PROJECT-1

Arctitecture

VPC

Your VPCs (1/1) Info

Actions ▼ | Create VPC

Filter VPCs

Name	VPC ID	State	IPv4 CIDR	IPv6 CIDR	DHCI
D-VPC	vpc-0f86a7b4f5b69fdf7	Available	172.31.0.0/16	–	dopt-

Subnet used(1a & 1b)

Subnets (3) Info

Actions ▼ | Create subnet

Filter subnets

Name	Subnet ID	State	VPC	IPv4 CIDR	IPv6 CIDR
D-PUBLIC-AZ1	subnet-025fe94c95230afa3	Available	vpc-0f86a7b4f5b69fdf7 | D-VPC	172.31.32.0/20	–
D-PUBLIC-AZ2	subnet-090968c4da7702e9f	Available	vpc-0f86a7b4f5b69fdf7 | D-VPC	172.31.16.0/20	–
D-PUBLIC-AZ3	subnet-093457fcdb33ad850	Available	vpc-0f86a7b4f5b69fdf7 | D-VPC	172.31.0.0/20	–

Route table

Route tables (1/1) Info

Actions ▼ | Create route table

Filter route tables

Name	Route table ID	Explicit subnet associat...	Edge associations	Main	VPC
PUBLIC-ROUTE	rtb-07e9d1439e5004003	3 subnets	–	Yes	vpc-0f86a7b4f5b69fdf7 | D-VPC

Default Gateway

Internet gateways (1/1) Info

Actions ▼ | Create internet gateway

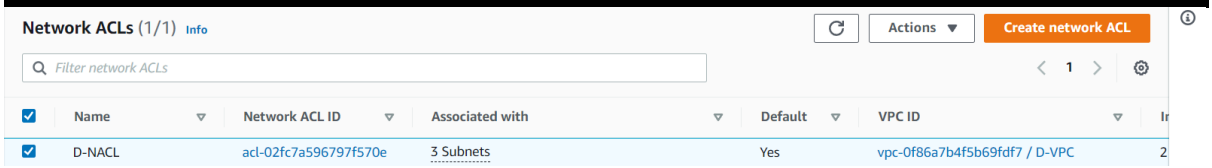
Filter internet gateways

Name	Internet gateway ID	State	VPC ID	Owner
D-IGW	igw-086877cd89921363d	Attached	vpc-0f86a7b4f5b69fdf7 | D-VPC	659202326636

