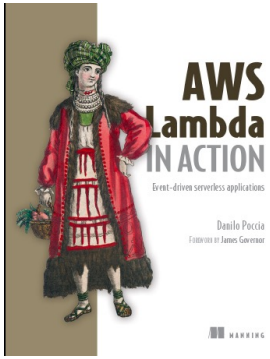
Function as a Service (FaaS) aka Serverless Computing



AWS Lambda in Action

Danilo Poccia

Manning

Function as a Service (FaaS)

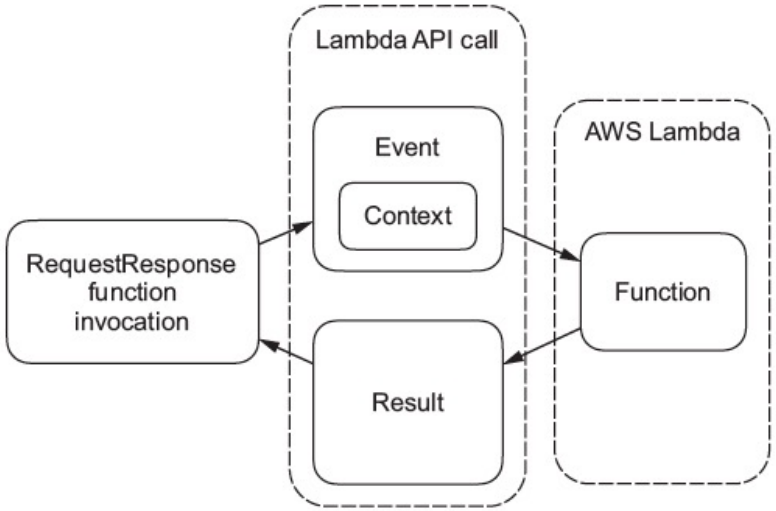
- **Cloud provider**
  - Manages infrastructure
  - Manages software stack (OS, runtime)
  - Handles provisioning
  - Availability
  - Scalability
- **Developer**
  - Implements application as set of functions
  - Functions run when certain events are triggered
    - Web request
    - File upload
    - Alarm
    - Database update
- **Examples:**
  - AWS Lambda, Google Cloud Functions, IBM OpenWhisk, MSFT Azure Functions

Lambda Execution Environment

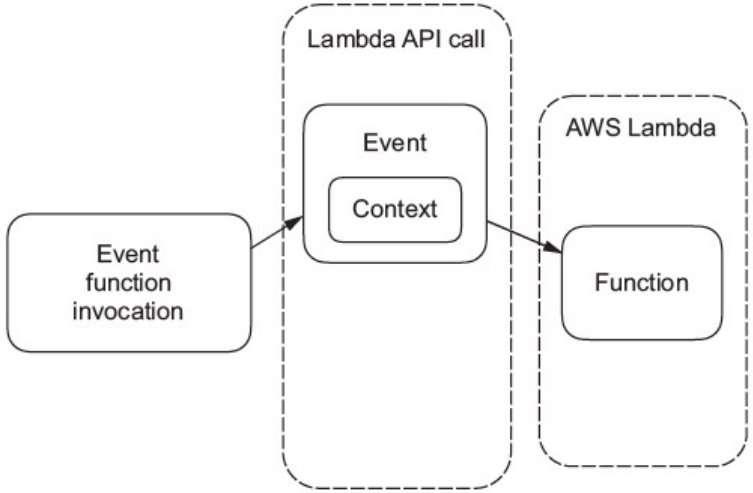
- **Function runs inside an operating system container**
  - Single OS runs multiple containers
  - Isolation ensures that only processes inside container are visible
  - Has its own file system /tmp
  - Runtimes: C#, Node.js, Java, Python
- **A container handles a single function/event at a time**
- **Concurrent execution by container replication**
- **Containers may be reused for subsequent function executions**
- **Containers may be terminated at any time**
  - Stateless
  - Store all persistent state outside of container

