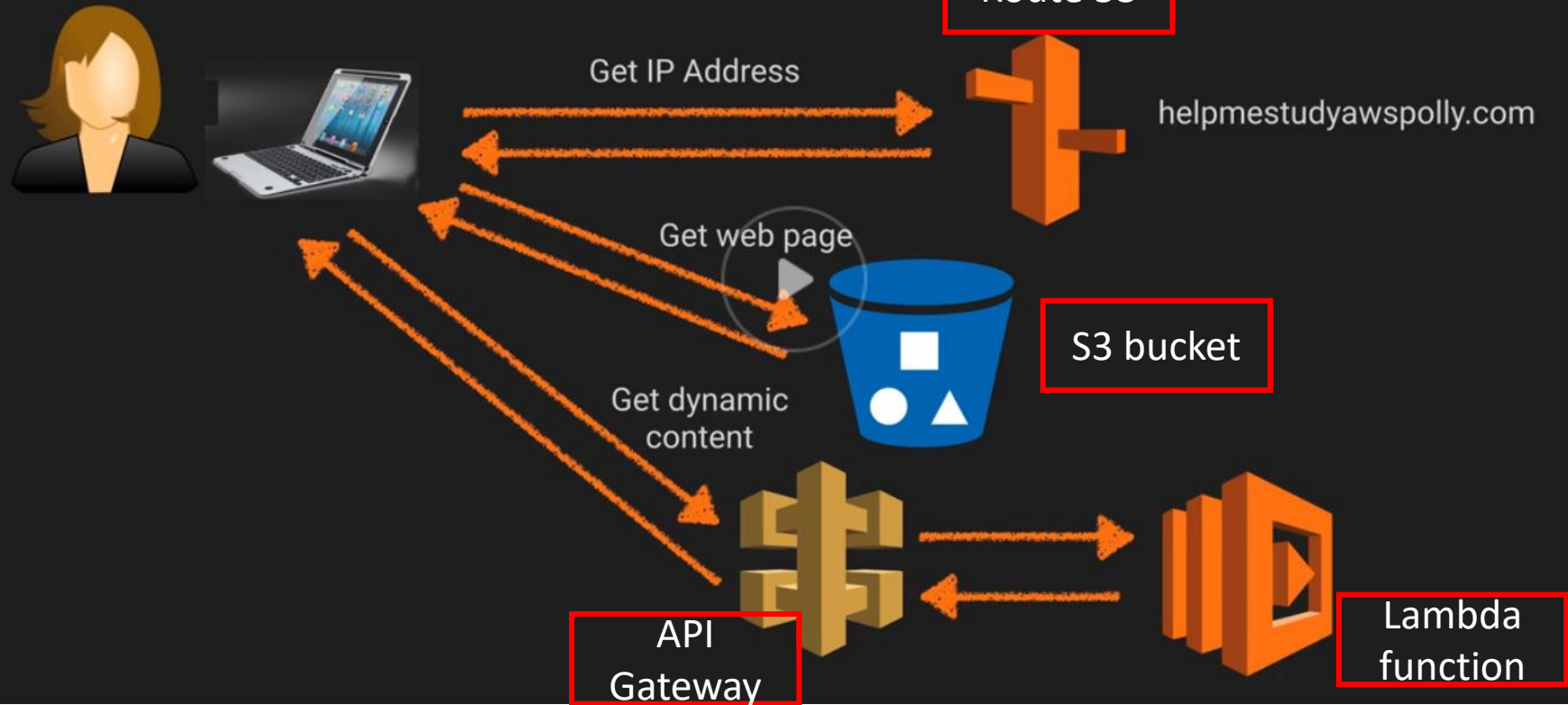


Overview Of Our Serverless Website



When we register domain name using static website in the s3, we need to look for 2 things – domain name is available and bucket name is available. Domain name and bucket name should be the same.

Create s3 static website bucket with index.html and error.html


Services → networking and content delivery → route 53 → register domain name → make sure the domain is available.


