

MODULE 3

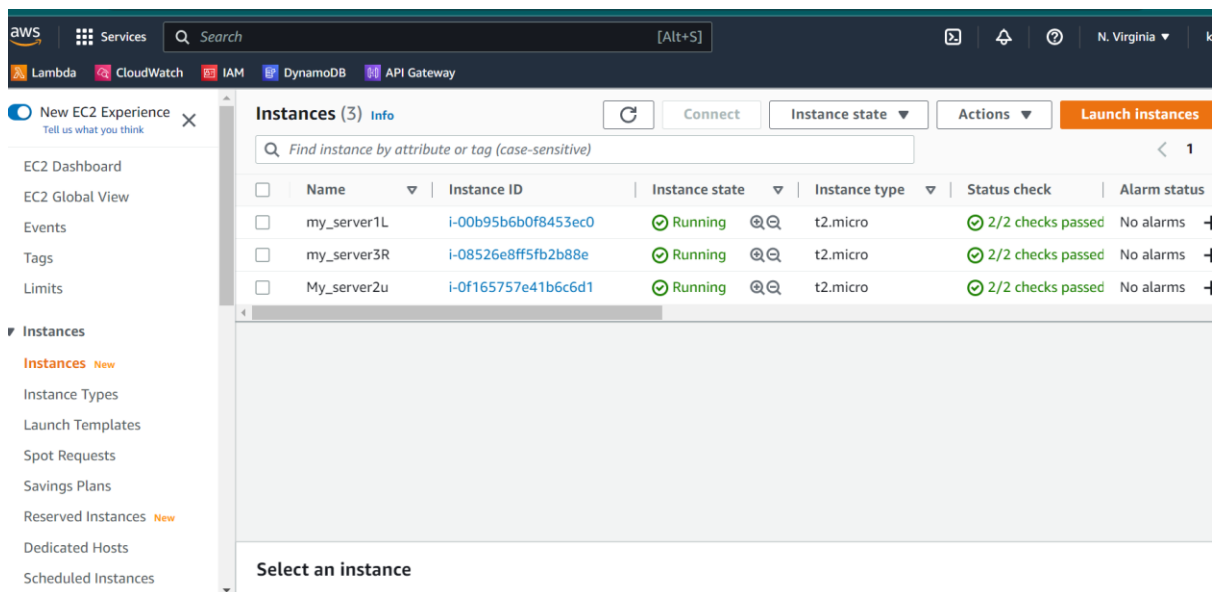
ASSIGNMENT 1

Module-3: ELB Assignment - 1

You have been asked to:

1. Create a Classic Load Balancer and register 3 EC2 instances with different web pages running in them
 2. Migrate the Classic Load Balancer into an Application Load Balancer
-

1. Created 3 instances with redhat,linux,ubuntu



Welcome to nginx!

If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

For online documentation and support please refer to nginx.org.
Commercial support is available at nginx.com.

Thank you for using nginx.

➔ Default page of the AMI

hi team

Step 1: Define Load Balancer

Basic Configuration

This wizard will walk you through setting up a new load balancer. Begin by giving your new load balancer a unique name so that you can identify it from other load balancers you might create. You will also need to configure ports and protocols for your load balancer. Traffic from your clients can be routed from any load balancer port to any port on your EC2 instances. By default, we've configured your load balancer with a standard web server on port 80.

Load Balancer name:

Create LB Inside:

Create an internal load balancer: ☐ [\(what's this?\)](#)

Enable advanced VPC configuration: ☐

Listener Configuration:

Load Balancer Protocol	Load Balancer Port	Instance Protocol	Instance Port
<input type="text" value="HTTP"/>	<input type="text" value="80"/>	<input type="text" value="HTTP"/>	<input type="text" value="80"/>

[Cancel](#) [Next: Assign Security Groups](#)

➔ Created Load Balancer

EC2 | Load balancers | loadBalancer

Load balancer: loadBalancer

Description Instances Health check Listeners Monitoring Tags Migration

Basic Configuration

Name	loadBalancer	Creation time	November 25, 2022 at 6:03:29 PM UTC+5:30
* DNS name	loadBalancer-1036903214.us-east-1.elb.amazonaws.com (A Record)	Hosted zone	Z35SXDTRQ7X7K
Type	Classic (Migrate Now)	Status	0 of 3 instances in service
Scheme	internet-facing	VPC	vpc-0839c41cfd7d8746c
Availability Zones	subnet-02c1d0ed25926459f - us-east-1f, subnet-03c9030bae9ebb891 - us-east-1c, subnet-0a43ae1495deda633 - us-east-1b, subnet-0a78bbc85295fdd94 - us-east-1d, subnet-0b1c625e5533aa1f8 - us-east-1a, subnet-0f12961768daa5d41 - us-east-1e		

Port Configuration

➔ Migrated to Application Load Balancer

➔ As the Classic load balancer is retired, created an Application load balancer for this assignment.

