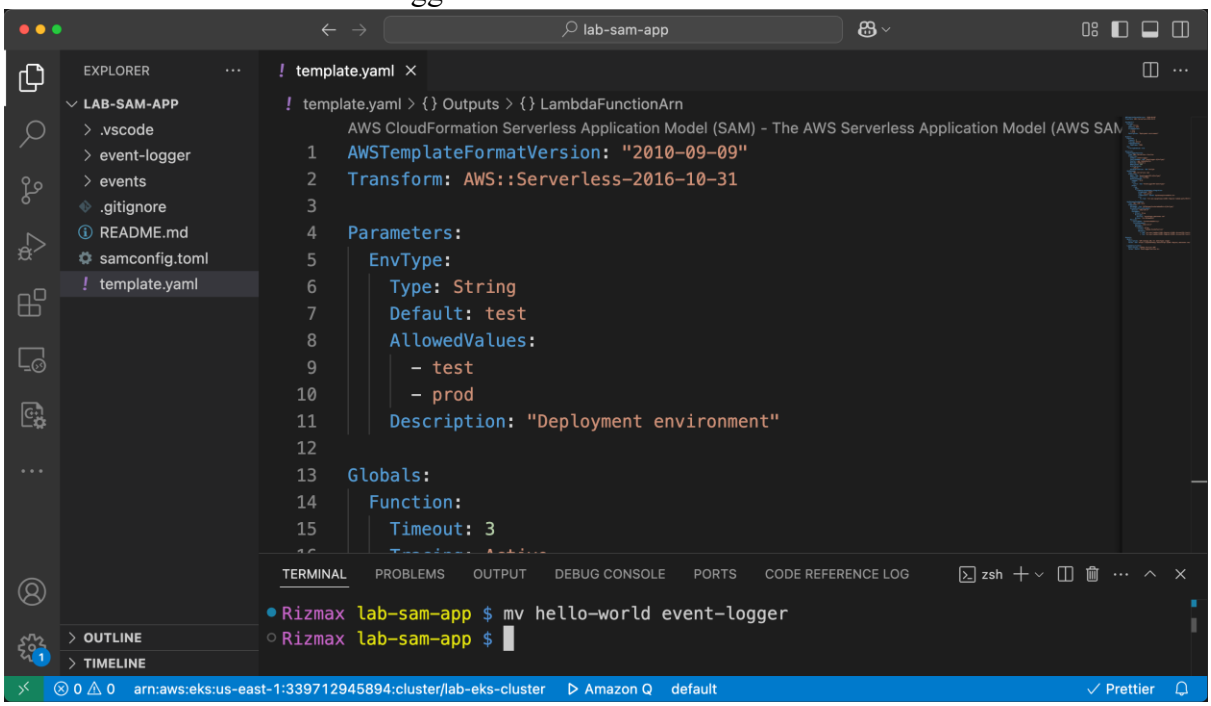
1. sam init --name lab-sam-app --runtime nodejs22.x

- Choose:
  - (1) AWS Quick Start Templates
  - (1) Hello World Example
  - (1) Hello World Example as your starter template
  - (y) Enable X-Ray tracing
  - (N) CloudWatch Application Insights
  - (y) Structured logging in JSON format
- Navigate into the project folder and open it VS Code.

1. cd lab-sam-app
2. code .

- Rename the hello-world directory to event-logger.

