

LAMBDA HANDSON 2

Thursday, 12 August 2021 20.12

Hands-on Lambda-02 : Setting S3 Bucket, CORS event, Lambda and API Gateway

The topics for this hands-on session will be AWS Lambda, function as a service (FaaS). During this playground you will create a website hosted on AWS S3 using AWS Lambda and Amazon API Gateway to add dynamic functionality to the site.

Learning Outcomes

At the end of this hands-on training, students will be able to;

- Create a static website on S3 bucket.
- Create a lambda function, that generates a random number and another function that processes form GET and PUT requests.
- Use API Gateway to expose lambda function to static website hosted on S3 bucket

Outline

- Part 1 - Prep - Creating a static web hosting on S3
- Part 2 - Creating a CORS event on S3 Bucket
- Part 3 - Create a Lambda Function with API Gateway

FARKLI bucketlar olusturup ayaga kadiracvgioz birindeki dosyayı digerde kullanacagiz sonunda lambda ile s3 birllestirecegiz.

Part 1 - Prep - Launching an Instance

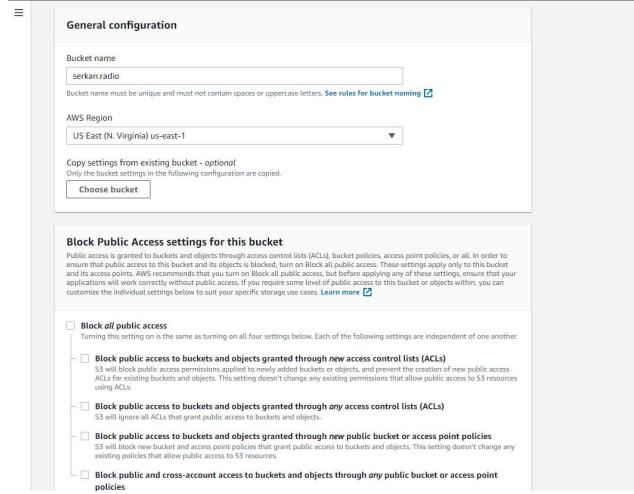
STEP 1 : Prep - Creating a static web hosting on S3

! [lab-001 Static Website]

(<https://raw.githubusercontent.com/ForestTechnologiesLtd/devopsplayground11-lambda/master/diagrams/pg11-lab-001.png>)

- Go to S3 menu using AWS console
- Create a bucket of `clarusway.broadcast` with following properties,

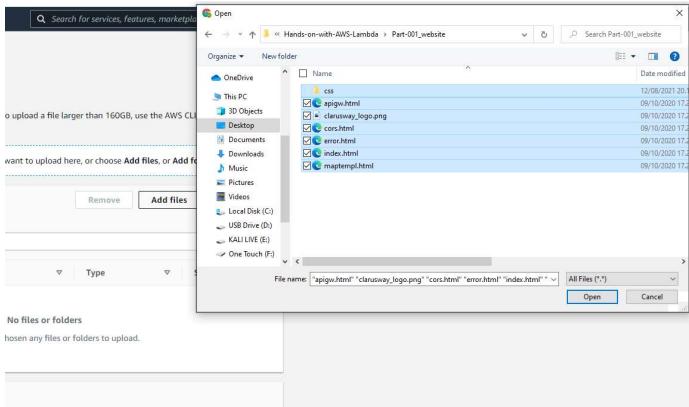
```
```text
Region : US East (N.Virginia)
Versioning : Disable
Server access logging : Disabled
Tagging : 0 Tags
Object-level logging : Disabled
Default encryption : None
Cloudwatch request metrics : Disabled
Object lock : Disabled
Block all public access : Unchecked (Public)
PS: Please, do not forget to select "US East (N.Virginia)" as Region
```



Create bucket diyoruz.

- Upload files from folder part-001\_website

- ```
``` 
1. Select files
- (all files minus css folder)
- Click 'Next'
2. Set Permissions
- Click 'Next'
3. Set Properties
- Click 'Upload'
```



No files or folders

hosted any files or folders to upload.

Name	Folder	Type	Size
960.css	css/	text/css	7.2 KB
960_12_col.css	css/	text/css	2.9 KB
960_12_col rtl.css	css/	text/css	2.9 KB
960_16_col.css	css/	text/css	3.7 KB
960_16_col rtl.css	css/	text/css	3.8 KB
960_24_col.css	css/	text/css	5.4 KB
960_24_col rtl.css	css/	text/css	5.5 KB
960_rtl.css	css/	text/css	5.7 KB
apigw.html	-	text/html	1.9 KB
clarusway_logo.png	-	image/png	4.2 KB

**Destination**

s3://serkan.radio

▶ Destination details

Bucket settings that impact new objects stored in the specified destination.

▶ Permissions

Grant public access and access to other AWS accounts.

▶ Properties

Specify storage class, encryption settings, tags, and more.

Cancel Upload

Hepsini yukledik.

S3 public yapmak icin policysinde bir degisiklik yapacagiz.

Create subfolder css using S3 web console

- Goto bucket "clarusway.broadcast"
- Click Create folder
- Name: css
- Click 'Save'
- Upload css folder.

- Set the static website bucket policy as shown below (`PERMISSIONS` >> `BUCKET POLICY`) and change `bucket-name` with your own bucket.

```
```json
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "PublicReadGetObject",
            "Effect": "Allow",
            "Principal": "*",
            "Action": "s3:GetObject",
            "Resource": "arn:aws:s3:::don't forget to change me/*"
        }
    ]
}
```

```

The screenshot shows the AWS S3 Bucket Policy editor. At the top, there's a navigation bar with tabs like 'Services', 'Search for services, features, marketplace products, and docs', and a search bar. Below the navigation is the title 'Edit bucket policy' with a 'Info' link. A 'Bucket policy' section explains what it does: 'The bucket policy, written in JSON, provides access to the objects stored in the bucket. Bucket policies don't apply to objects owned by other accounts.' It includes links to 'Learn more' and 'Policy examples'. The main area contains the JSON policy code:

```

1 {
2 "version": "2012-10-17",
3 "Statement": [
4 {
5 "Sid": "PublicReadGetObject",
6 "Effect": "Allow",
7 "Principal": "*",
8 "Action": "s3:GetObject",
9 "Resource": "arn:aws:s3:::serkan.radio/*"
10 }
11]
12 }
13

```

Below the policy is a 'Preview external access' button. At the bottom, there are 'Feedback' and 'English (US)' buttons, and a copyright notice: '© 2008 - 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.'