S3 bucket name and domain name should be same.


Services → compute → lambda → create lambda function

Lambda > Functions > Create function

Create function

Author from scratch
Start with a simple "hello world" example.


Blueprints
Choose a preconfigured template as a starting point for your Lambda function.


Serverless Application Repository
Find and deploy serverless apps published by developers, companies, and partners on AWS.


Author from scratch [info](#)

Name

Runtime

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.

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.

This new role will be scoped to the current function. To use it with other functions, you can modify it in the IAM console.

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.

Cancel **Create function**

3 ways of creating lambda function

Lambda > Functions > MyServerlessWebsite

ARN - arn:aws:lambda:us-east-1:673164704829:function:MyServerlessWebsite

MyServerlessWebsite

2

Throttle Qualifiers Actions Select a test event.. Test Save

✔ Congratulations! Your Lambda function "MyServerlessWebsite" has been successfully created. You can now change its code and configuration. Click on the "Test" button to input a test event when you are ready to test your function.

Configuration Monitoring

▼ Designer

Add triggers

Click on a trigger from the list below to add it to your function.

- API Gateway
- AWS IoT
- Alexa Skills Kit
- Alexa Smart Home
- CloudFront

Add triggers from the list on the left

MyServerlessWebsite

Amazon CloudWatch Logs

Amazon DynamoDB

Resources the function's role has access to will be shown here

Lambda > Functions > MyServerlessWebsite

ARN - arn:aws:lambda:us-east-1:673164704829:function:MyServerlessWebsite

MyServerlessWebsite

4

Throttle Qualifiers Actions Select a test event.. Test Save

✔ Congratulations! Your Lambda function "MyServerlessWebsite" has been successfully created. You can now change its code and configuration. Click on the "Test" button to input a test event when you are ready to test your function.

Configuration Monitoring

▼ Designer

Add triggers

Click on a trigger from the list below to add it to your function.

- API Gateway
- AWS IoT
- Alexa Skills Kit
- Alexa Smart Home
- CloudFront

Add triggers from the list on the left

MyServerlessWebsite

API Gateway Configuration required

Amazon CloudWatch Logs

Amazon DynamoDB

Resources the function's role has access to will be shown here

MyServerlessWebsite

Function code info

Code entry type

Edit code inline

Runtime

Python 3.6

Handler info

lambda_function.lambda_handler

File Edit Find View Goto Tools Window

```
1 def lambda_handler(event, context):
2     print("In lambda handler")
3
4     resp = {
5         "statusCode": 200,
6         "headers": {
7             "Access-Control-Allow-Origin": "*",
8         },
9         "body": "Ryan Kroonenburg"
10    }
11
12    return resp
```

9:33 Python Spaces: 4

Configure triggers

5

We'll set up an API Gateway endpoint with a [proxy integration type](#) (learn more about the [input](#) and [output](#) format). Any method (GET, POST, etc.) will trigger your integration. To set up more advanced method mappings or subpath routes, visit the [Amazon API Gateway console](#).

API

Pick an existing API, or create a new one.

Create a new API

API name

Enter a name to uniquely identify your API.

myServerlessWebsite

Deployment stage

The name of your API's deployment stage.

prod

Security

Configure the security mechanism for your API endpoint.

Open

Warning: Your API endpoint will be publicly available and can be invoked by all users.

Lambda will add the necessary permissions for Amazon API Gateway to invoke your Lambda function from this trigger. [Learn more](#) about the Lambda permissions model.

Cancel Add

Exam question on triggers – which are not aws triggers?

