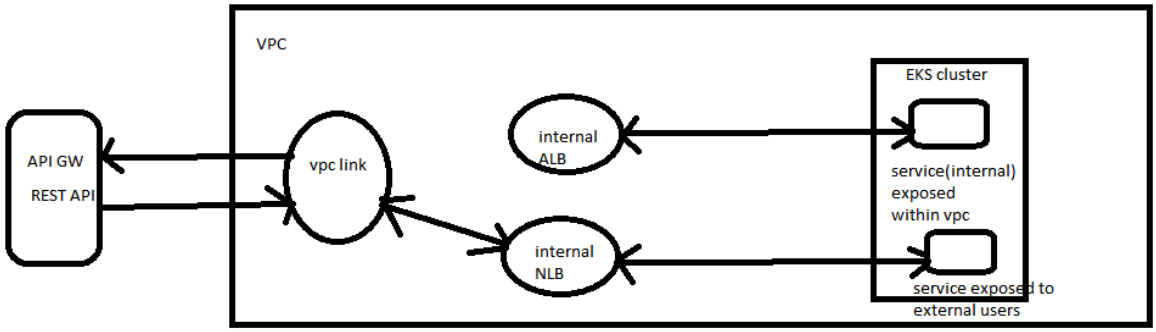
Assignment Documentation

1. Architecture



EKS cluster hosts a set of microservices,showing only two for demo, one service which is exposed internally and the other which is exposed to external users via API Gateway.

2. **Replication Steps**

Follow these steps to replicate the setup:

1. Prerequisites:
   - AWS CLI installed and configured
   - Terraform v0.14+ installed
   - kubectl installed
   - Docker installed

2. Clone the Repository:

   git clone https://github.com/Mahendra-Maddu/assignment-pf.git
   cd assignment-pf

3. Initialize Terraform:

   terraform init

4. Plan and Apply Terraform Configuration:

   terraform plan
   terraform apply

5. Configure kubectl:

aws eks update-kubeconfig --name <cluster-name>

6. Deploy ALB ingress controller on the EKS cluster. Use this link to do the same

7. Deploy Microservices:

kubectl apply -f internal-service.yaml
kubectl apply -f external-service.yaml

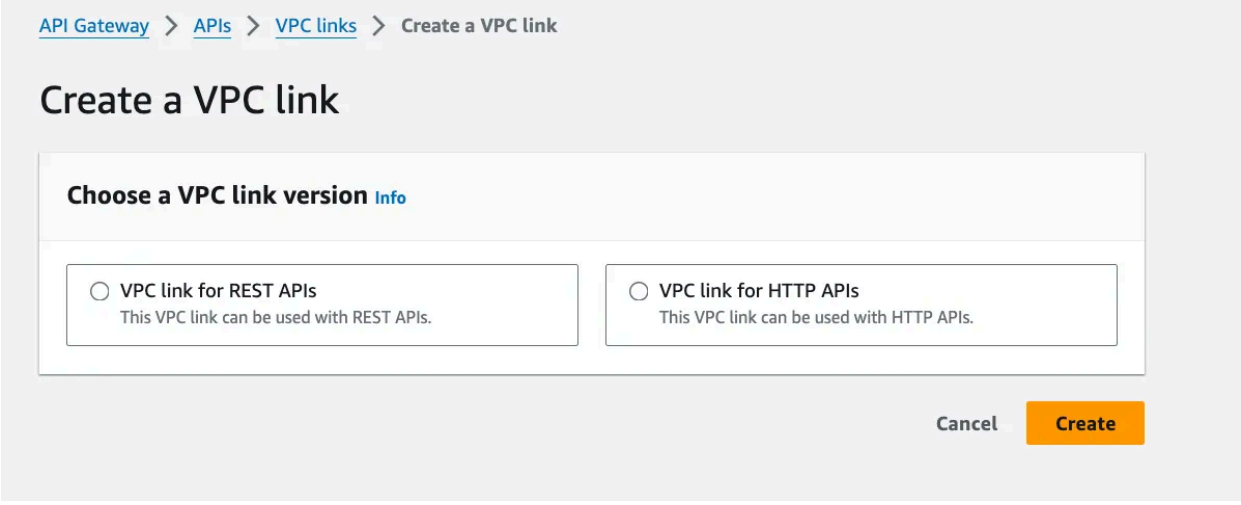
After deploying the services, you would have one NLB and ALB. Note down the DNS of the NLB created for the service to be exposed to the internet.