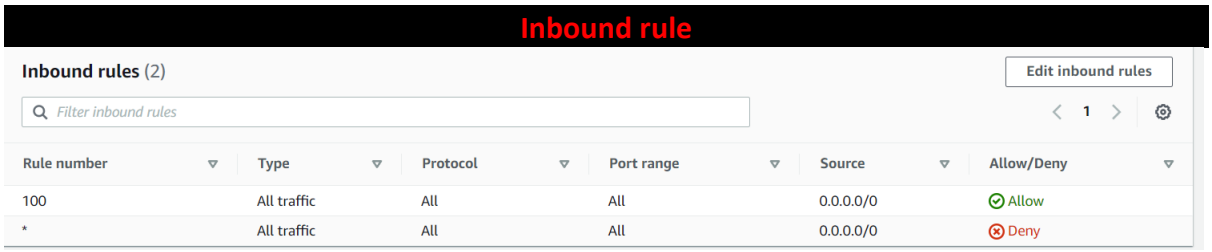
CREATED BY:- VIVEK GANDHI

PROJECT-1

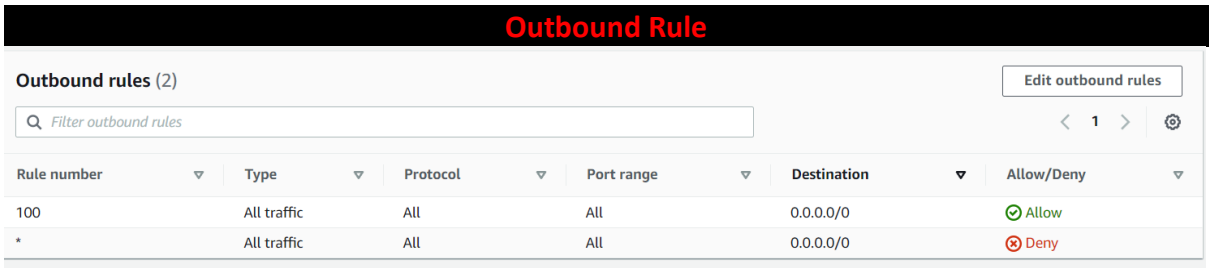
Default NACL



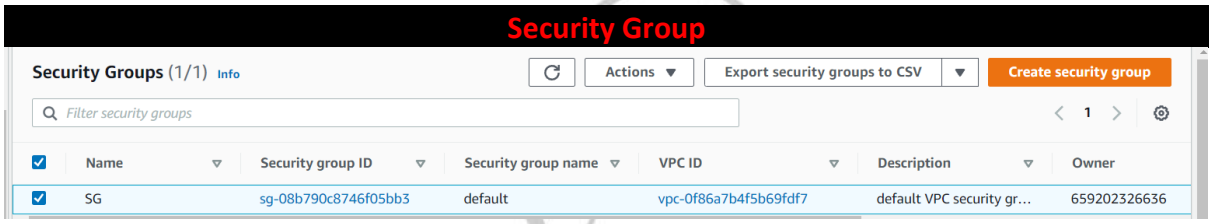
Inbound rule



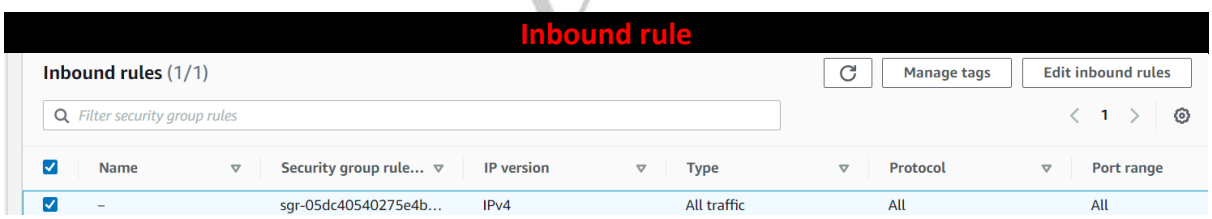
Outbound Rule



