

# API Gateway Resources

<https://www.youtube.com/watch?v=riU2-rxBq10>

ddb Query

@<https://docs.aws.amazon.com/amazondynamodb/latest/developerguide/GettingStarted.Python.04.html>

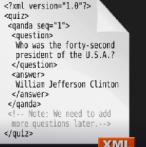
API Calls

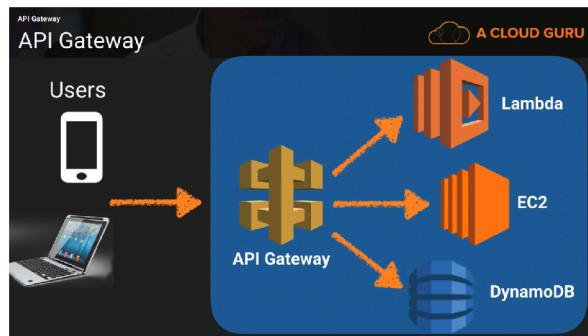
<https://boto3.amazonaws.com/v1/documentation/api/latest/guide/dynamodb.html>

<https://ra8z188344.execute-api.us-east-1.amazonaws.com/dev/parse4metadata-API>

## API 101

### Types of APIs

- REST APIs (**R**epresentational **S**tate **T**ransfer)
  - Uses JSON
- SOAP APIs (**S**imple **OA**ccess **P- Uses XML
- {  
  "\_id" : "51262c865ca358946be09d77",  
  "firstname" : "John",  
  "surname" : "Smith",  
}
-   
<?xml version="1.0"?>  
<!DOCTYPE quora>  
<quora seq="1">  
  <question>  
    Who was the forty-second president of the U.S.A.?</question>  
  <answer>  
    William Jefferson Clinton  
  </answer>  
  <meta>We need to add more questions later...</meta>  
</quora>
- XML**



### How do I configure API Gateway?

- Define an API (container)
- Define Resources and nested Resources (URL paths)
- For each Resource:
  - Select supported HTTP methods (verbs)
  - Set security
  - Choose target (such as EC2, Lambda, DynamoDB, etc.)
  - Set request and response transformations
- Deploy API to a Stage
- Uses API Gateway domain, by default
- Can use custom domain
- Now supports AWS Certificate Manager: free SSL/TLS certs!

### What is API Gateway?

**Amazon API Gateway** is a fully managed service that makes it easy for developers to publish, maintain, monitor, and secure APIs at any scale. With a few clicks in the AWS Management Console, you can create an API that acts as a "front door" for applications to access data, business logic, or functionality from your back-end services, such as applications running on Amazon Elastic Compute Cloud (Amazon EC2), code running on AWS Lambda, or any web application.



Hypertext Transfer Protocol Secure (HTTPS) is an extension of the Hypertext Transfer Protocol (HTTP). It is used for secure communication over a computer network, and is widely used on the Internet. In HTTPS, the communication protocol is encrypted using Transport Layer Security or, formerly, Secure Sockets Layer.

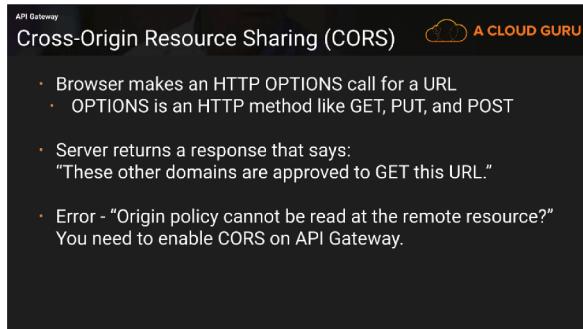
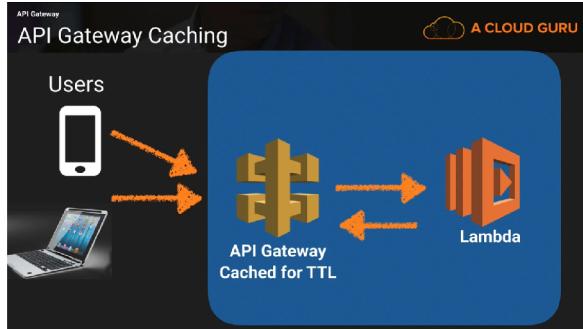
### What can API Gateway Do?