Load Balancer Creation Status



Successfully created load balancer

Load balancer [loadBalancer](#) was successfully created.

Note: It might take a few minutes for your load balancer to be fully set up and ready to route traffic, and for the targets to complete the registration process and pass the initial health checks.

After migration is complete, you can do the following as needed:

- Redirect traffic to your new load balancer (see [Migrate Traffic](#)).
- Change the deregistration delay (see [Deregistration Delay](#)). The default is 300 seconds.
- Change the idle connection timeout if needed (see [Connection Idle Timeout](#)). The default is 60 seconds.
- Enable access logs (see [Access Logs](#)).

Suggested next steps

- Discover other services that you can integrate with your load balancer. Visit the **Integrated services** tab within [loadBalancer](#)
- Consider using AWS Global Accelerator to further improve the availability and performance of your applications. [AWS Global Accelerator console](#)

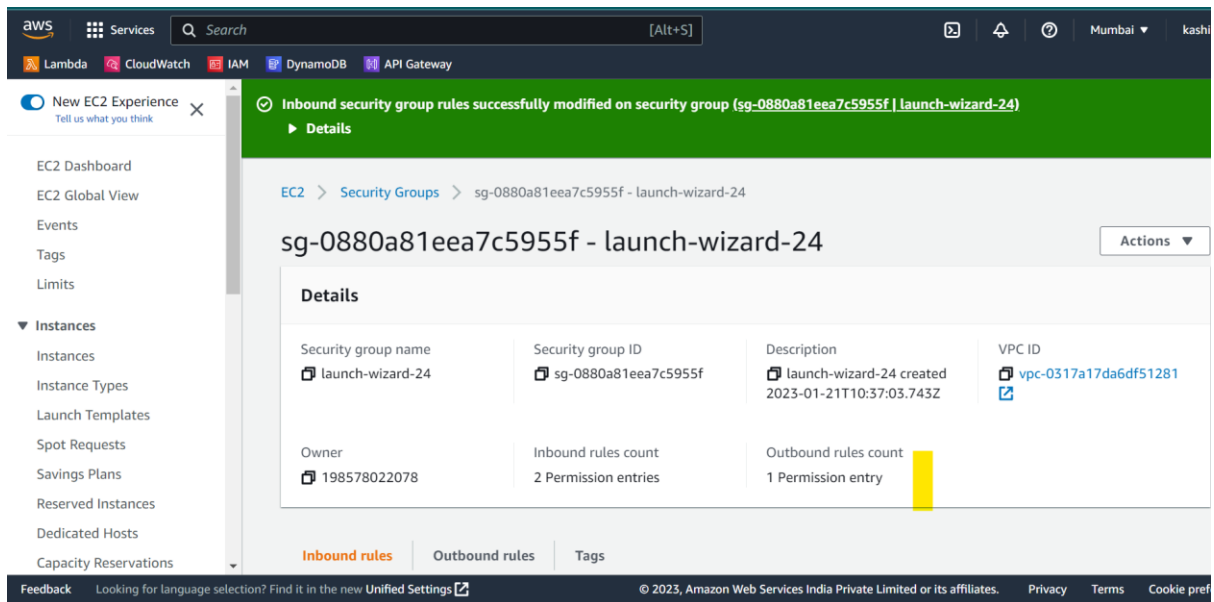
Close

ASSIGNMENT 2

Module-2: Auto Scaling Assignment - 2

You have been asked to:

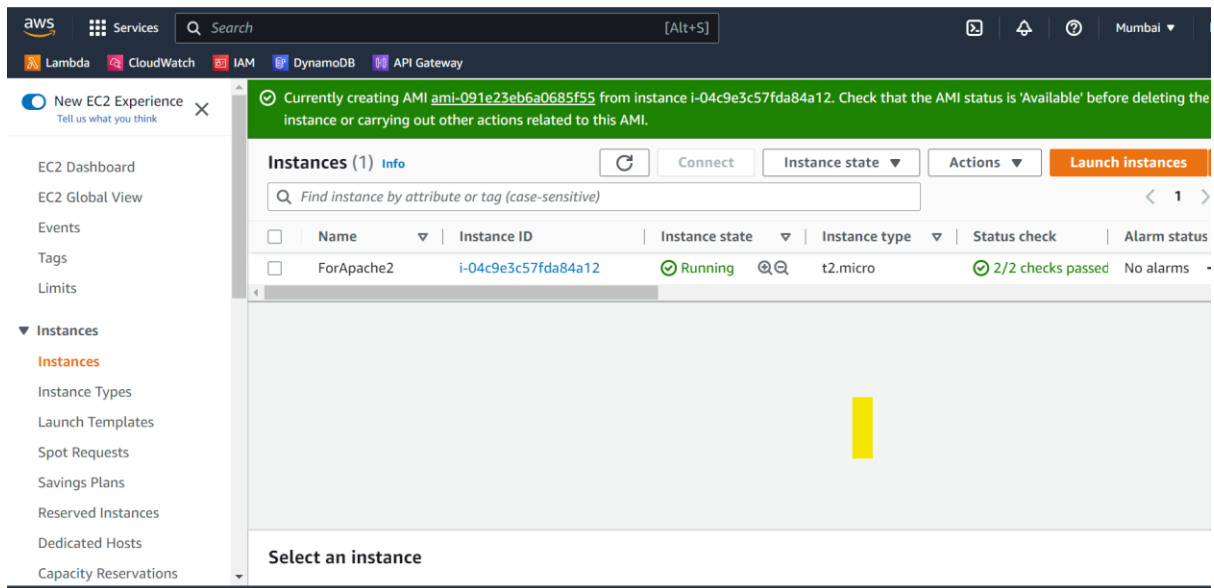
1. Create a Web Server AMI with Apache 2 server running in it
 2. Create a Launch Configuration with this AMI
 3. Use this Launch Configuration to create an Auto Scaling group with 1 minimum and 3 maximum instances
-