Bu policy napiyor.  
Source id : s3 bucketdan alma yetkisi  
Allow izin ver

- Herseye
- Resource kaynak /\* icindeki herseye

- Enable Static website hosting
- Properties > Static website hosting
- Select 'Use this bucket to host a Website'
- Index document: index.html
- Error document: error.html
- Click 'Save'

This screenshot shows the 'Static website hosting' configuration page for a bucket. It includes sections for enabling static hosting, choosing a hosting type (static website), and specifying index and error documents. A note about making content publically readable is present. The 'Index document' field is set to 'index.html'.

<http://serkan.radio.s3-website-us-east-1.amazonaws.com/>

- → C Not secure serkan.radio.s3-website-us-east-1.amazonaws.com Apps Gmail Lists Questions and... Your first Python pr... CLARUSWAY-dev-750... AWS Management... Python conditional... Home | Codewars 3. Data Structures... Python Tutorials

The screenshot shows the 'Clarusway AWS&DEVOPS PATH Lambda API Gateway Lab' interface. It features a navigation bar with 'Home', 'CORS', 'API Gateway', and 'Mapping Template'. The main content area has a large 'STATIC CONTENT' heading. Below it, a note says 's3 Sample Static website'.

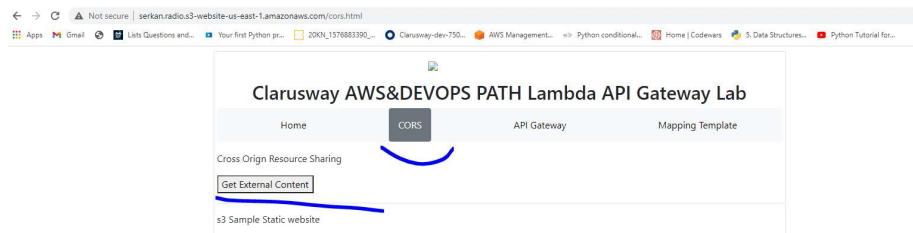
~ ~ ~

- Open the Endpoint in a web browser.  
<http://clarusway.broadcast.s3-website-us-east-1.amazonaws.com>  
 and click the sub-links and show the massages

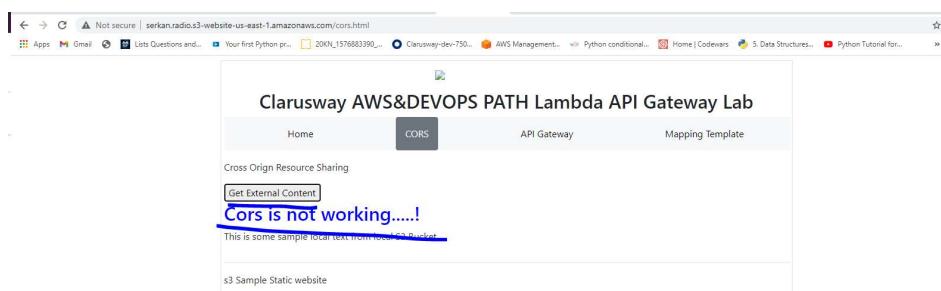
Burasi bizim home sayfamiz henuz bir sey yamadik

Bucket object bolumune tekrar geliyoruz.  
 Sonrada klasore gidiyoruz  
 Post2 corsa giriyoruz  
 Cors.html ve demo.txt yi yukluyoruz  
 Dah aonce vardi ama calismiyordu

```
Part 2 - Creating a CORS event on S3 Bucket
![Lab-002 Static Website using CORS]
(https://raw.githubusercontent.com/ForestTechnologiesLtd/devopsplayground11-lambda/master/diagrams/pg11-lab-002.png)
STEP 1: Upload new html files
- Go to S3 bucket named "clarusway.broadcast"
- Upload files from folder part-002_CORS:
 - cors.html
 - demo_text.html
```



- Open the Endpoint in a web browser and show CORS sub-link.  
<http://clarusway.broadcast.s3-website-us-east-1.amazonaws.com>  
 With a web browser visit the static website from Part-001 and click on the menu item 'CORS' the RED text "LAB NOT STARTED" should disappeared if not refresh the web page.  
 Click on the button Get External Content. Javascript code you edited at the start of this lab has now pull text from the file demo\_text.



Cors html degistirince yeni sayfada yukaridaki gibi oldu

- STEP 2 : Create Second S3 Bucket for using CORS
- Create a bucket of **clarusway.cors.broadcast** with following properties,

## Yeni bucket olusturacagiz

Create bucket [Info](#)  
Buckets are containers for data stored in S3. Learn more [?](#)

**General configuration**

Bucket name: cors.broadcast.serkan  
Bucket name must be unique and must not contain spaces or uppercase letters. See rules for bucket naming [?](#)

AWS Region: US East (N. Virginia) us-east-1

Copy settings from existing bucket - optional  
Only the bucket settings in the following configuration are copied.  
[Choose bucket](#)

**Block Public Access settings for this bucket**  
Public access is granted to buckets and objects through access control lists (ACLs), bucket policies, access point policies, or all. In order to ensure that public access to this bucket and its objects is blocked, turn on Block all public access. These settings apply only to this bucket and its contents. Note that you can still grant public access to an object by applying any of these settings, ensure that your application will work correctly without public access. If you require some level of public access to this bucket or objects within, you can customize the individual settings below to suit your specific storage use cases. [Learn more \[?\]\(#\)](#)

**Block all public access**  
Turning this setting on is the same as turning on all four settings below. Each of the following settings are independent of one another.

**Block public access to buckets and objects granted through new access control lists (ACLs)**  
S3 will block public access permissions applied to newly added buckets or objects, and prevent the creation of new public access ACLs for existing buckets and objects. This setting doesn't change any existing permissions that allow public access to S3 resources using ACLs.

**Block public access to buckets and objects granted through any access control lists (ACLs)**  
S3 will ignore all ACLs that grant public access to buckets and objects.