- Expose HTTPS endpoints to define a RESTful API
- Serverlessly connect to services like Lambda & DynamoDB
- Send each API endpoint to a different target
- Run efficiently with low cost
- Scale effortlessly
- Track and control usage by API key
- Throttle requests to prevent attacks
- Connect to CloudWatch to log all requests for monitoring
- Maintain multiple versions of your API

### What is API Caching?

You can enable **API caching** in Amazon API Gateway to cache your endpoint's response. With caching, you can reduce the number of calls made to your endpoint and also improve the latency of the requests to your API. When you enable caching for a stage, API Gateway caches responses from your endpoint for a specified time-to-live (TTL) period, in seconds. API Gateway then responds to the request by looking up the endpoint response from the cache instead of making a request to your endpoint.





## Debug Lambda

## Debug

```
Generate template: 
```

```
1+ {
2   "TableName": "PII_Metadata",
3   "PrimaryKey": "ID",
4   "KeyConditionExpression": "ID = :v1",
5   "ExpressionAttributeValues": {
6     ":v1": {
7       "Key": "$input.path('$.Item.1')
```

```
Generate template: <input type="text" value="1- {<br>2   \"TableName": "PII_Metadata",<br>3   \"PrimaryKey": "ID",<br>4   \"KeyConditionExpression": "ID = :v1",<br>5   \"ExpressionAttributeValues": {<br>6     ":v1": {<br>7       "Key": "$input.params('ID')"}<br>8   }<br>9 }<br>10 }" />
```

Client connects you to API Gateway to backend (like a front door) for services

Don't use it directly with DynamoDB, and with DynamoDB there, there is a very small range of what you can do. some

## filters

Don't combine things.

THIS WORKED

```
{  
    "TableName": "PII_Metadata",  
    "PrimaryKey": "ID",  
    "KeyConditionExpression": "ID = :v1",  
    "ExpressionAttributeValues" : {  
        ":v1": {  
            "S": "$input.params('ID')"  
        }  
    }  
}
```

# debugging

lambda\_function

```

101 table = dynamodb.Table('PII')
102 def lambda_handler(event, context):
103     bucketName = event['Records'][0]['s3']['bucket']['name']#bucket name
104     s3fileName = event['Records'][0]['s3']['object']['key']#s3 file name
105     response = s3_client.get_object(Bucket=bucketName,Key=s3fileName)
106     data = response['Body'].read().decode("utf-8")
107     # splitData = data.split("\r\n")
108     # print(type(data))
109     # data = response['Body']
110     print(data)
111
112
113     # s3_file_key = 'data/test.csv'
114     # bucket = 'data-bucket'
115

```

Execution Result

Execution results

Response: null

Status: Succeeded | Max Memory Used: 82 MB | Time: 347.15 ms

Request ID: "3eec2218-554c-4431-896f-e7a526554660"

Function logs:

```

Id: 3eec2218-554c-4431-896f-e7a526554660 Version: $LATEST
SSN,gender,birthdate,maiden name,last name,first name,address,city,state,zip,phone,email,cc_type,CVN,cc_cvc,cc_expiredate
514-30-2668,f,5/27/86,Nicholson,Russell,Jacki,3097 Better Street,Kansas City,MO,66215,913-227-6106,jrussell@domain.com,a,3.4539E+14,232,1/1
505-88-5714,f,9/23/63,Mcclain,Venson,Lillian,539 Kyle Street,Wood River,NE,68883,308-583-8759,lvenson@domain.com,d,3.02049E+13,471,12/1/11
690-05-5315,m,10/2/69,Kings,Conley,Thomas,570 Nancy Street,Morrisville,NC,27560,919-656-6779,tconley@domain.com,v,4916 4811 5814 8111,731,10
646-44-9061,M,1/12/78,Kurtz,Jackson,Charles,1074 Small Street,New York,NY,10011,212-847-4915,cjackson@domain.com,m,5218 0144 2703 9266,892,1
431-37-1800,f,1/3/80,Binkley,Davis,Susan,1200 California Street,Superior,IL,60601,200-221-0100,susan.binkley@domain.com,v,4916 4034 9269 8783,33,4/1/11