API Gateway 6

myServerlessWebsite
arn:aws:execute-api:us-east-1:673164704829:g13hekaz2l/*/*/MyServerlessWebsite

Details

- Security: NONE
- Method: ANY
- Resource path: /MyServerlessWebsite
- API name: myServerlessWebsite
- API: api-gateway/g13hekaz2l/*/*/MyServerlessWebsite
- Authorization: NONE
- Invoke URL: <https://g13hekaz2l.execute-api.us-east-1.amazonaws.com/prod/MyServerlessWebsite>
- Stage: prod

Resources **Actions** /MyServerlessWebsite - ANY - Method Execution 7

/MyServerlessWebsite
ANY

TEST

Client

Method Request
Auth: NONE
ARN: arn:aws:execute-api:us-east-1:673164704829:g13hekaz2l/*/*/MyServerlessWebsite

Integration Request
Type: LAMBDA_PROXY

Method Response
HTTP Status: Proxy
Models:

Integration Response
Proxy integrations cannot be configured to transform responses.

Lambda MyServerlessWebsite

Resources **Actions** /MyServerlessWebsite - GET - Method Execution 11

/MyServerlessWebsite
GET

TEST

Client

Method Request
Auth: NONE
ARN: arn:aws:execute-api:us-east-1:673164704829:g13hekaz2l/*/*/GET/MyServerlessWebsite

Integration Request
Type: LAMBDA_PROXY

Method Response
HTTP Status: Proxy
Models: application/json => Empty

Integration Response
Proxy integrations cannot be configured to transform responses.

Lambda MyServerlessWebsite

Resources **Actions** /MyServerlessWebsite - ANY - Method Execution 8

/MyServerlessWebsite
ANY

METHOD ACTIONS
Edit Method Documentation
Delete Method
Create Method
Create Resource
Enable CORS
Edit Resource Documentation
Delete Resource

RESOURCE ACTIONS
Deploy API
Import API
Edit API Documentation
Delete API

Method Request
Auth: NONE
ARN: arn:aws:execute-api:us-east-1:673164704829:g13hekaz2l/*/*/MyServerlessWebsite

Method Response
HTTP Status: Proxy
Models:

Resources **Actions** /MyServerlessWebsite 9

/MyServerlessWebsite

ANY
DELETE
GET
HEAD
OPTIONS
PATCH
POST
PUT

Resources **Actions** /MyServerlessWebsite - GET - Setup 10

/MyServerlessWebsite
GET

Choose the integration point for your new method.

Integration type ☒ Lambda Function ⓘ
☐ HTTP ⓘ
☐ Mock ⓘ
☐ AWS Service ⓘ
☐ VPC Link ⓘ

Use Lambda Proxy integration ☒ ⓘ

Lambda Region us-east-1

Lambda Function MyServerlessWebsite ⓘ


Use Default Timeout ☒ ⓘ

Save

Add Permission to Lambda Function

You are about to give API Gateway permission to invoke your Lambda function:
arn:aws:lambda:us-east-1:673164704829:function:MyServerlessWebsite

Cancel **OK**

Resources **Actions** 

METHOD ACTIONS

- Edit Method Documentation
- Delete Method

RESOURCE ACTIONS

- Create Method
- Create Resource
- Enable CORS
- Edit Resource Documentation
- Delete Resource

API ACTIONS

- Deploy API
- Import API
- Edit API Documentation
- Delete API

Method Request

Auth: NONE

ARN: arn:aws:execute-api:us-east-1:673164704829:g13hekaz2i/~/MyServerlessWebsite

Method Response

HTTP Status: Proxy

Models:

12

Deploy API 

13

Choose a stage where your API will be deployed. For example, a test version of your API could be deployed to a stage named beta.

Deployment stage

prod

Deployment description

My First Prod Deployment

Cancel

Deploy

APIs

myServerlessWebsite

Resources

Stages

Authorizers

Gateway Responses

Models

Resource Policy

Documentation

Stages

Create

prod - GET - /MyServerlessWebsite

Invoke URL: <https://g13hekaz2i.execute-api.us-east-1.amazonaws.com/prod/MyServerlessWebsite>

Use this page to override the **prod** stage settings for the GET to /MyServerlessWebsite method.