Lambda Invocation Modalities

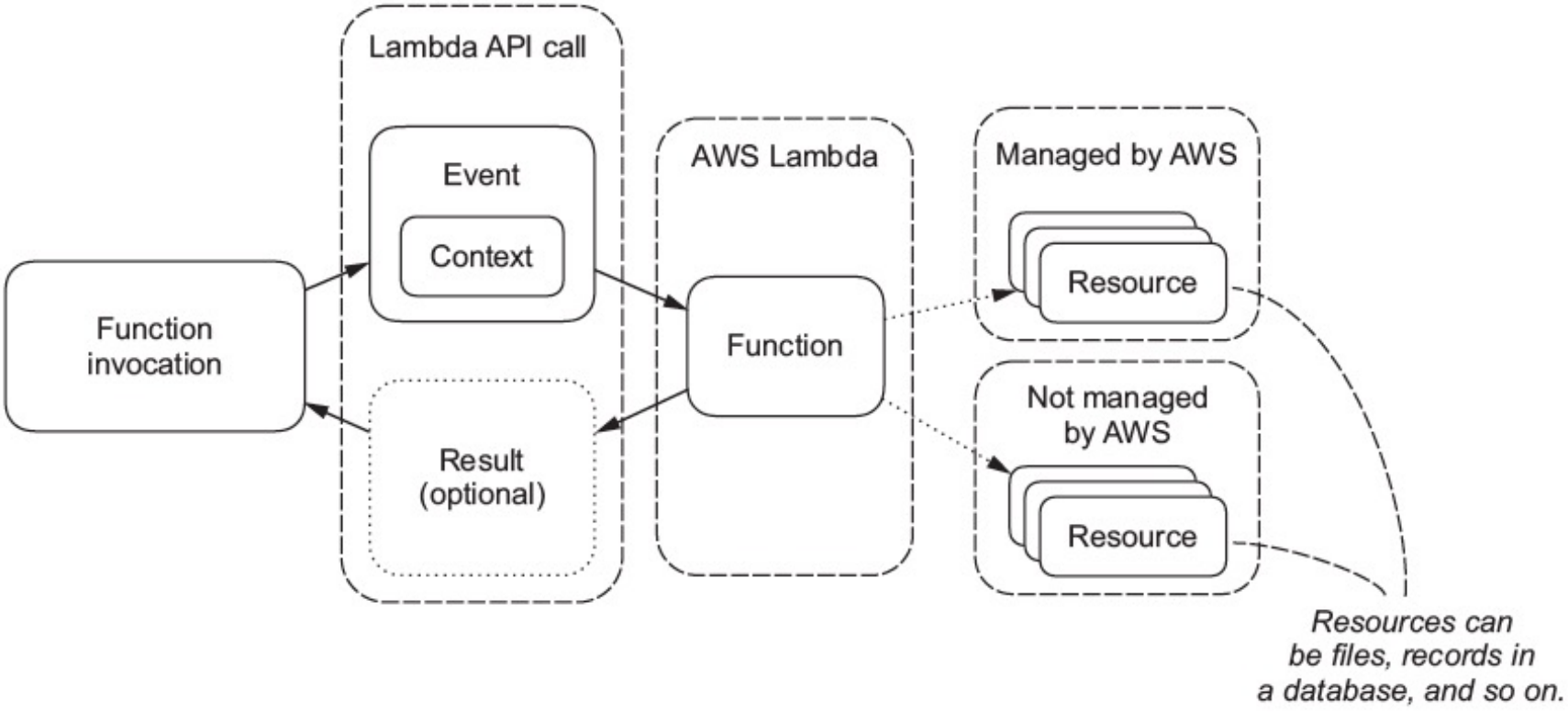
- **Synchronous (RequestResponse)**



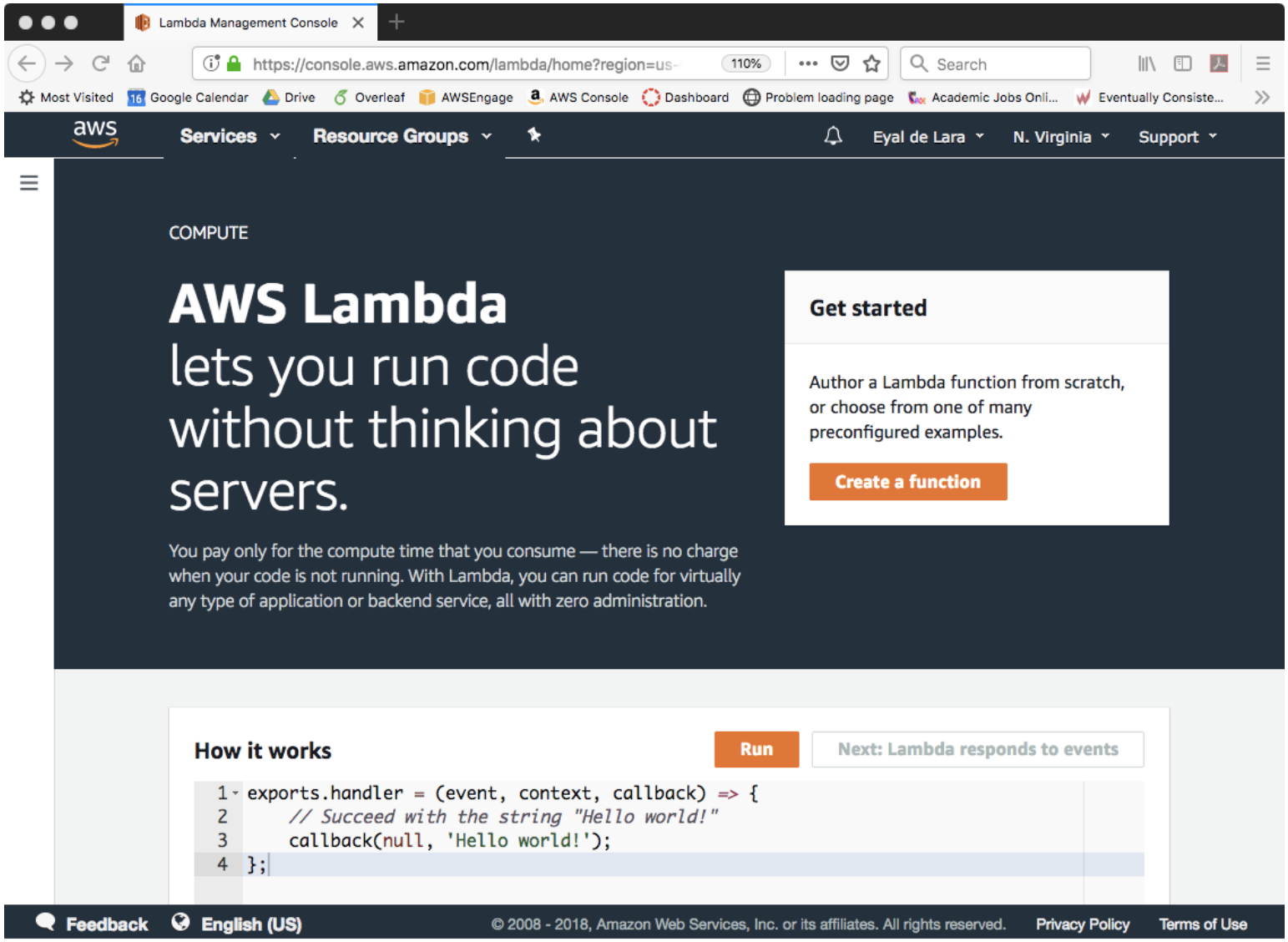
- **Asynchronous (Event)**



Lambda Application Architecture



1st Lambda Function



1st Lambda Function

Lambda

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

aws Services ▼ Eyal de Lara ▼ N. Virginia ▼ Support ▼

Lambda > Functions > Create function

Create function Info

Choose one of the following options to create your function.

Author from scratch
Start with a simple Hello World example.

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

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.

helloWorld

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

Runtime Info
Choose the language to use to write your function.

Python 3.8

Feedback English (US) ▼ Privacy Policy Terms of Use

1st Lambda Function

Lambda

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

aws Services Eyal de Lara N. Virginia Support

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

helloWorld

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

Role creation might take a few minutes. Please do not delete the role or edit the trust or permissions policies in this role.

Lambda will create an execution role named helloWorld-role-zsteinrm, with permission to upload logs to Amazon CloudWatch Logs.

Feedback English (US) © 2008 - 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

1st Lambda Function

aws Services Resource Groups Eyal de Lara N. Virginia Support

Successfully created the function **helloWorld**. You can now change its code and configuration. To invoke your function with a test event, choose "Test".

Lambda > Functions > helloWorld

ARN - arn:aws:lambda:us-east-1:470901565805:function:helloWorld

helloWorld

Throttle | Qualifiers | Actions | Select a test event | Test | Save

Configuration | Permissions | Monitoring

Designer

helloWorld

Layers (0)

+ Add trigger

+ Add destination

Feedback English (US)

Privacy Policy Terms of Use

1st Lambda Function

helloWorld - Lambda

console.aws.amazon.com/lambda/home?region=us-east-1#/functions/helloWorld?newFunction=true&tab=confi...

aws Services ▼ Eyal de Lara ▼ N. Virginia ▼ Support ▼

Successfully created the function **helloWorld**. You can now change its code and configuration. To invoke your function with a test event, choose "Test".

helloWorld

Throttle | Qualifiers ▼ | Actions ▼ | Select a test event ▼ | Test

Function code Info

Deploy | Actions ▼

File Edit Find View Go Tools Window Test ▼ Deploy

Environment

helloWorld - /

lambda_function.py

lambda_function

```
1  import json
2
3  def lambda_handler(event, context):
4      print('Received event: ' + json.dumps(event,indent=2))
5      if 'name' in event and len(event['name']) > 0:
6          name = event['name']
7      else:
8          name = "World"
9          greetings = "Hello " + name + '!'
10         print(greetings)
11     return greetings
12
```

11:5 Python Spaces: 4

Execution Result

Execution results

No execution results yet

Feedback English (US) ▼

Privacy Policy Terms of Use

1st Lambda Function

Lambda Management Consol

https://console.aws.amazon.com/lambda/home?region=us-east-1#/functions/helloWorld?tab

Most Visited ECE1779 CSC2228 Google Calendar AWSEngage AWS Console PCRS

aws Services Resource Groups Eyal de Lara N. Virginia Support

Execution role

Defines the permissions of your function. Note that new roles may not be available for a few minutes after creation. Learn more about Lambda execution roles.

Choose an existing role

Existing role

You may use an existing role with this function. Note that the role must be assumable by Lambda and must have Cloudwatch Logs permissions.

lambda_basic_execution

Basic settings

Memory (MB) Info

Your function is allocated CPU proportional to the memory configured.