```

s3toDynamoDB - Lambda

File Edit Find View Go Tools Window Save Test ▾

Environment

λ lambda\_function

```

101 table = dynamodb.Table('PII')
102 def lambda_handler(event, context):
103     bucketName = event['Records'][0]['s3']['bucket']['name']#bucket name
104     s3fileName = event['Records'][0]['s3']['object']['key']#s3 file name
105     response = s3_client.get_object(Bucket=bucketName,Key=s3fileName)
106     data = response['Body'].read().decode("utf-8")
107     splitData = data.split("\r\n")
108     print(splitData)
109     # print(type(data))
110     # data = response['Body']
111     # print(data)
112
113     # s3_file_key = 'data/test.csv'
114     # bucket = 'data-bucket'
115

```

Execution Result

Execution results

Status: Succeeded | Max Memory Used: 81 MB | Time: 195.34 ms

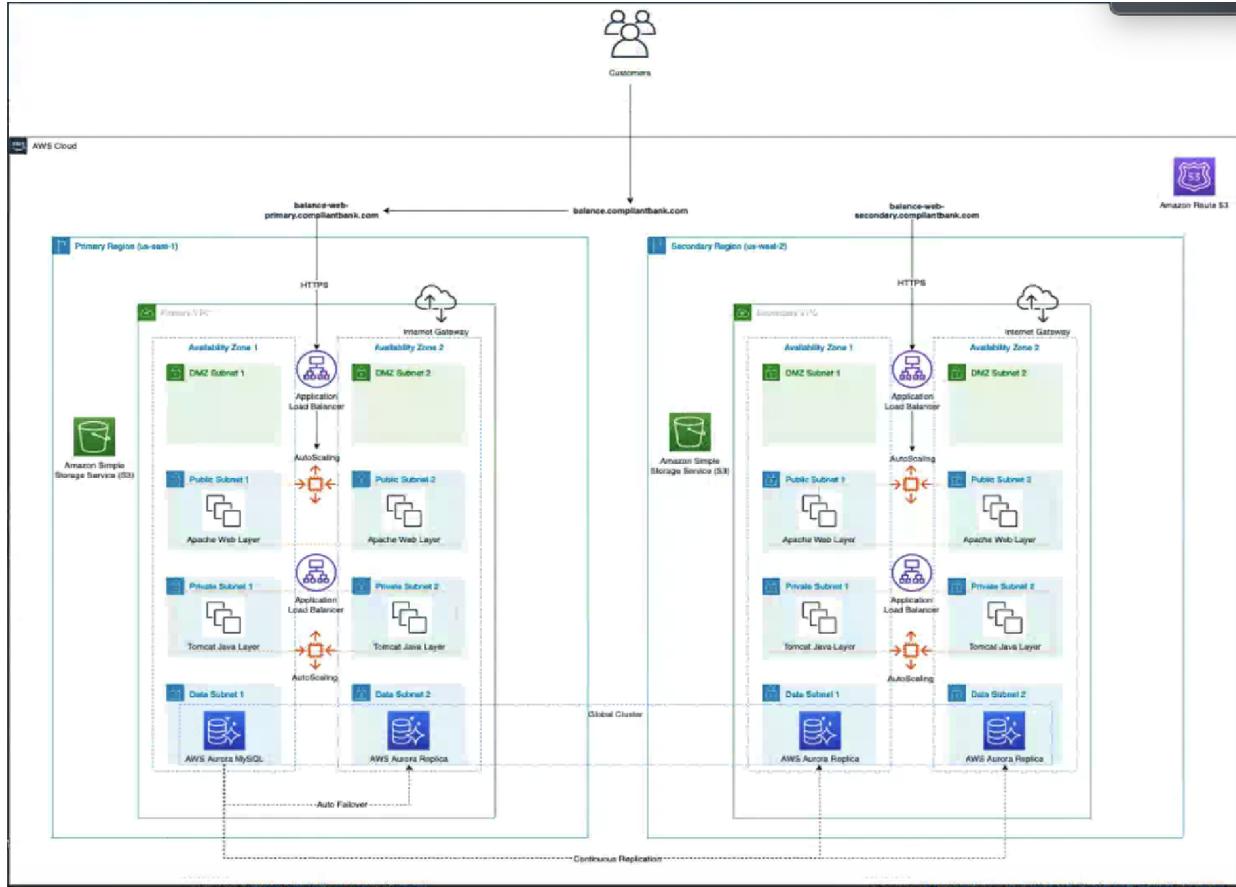
Request ID: "ce793840-3b2a-4a23-a412-16482818fb84"