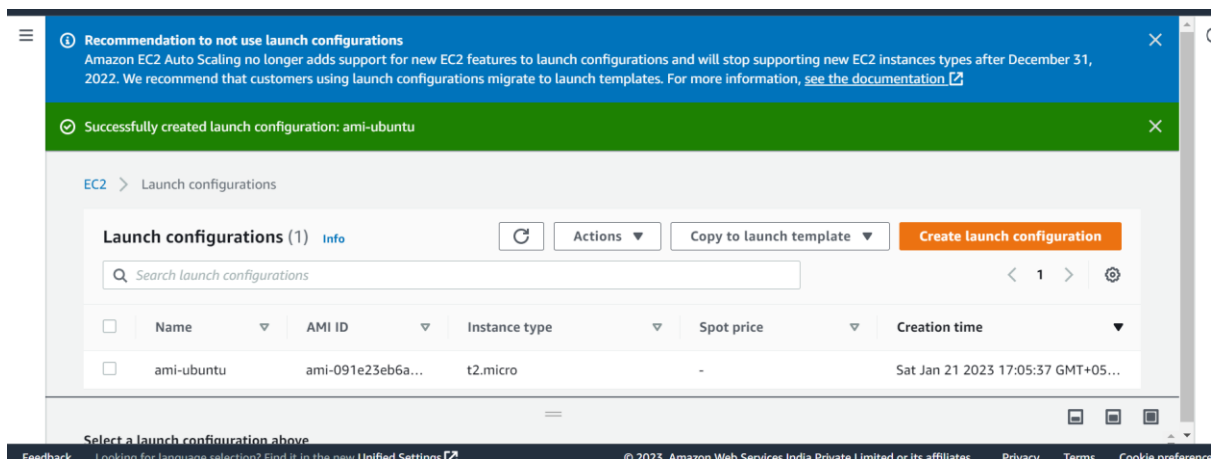
➔ Set up ec2 and security group



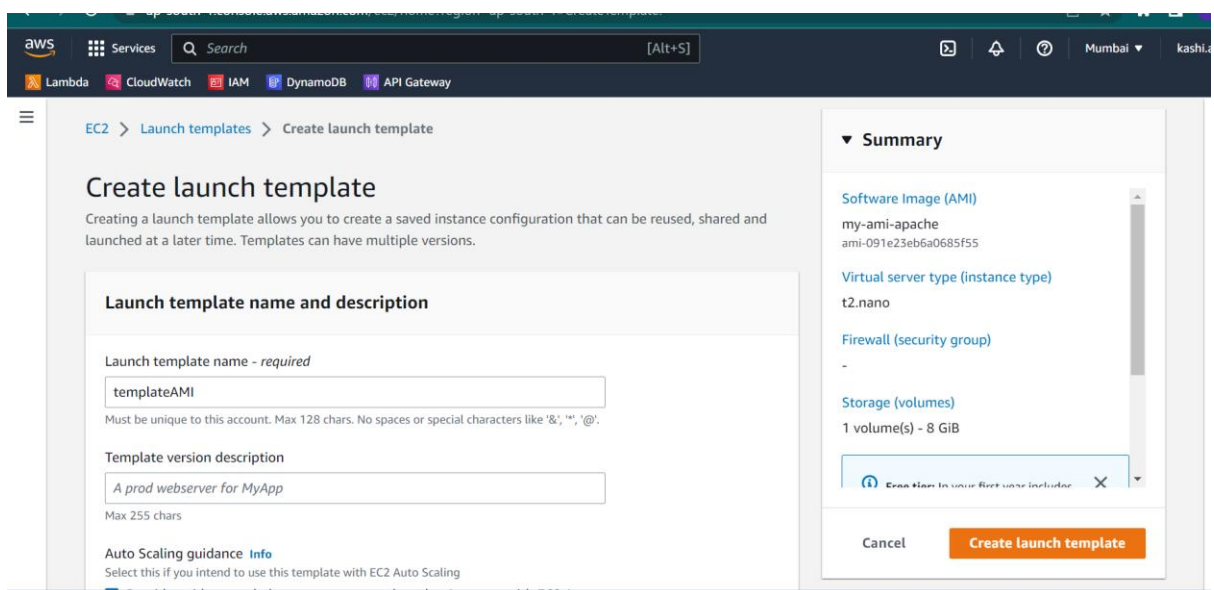
➔ Installed Apache 2 on server

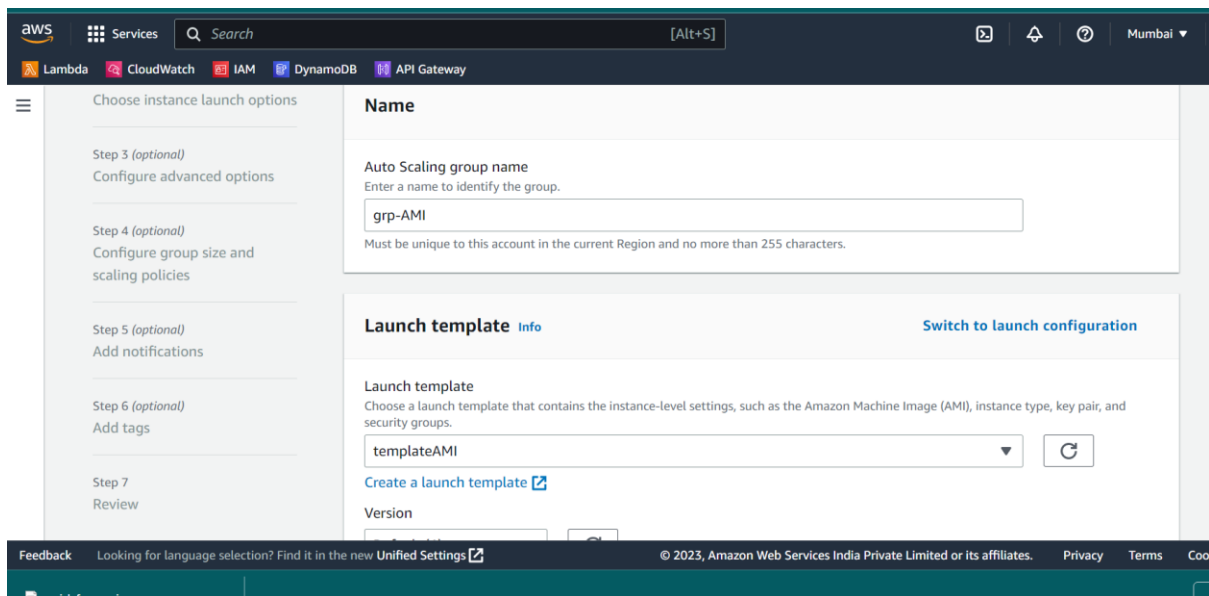


➔ Created AMI from Instance

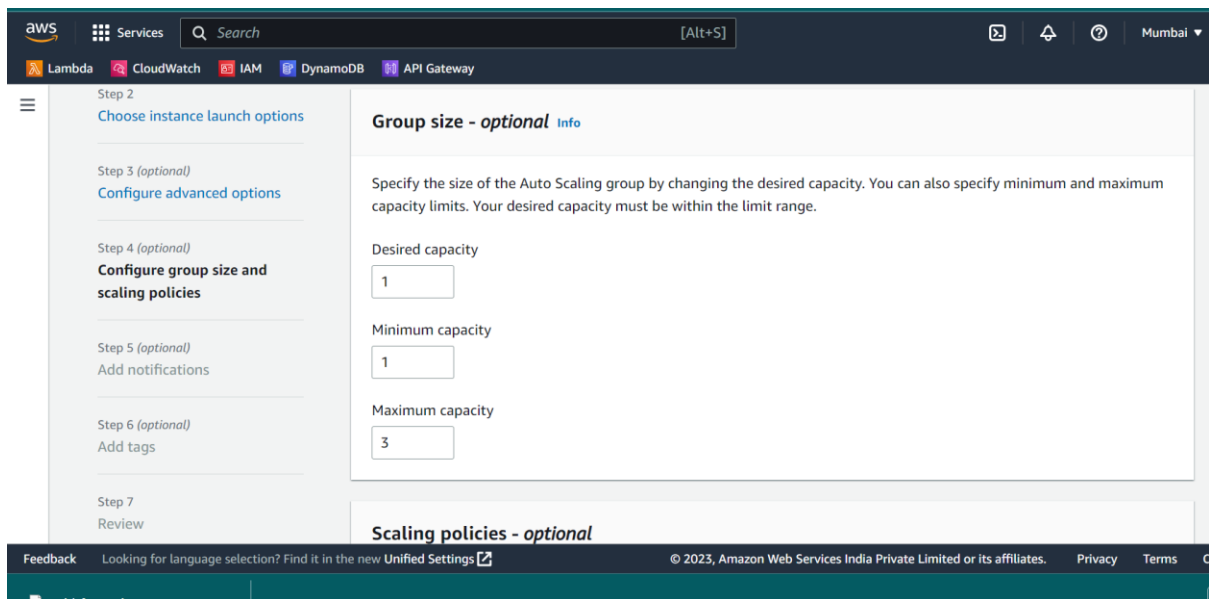


➔ As Auto scaling does not support launch configuration for ec2 have used launch template





➔ Created Auto-scaling Group with min capacity 1 and max capacity 3



ASSIGNMENT 3



Module-3: Route 53 Assignment - 3

You have been asked to:

1. Use the Route 53 Hosted Zone created in the Assignment
2. Route the traffic to an EC2 instance with an Apache web server running in it using it's IP address