128 MB

Timeout Info

0 min 3 sec

Description

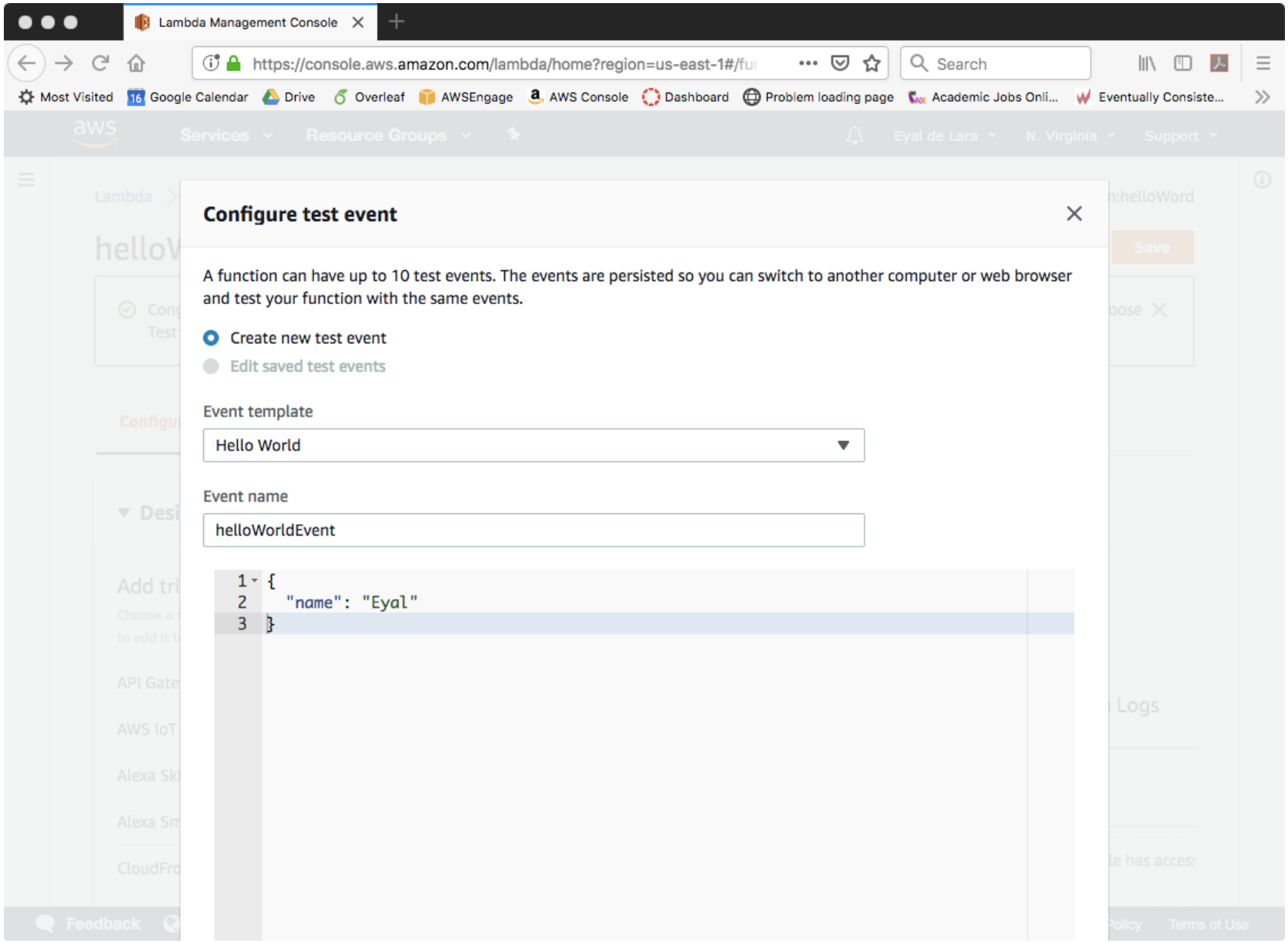
Network

Debugging and error handling

Feedback English (US)

Privacy Policy Terms of Use

Lambda Function Testing Arguments



Lambda Function Testing

Lambda Management Console | Lambda Management Console

https://console.aws.amazon.com/lambda/home?region=us-east-1#/fu

Most Visited | Google Calendar | Drive | Overleaf | AWSEngage | AWS Console | Dashboard | Problem loading page | Academic Jobs Onli... | Eventually Consiste...

aws | Services | Resource Groups | Eyal de Lara | N. Virginia | Support

Lambda > Functions > helloWord

ARN - arn:aws:lambda:us-east-1:470901565805:function:helloWord

helloWord

Throttle | Qualifiers | Actions | helloWorldEvent | Test | Save

Execution result: succeeded (logs)

Details

The section below shows the result returned by your function execution.

```
"Hello World!"
```

Summary

Code SHA-256 D8wUy77Ts8/Mg1nd5uC9+kfKSuJdxvApwHaKhwi6DjA=	Request ID 5c2a854b-dd2c-11e8-9a1f-138b16ec1863
Duration 0.47 ms	Billed duration 100 ms
Resources configured 128 MB	Max memory used 21 MB

Log output

The section below shows the logging calls in your code. These correspond to a single row within the CloudWatch log group corresponding to this Lambda function. Click here to view the CloudWatch log group.

```
START RequestId: 5c2a854b-dd2c-11e8-9a1f-138b16ec1863 Version: $LATEST
Received event: {
  "other": "Eyal"
}
Hello World!
```

Feedback | English (US)

Privacy Policy | Terms of Use

Lambda Command Line Interface (CLI)