Function logs:

```

'uffeffSSN,gender,birthdate,maiden name,last name,first name,address,city,state,zip,phone,email,cc_type,CVN,cc_cvc,cc_expiredate', '514-30-2668
END RequestId: ce793840-3b2a-4a23-a412-16482818fb84
REPORT RequestId: ce793840-3b2a-4a23-a412-16482818fb84 Duration: 195.34 ms Billed Duration: 200 ms Memory Size: 128 MB Max Memory Used: 81 MB

```



**lambda\_function**

```

101  table = dynamodb.Table('PII')
102  def lambda_handler(event, context):
103      bucketName = event['Records'][0]['s3']['bucket']['name']#bucket name
104      s3fileName = event['Records'][0]['s3']['object']['key']#s3 file name
105      response = s3_client.get_object(Bucket=bucketName,Key=s3fileName)
106      data = response['Body'].read().decode("utf-8")
107      # splitData = data.split("\r\n")
108      # print(type(data))
109      # data = response['Body']
110      print(data)
111
112
113      # s3_file_key = 'data/test.csv'
114      # bucket = 'data-bucket'
115

```

**Execution Result**

Status: Succeeded | Max Memory Used: 82 MB | Time: 347.15 ms

**Execution results**

Response:  
null

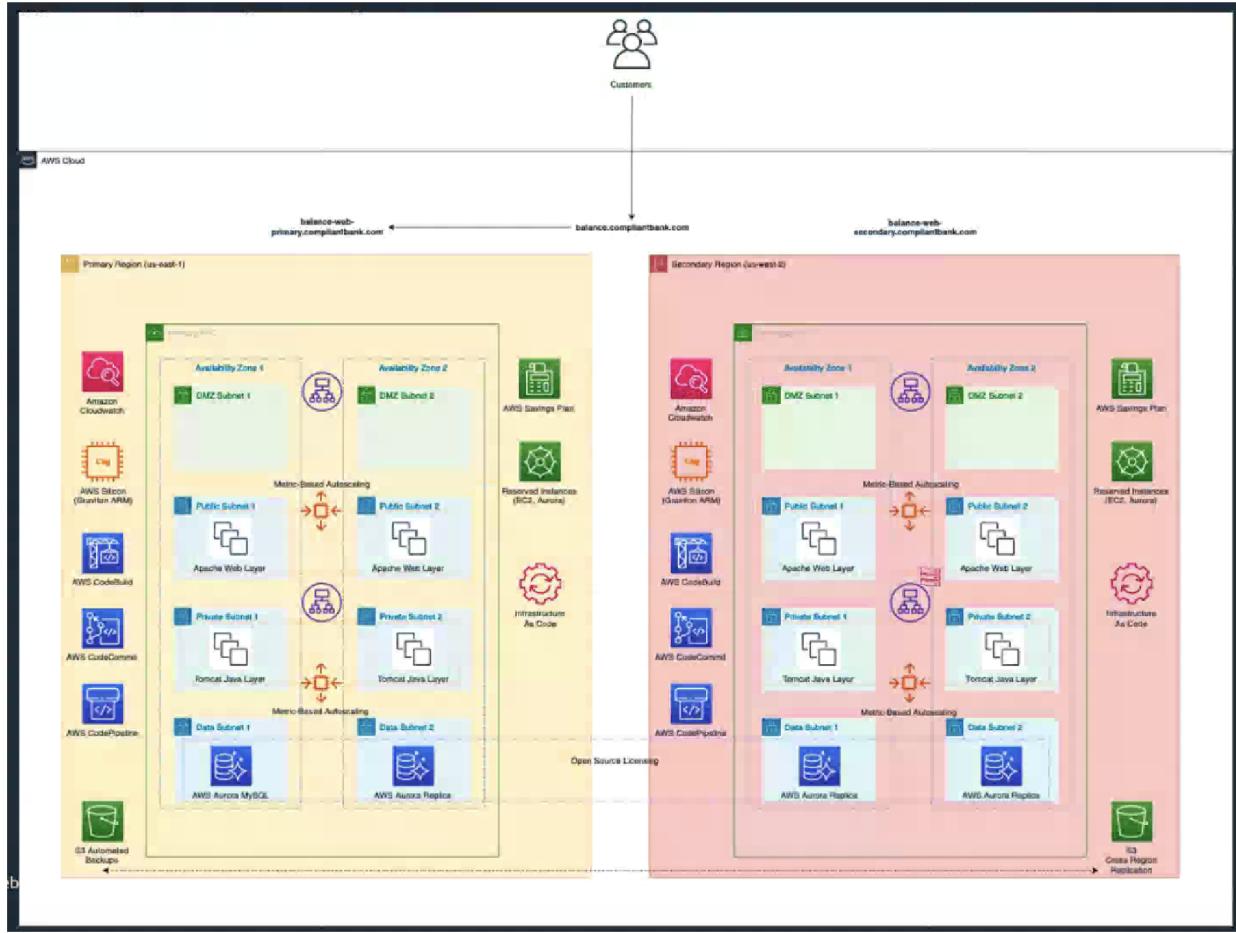
Request ID:  
"3eec2218-554c-4431-896f-e7a526554660"