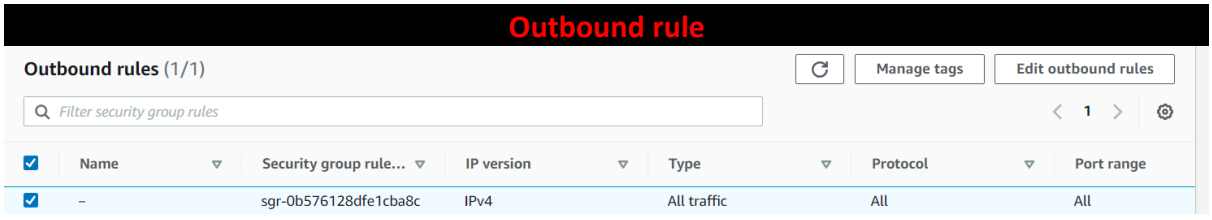
Security Group



Inbound rule



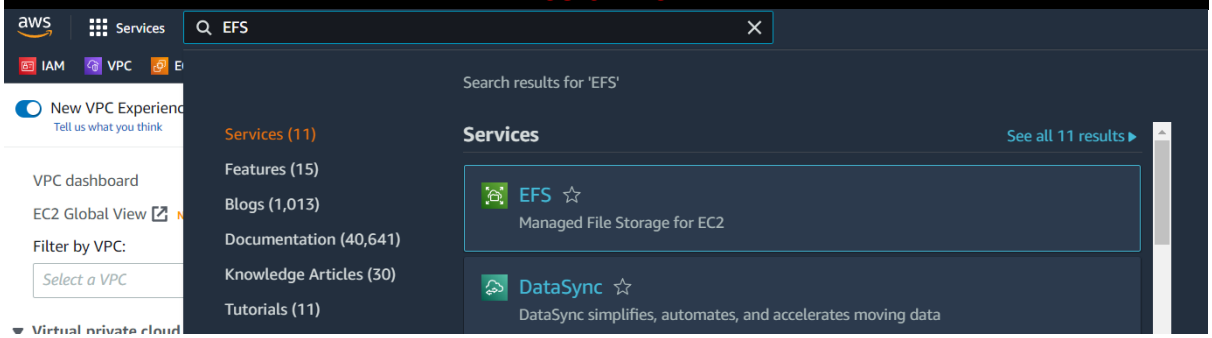
Outbound rule



PROJECT-1

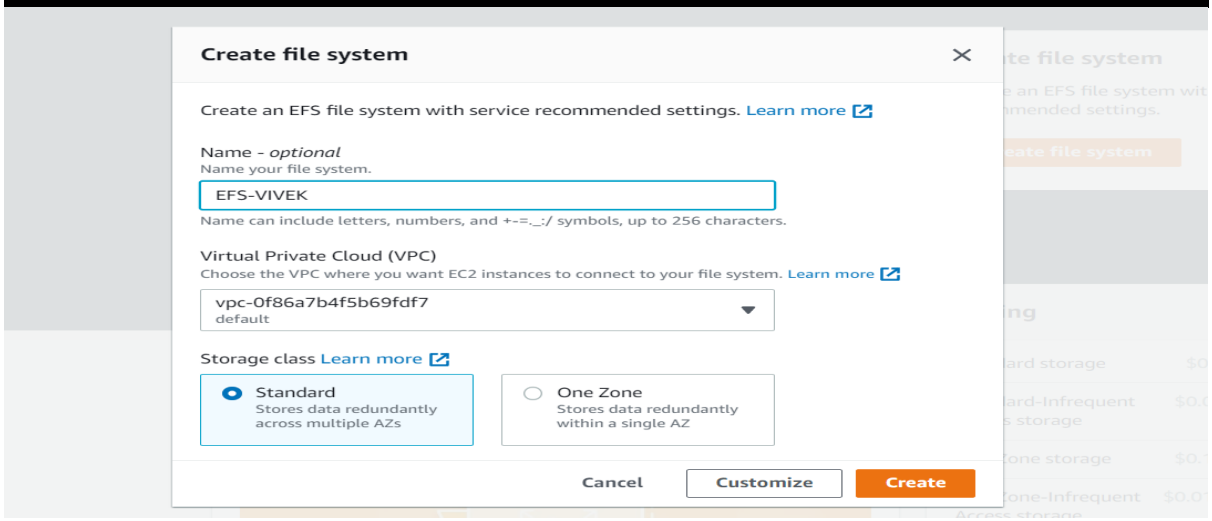
1. Create EFS

Serch EFS

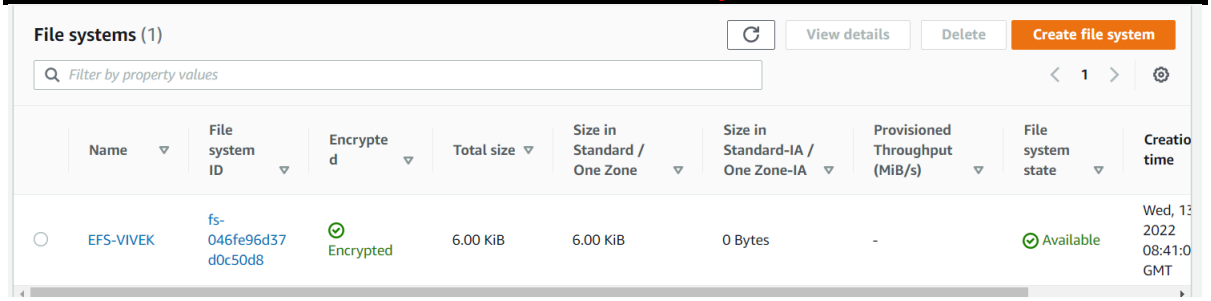