- **Download from http://aws.amazon.com/cli**

- **Configure credentials**

  Create access key on AWS Console

  > aws configure

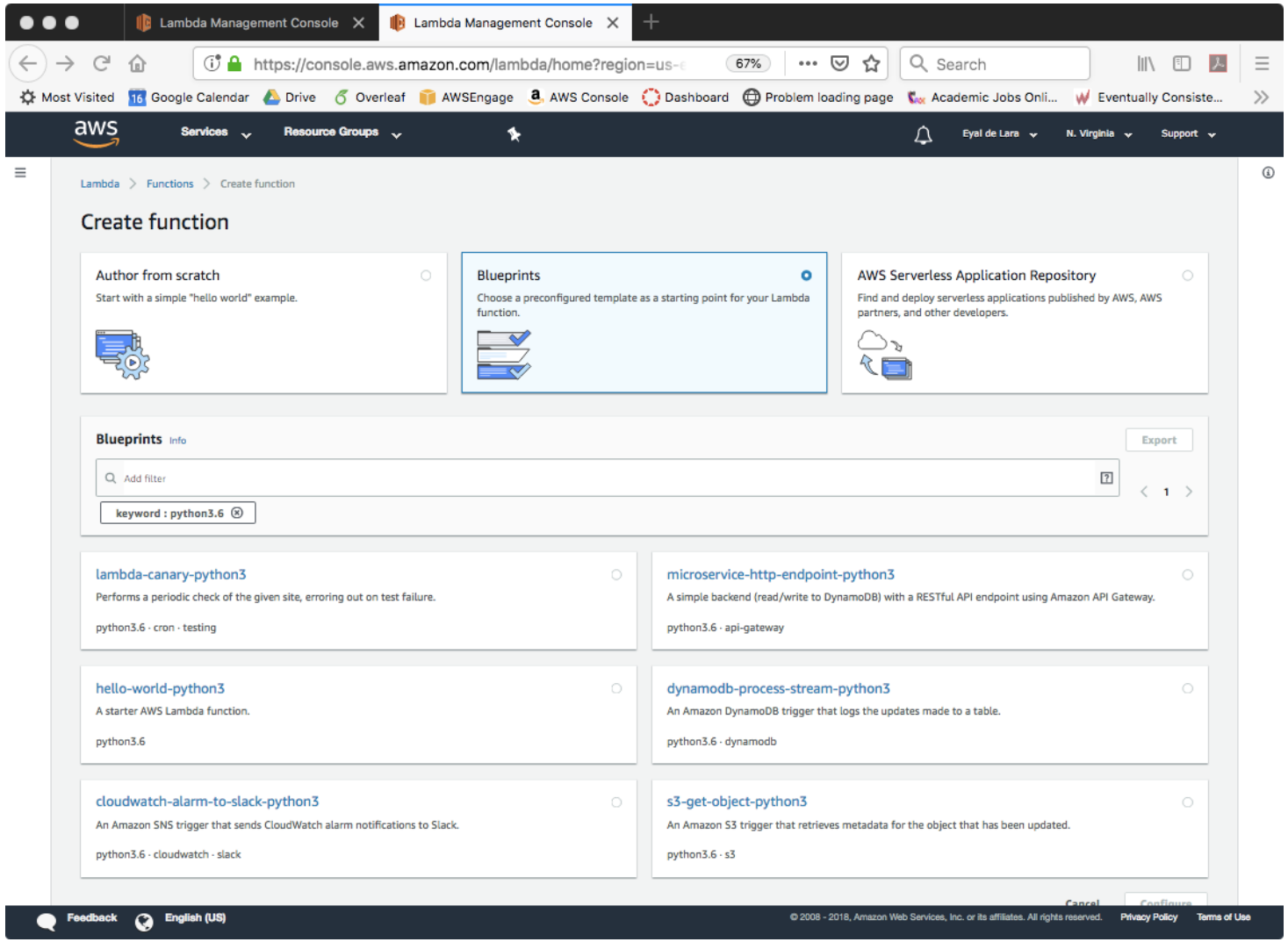
- **Call function**

  > aws lambda list-functions

  > aws lambda invoke --function-name helloWorld --payload '{}' output.txt

  > aws lambda invoke --function-name helloWorld --payload '{"name":"Eyal"}' output.txt

Reacting to S3 Events



Reacting to S3 Events

Lambda Management Consol

https://console.aws.amazon.com/lambda/home?region=us-east-1#/create/new?bp=s3-get-

Most Visited ECE1779 CSC2228 Google Calendar AWSEngage AWS Console PCRS

aws Services Resource Groups Eyal de Lara N. Virginia Support

Lambda > Functions > Create function > Using blueprint s3-get-object-python

Basic information Info

Name*

s3EventHandler

Role*

Defines the permissions of your function. Note that new roles may not be available for a few minutes after creation. Learn more about Lambda execution roles.

Create new role from template(s)

Lambda will automatically create a role with permissions from the selected policy templates. Note that basic Lambda permissions (logging to CloudWatch) will automatically be added. If your function accesses a VPC, the required permissions will also be added.

Role name*

Enter a name for your new role.

s3ReadRole

Policy templates

Choose one or more policy templates. A role will be generated for you before your function is created. Learn more about the permissions that each policy template will add to your role.

S3 object read-only permissions

Feedback English (US) Privacy Policy Terms of Use

Reacting to S3 Events

Lambda Management Console | Lambda Management Console

https://console.aws.amazon.com/lambda/home?region=us-

Most Visited | Google Calendar | Drive | Overleaf | AWSEngage | AWS Console | Dashboard | Problem loading page | Academic Jobs Onli... | Eventually Consiste...

aws | Services | Resource Groups | Eyal de Lara | N. Virginia | Support

S3 trigger

Remove

Bucket

Please select the S3 bucket that serves as the event source. The bucket must be in the same region as the function.

ece1779fall2018

Event type

Select the events that you want to have trigger the Lambda function. You can optionally set up a prefix or suffix for an event. However, for each bucket, individual events cannot have multiple configurations with overlapping prefixes or suffixes that could match the same object key.

PUT

Prefix

Enter a single optional prefix to limit the notifications to objects with keys that start with matching characters.

e.g. images/

Suffix

Enter a single optional suffix to limit the notifications to objects with keys that end with matching characters.

e.g. .jpg

Lambda will add the necessary permissions for Amazon S3 to invoke your Lambda function from this trigger. Learn more about the Lambda permissions model.

☐ Enable trigger

Enable the trigger now, or create it in a disabled state for testing (recommended).

Lambda function code

Code is preconfigured by the chosen blueprint. You can configure it after you create the function.

Feedback | English (US) | © 2008 - 2018, Amazon Web Services, Inc. or its affiliates. All rights reserved. | Privacy Policy | Terms of Use

Reacting to S3 Events

Lambda Management Consol | AWS Management Console

https://console.aws.amazon.com/lambda/home?region=us-east-1#/functions/s3EventHandler?tab=c

Most Visited | ECE1779 | CSC2228 | Google Calendar | AWSEngage | AWS Console | PCRS

aws | Services | Resource Groups | Eyal de Lara | N. Virginia | Support

Function code

Code entry type: Edit code inline

Runtime: Python 2.7

Handler Info: lambda_function.lambda_handler

lambda_function.py

```python
from __future__ import print_function

import json
import urllib
import boto3

print('Loading function')

s3 = boto3.client('s3')


def lambda_handler(event, context):
    print("Received event: " + json.dumps(event, indent=2))

    # Get the object from the event and show its content type
    bucket = event['Records'][0]['s3']['bucket']['name']
    key = urllib.unquote_plus(event['Records'][0]['s3']['object']['key'].encode('utf8'))
    try:
        response = s3.get_object(Bucket=bucket, Key=key)
        print("CONTENT TYPE: " + response['ContentType'])
        return response['ContentType']
    except Exception as e:
        print(e)
        print('Error getting object {} from bucket {}. Make sure they exist and your bucket is in the
        raise e
```

Feedback | English (US) | | Privacy Policy | Terms of Use

Reacting to S3 Events

aws Services Resource Groups Eyal de Lara N. Virginia Support

s3EventHandler

Throttle | Qualifiers | Actions | Select a test event | Test | Save

Congratulations! Your Lambda function "s3EventHandler" has been successfully created and configured with ece1779winter2020 as a trigger in a disabled state. We recommend testing the function behavior before enabling the trigger.