8. Create VPC link for Rest API and Integrate it with NLB of the EKS service.

To access EKS service that is exposed via the NLB within the VPC through API Gateway, We have to create a VPC Link resource targeted for our VPC and then integrate an API method with a private integration that uses the VpcLink.

A VPC link encapsulates connections between API Gateway and targeted our Network Load Balancer . When you create a VPC link, API Gateway creates and manages elastic network interfaces for the VPC link in your account.

To Create VPC Link In the API Gateway console, go to “VPC Links” and create a new VPC Link for REST API and add the Target NLB of EKS Cluster service



API Gateway > APIs > VPC links > Create a VPC link

Create a VPC link

Choose a VPC link version Info

- (●) VPC link for REST APIs
  This VPC link can be used with REST APIs.
- ( ) VPC link for HTTP APIs
  This VPC link can be used with HTTP APIs.

VPC link details

Name

restapi-[redacted]-vpc-link

Description (optional)

vpclink to access the EKS NLB service

Target NLB

[redacted]

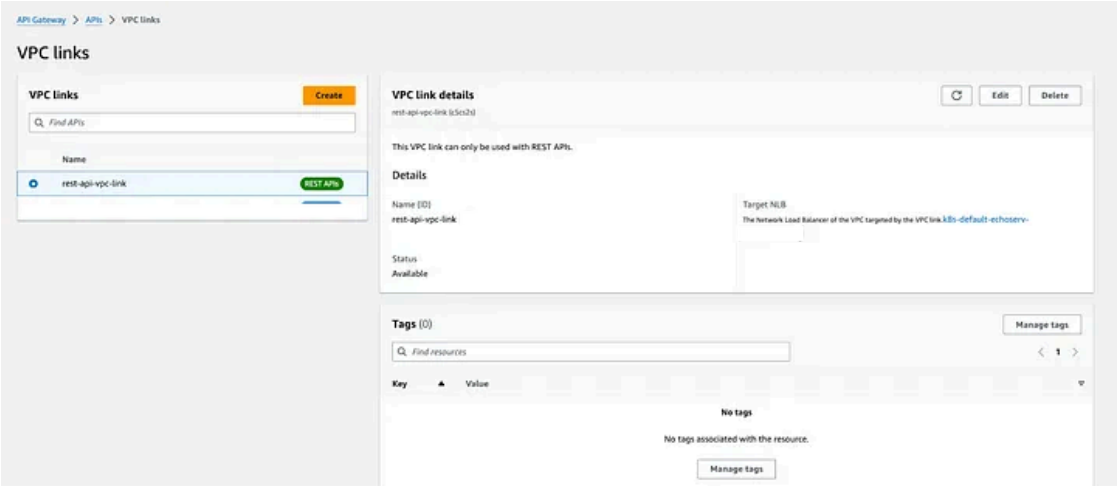
Tags

A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

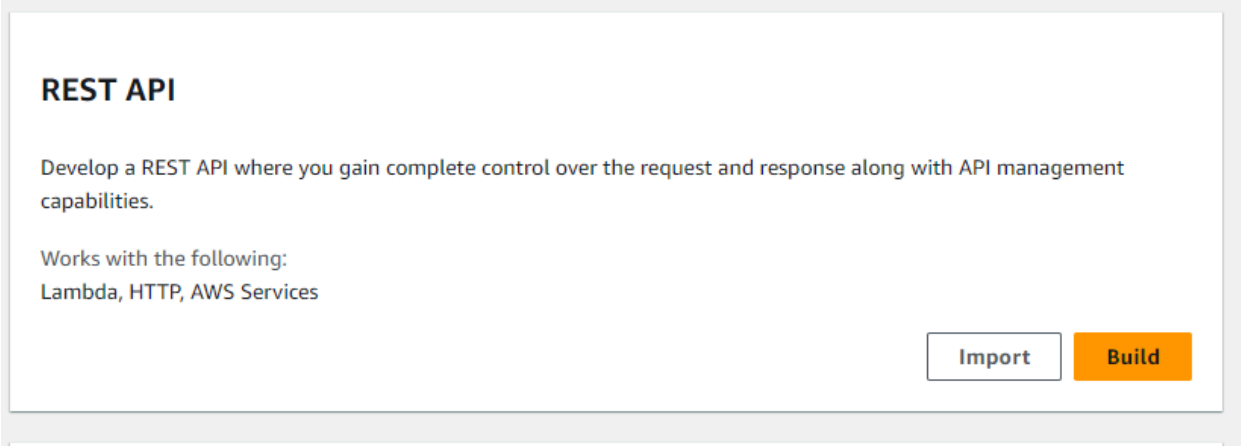
No tags associated with the resource.