1. mv hello-world event-logger

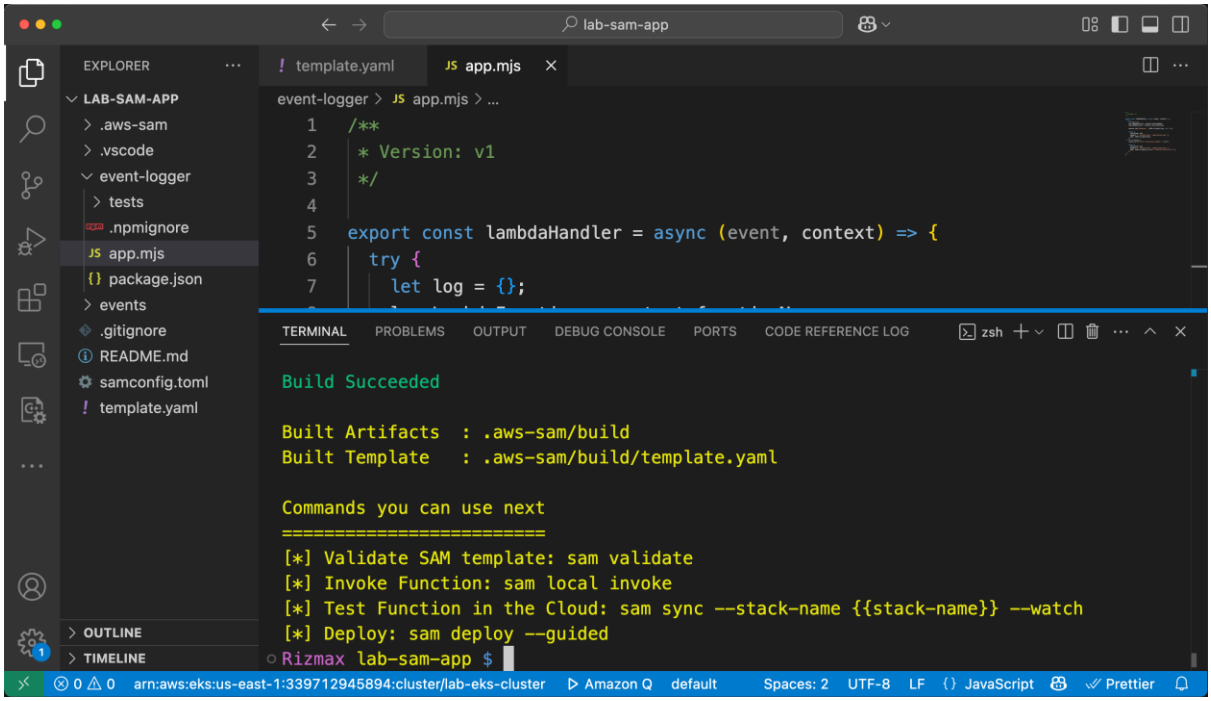


Task 2 - Deploy the AWS SAM Application

Now that you have defined the infrastructure, you will build and deploy it using AWS SAM.
Build and Deploy the Application

- Make sure you have AWS CLI configured or have set AWS temporary credentials in your terminal session.
- Build the application:

1. sam build



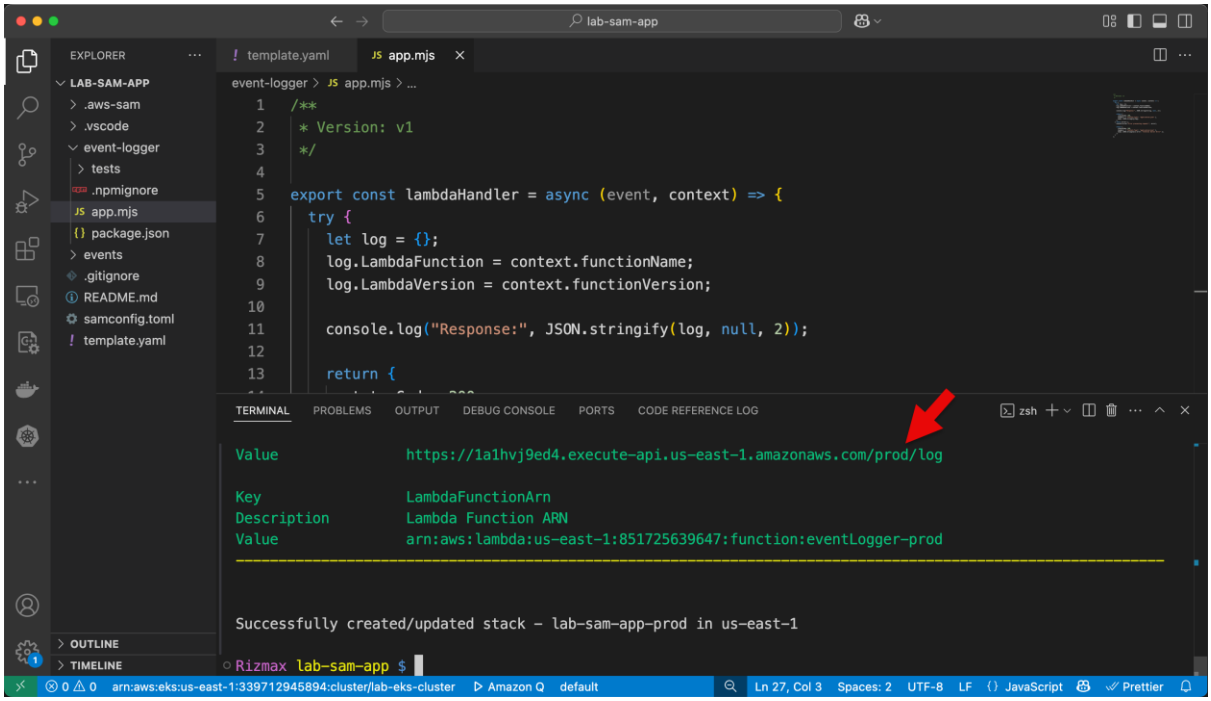
- Deploy the infrastructure for test environment. Enter y for any confirmation prompts.

1. ENV_TYPE=test
2. sam deploy --stack-name=lab-sam-app-${ENV_TYPE} --capabilities CAPABILITY_NAMED_IAM --parameter-overrides EnvType=${ENV_TYPE}