**Block public access to buckets and objects granted through new public bucket or access point policies**  
S3 will block new bucket and access point policies that reveal public access to buckets and objects. This setting doesn't change any

```
```text
Region : US East (N.Virginia)
Versioning : Disable
Server access logging : Disabled
Tagging : 0 Tags
Object-level logging : Disabled
Default encryption : None
CloudWatch request metrics : Disabled
Object lock : Disabled
Block all public access : Unchecked (Public)
PS: Please, do not forget to select "US East (N.Virginia)" as Region
```

```

## Bucketimiz olustu

Amazon S3 [X](#)

Buckets [cors.broadcast007](#)

Access Points

Object Lambda Access Points

Batch Operations

Access analyzer for S3

Block Public Access settings for this account

Storage Lens

Dashboards

AWS Organizations settings

Feature spotlight

AWS Marketplace for S3

**cors.broadcast007** [Info](#)

Objects Properties Permissions Metrics Management Access Points

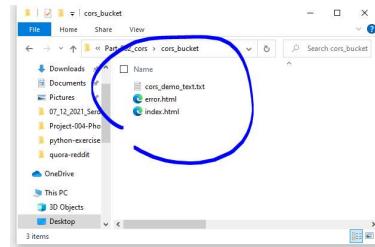
**Objects (0)**  
Objects are the fundamental entities stored in Amazon S3. You can use [Amazon S3 inventory](#) to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. [Learn more \[?\]\(#\)](#)

[Actions](#) [Create folder](#) [Upload](#)

Find objects by prefix

| Name       | Type | Last modified | Size | Storage class |
|------------|------|---------------|------|---------------|
| No objects |      |               |      |               |

Objekty yukluyoruz part2 deki cors\_bucket klasoru tum dosyalarini yukluyoruz



Upload succeeded  
View details below.

**Upload: status**

The information below will no longer be available after you navigate away from this page.

**Summary**

| Destination            | Succeeded                  | Failed            |
|------------------------|----------------------------|-------------------|
| s3://cors.broadcast007 | 3 files, 653.0 B (100.00%) | 0 files, 0 B (0%) |

**Files and folders** (3 Total, 653.0 B)

| Name               | Folder | Type       | Size    | Status    | Error |
|--------------------|--------|------------|---------|-----------|-------|
| cors_demo_text.txt | -      | text/plain | 158.0 B | Succeeded | -     |
| error.html         | -      | text/html  | 254.0 B | Succeeded | -     |
| index.html         | -      | text/html  | 241.0 B | Succeeded | -     |

Set the static website bucket policy as shown below (`PERMISSIONS` >> `BUCKET POLICY`) and change `bucket-name` with your own bucket.

```
```json
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "PublicReadGetObject",
            "Effect": "Allow",
            "Principal": "*",
            "Action": "s3:GetObject",
            "Resource": "arn:aws:s3:::don't forget to change me/*"
        }
    ]
}
```

```

Amazon S3 > cors.broadcast007 > Edit bucket policy

Edit bucket policy [Info](#)

**Bucket policy**

The bucket policy, written in `JSON`, provides access to the objects stored in the bucket. Bucket policies don't apply to objects owned by other accounts. Learn more [\[?\]](#)

[Policy examples](#) [\[?\]](#) [Policy generator](#) [\[?\]](#)

**Bucket ARN**

arnaws3:::cors.broadcast007

**Policy**

```

1: {
2: "Version": "2012-10-17",
3: "Statement": [
4: {
5: "Sid": "PublicReadGetObject",
6: "Effect": "Allow",
7: "Principal": "*",
8: "Action": "s3:GetObject",
9: "Resource": "arn:aws:s3:::cors.broadcast007/*"
10: }
11:]
12: }

```

[Preview external access](#)

Okuma ve obje almaya izin veriyor

Properties e gidip, static website enable ediyoruz.  
Index.html ekliyoruz

Use this bucket to host a website or redirect requests. [Learn more](#)

Disable  
 Enable

**Hosting type**

Host a static website  
 Use the bucket endpoint as the web address. [Learn more](#)

Redirect requests for an object  
 Redirect requests to another bucket or domain. [Learn more](#)

**For your customers to access content at the website endpoint, you must make all your content publicly readable.** To do so, you can edit the S3 Block Public Access settings for the bucket. For more information, see [Using Amazon S3 Block Public Access](#)

**Index document**  
 Specify the home or default page of the website.

**Error document**  
This is returned when an error occurs.

**Redirection rules – optional**  
 Redirection rules, written in JSON, automatically redirect webpage requests for specific content. [Learn more](#)

1 |

- Upload files from the folder Part-002\_cors\cors\_bucket to your new bucket "clarusway.cors.broadcast"
  - cors\_demo\_text.txt
  - index.html
  - error.html
- Accept defaults, click 'Upload'
- Enable Static website hosting

```
```text
```

Properties > Static website hosting

Select 'Use this bucket to host a Website'

Index document: index.html

Error document: error.html

Click 'Save'

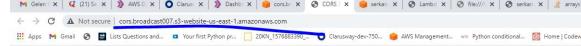
```
```
```

- Open the Endpoint in a web browser.

<http://clarusway.cors.broadcast.s3-website-us-west-2.amazonaws.com>

show that CORS option still inactive but page is running

Properties deki sondaki endpointi calisitiriyoruz



Sample CORS Page

Bu sayfa geliyor.

Corsdemo\_text.txt dosyasına tıklıyoruz bucketimiz içinden  
 Object url kopyaliyoruz

Object overview

Owner: sersal95  
AWS Region: US East (N. Virginia) us-east-1  
Last modified: August 12, 2021, 20:36:02 (UTC+03:00)  
Size: 158.0 B  
Type: txt  
Key: cors\_demo\_text.txt

S3 URI: s3://cors.broadcast007/cors\_demo\_text.txt  
Amazon Resource Name (ARN): arn:aws:s3:::cors.broadcast007/cors\_demo\_text.txt  
Entity tag (Etag): f5ca8be76c7146aad99cf0006794cd3  
Object URL: https://s3.amazonaws.com/cors.broadcast007/cors\_demo\_text.txt

Object management overview

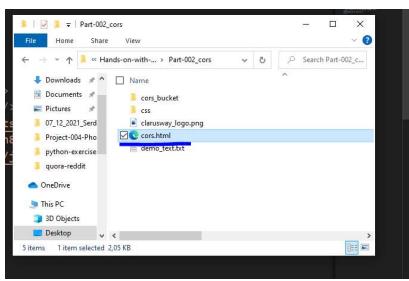
Bucket properties

Bucket Versioning: When enabled, multiple variants of an object can be stored in the bucket to easily recover from unintended changes.