Add tag

Cancel Create



9.Create a Private REST API using the API Gateway



API details

- (●) New API
  Create a new REST API.
- ( ) Clone existing API
  Create a copy of an API in this AWS account.
- ( ) Import API
  Import an API from an OpenAPI definition.
- ( ) Example API
  Learn about API Gateway with an example API.

API name

My REST API

Description - *optional*

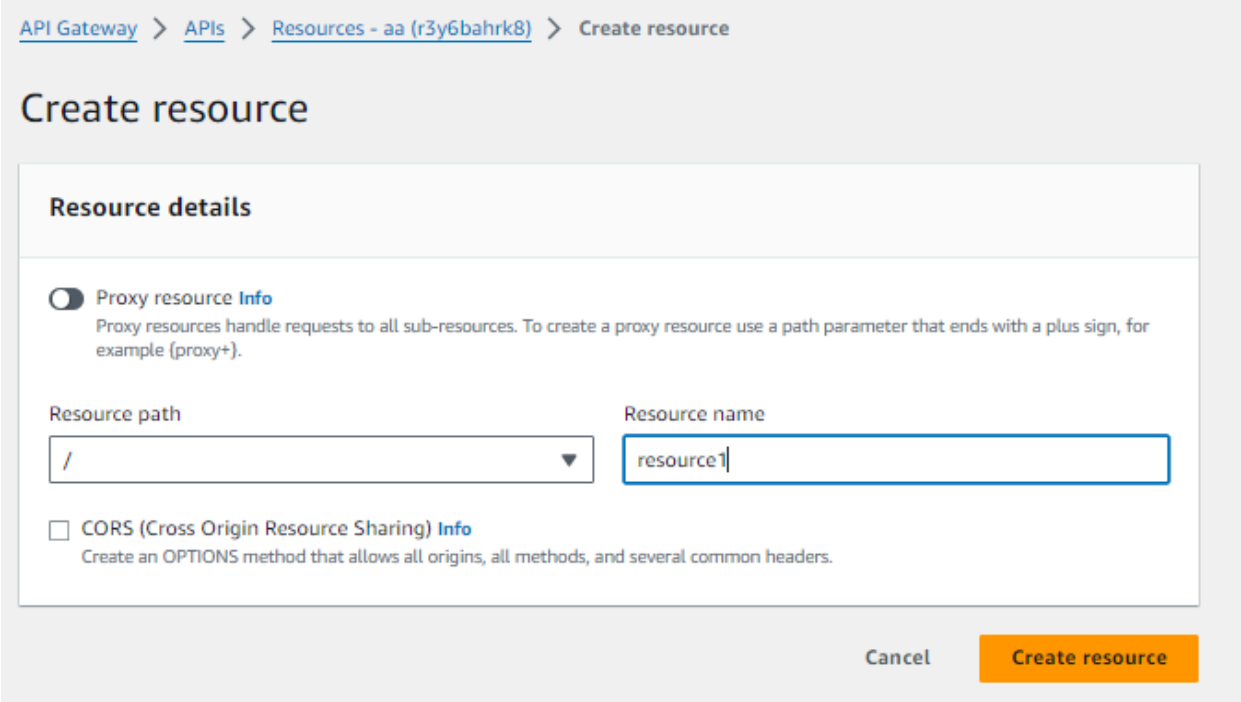
API endpoint type

Regional APIs are deployed in the current AWS Region. Edge-optimized APIs route requests to the nearest CloudFront Point of Presence. Private APIs are only accessible from VPCs.