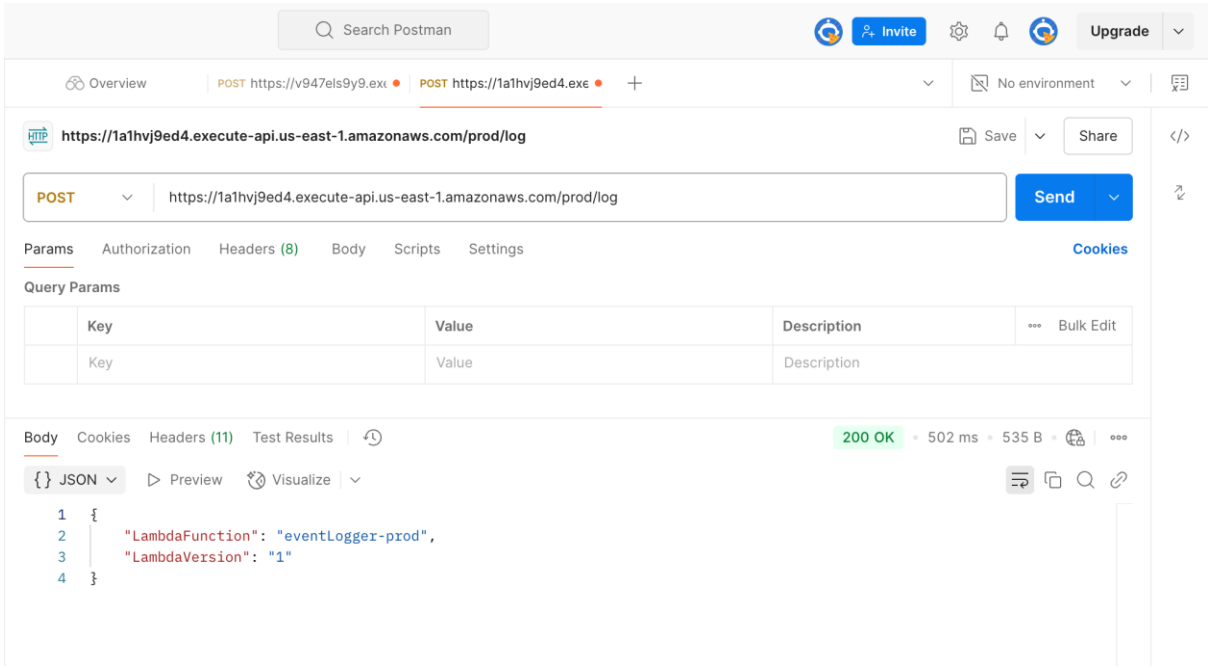
- Deploy the infrastructure for prod environment. Enter y for any confirmation prompts.

1. ENV_TYPE=prod
2. sam deploy --stack-name=lab-sam-app-${ENV_TYPE} --capabilities CAPABILITY_NAMED_IAM --parameter-overrides EnvType=${ENV_TYPE}

- Note the API Gateway method endpoint URL from the output.



- Test using postman to see both test and prod endpoints using the latest Lambda version



Task 3 - Publish Lambda Versions

Now that the infrastructure is deployed, you will create and test Lambda versioning with traffic shifting.

Publish New Lambda Version

- Make a small change in app.js, such as changing a comment e.g. change v1 to v2:

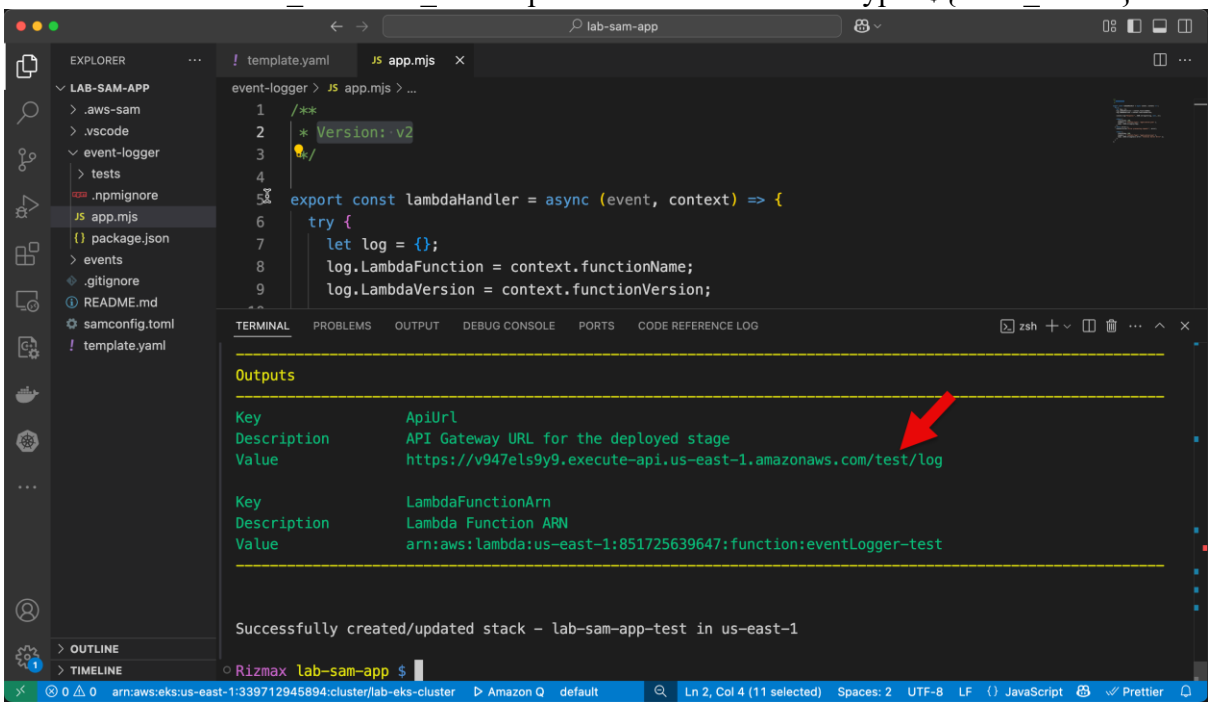
1. /**
2. * Version: v2
3. */
4. export const lambdaHandler ...