Management configurations

Replication status: When a replication rule is applied to an object, the replication status indicates the progress of the operation.

- Open Part-002\_cors\cors.html



Bunu aciyoruz vscode ile

- find the "demo\_text.txt" script in the html file
- REPLACE: demo\_text.txt WITH: [http://clarusway.cors.broadcast.s3-website-us-east-1.amazonaws.com/cors\\_demo\\_text.txt](http://clarusway.cors.broadcast.s3-website-us-east-1.amazonaws.com/cors_demo_text.txt) (URL of the cors\_demo\_text.txt file in the clarusway.cors.broadcast)

```
<!DOCTYPE html>
<html lang="en">

<head>
<meta charset="utf-8" />
<title>Lambda Lab</title>
<link rel="stylesheet" href="css/reset.css" />
<link rel="stylesheet" href="css/text.css" />
<link rel="stylesheet" href="css/960_24_col.css" />
<link rel="stylesheet" href="css/meetup_base.css" />
<link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css" integrity="sha384-JcKqj8iFw4ZqkxJbP0LSpRR0PZUqE4Lz7H1BtWf6t7oH3XoK6VUcAtHtWkXg" crossorigin="anonymous"/>
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.2.1/jquery.min.js" type="text/javascript"></script>
<script type="text/javascript">
$(document).ready(function () {
 $("button").click(function () {
 $("#s3_cors").load("https://s3.amazonaws.com/cors.broadcast007/cors_demo_text.txt");
 });
});
</script>
</head>

<body>
<div class="container_24 mt-2">
<div class="d-flex flex-column border mb-3">

<h2 class="text-center mt-3 ml-3 rounded">
 Clarusway AWSDEVOPS PATH Lambda API Gateway Lab
</h2>
<div class="grid_24 nav">
<p>
```

Ln 17, Col 90 Spaces: 2 UTF-8 LF Django Template R

Url i buraya yapistiriyoruz

Cors.html tekrar yukluyoruz

Sag tiklayip inspecte basiyoruz

```
- upload this file to the S3 bucket named "clarusway.broadcast"
- Click on the button Get External Content in the sub-link CORS. You will notice that nothing happens or sometimes it can be raised an error.
```text
This occurs because web browsers expect resources to be requested from the same domain. To resolve this issue AWS S3 has a feature called CORS (Cross Origin Resource Sharing) if you enable this feature this will allow the webpage to request the content from another bucket.
```
- With bucket clarusway.cors.broadcast enable CORS configuration, add a new policy
y.
```text
Permissions > CORS configuration
NOTE: Replace url in <AllowedOrigin> tag with your static website link from Part-1.
```
```bash
<?xml version="1.0" encoding="UTF-8"?>
<CORSConfiguration xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
<CORSRule>
  <AllowedOrigin>http://clarusway.broadcast.s3-website-us-west-2.amazonaws.com/</AllowedOrigin>
  <AllowedMethod>GET</AllowedMethod>
  <MaxAgeSeconds>3000</MaxAgeSeconds>
  <AllowedHeader>Authorization</AllowedHeader>
</CORSRule>
</CORSConfiguration>
```
- Click "Save"
- go to browser and refresh the web-page and show the CORS sub-link.
```

### Ikinci cors bucketa geliyoruz

Permisjon altına copy past yapacagiz

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <CORSConfiguration xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
3 <CORSRule>
4 <AllowedOrigin>http://clarusway.broadcast.s3-website-us-west-2.amazonaws.com/</AllowedOrigin>
5 <AllowedMethod>GET</AllowedMethod>
6 <MaxAgeSeconds>3000</MaxAgeSeconds>
7 <AllowedHeader>Authorization</AllowedHeader>
8 </CORSRule>
9 </CORSConfiguration>
```

Burdaki metni degisitirip asagidaki metni yazacagiz ve kaynagi kendi kaynagini yazacagiz  
Radio bucket endpointimizi ekleyecegiz

**Object Lock**  
Store objects using a write-once-read-many (WORM) model to help you prevent objects from being deleted or overwritten for a fixed amount of time or indefinitely. [Learn more](#)

Object Lock  
Disabled

Amazon S3 currently does not support enabling Object Lock after a bucket has been created. To enable Object Lock for this bucket, contact Customer Support.

**Requester pays**  
When enabled, the requester pays for requests and data transfer costs, and anonymous access to this bucket is disabled. [Learn more](#)

Requester pays  
Disabled

**Static website hosting**  
Use this bucket to host a website or redirect requests. [Learn more](#)

Static website hosting  
Enabled

Hosting type  
Bucket hosting

Bucket website endpoint  
When you configure your bucket as a static website, the website is available at the AWS Region-specific website endpoint of the bucket. [Learn more](#)

<http://serkan.radio.s3-website-us-east-1.amazonaws.com>

```
[
 {
 "AllowedHeaders": [
 "Authorization"
],
 "AllowedMethods": [
 "GET"
],
 "AllowedOrigins": [
 "http://serdar.clarusway.broadcast.s3-website-us-east-1.amazonaws.com"
],
 "MaxAgeSeconds": 3000
 }
]
```

From <<https://app.slack.com/client/T0227UVRJU8/C021BG84YJJ>>

**Cross-origin resource sharing (CORS)**  
The CORS configuration, written in JSON, defines a way for client web applications that are loaded in one domain to interact with resources in a different domain. [Learn more](#)

```

1+ [
2+ {
3+ "AllowedHeaders": [
4+ "Authorization"
5+],
6+ "AllowedMethods": [
7+ "GET"
8+],
9+ "AllowedOrigins": [
10+ "http://serdar.clarusway.broadcast.s3-website-us-east-1.amazonaws.com"
11+],
12+ "MaxAgeSeconds": 3000
13+
14]

```

Cancel **Save changes**

Daha önceki raido bucktimizin endpointine girip corsu ve ardindan get external contenti tiklayinca yukardaki neticeyi aliyoruz

Orjinler arası kaynak paylasimi sayesinde bunu yapabildik  
Bunu bize cors sagliyor.

Cross Origin Resource Sharing

From <<http://serkan.radio.s3-website-us-east-1.amazonaws.com/cors.html>>

Domainler arası kaynaklý áaylasimini CORS sagliyor.

**Cizli yere yapistiriyoruz**

**## Part 3 - Create a Lambda Function with API Gateway**

**SIMDI LAMBDA SAYFASINA GELIYROUZ KONSOLDAN**

**Lambda create function source hangisi degildir sinav sorusu**