Regional

Cancel | Create API

Create a Resource for the NLB:



Create Method for the resource . Copy DNS of the NLB in the endpoint URL. http://<NLB-DNS>

Create method

Method details

Method type

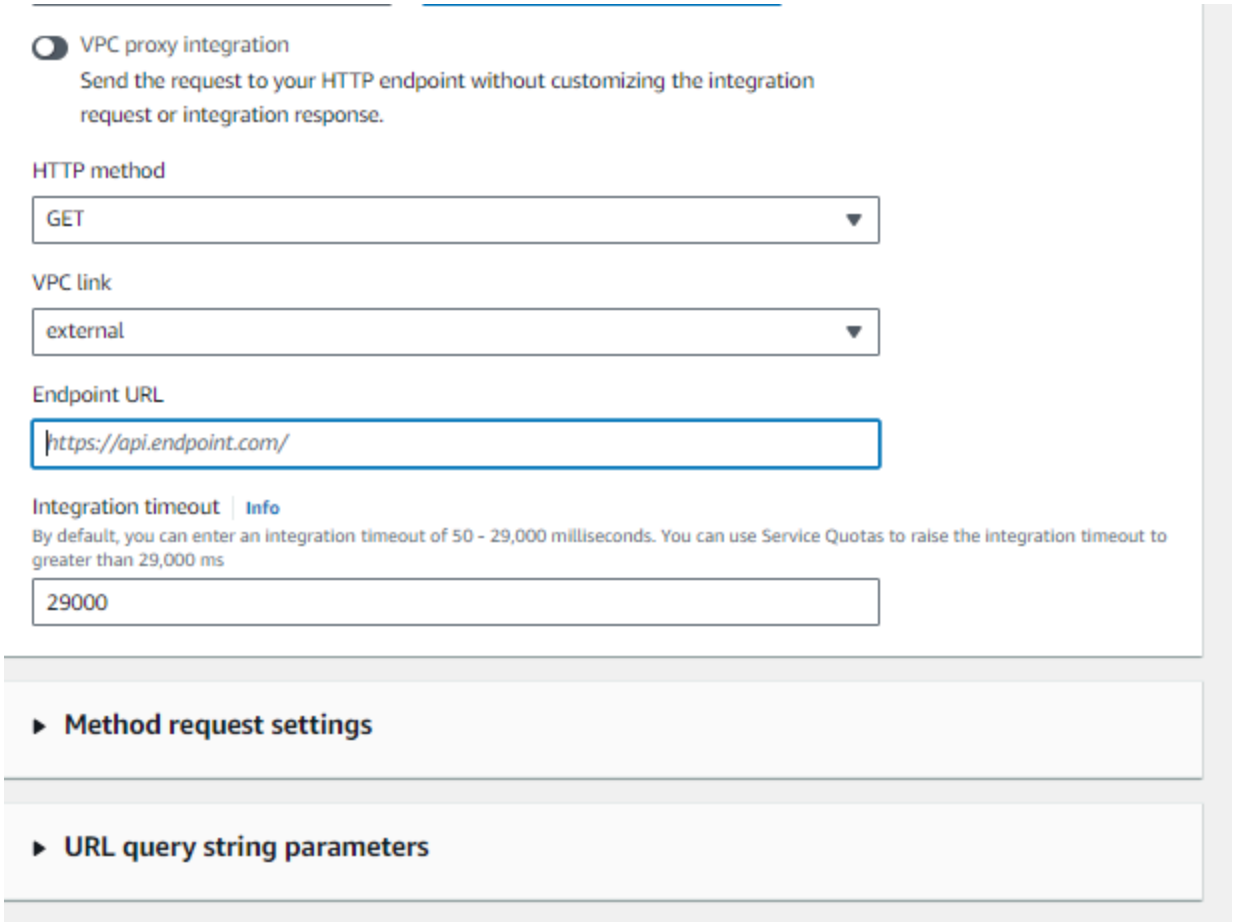
GET

Integration type

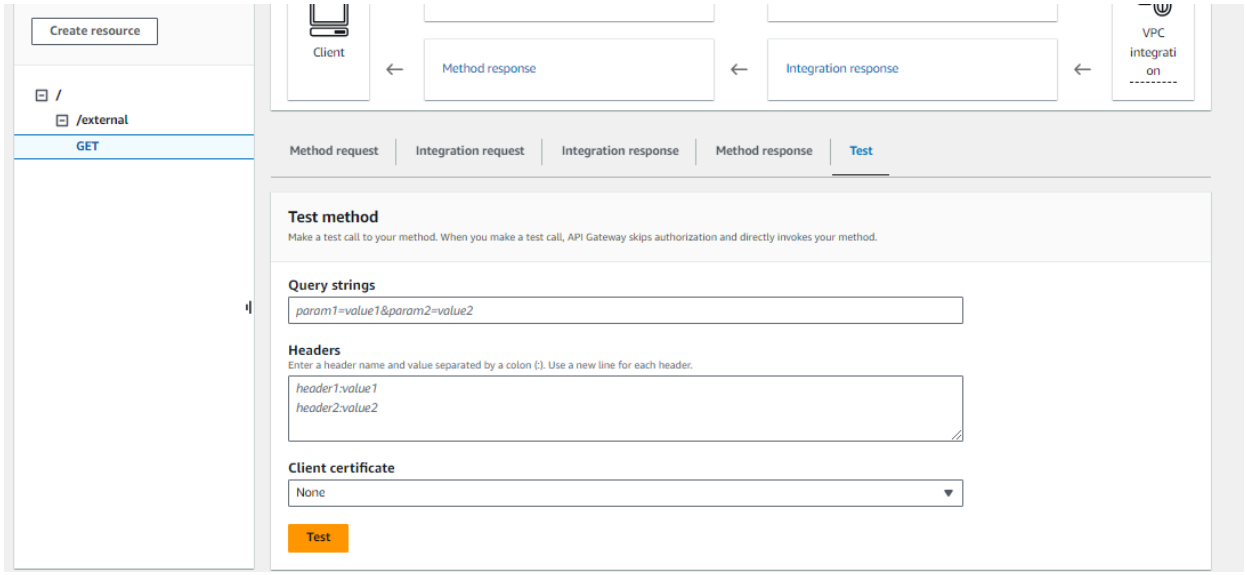
- Lambda function
  Integrate your API with a Lambda function.
- HTTP
  Integrate with an existing HTTP endpoint.