Configuration | Permissions | Monitoring

Designer

s3EventHandler

Layers (0)

S3 (2)

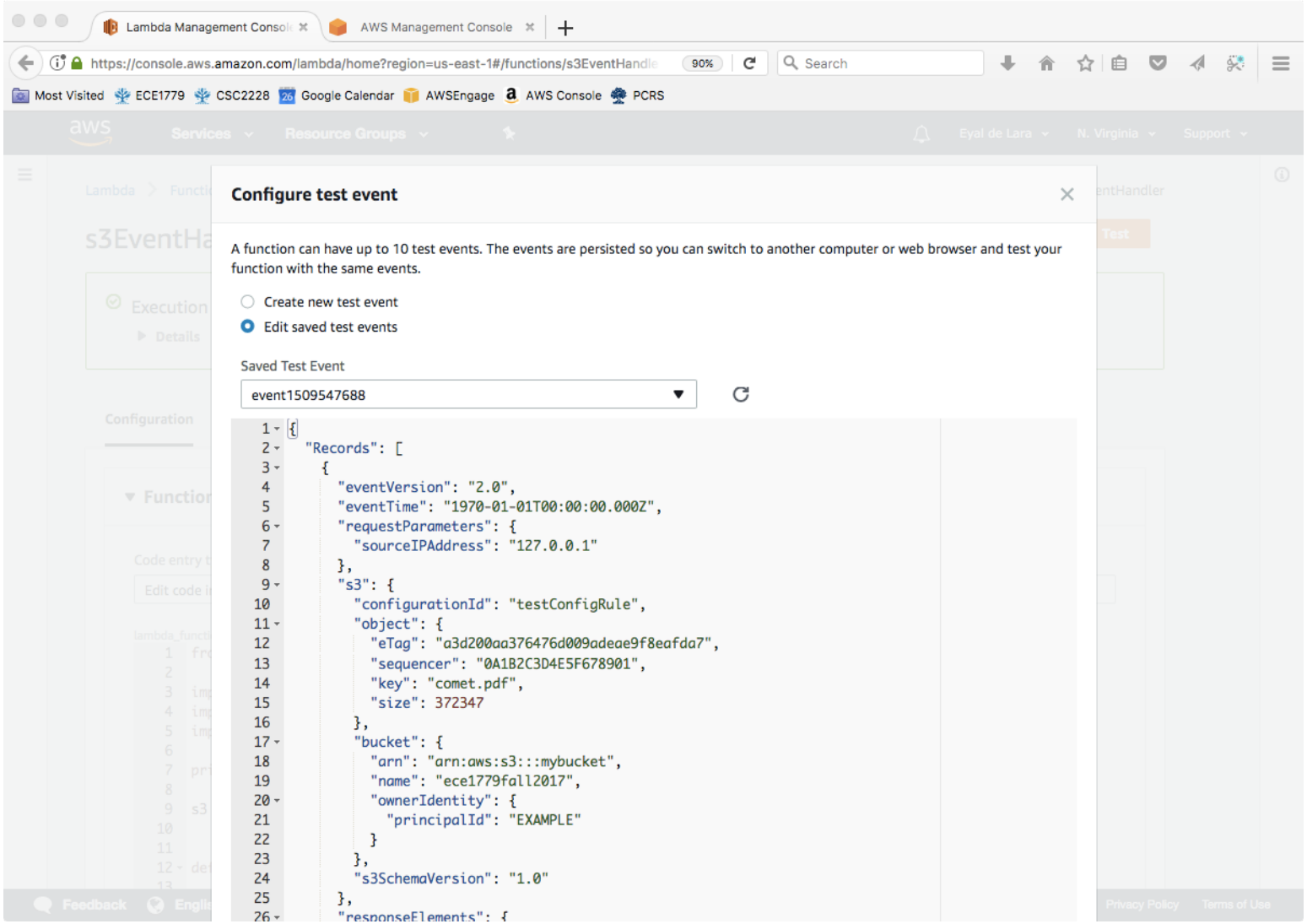
+ Add trigger

+ Add destination

Feedback English (US)

Privacy Policy Terms of Use

Reacting to S3 Events



Reacting to S3 Events

Lambda Management Console | AWS Management Console

https://console.aws.amazon.com/lambda/home?region=us-east-1#/functions/s3EventHandle 90% Search

Most Visited ECE1779 CSC2228 Google Calendar AWSEngage AWS Console PCRS

aws Services Resource Groups Eyal de Lara N. Virginia Support

Lambda > Functions > s3EventHandler

ARN - arn:aws:lambda:us-east-1:470901565805:function:s3EventHandler

s3EventHandler

Qualifiers Actions event1509547688 Test

Execution result: succeeded (logs)

Details

The area below shows the result returned by your function execution.

```
"application/pdf"
```

Summary

Code SHA-256

posOGV7qe1JfL0Th1BMw
RxvQ0XRrb2pGgI4mcsDr
N1U=

Request ID

bdcb6aec-bf16-11e7-
ba00-c9e52ddf0492

Duration

410.76 ms

Billed duration

500 ms

Resources configured

128 MB

Log output

The area below shows the logging calls in your code. These correspond to a single row within the CloudWatch log group corresponding to this Lambda function. Click here to view the CloudWatch log group.

```
START RequestId: bdcb6aec-bf16-11e7-ba00-c9e52ddf0492 Version: $LATEST
Received event: {
  "Records": [
    {
      "eventVersion": "2.0",
      "eventTime": "1970-01-01T00:00:00.000Z",
      "requestParameters": {
        "sourceIPAddress": "127.0.0.1"
      },
      "s3": {
        "configurationId": "testConfigRule",
```

Feedback English (US)

 Privacy Policy Terms of Use

Reacting to S3 Events

Lambda Management Consol | CloudWatch Management Co | AWS Management Console

https://console.aws.amazon.com/cloudwatch/home?region=us-east-1#logEventViewer:grou

Most Visited | ECE1779 | CSC2228 | Google Calendar | AWSEngage | AWS Console | PCRS

aws | Services | Resource Groups | Eyal de Lara | N. Virginia | Support

CloudWatch
Dashboards
Alarms
ALARM 0
INSUFFICIENT 5
OK 0
Billing
Events
Rules
Event Buses NEW
Logs
Metrics

CloudWatch > Log Groups > /aws/lambda/s3EventHandler > 2017/11/01/[$LATEST]2e1b285cf5d34665b7cada673a3585e5

Expand all | Row | Text

Filter events | all | 30s | 5m | 1h | 6h | 1d | 1w | custom

Time (UTC +00:00)	Message
2017-11-01	
15:10:05	},
15:10:05	"name": "ece1779fall2017",
15:10:05	"arn": "arn:aws:s3:::mybucket"
15:10:05	},
15:10:05	"s3SchemaVersion": "1.0"
15:10:05	},
15:10:05	"responseElements": {
15:10:05	"x-amz-id-2": "EXAMPLE123/5678abcdefghijklambdaisawesome/mnopqrstuvwxyzABCDEFGH",
15:10:05	"x-amz-request-id": "EXAMPLE123456789"
15:10:05	},
15:10:05	"awsRegion": "us-east-1",
15:10:05	"eventName": "ObjectCreated:Put",
15:10:05	"userIdentity": {
15:10:05	"principalId": "EXAMPLE"
15:10:05	},
15:10:05	"eventSource": "aws:s3"
15:10:05	}
15:10:05	]
15:10:05	}
15:10:05	CONTENT TYPE: application/pdf
15:10:05	END RequestId: bdcb6aec-bf16-11e7-ba00-c9e52ddf0492
15:10:05	REPORT RequestId: bdcb6aec-bf16-11e7-ba00-c9e52ddf0492 Duration: 410.76 ms Billed Duration: 500 ms Memory Size: 128 MB Ma