Function logs:

```

Id: 3eec2218-554c-4431-896f-e7a526554660 Version: $LATEST
SSN,gender,birthdate,maiden name,last name,first name,address,city,state,zip,phone,email,cc_type,CCN,cc_cvc,cc_expiredate
514-30-2668,f,5/27/86,Nicholson,Russell,Jacki,3097 Better Street,Kansas City,MO,66215,913-227-6106,jrussell@domain.com,a,3.4539E+14,232,1/1/
505-88-5714,f,9/23/63,Mcclain,Venson,Lillian,539 Kyle Street,Wood River,NE,68883,308-583-8759,lvenson@domain.com,d,3.02049E+13,471,12/1/11
690-05-5315,m,10/2/69,Kings,Conley,Thomas,570 Nancy Street,Morrisville,NC,27560,919-656-6779,tconley@domain.com,v,4916 4811 5814 8111,731,10
646-44-9061,M,1/12/78,Kurtz,Jackson,Charles,1074 Small Street,New York,NY,10011,212-847-4915,cjackson@domain.com,m,5218 0144 2703 9266,892,1
400-07-1886,f,7/10/80,Smith,Davis,Claire,1000 Bedford St,New York,NY,10001,212-555-0100,claire@domain.com,v,4916 4034 9269 8783,33,4/1/11

```



```

40
47 def lambda_handler(event, context):
48     # table = create_sample_table()
49     table = dynamodb.Table('PII')
50     bucketName = event['Records'][0]['s3']['bucket']['name']#bucket name
51     s3fileName = event['Records'][0]['s3']['object']['key']#s3 file name
52     response = s3_client.get_object(Bucket=bucketName,Key=s3fileName)
53     data = response['Body'].read().decode("utf-8")
54
55
56 ##### cleaning up each line #####
57 line = data.split("\r\n")
58
59 ##### entering data into the table #####
60 int = -1
61 for lin in line:
62     if int == -1:
63         int += 1
64         continue
65     else:
66         int += 1
67         cell = lin.split(",")
68         #for x in range (15):
69             #print(cell[x])
70
71         table.put_item(
72             Item={
73                 'ID' : int,
74                 'SSN': cell[0],
75                 'gender': cell[1],
76                 'birthdate': cell[2],
77                 'maiden name': cell[3],
78                 'last name': cell[4],
79                 'first name': cell[5],
80                 'address': cell[6],
81                 'city': cell[7],
82                 'state': cell[8],

```

## Macie Set up

<https://wisdom.corp.amazon.com/Pages/amazon-macie-demo.aspx>

Installed homebrew, AWS CLI, Python3, boto3, pip3

Ran CreateMacieDemo.py by “python3 CreateMacieDemo.py”