- Mock
  Generate a response based on API Gateway mappings and transformations.
- AWS service
  Integrate with an AWS Service.
- VPC link
  Integrate with a resource that isn't accessible over the public internet.

VPC proxy integration

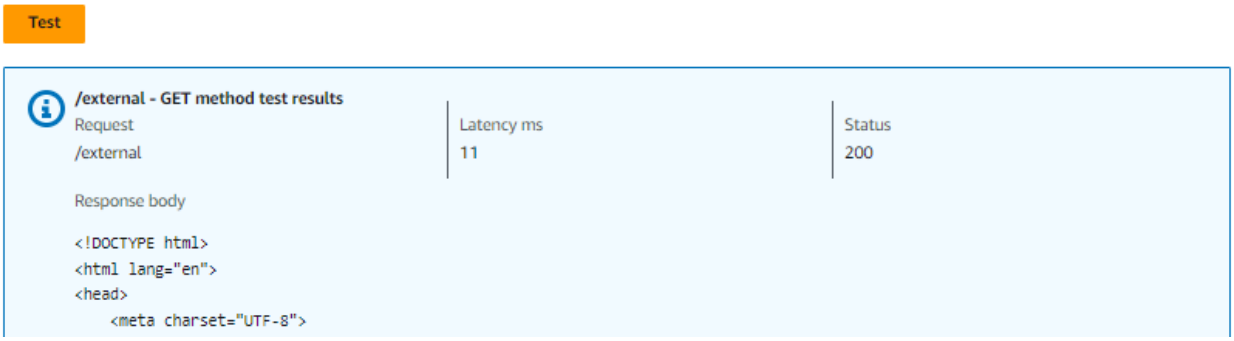
Send the request to your HTTP endpoint without customizing the integration request or integration response.