Settings

- ☒ Inherit from stage
- ☐ Override for this method

Save Changes

14

15

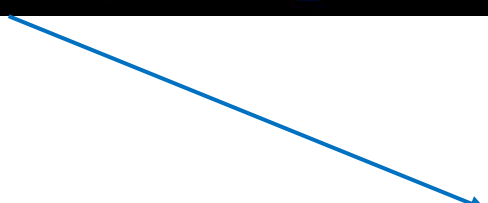
Ryan Kroonenburg

Accessing API gateway using ec2 cli

```
[root@ip-172-31-25-145 ec2-user]# curl "https://g13hekaz2i.execute-api.us-east-1.amazonaws.com/prod/MyServerlessWebsite" > myname.txt
```

| % Total | % Received | % Xferd | Average | Speed | Time | Time | Time |
|---------|------------|---------|---------|--------|-------|-------|----------|
| Current | | | Dload | Upload | Total | Spent | Left |
| Speed | | | | | | | |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | --:--:-- |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | --:--:-- |
| 100 | 16 | 100 | 16 | 0 | 0 | 82 | --:--:-- |
| 82 | | | | | | | |

```
[root@ip-172-31-25-145 ec2-user]# nano myname.txt
```



Ryan Kroonenburg



```
1 <html>
2 <head>
3 <script>
4   function myFunction() {
5     var xhttp = new XMLHttpRequest();
6     xhttp.onreadystatechange = function() {
7       if (this.readyState == 4 && this.status == 200) {
8         document.getElementById("my-demo").innerHTML = this.responseText;
9       }
10    };
11    xhttp.open("GET", "https://g13hekaz2i.execute-api.us-east-1.amazonaws.com/prod/MyServerlessWebsite", true);
12    xhttp.send();
13  }
14 </script>
15 </head>
16 <body>
17 <div align="center">
18   <br>
19   <br>
20   <br>
21   <br>
22   <br>
23   <h1>Hello <span id="my-demo">Cloud Gurus!</span></h1>
24   <button onclick="myFunction()">Click me</button>
25   <br>
26   
27 </div>
28 </body>
29 </html>
```

Add the api gateway url.

Upload the index.html and error.html to the s3 bucket.

Services → networking and content delivery → route 53 → hosted zones → selected the domain name

The screenshot shows the AWS Route 53 console. At the top, there are buttons: 'Back to Hosted Zones', 'Create Record Set' (highlighted with a red box), 'Import Zone File', 'Delete Record Set', and 'Test Record Set'. Below these is a search bar for 'Record Set Name' and a filter dropdown set to 'Any Type'. A table lists record sets for the domain 'helpmestudyawspolly.com'. The first record set is of type 'NS' with values 'ns-1427.awsdns-50.org.', 'ns-1652.awsdns-14.co.uk.', 'ns-321.awsdns-40.com.', and 'ns-967.awsdns-56.net.'. The second record set is of type 'SOA' with value 'ns-1427.awsdns-50.org. awsdns-hostmaster.amazon.com.'. A large play button is overlaid on the table.

Create Record Set

Name: helpmestudyawspolly.com

Type: A – IPv4 address

Alias: ☒ Yes ☐ No

Alias Target: s3-website-us-east-1.amazonaws.com

Alias Hosted Zone ID: Z3AQBSTGFYJSTF

You can also type the domain name for the resource. Examples:

- CloudFront distribution domain name: d1111111abcdef8.cloudfront.net
- Elastic Beanstalk environment CNAME: example.elasticbeanstalk.com
- ELB load balancer DNS name: example-1.us-east-1.elb.amazonaws.com
- S3 website endpoint: s3-website-us-east-2.amazonaws.com
- Resource record set in this hosted zone: www.example.com

[Learn More](#)

Routing Policy: Simple

Route 53 responds to queries based only on the values in this record. [Learn More](#)

Evaluate Target Health: ☐ Yes ☒ No

We should see the s3 bucket name. else, it means our s3 bucket name and domain name aren't same.

Hello Cloud Gurus!

Click me

helpmestudyawspolly.com

Click me hits the api gateway and this triggers the lambda function and returns the response back.