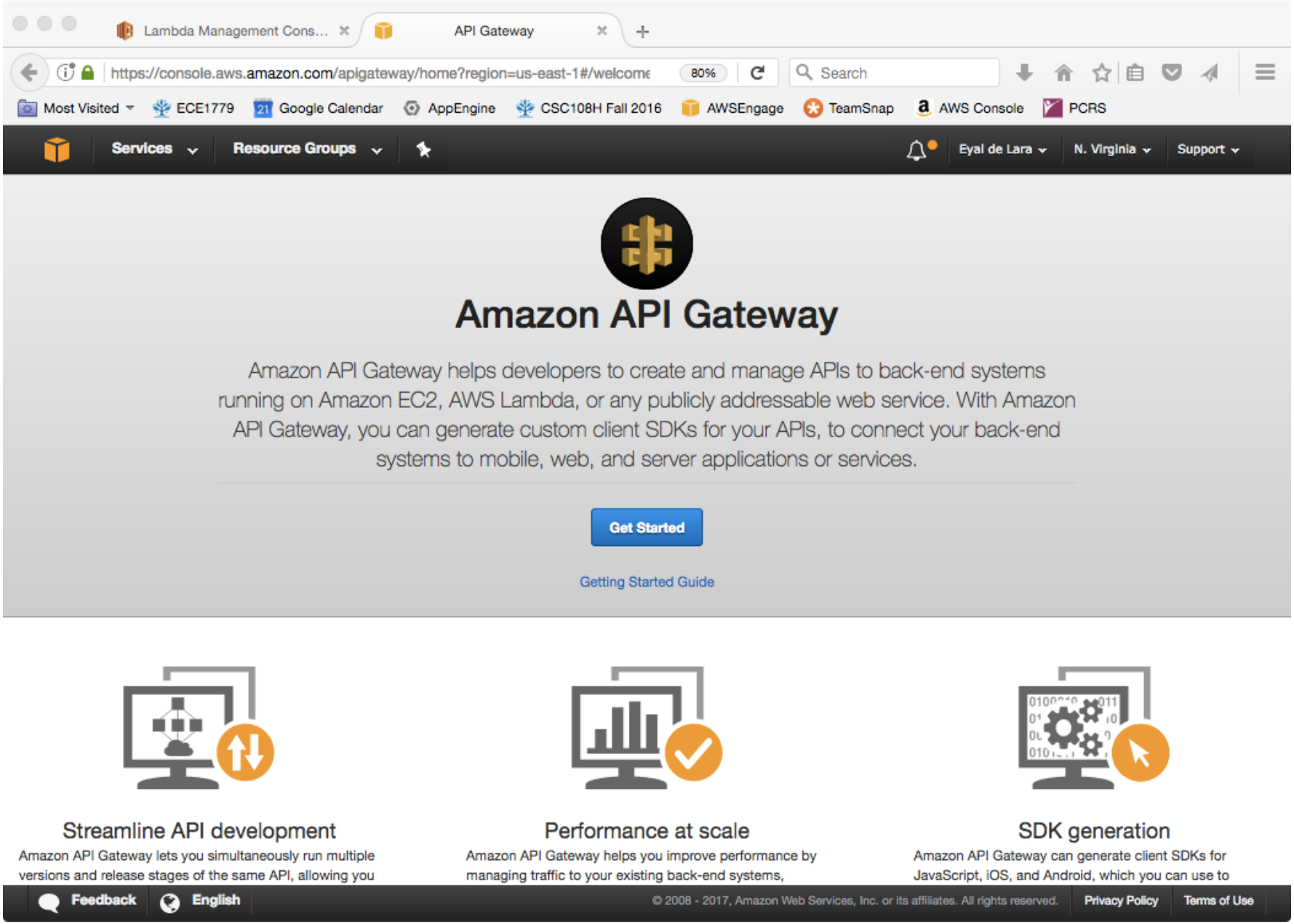
No newer events found at the moment. Retry.

Feedback | English (US) | | Privacy Policy | Terms of Use

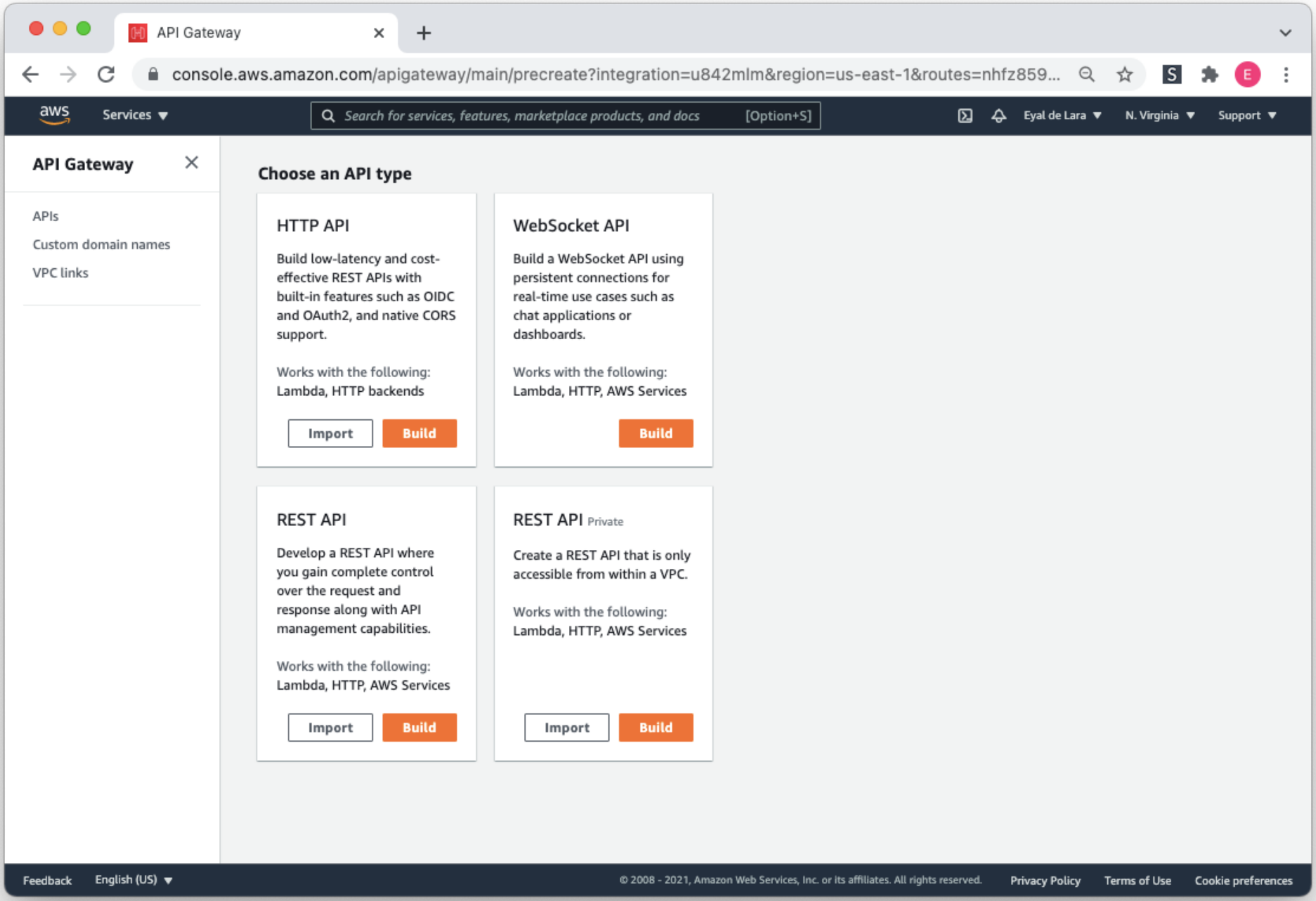
Function as Web API

- Amazon API Gateway can be integrated with Lambda
- Turn http request into Lambda events
- Maps each Lambda Event into a URL endpoint