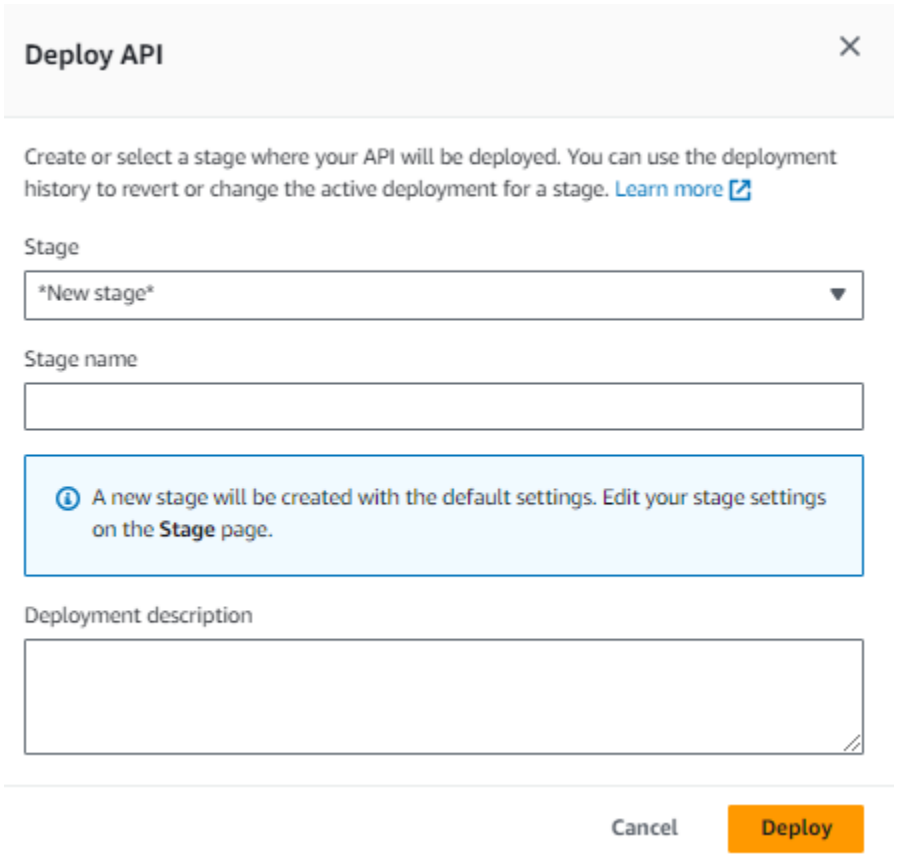
Once you create the Method, test it before deploying .