- Run the following command to update the Lambda function and deploy again to the test environment.
- Build the application:

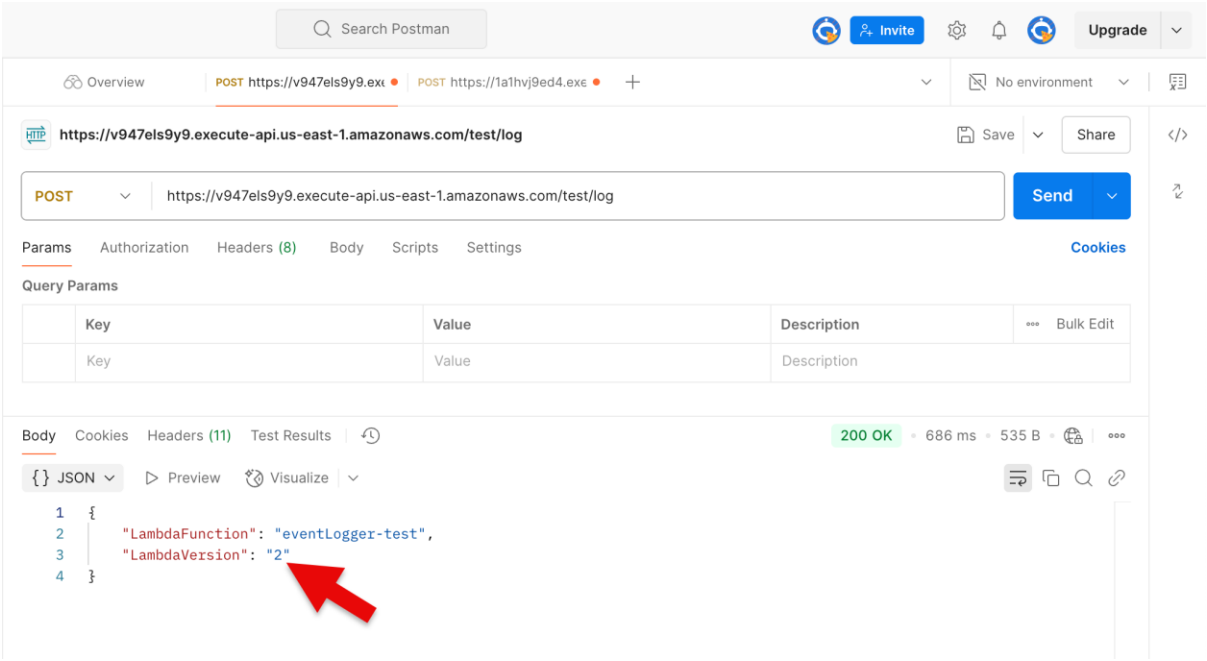
1. sam build

- Deploy the infrastructure for test environment

1. ENV_TYPE=test
2. sam deploy --stack-name=lab-sam-app-${ENV_TYPE} --capabilities CAPABILITY_NAMED_IAM --parameter-overrides EnvType=${ENV_TYPE}



- Test the API using Postman to verify that the test endpoint now uses the new Lambda version