```
![lab-003 Static Website with API Gateway and Lambda]
(https://raw.githubusercontent.com/ForestTechnologiesLtd/devopsplayground11-lambda/master/diagrams/pg11-lab-003.png)
STEP 1: Create Lambda Function
- Go to Lambda Service on AWS Console
- Functions ----> Create Lambda function
```text
1. Select Author from scratch
Name: NumberGenerator
Runtime: Python 2.7
Role: Create a new role with basic Lambda permissions
Click 'Create function'
```

```

Create function dedik

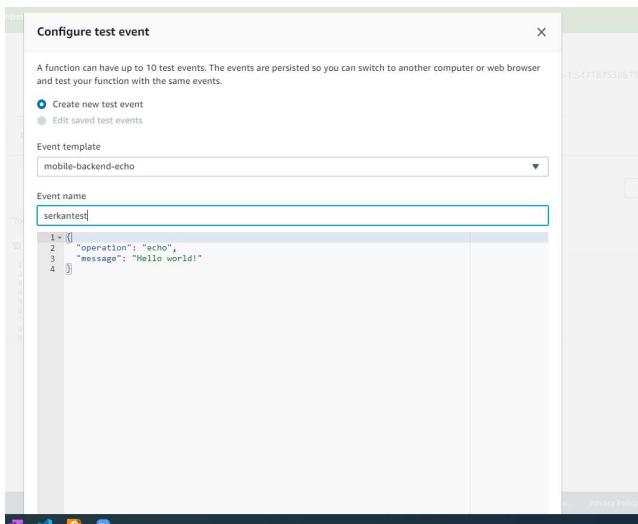
- Configuration of Function Code
- In the sub-menu of configuration go to the "Function code section" and paste code seen below

```
```python
from __future__ import print_function
from random import randint
print('Loading function')
def lambda_handler(event, context):
    myNumber = randint(0,1000)
    print("Random No. [ %s ]" % myNumber)
    return myNumber
```

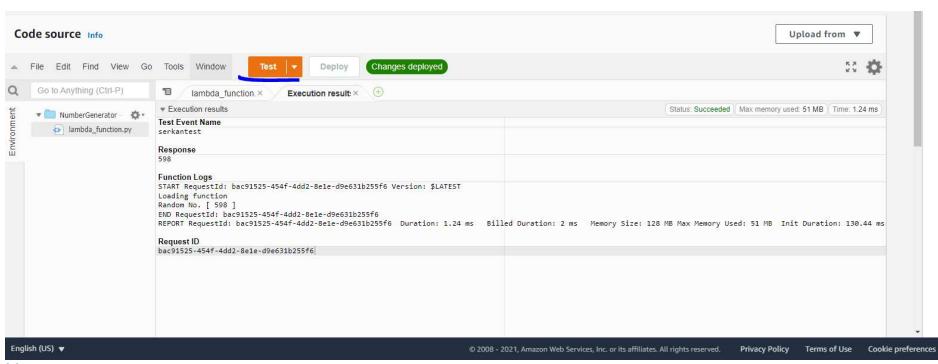
```

From <<https://app.slack.com/client/T0227UVRJU8/C021BG84YJJ>>

- ~~~
- Click "DEPLOY" button



- In the sub-menu of configuration go to the "Basic settings section" and click edit  
````  
Description : Function that generates a random number between 0 and 1000
Accept Defaults for other settings



STEP 2: Testing your function - Create test event
Click 'Test' button and opening page Configure test events
````

Select: Create new test event

Event template: Hello World

Event name: emptyClarus

Input test event as;

{}

Click 'Create'

Click 'Test'

You will see the message Execution result: succeeded(logs) and a random number in a box with a dotted line.

#### TESTIMIZ VE YENİ KODUMUZ CALISTI

STEP 3 : Create New 'API'

- Go to API Gateway on AWS Console
- Click "Create API"
- Select REST API ----> Build

AWS Lambda, or any publicly addressable web service. With Amazon API Gateway, you can generate custom client SDKs for your APIs, to connect your back-end systems to mobile, web, and server applications or services.

### Choose an API type

|                                                                                                                                                                                                                                                                                          |                                                                                                                                                                                                                                                    |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>HTTP API</b><br>Build low-latency and cost-effective REST APIs with built-in features such as OAuth2 and native CORS support.<br>Works with the following:<br>Lambda, HTTP backends<br><br><input type="button" value="Import"/> <input type="button" value="Build"/>                 | <b>WebSocket API</b><br>Build a WebSocket API using persistent connections for real-time use cases such as chat applications or dashboards.<br>Works with the following:<br>Lambda, HTTP, AWS Services<br><br><input type="button" value="Build"/> |
| <b>REST API</b><br>Develop a REST API where you gain complete control over the request and response along with API management capabilities.<br>Works with the following:<br>Lambda, HTTP, AWS Services<br><br><input type="button" value="Import"/> <input type="button" value="Build"/> | <b>REST API Private</b><br>Create a REST API that is only accessible from within a VPC.<br>Works with the following:<br>Lambda, HTTP, AWS Services<br><br><input type="button" value="Import"/> <input type="button" value="Build"/>               |

Choose the protocol : REST

Create new API : New API

Settings :

- Name: ClarusAPI
- Description: test lab api
- Endpoint Type: Regional
- Click 'Create API'

Amazon API Gateway APIs > Create

### Choose the protocol

Select whether you would like to create a REST API or a WebSocket API.

REST  WebSocket

### Create new API

In Amazon API Gateway, a REST API refers to a collection of resources and methods that can be invoked through HTTPS endpoints.

New API  Import from Swagger or Open API 3  Example API

### Settings

Choose a friendly name and description for your API.

|               |          |
|---------------|----------|
| API name*     | My API   |
| Description   |          |
| Endpoint Type | Regional |

\* Required

STEP 4 : Exposing Lambda via API Gateway

- 1. Add a New Child Resource
- APIs > ClarusAPI > Resources > /
- Click on Actions > 'Create Resource'

APIs Services ▾ APIs > SerkanAPI (974mj7aky7) > Resources > / (32x1vhjhg)

Actions ▾ Methods

No methods defined for the resource.