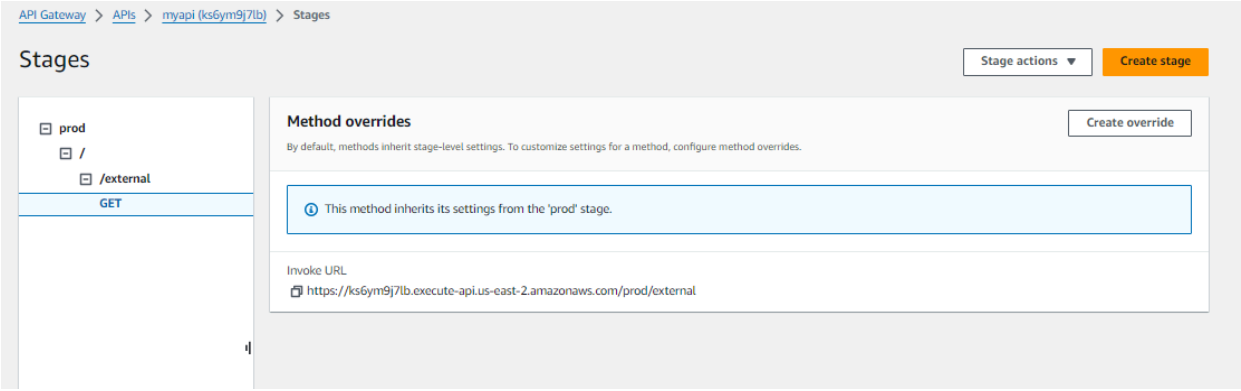
Test should be successful



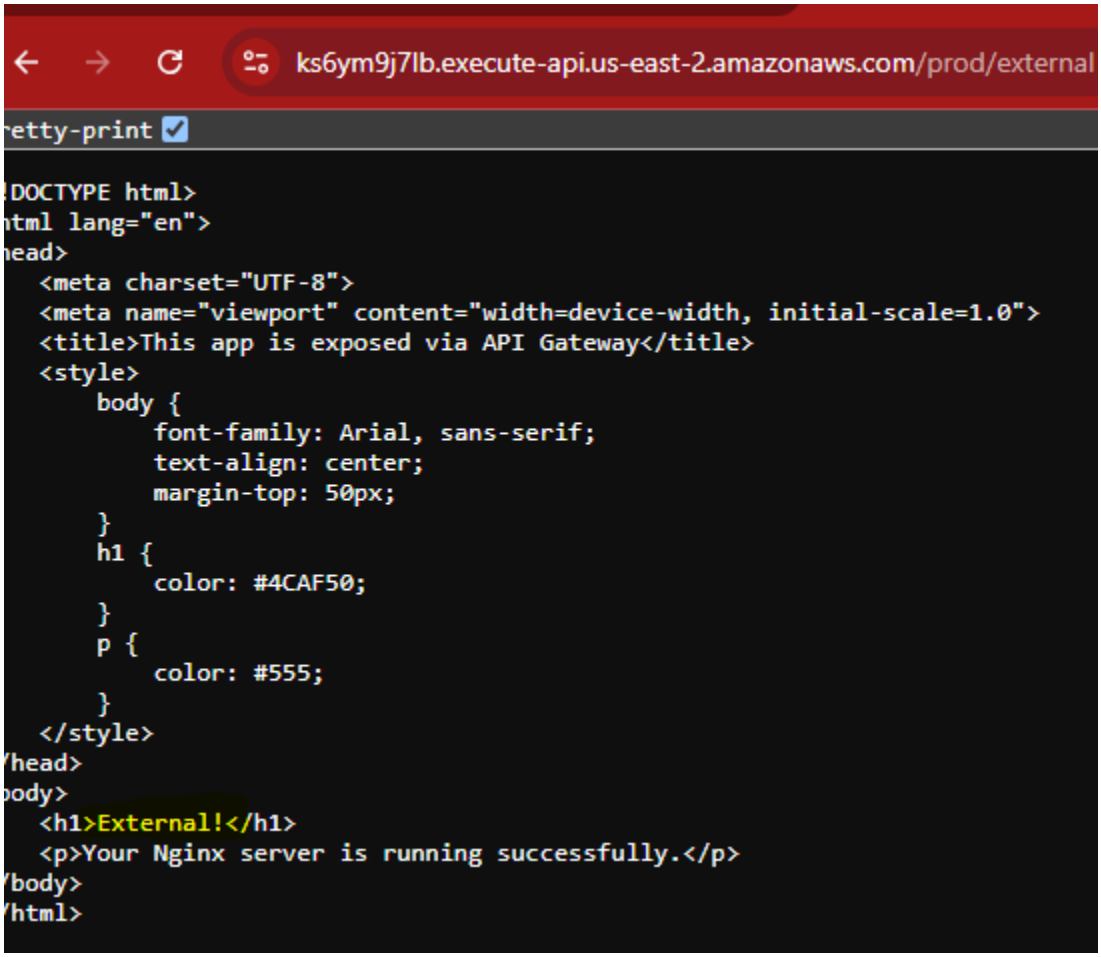
Deploy the API upon successful validation



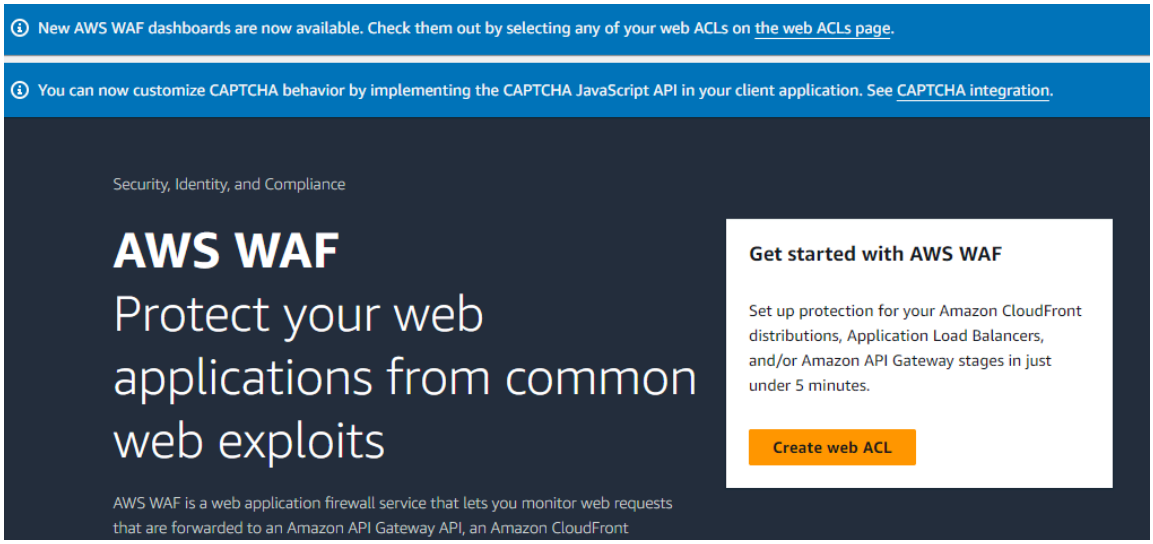
Goto stages, and get the URL of the resource you created for the kubernetes service