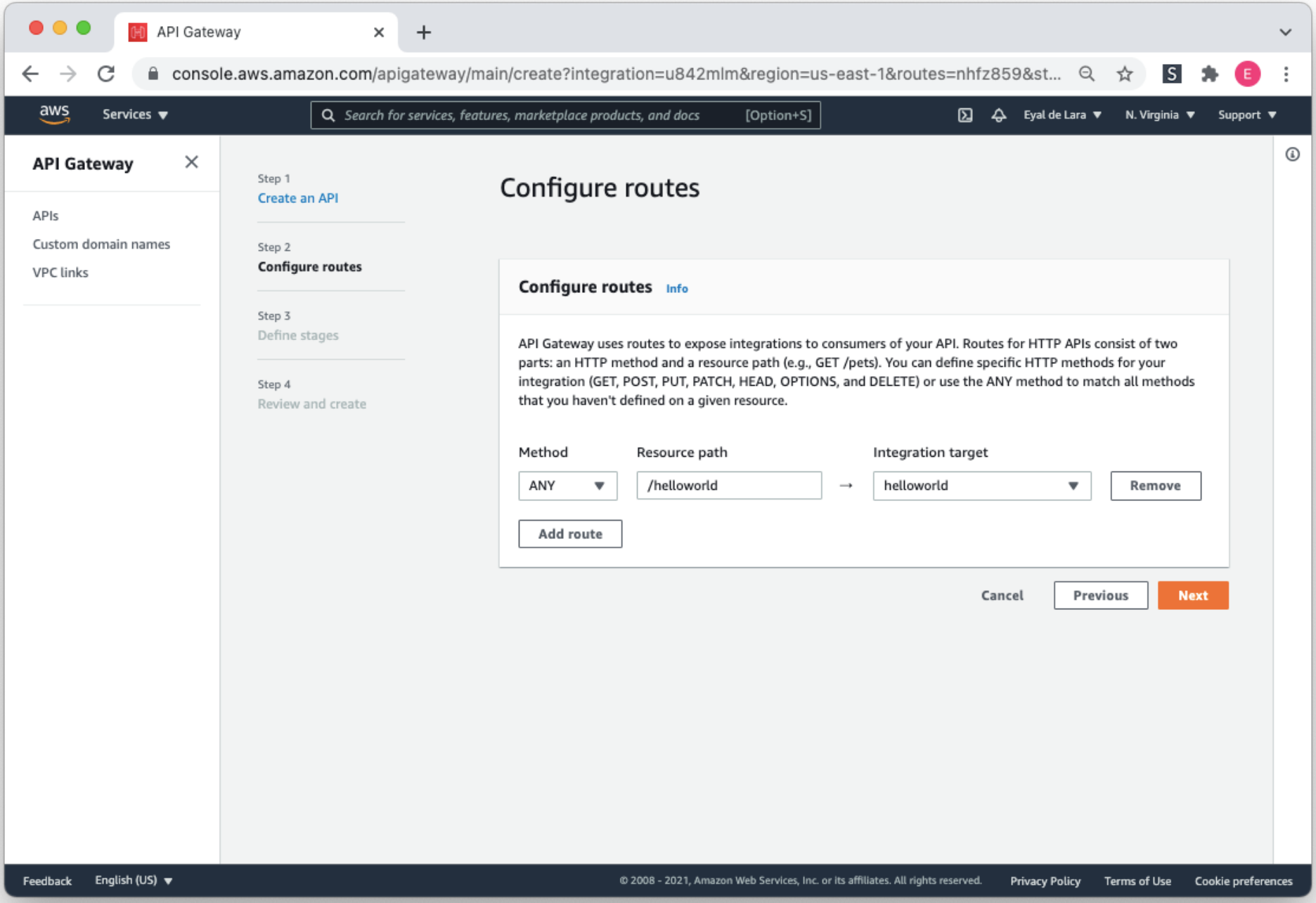
Function as Web API



Function as Web API



Function as Web API



Function as Web API

API Gateway

console.aws.amazon.com/apigateway/main/create?integration=u842mlm&region=us-east-1&routes=nhfz859&st...

aws Services Search for services, features, marketplace products, and docs [Option+S] Eyal de Lara N. Virginia Support

API Gateway

APIs
Custom domain names
VPC links

Step 1
Create an API

Step 2
Configure routes

Step 3
Define stages

Step 4
Review and create

Define stages

Configure stages Info

Stages are independently configurable environments that your API can be deployed to. You must deploy to a stage for API configuration changes to take effect, unless that stage is configured to autodeploy. By default, all HTTP APIs created through the console have a default stage named $default. All changes that you make to your API are autodeployed to that stage. You can add stages that represent environments such as development or production.

Stage name	Auto-deploy	
dev	ON	Remove

Add stage

Cancel Previous Next

Feedback English (US)

Privacy Policy Terms of Use Cookie preferences

Function as Web API

API Gateway

console.aws.amazon.com/apigateway/main/create?integration=u842mlm&region=us-east-1&routes=nhfz859&st...

aws Services ▼ Search for services, features, marketplace products, and docs [Option+S] Eyal de Lara ▼ N. Virginia ▼ Support ▼

API Gateway

- APIs
- Custom domain names
- VPC links

Step 1
Create an API

Step 2
Configure routes

Step 3
Define stages

Step 4
Review and create

Review and create

API name and integrations Edit

API name
helloworldapi
Integrations
helloworld (Lambda)

Routes Edit

Routes
ANY /helloworld → helloworld (Lambda)