➔ Created ec2 instance with apache 2 on it

```
Processing triggers for libc-bin (2.35-0ubuntu3.1) ...
Scanning processes...
Scanning linux images...

Running kernel seems to be up-to-date.

No services need to be restarted.

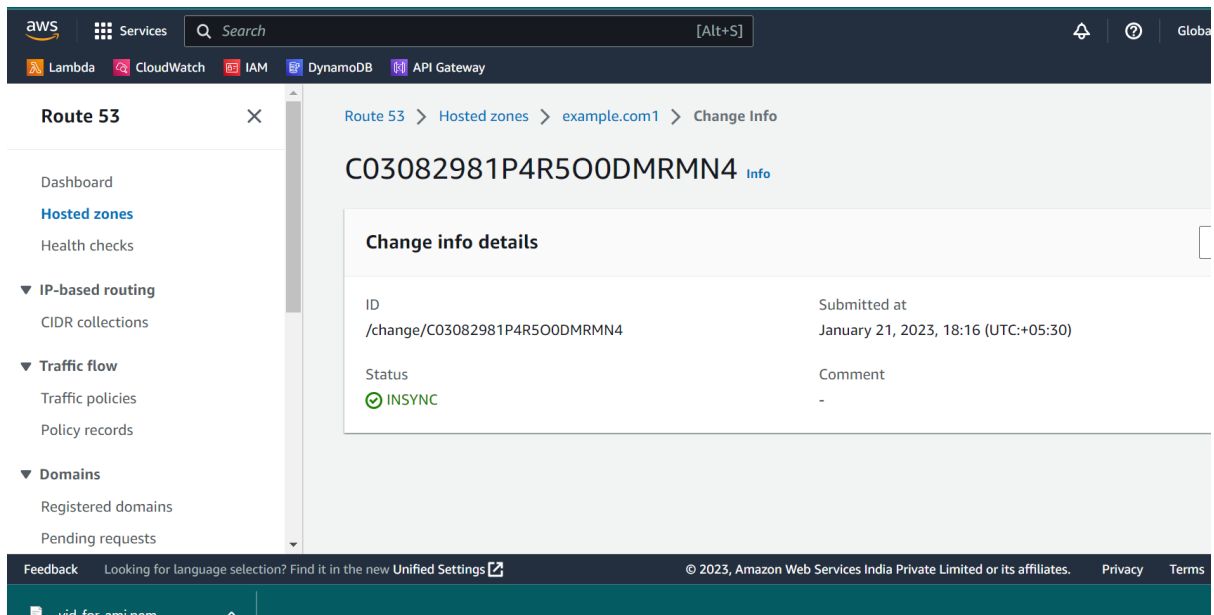
No containers need to be restarted.

No user sessions are running outdated binaries.

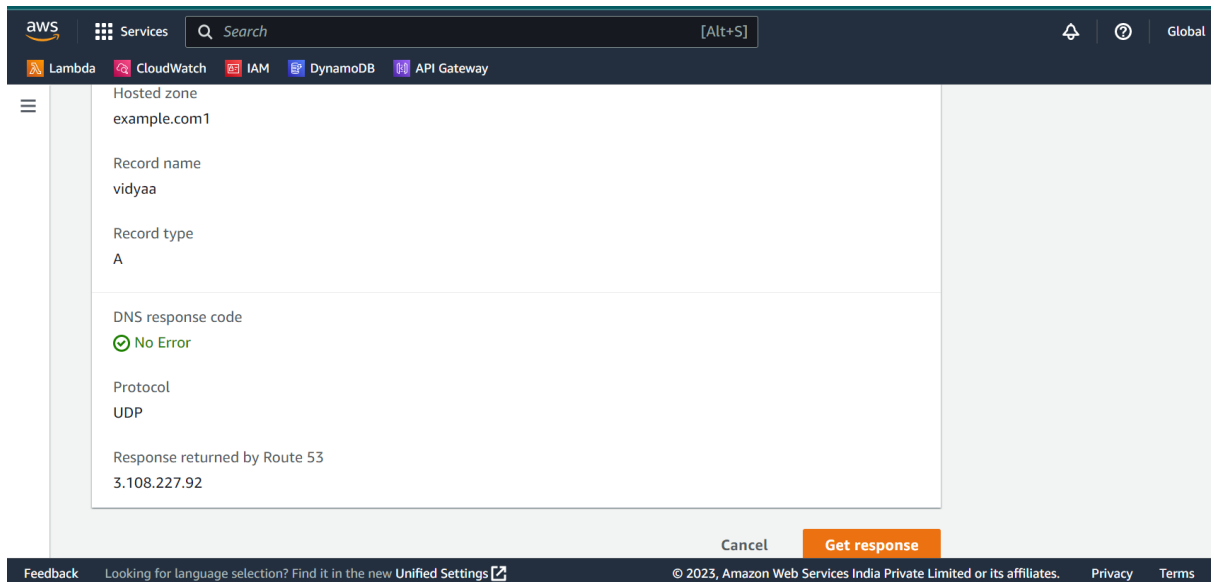
No VM guests are running outdated hypervisor (qemu) binaries on this host.
root@ip-172-31-36-122:/home/ubuntu# sudo apt --version
apt 2.4.8 (amd64)
root@ip-172-31-36-122:/home/ubuntu# sudo apt ubuntu --version
apt 2.4.8 (amd64)
root@ip-172-31-36-122:/home/ubuntu#
```

i-08f74b2277d0ecc45 (my-ec2)
PublicIPs: 3.108.227.92 PrivateIPs: 172.31.36.122

➔ Saved public ip



➔ Created record named vidya under zone example.com



➔ Got response from ec2 IP