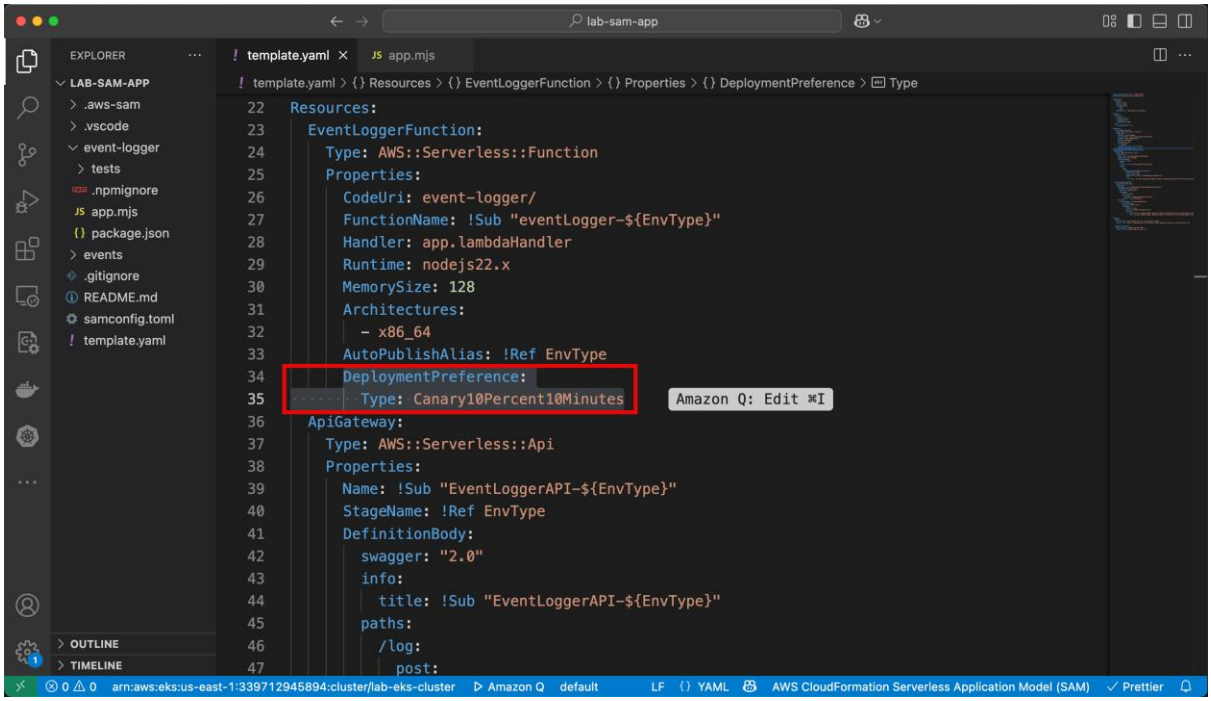


Task 4 - Test Traffic Shifting with CodeDeploy Deployment

Now, you will test the traffic shifting setup.

- Test the prod API endpoint using Postman.
- Test multiple times. You should see the same LambdaVersion in the response every time.
- Configure DeploymentPreference for the Lambda function, like so:

```
Resources:
  EventLoggerFunction:
    Type: AWS::Serverless::Function
    Properties:
      CodeUri: event-logger/
      FunctionName: !Sub "eventLogger-${EnvType}"
      Handler: app.lambdaHandler
      Runtime: nodejs22.x
      MemorySize: 128
      Architectures:
        - x86_64
      AutoPublishAlias: !Ref EnvType
      DeploymentPreference:
        Type: Canary10Percent10Minutes
```

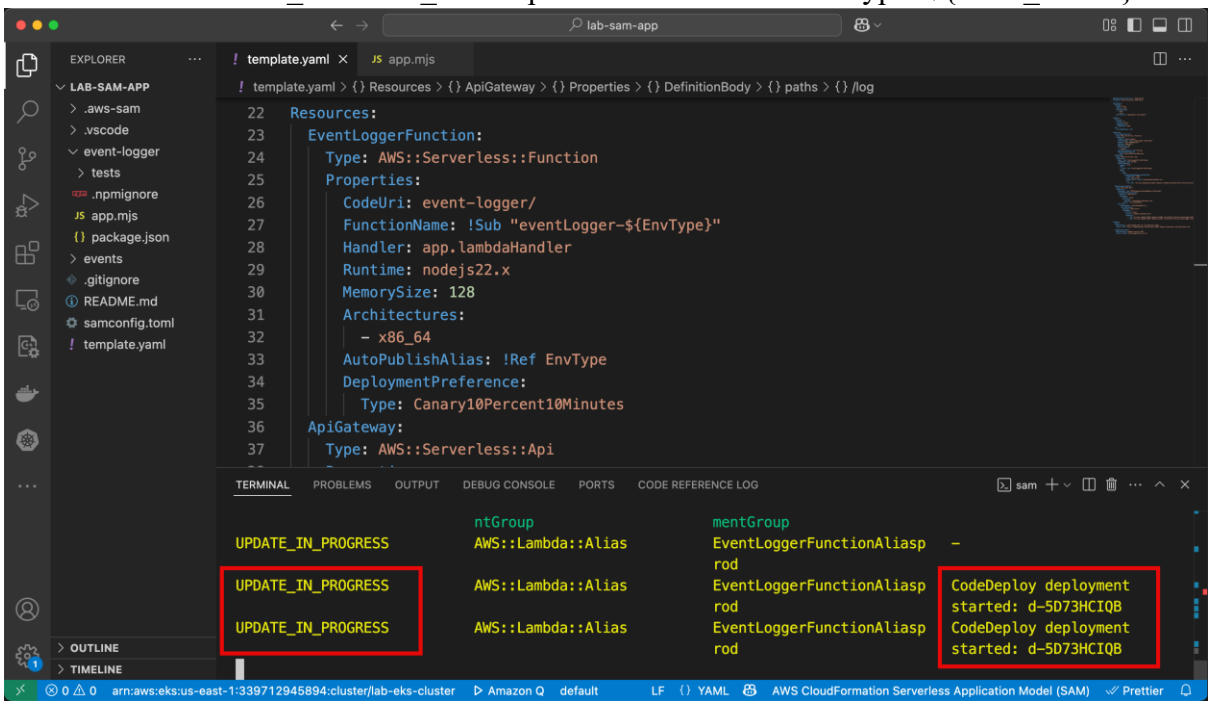


- You can either use Canary deployment or Linear deployment option. Learn more here.
- Build the application:

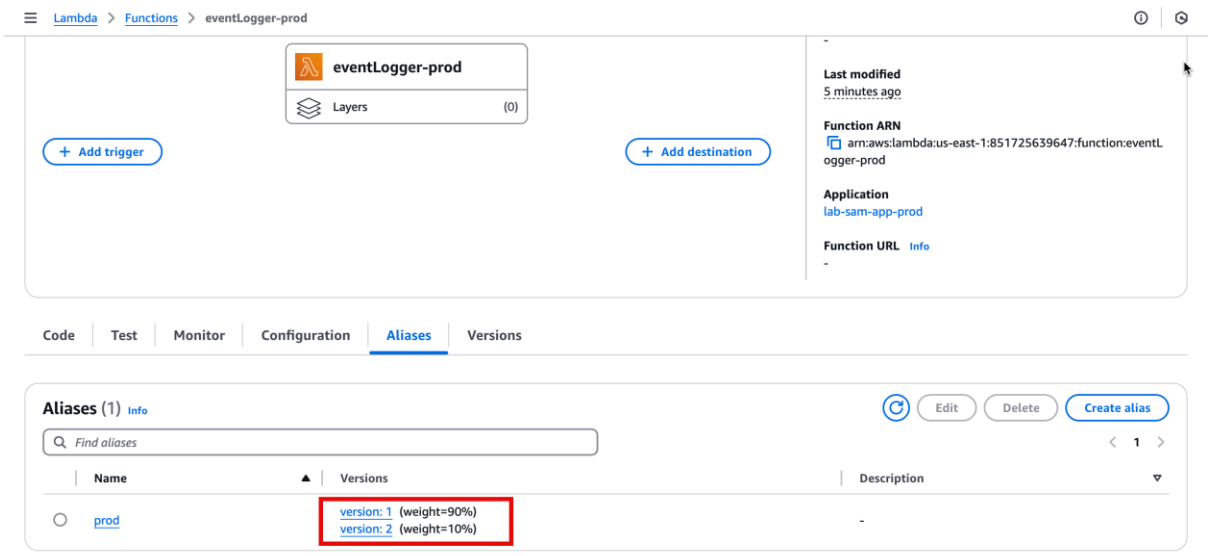
1. sam build

- Deploy the infrastructure for prod environment

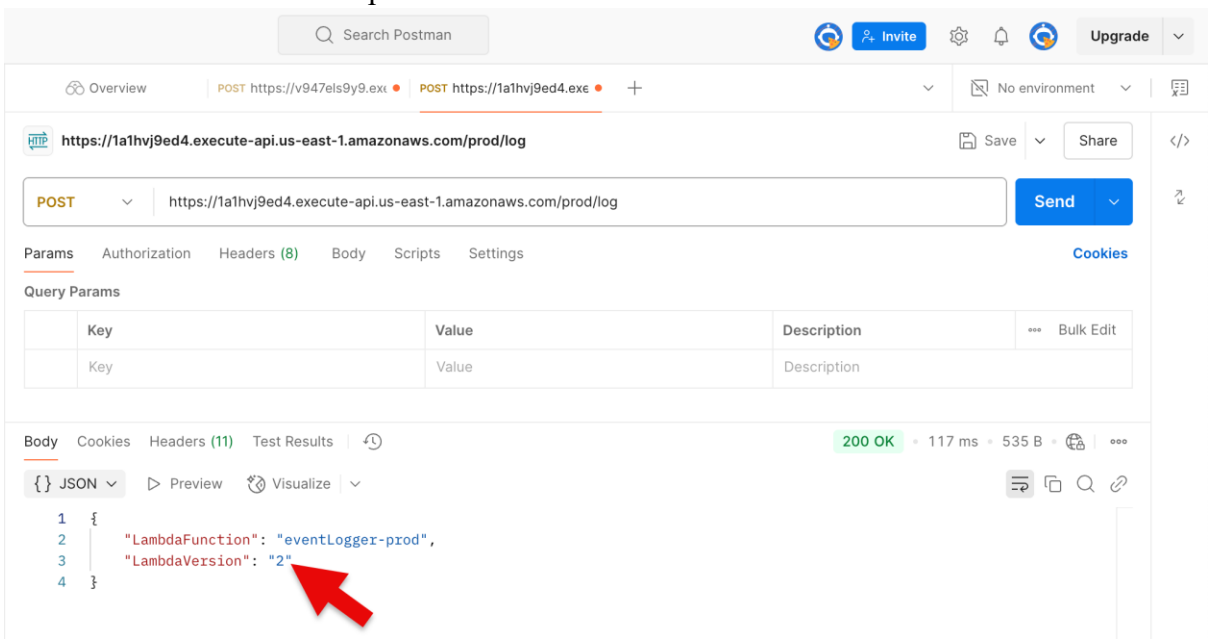
1. ENV_TYPE=prod
2. sam deploy --stack-name=lab-sam-app-${ENV_TYPE} --capabilities CAPABILITY_NAMED_IAM --parameter-overrides EnvType=${ENV_TYPE}