**RESOURCES ACTIONS**

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

**ACTIONS**

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

#### New Child Resource

- Configure as proxy resource: Leave blank
- Resource Name: Random Number
- Resource Path: /random-number
- Enable API Gateway CORS: Yes
- Click 'Create Resource' button

APIs Services ▾ APIs > SerkanAPI (974mj7aky7) > Resources > / (32x1vhjhg) > Create

Use this page to create a new child resource for your resource.

Configure as  proxy resource

Resource Name\*

Resource Path\*

You can add path parameters using brackets. For example, the resource path `(username)` represents a path parameter called `username`. Configuring `(proxy)` as a proxy resource catches all requests to its sub-resources. For example, it works for a GET request to `/foo`. To handle requests to `/`, add a new ANY method on the `/` resource.

Enable API Gateway CORS

\* Required

Cancel **Create Resource**

Enable Gateway corsu simdi degil sonra yapacagiz

APIs Services ▾ APIs > SerkanAPI (974mj7aky7) > Resources > /random (ks2gm3)

APIs Resources Actions ▾ /random Methods

No methods defined for the resource.

- 2. Add a GET method to resource /
- Actions > Create Method
- Under the resource a drop down will appear select GET method and click the 'tic k'.

The screenshot shows the AWS API Gateway console. On the left, the navigation bar includes 'APIs', 'Custom Domain Names', and 'VPC Links'. Under 'APIs', 'SerkanAPI' is selected. In the main area, 'Resources' is selected, and the path is shown as 'APIs > SerkanAPI (974mj7aky7) > Resources > /random (ks2gm3)'. A context menu is open over the '/random' resource, with 'Actions' expanded. The 'Create Method' option is highlighted. Other options in the 'RESOURCE ACTIONS' section include 'Create Resource', 'Enable CORS', 'Edit Resource Documentation', and 'Delete Resource'. Below this, the 'API ACTIONS' section includes 'Deploy API', 'Import API', 'Edit API Documentation', and 'Delete API'.

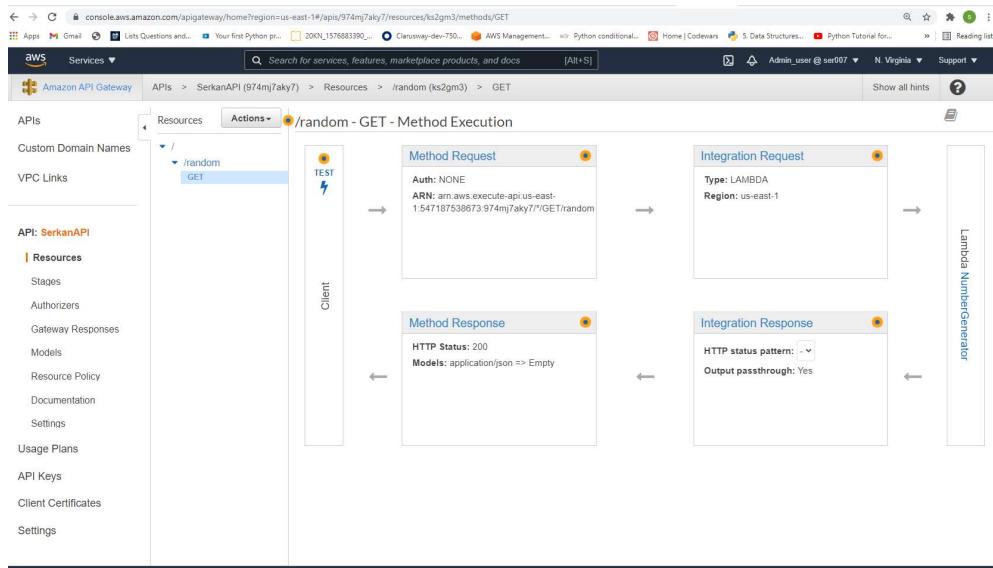
### - 3. / GET - Method Execution

- Integration type: Lambda Function
- Use Lambda Proxy integration: Leave blank
- Lambda Region: us-east-1
- Lambda Function: generateNumber
- Click 'Save'
- Confirm the dialog 'Add Permission to Lambda Function', Click 'OK'

The screenshot shows the configuration of the GET method for the '/random' resource. The 'Integration type' is set to 'Lambda Function'. The 'Lambda Region' is 'us-east-1'. The 'Lambda Function' field contains 'NumberGenerator'. A note at the bottom states: 'You do not have any Lambda Functions in us-east-1. Create a Lambda Function in your current account, or pr Access.' The 'Use Default Timeout' checkbox is checked. The 'Save' button is visible at the bottom right.

The screenshot shows the completed configuration of the GET method for the '/random' resource. The 'Integration type' is still set to 'Lambda Function'. The 'Lambda Region' is 'us-east-1'. The 'Lambda Function' field now contains 'NumberGenerator'. The 'Use Default Timeout' checkbox is checked. The 'Save' button is visible at the bottom right.

Lambda Functionı seçilip save yapıyorum



Musterinin requestini lambda functionuyla get methoduyla tekrar mustureliye donduruyoruz

#### STEP 5: Testing Lambda via API Gateway

- Click GET method under /random-number
- Click TEST link in the box labeled 'Client' At the bottom of the new view Click 'Test' button
- Under Response Body you should see a random number. Click the blue button labeled 'Test' again at the bottom of the screen and you will see a new number appear
- Test completed successfully

APIs > SerkanAPI > Resources > /random > GET

Method Execution /random - GET - Method Test

Request: /random

Status: 200

Latency: 271 ms

Response Body

```
{"x-amzn-trace-id": "Root=1-611569cd-e4722840b86637e1ad595575;Sampled=0", "Content-Type": "application/json"}
```

Logs

```
Execution log for request 330bb33a-6354-4c06-82f0-01b34f029f73
Thu Aug 12 18:34:53 UTC 2021 : Starting execution for request: 330bb33a-6354-4c06-82f0-01b34f029f73
Thu Aug 12 18:34:53 UTC 2021 : HTTP Method: GET, Resource Path: /random
Thu Aug 12 18:34:53 UTC 2021 : Method request path: {}
Thu Aug 12 18:34:53 UTC 2021 : Method request query string: {}
Thu Aug 12 18:34:53 UTC 2021 : Method request headers: {}
Thu Aug 12 18:34:53 UTC 2021 : Method request body before transformations:
Thu Aug 12 18:34:53 UTC 2021 : Endpoint request URI: https://lambda.us-east-1.amazonaws.com/2015-03-31/functions/arn:aws:lambda:us-east-1:54718753673:function:numberGenerator/invocations
Thu Aug 12 18:34:53 UTC 2021 : Endpoint request headers: {x-amzn-lambda-integ...
```