Select Create file system

ENTER name and select VPC &AZ



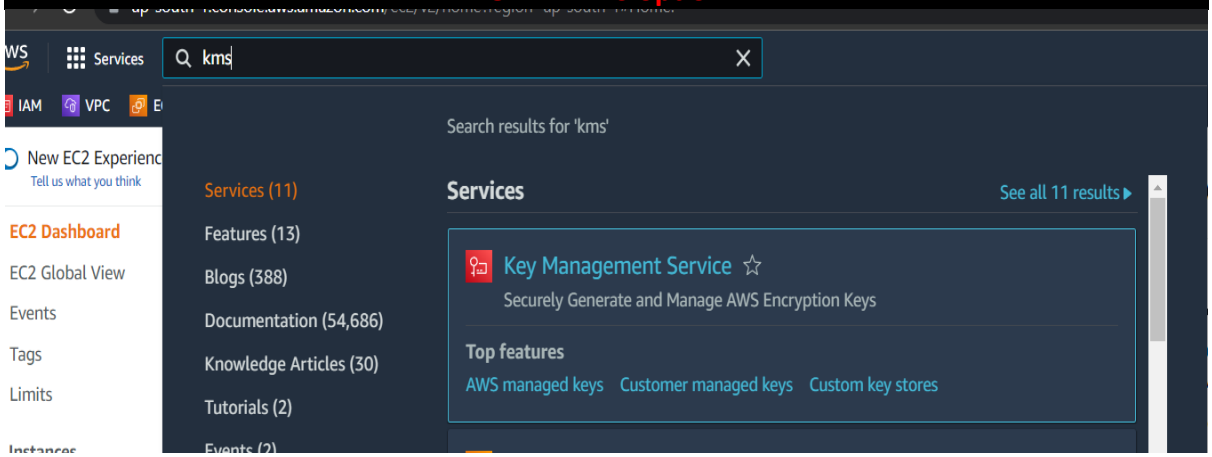
Create successfully



PROJECT-1

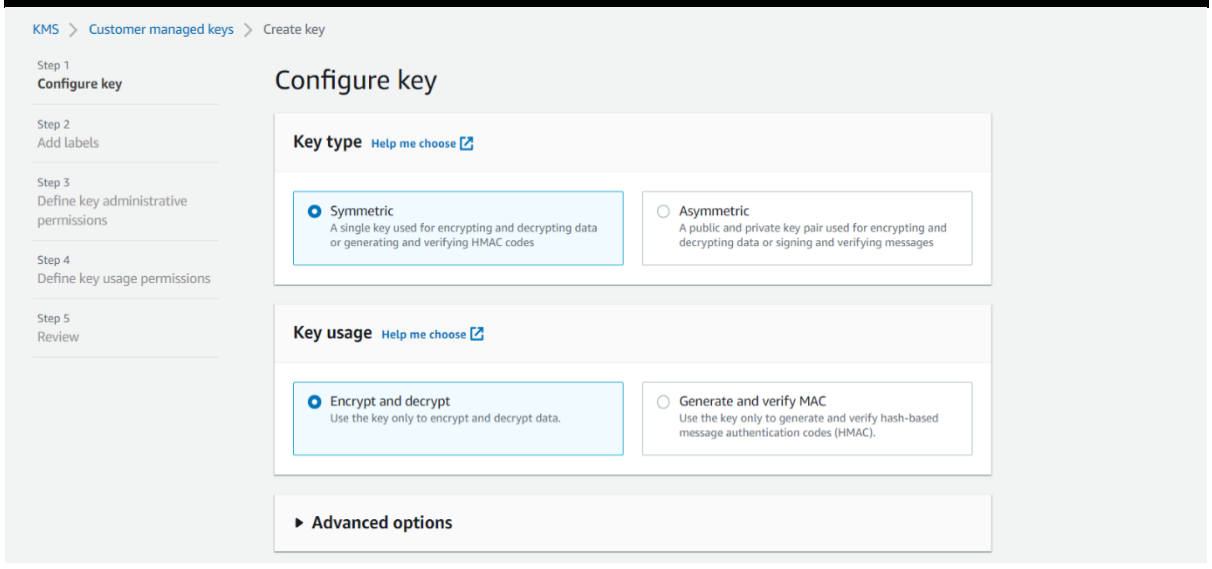
2. Create KMS (if configure Autoscaling not create instance)