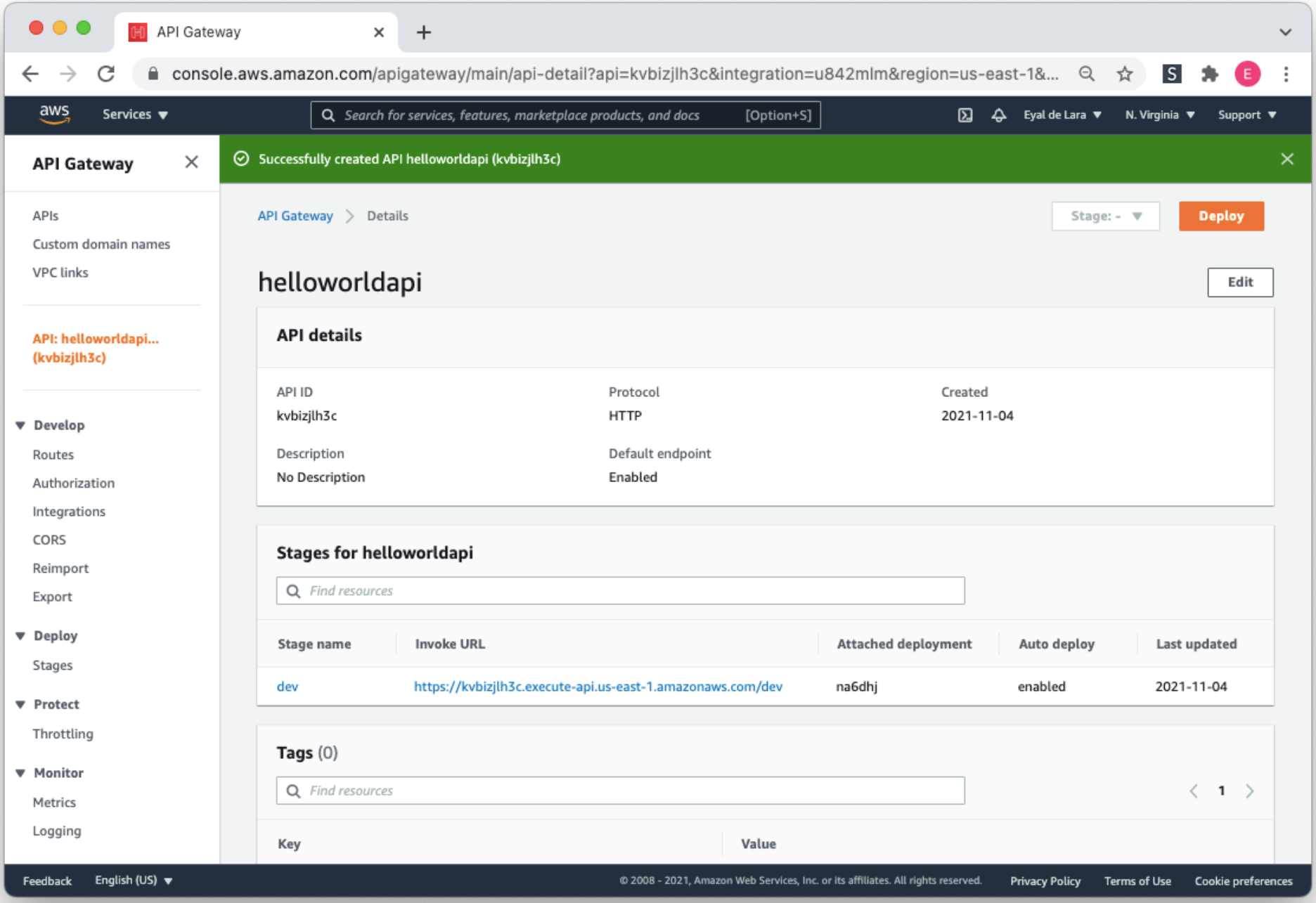
Stages Edit

Stages
dev (Auto-deploy: enabled)

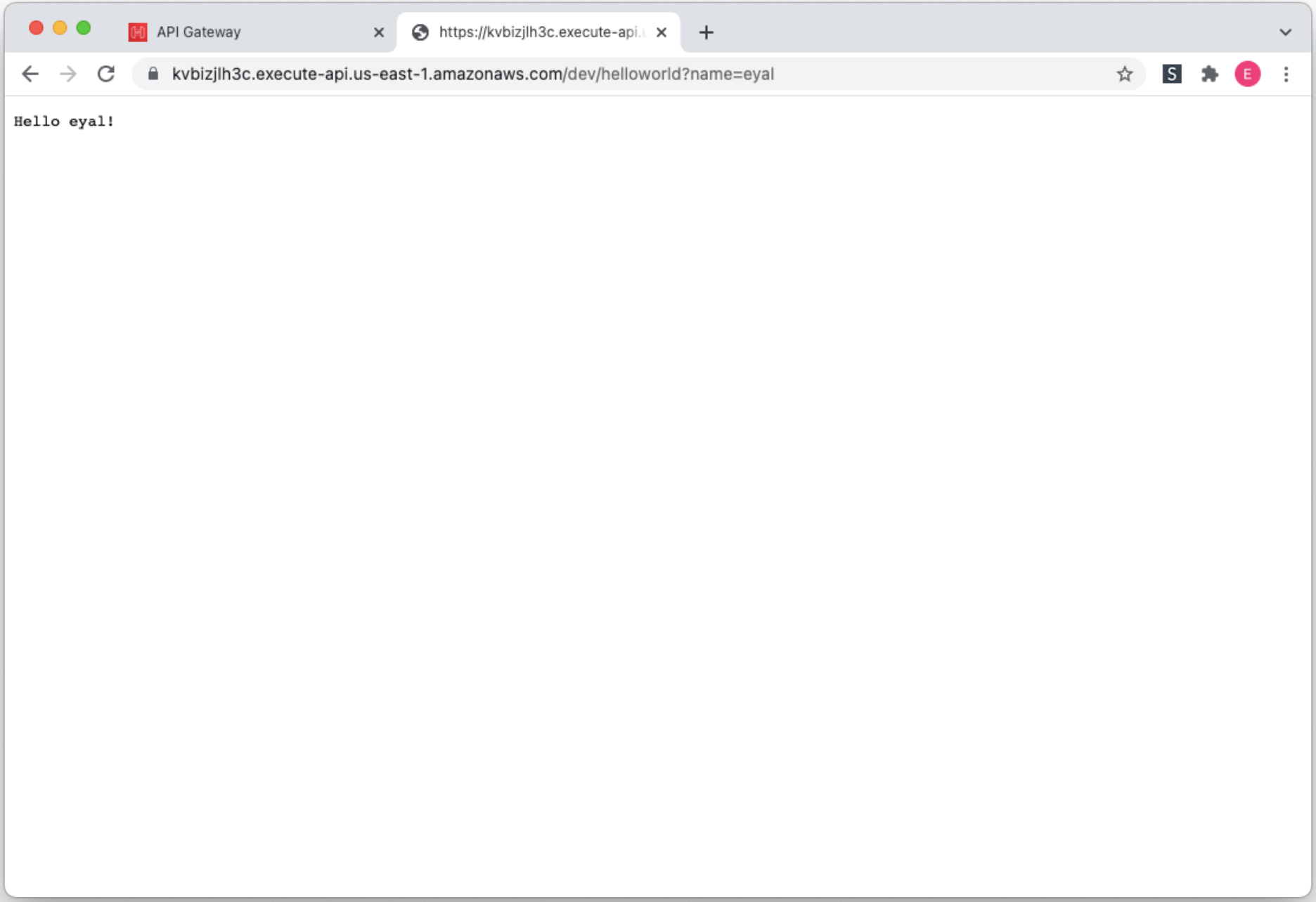
Cancel Previous Create

Feedback English (US) ▼ Privacy Policy Terms of Use Cookie preferences

Function as Web API



Function as Web API



Zappa

- Deploy Python WSGI applications on AWS Lambda + API Gateway.
- https://github.com/Miserlou/Zappa
- Support limited to Python 3.6, 3.7, 3.8

- Steps:
  - Install and configure AWS CLI
  - Create flask directory structure
  - Create a python virtual environment:

    > virtualenv -p python3.8 venv

    > source venv/bin/activate
  - Install flask

    > pip install flask
  - Install zappa

    > pip install zappa

    > zappa init

Zappa

- zappa_settings.json

```
{
    "dev": {
        "project_name": "flask_zappa",
        "keep_warm": false,
        "debug": true,
        "log_level": "DEBUG",
        "s3_bucket": "ece1779fall2017",
        "app_function": "app.webapp",
        "http_methods": ["GET","POST"],
        "parameter_depth": 1,
        "timeout_seconds": 300,
        "memory_size": 128,
        "use_precompiled_packages": true
    }
}
```

- Deploy application to AWS Lambda

  > zappa deploy dev