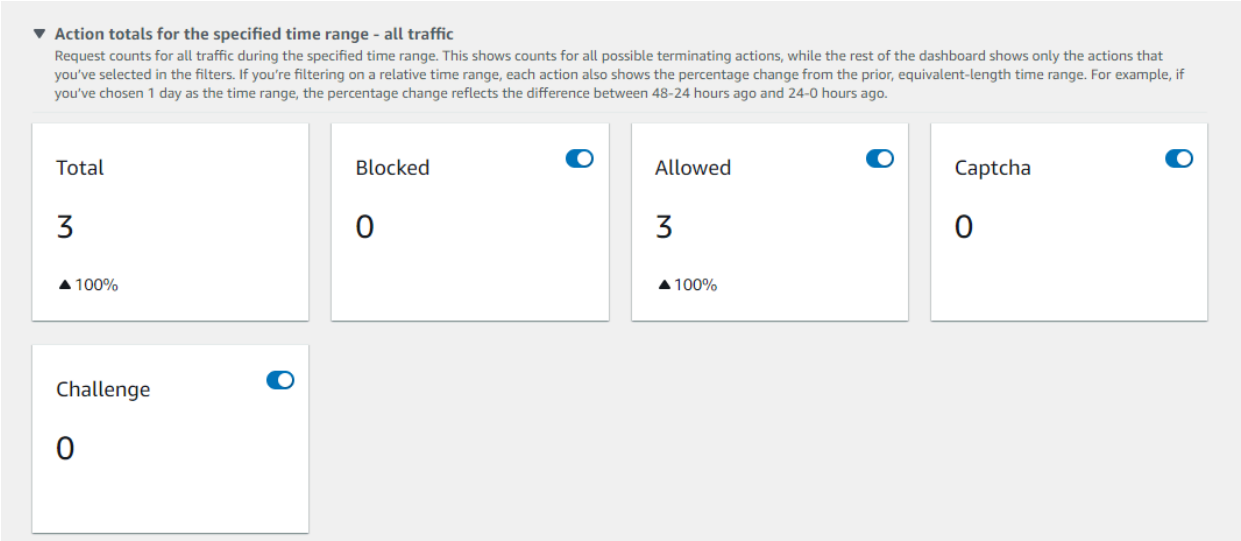
With this URL, your service can be accessed. Sorry for poor HTMLing.



10. Secure the API gateway with WAF from potential SQL attacks. Goto WAF & Shield service and
Associate your API with WAF. Use this link for teh instructions : link



Once the WAF is associated, you will see the incoming traffic to API in the dash board.



10. API Gateway Justification : I chose REST API because of its ability to add security layer(WAF) . assuming this application is having potential risks.
11. Challenges and Resolutions: After deploying the services in , NLB is created but service is not responding despite VPC and security groups are configured correctly. After checking the below logs, its found out the alb-ingress -controller does not have "ec2:DescribeAvailabilityZones". So edited the policy of IAM role of the ingress controller, which solved the issue.

kubectl logs -n kube-system deployment/aws-load-balancer-controller
kubectl describe service external-nginx-service

12. Source Code : refer to the github repo given above
13.Demo : API Gateway functioning is recorded and uploaded here : assignment_recording