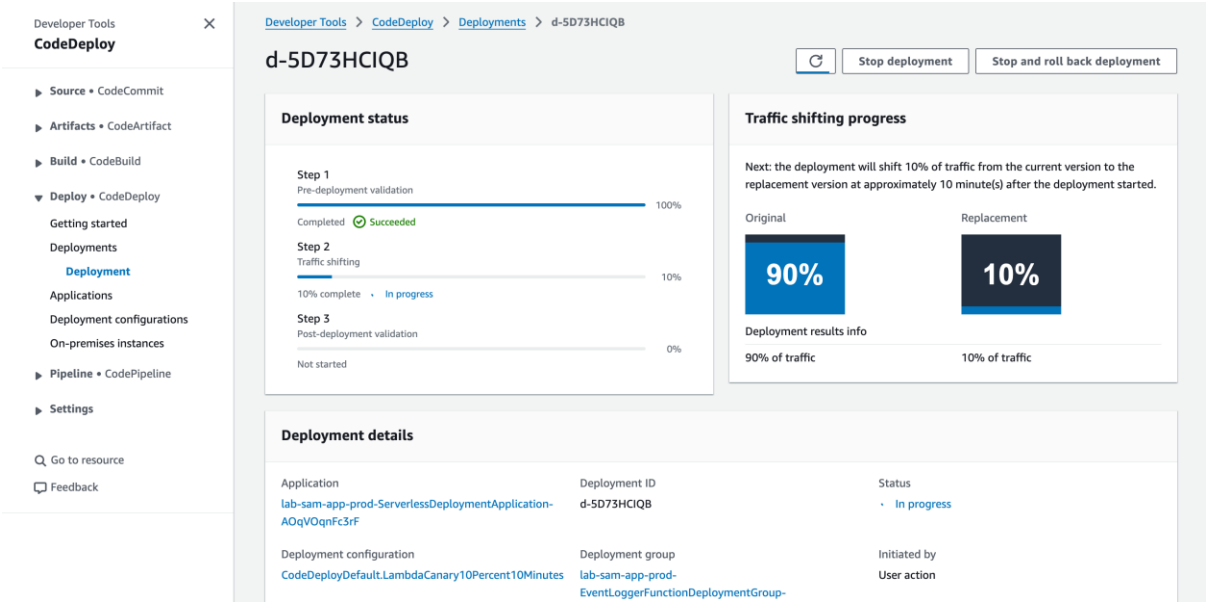
- While the deployment is in progress, review the Lambda function Aliases tab for the eventLogger-prod function. You should see traffic being shifted between two versions.



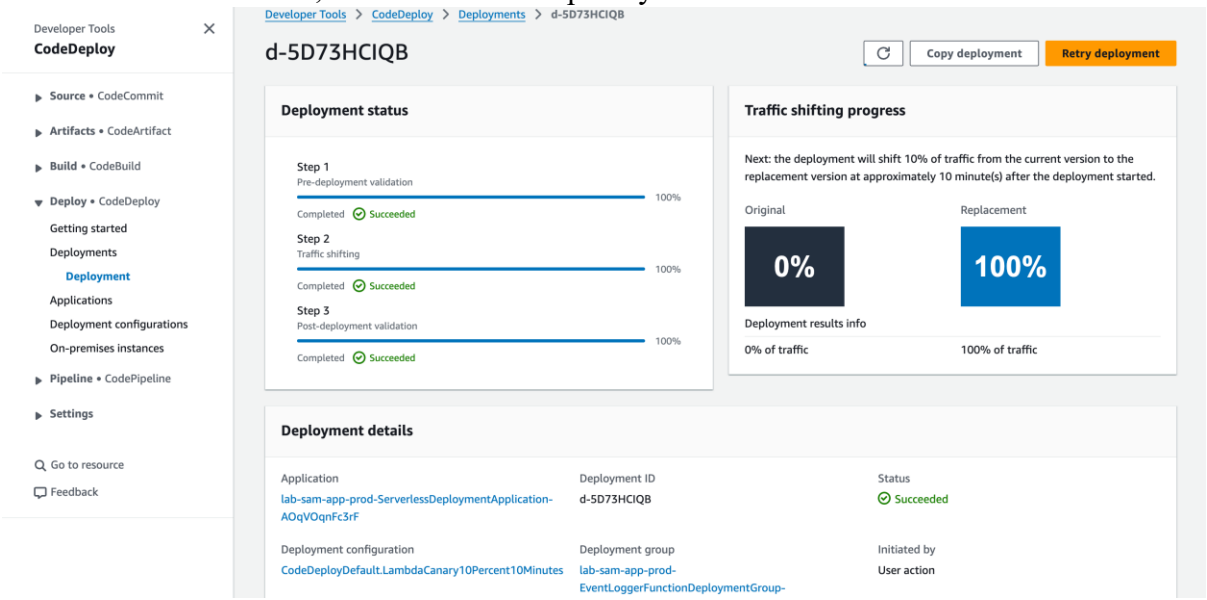
- Now, test the prod API endpoint using Postman.
- Test multiple times. Observe that 10% of responses show new Lambda version, while 90% show the previous version.