#### STEP 6 : Deploy API

- Click Resources select /random-number
- Select Actions and select Deploy API

APIs > SerkanAPI > Resources > /random > GET

METHOD ACTIONS

- Edit Method Documentation
- Delete Method**

RESOURCE ACTIONS

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

ACTIONS

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

Logs

Deployment stage: [New Stage]  
 Stage Name: dev  
 Stage description: Generate Number stage

Deployment stage: [New Stage]

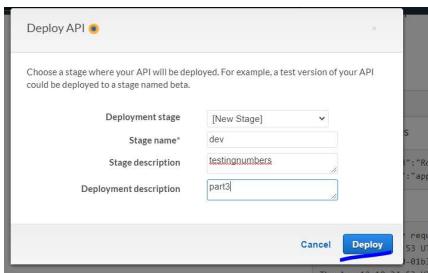
Stage name\*: dev

Stage description: Generate Number stage

Deployment description: Generate Number stage

Cancel Deploy

Deployment description: Part-3



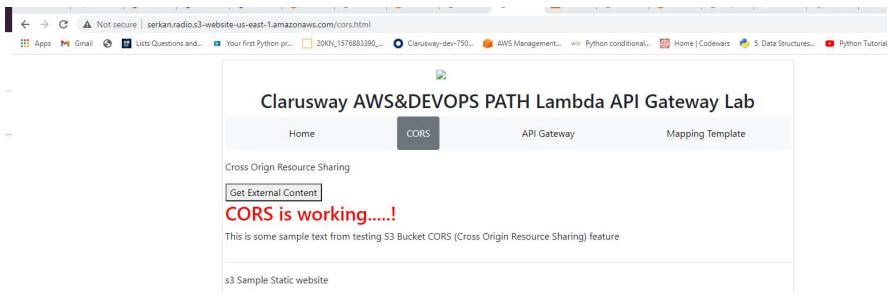
Lambda fonksiyonunu entegre ettigimiz API linki- it can be seen invoke URL on top like;  
["https://d3w0w4ajyh.execute-api.eu-central-1.amazonaws.com/dev"](https://d3w0w4ajyh.execute-api.eu-central-1.amazonaws.com/dev) and note it down somewhere.

- Entering the Invoke URL into the web browser and add "/random-number" at the end of the URL. Show the generated random number with refreshing the page in the web browser.



Dev/random yazip caistirdik

Numbergeneratorin API gateway e tiklayinca calismasini istiyoruz



STEP 7: Implementation of the API

C:\Users\serkan\Desktop\MYWORKSPACE\clarusway-aws-8-21\aws\hands-on\10\_10\_2020-Guile-S3-APIGW-Lambda\Hands-on-with-AWS-Lambda\Hands-on-with-AWS-Lambda\Part-003\_lambda\apigw.html

```

1 <!DOCTYPE html>
2 <html lang="en">
3
4 <head>
5 <meta charset="utf-8" />
6 <title>Lambda Lab</title>
7 <link rel="stylesheet" href="css/reset.css" />
8 <link rel="stylesheet" href="css/text.css" />
9 <link rel="stylesheet" href="css/960_24_col.css" />
10 <link rel="stylesheet" href="css/meetup_base.css" />
11 <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css"
12 integrity="sha384-JcKbq3iqJ61gNV9KGb8thSsNjpSl0n8PARnHuZOnIxN0hoP+VmmDGmn5t9UJ0Z" crossorigin="anonymous">
13 <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.2.1/jquery.min.js" type="text/javascript"></script>
14 <script type="text/javascript">
15 $(document).ready(function () {
16 $("button").click(function () {
17 $("#apigw").load("MY_API_GW_REQUEST");
18 });
19 });
20
21 </script>
22 </head>
23
24 <body>
25 <div class="container_24">
26 <div class="d-flex flex-column border mb-3">
27
28 <h2 class="text-center mt-3 ml-3 rounded">
29 Clarusway AWS&DEVOPS PATH Lambda API Gateway Lab
30 </h2>
31 </div>
32 <div class="grid_24 nav">
33 <p>
34 <ul class="bg_light col-12 d-flex justify-content-around">
35 Home
36 CORS
37 API Gateway
38 Mapping Template
39 <!--Kinesis-->

```

- Open and Edit the file "apigw.html" in the Part-3 folder change the link and replace the string 'MY\_API\_GW\_REQUEST' with the API Gateway Invoke URL e.g. "<https://d3w0w4ajyh.execute-api.eu-central-1.amazonaws.com/dev>"

```

apigw.html
1 <!DOCTYPE html>
2 <html lang="en">
3
4 <head>
5 <meta charset="utf-8" />
6 <title>Lambda Lab</title>
7 <link rel="stylesheet" href="css/reset.css" />
8 <link rel="stylesheet" href="css/text.css" />
9 <link rel="stylesheet" href="css/960_24_col.css" />
10 <link rel="stylesheet" href="css/meetup_base.css" />
11 <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css"
12 integrity="sha384-JCkb8j3iqJ61gNV9KGb8thsNjpSL0n8PARn9HuZOnIxNhoP+VmnDGMS5t9Uj0Z" crossorigin="anonymous">
13 <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.2.1/jquery.min.js" type="text/javascript"></script>
14 <script type="text/javascript">
15 $(document).ready(function () {
16 $('#button').click(function () {
17 $('#apigw').load("https://974mj7aky7.execute-api.us-east-1.amazonaws.com/dev/random");
18 });
19 });
20 </script>
21 </head>
22
23 <body>
24 <div class="container_24">
25 <div class="d-flex flex-column border mb-3">
26
27 <h2 class="text-center mt-3 ml-3 rounded">
28 Clarusway AWS&DEVOPS PATH Lambda API Gateway Lab
29 </h2>
30 </div>
31 <div class="grid_24 nav">
32 <p>
33 <ul class="bg-light col-12 d-flex justify-content-around">
34 Home
35 CORS
36 API Gateway
37 Mapping Template
38
39
40 </p>
41 </div>
42 </div>
43 </body>

```

- Go to S3 Bucket menu and select "clarusway.broadcast" bucket ---> Upload "apigw.html" file into the bucket.