[https://drive-render.corp.amazon.com/view/rizra@/Isengard-cli/docs/installation\\_mac.html](https://drive-render.corp.amazon.com/view/rizra@/Isengard-cli/docs/installation_mac.html)

```

cd ~ > tar xvzf ~/Downloads/isengard-cli-mac.tar.gz > mv /usr/local/bin/isengard /usr/local/bin/bak.isengard >ln -s ~/isengard/isengard /usr/local/bin/isengard > which isengard > xattr -r -d com.apple.quarantine ~/isengard> isengard -help > isengard credentials --awscli suisun@amazon.com -role Admin > Midway Auth

```

```
suiyun@3c22fb074e68 ~ % mkdir .aws
suiyun@3c22fb074e68 ~ % cd .aws
suiyun@3c22fb074e68 .aws % vim config
suiyun@3c22fb074e68 .aws % cat config
[profile default]
region = us-east-1
output = json
credential_process = eisengard credentials --awscli suiyun@amazon.com --role Admin
```

SSO - user single sign on (like midway)

Eisengard - sign into AWS with amazon credentials

Press [esc], :(use shift not command for scolon) wq - saves and quits out of the doc on terminal

```
[profile mb]
region = us-east-1
source_profile = default
role_arn = arn:aws:iamp:040515623297:role/Admin
suiyun@3c22fb074e68 .aws %
suiyun@3c22fb074e68 .aws % cd Macie
cd: no such file or directory: Macie
suiyun@3c22fb074e68 .aws % cd ..
suiyun@3c22fb074e68 ~ % cd Desktop
[suiyun@3c22fb074e68 Desktop % cd Macie
```

```
[suiyun@3c22fb074e68 Macie %
curl https://bootstrap.pypa.io/get-pip.py -o get-pip.py
[
  % Total    % Received % Xferd  Average Speed   Time     Time      Time  Current
          Dload  Upload Total   Spent    Left  Speed
100 1825k  100 1825k    0     0  1131k      0  0:00:01  0:00:01  --:--:-- 1130k
suiyun@3c22fb074e68 Macie % python3 get-pip.py
Collecting pip
  Downloading pip-20.1.1-py2.py3-none-any.whl (1.5 MB)
    |██████████| 1.5 MB 1.4 MB/s
Installing collected packages: pip
  Attempting uninstall: pip
    Found existing installation: pip 20.0.2
  Uninstalling pip-20.0.2:
    Successfully uninstalled pip-20.0.2
Successfully installed pip-20.1.1
```

```
suiyun@3c22fb074e68 Macie % pip3 install boto3 --user
Collecting boto3
  Downloading boto3-1.14.19-py2.py3-none-any.whl (128 kB)
    |██████████| 128 kB 1.7 MB/s
```

```
suiyun@3c22fb074e68 Macie % python3 CreateMacieDemo.py
This script will create the Macie demo environment in us-east-1
Type yes, no or region name to continue: yes
Continuing with demo build in us-east-1
My Account ID is 040515623297
Traceback (most recent call last):
  File "CreateMacieDemo.py", line 147, in <module>
    userArn = boto3.client('iam').get_user()['User']['Arn']
  File "/Users/suiyun/Library/Python/3.8/lib/python/site-packages/botocore/client.py", line 316, in _api_call
    return self._make_api_call(operation_name, kwargs)
  File "/Users/suiyun/Library/Python/3.8/lib/python/site-packages/botocore/client.py", line 635, in _make_api_call
    raise error_class(parsed_response, operation_name)
botocore.exceptions.ClientError: An error occurred (ValidationError) when calling the GetUser operation: Must specify userName when calling with non-User credentials
suiyun@3c22fb074e68 Macie %
```

In the python script, change profile\_name to #ERROR! WHY?

what we did was we 1) ran the isengrard credentials command to authenticate with your midway token to then get access to the Admin role in your AWS account 2) we then assumed role into your account with Admin credentials from AWS (the profile is called 'mb' in my config file) \*\*if you want to change it to be 'my-account' or something, you can. [profile mb] should be changed to [profile my-account]

IAM > created a user. added into the config file  
changed profilename to the username

```
aws sts get-caller-identity --profile test
aws s3 ls --profile test
```