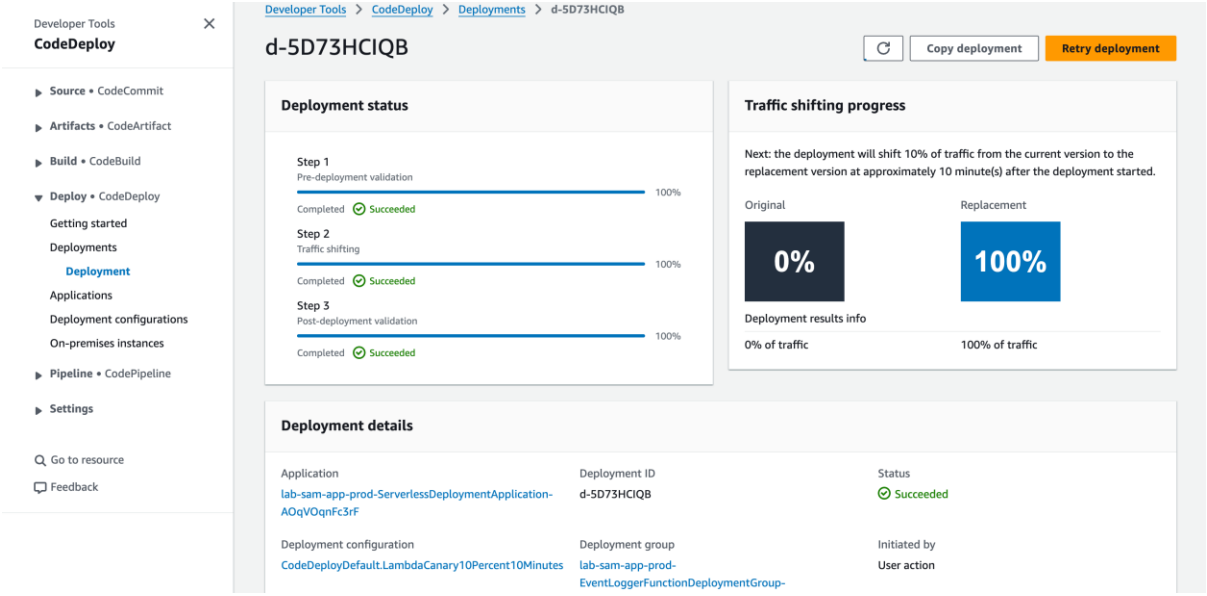
- Open CodeDeploy console, and open the running deployment to observe traffic shifting progress. If the deployment did not go as expected, it can be stopped or rolled back from here.



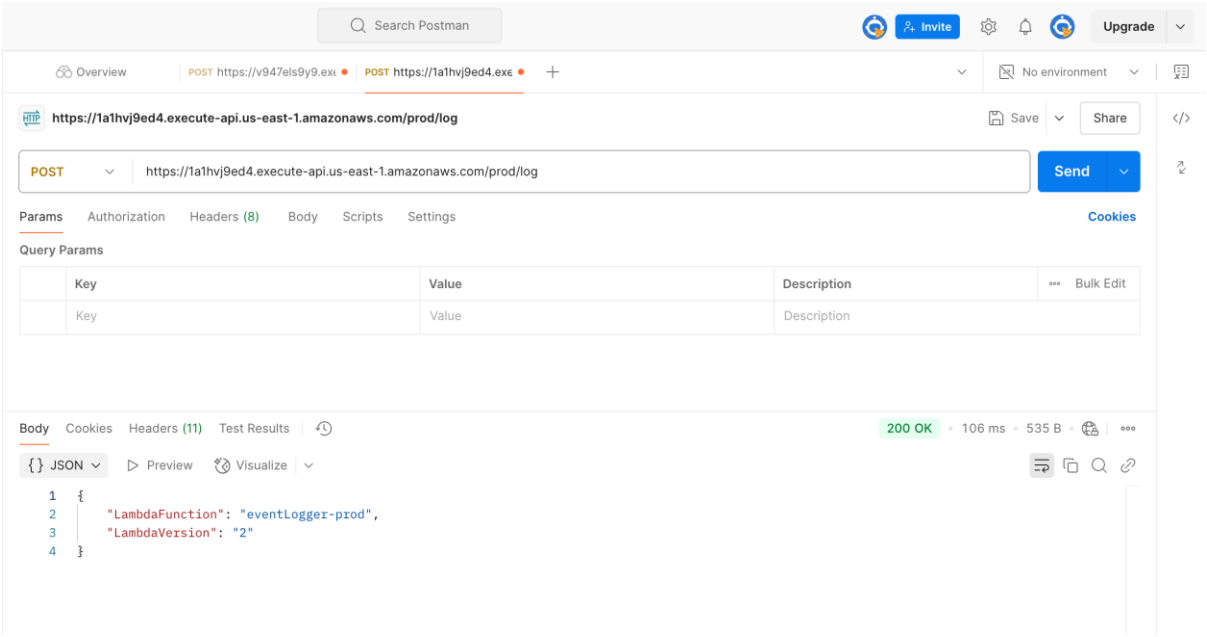
- After 10 minutes, traffic will be completely directed to the new Lambda version.



- Review the Lambda function Aliases tab in Lambda console. You should see traffic being shifted completely to the new Lambda version.



- Test the prod API endpoint using Postman, and you should see the new version 100% of the times.



- You may also experiment with Linear10PercentEvery1Minute as DeploymentPreference which would shift traffic to the new version in 10% increments every minute.

Task 6 - Clean Up Resources

As a best-practice, you may want to delete the resources created during the lab if no longer needed.

Delete the AWS SAM Stack

- Delete all deployed resources using the following commands. Enter y for any confirmation prompts.

1. ENV_TYPE=test
2. sam delete --stack-name=lab-sam-app-${ENV_TYPE}

1. ENV_TYPE=prod
2. sam delete --stack-name=lab-sam-app-${ENV_TYPE}