<https://s3.console.aws.amazon.com/s3/upload/serkan/radio?region=us-east-1>

The screenshot shows the AWS S3 'Upload' interface. On the left, there's a file selection area with a 'Drag and drop files and folders' button and a 'Find by name' search bar. Below it is a table titled 'Files and folders (1 Total, 2.0 KB)' showing the uploaded file 'apigw.html'. On the right, there's a file browser window titled 'Part-003\_Lambda' showing the same file 'apigw.html' selected. The 'Destination' section at the bottom shows the target bucket as 's3://serkan.radio'.

The screenshot shows the Clarusway AWS&DEVOPS PATH Lambda API Gateway Lab website. The main heading is 'API Gateway'. Below it, there are tabs for 'Home', 'CORS', 'API Gateway' (which is highlighted), and 'Mapping Template'. A blue bracket points from the 'Get External Content' button in the footer to the 'API Gateway' tab above. The footer also includes the text '\$3 Sample Static website'.

The screenshot shows the AWS API Gateway console interface. On the left, the navigation pane lists various API-related sections like APIs, Stages, Authorizers, and Documentation. The main area shows a tree structure for an API named "SerkanAPI". A context menu is open over a "GET" method under the "/random" resource. The menu has three main sections: METHOD ACTIONS (Edit Method Documentation, Delete Method), RESOURCE ACTIONS (Create Method, Create Resource, Enable CORS, Edit Resource Documentation, Delete Resource), and API ACTIONS (Deploy API, Import API, Edit API Documentation, Delete API). To the right of the menu, four boxes represent the execution flow: "Method Request" (Auth: NONE, ARN: arn:aws:execute-api:us-east-1:54718753867:974mj7aly7"/GET/random), "Method Response" (HTTP Status: 200, Models: application/json => Empty), "Integration Request" (Type: LAMBDA, Region: us-east-1), and "Integration Response" (HTTP status pattern: -, Output passthrough: Yes).

aws Services ▾

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

Amazon API Gateway APIs > SerkanAPI (974mj7aky7) > Resources > /random (ks2gm3) > Enable CORS Admin\_user @ ser007 N Virginia Support ▾

Show all hints ?

**APIs**

Resources Actions Enable CORS

Custom Domain Names

VPC Links

---

**API: SerkanAPI**

Resources Stages Authorizers Gateway Responses Models Resource Policy Documentation Dashboard Settings Usage Plans

Gateway Responses for SerkanAPI API  DEFAULT 4XX  DEFAULT 5XX

Methods  GET  OPTIONS

Access-Control-Allow-Methods: GET, OPTIONS

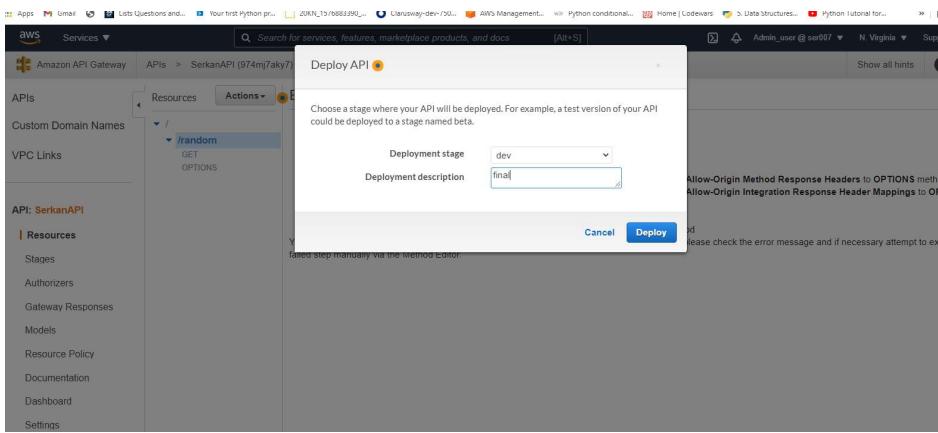
Access-Control-Allow-Headers: Content-Type,X-Amz-Date,Authorization

Access-Control-Allow-Origin:

Advanced

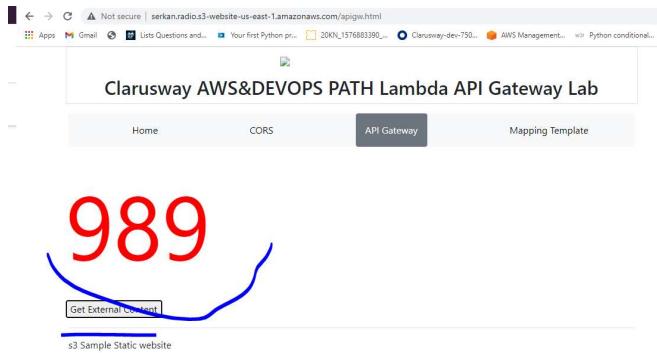
Enable CORS and replace existing CORS header

Herkes degil de tek source ulasın dersek access control e o adresi yazıyoruz



Copy Bucket's static website Endpoint and paste it to browser.

- Show the sub-link "API Gateway" and click the button of "Get External Content" ----> It will fail.
- View Javascript in your website and you'll see message like CORS header 'Access-Control-Allow-Origin' missing.
- STEP 8: Active CORS on API**
- Previous part we did learned what CORS. You need to enable the API here so the website can access the link.
- Go to the API /random-number, GET method then Actions > Enable CORS
- Click 'Enable CORS and replace existing CORS headers' button and keep the rest of page as is.
- Confirm dialog 'Yes, replace existing values'
- Watch the animated ticks appear on the AWS console
- You need to redeploy the API;
  - Select / ---> Actions ---> Deploy API
    - Development stage: dev
    - Development description: Part-3 CORS enabled GET request
    - Click 'Deploy'



- Refresh the web page and go to API Gateway sub-link
- Press button Get External Content and show the random number created by Lambda function.

Not secure | serkan.radio.s3-website-us-east-1.amazonaws.com/apigw.html

Apps Gmail Lists Questions and... Your first Python pr... Clarusway-dev-750... AWS Management... Python conditional... Home | Codewars 5. Data Structures... Python Tutorial for... 200N\_1576882390...

Clarusway AWS&DEVOPS PATH Lambda API Gateway Lab

Home CORS API Gateway Mapping Template

645

Get External Content

s3 Sample Static website