OPEN KMS option



Select option create KMS

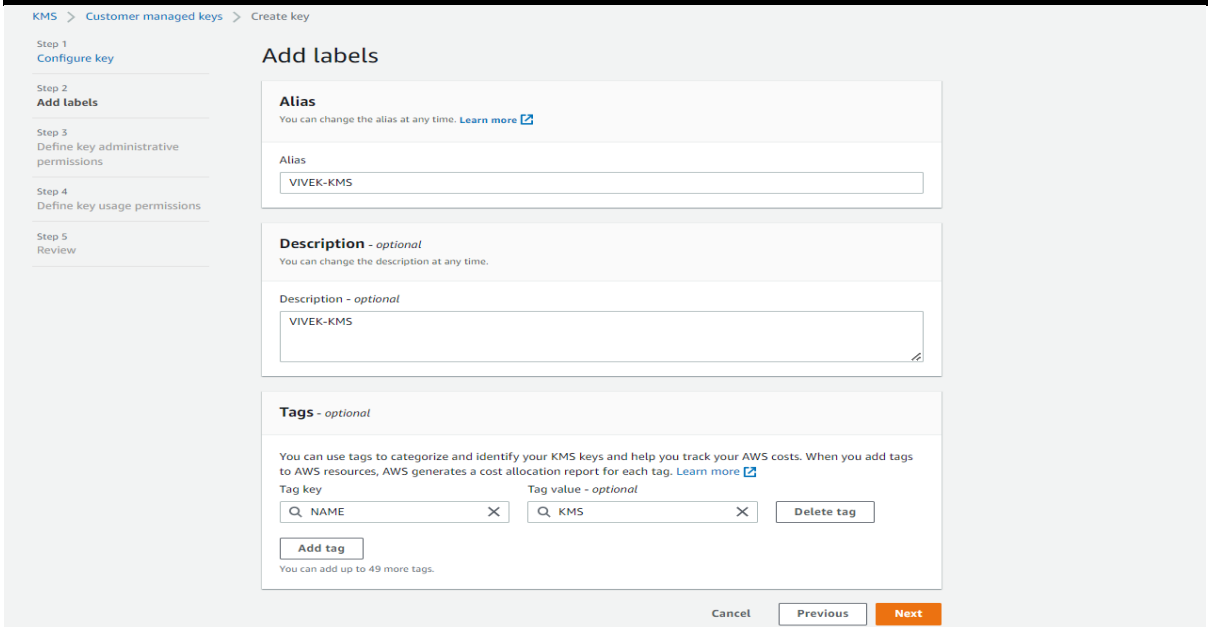
CONFIGURE STEPS



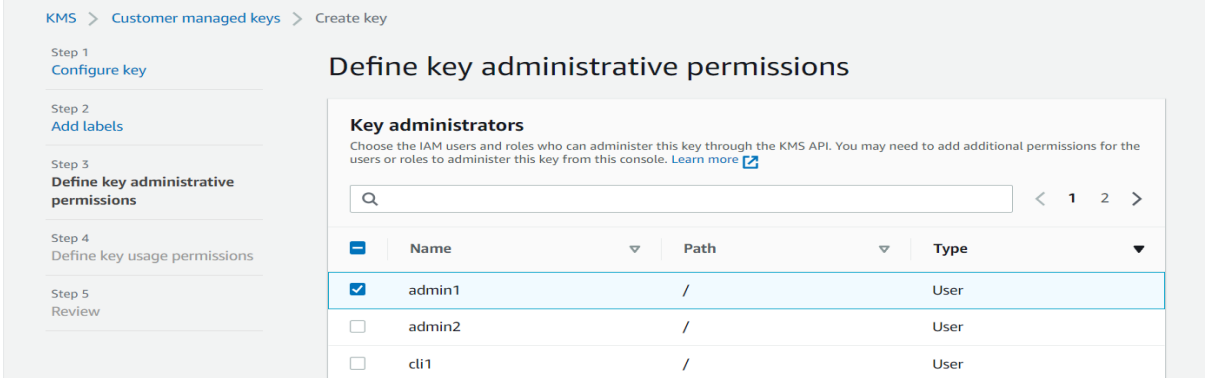
CREATED BY:- VIVEK GANDHI

PROJECT-1

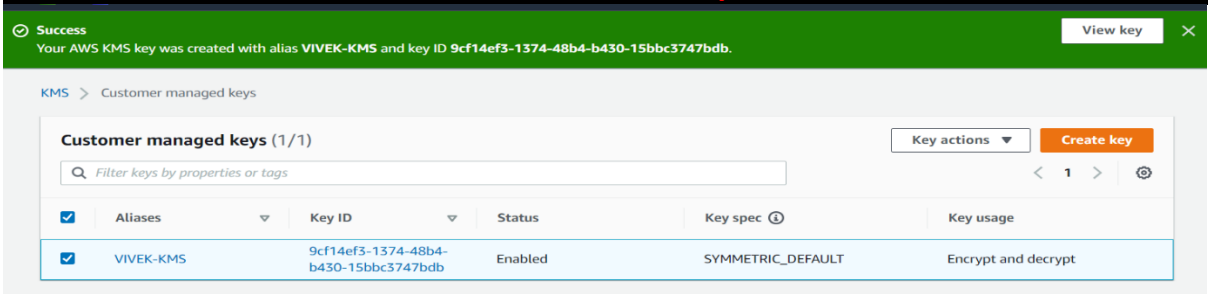
Enter alias name



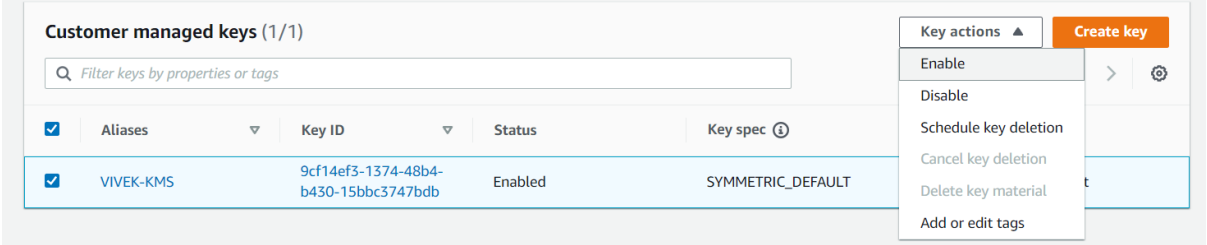
Select user



Create successfully



Enable KMS

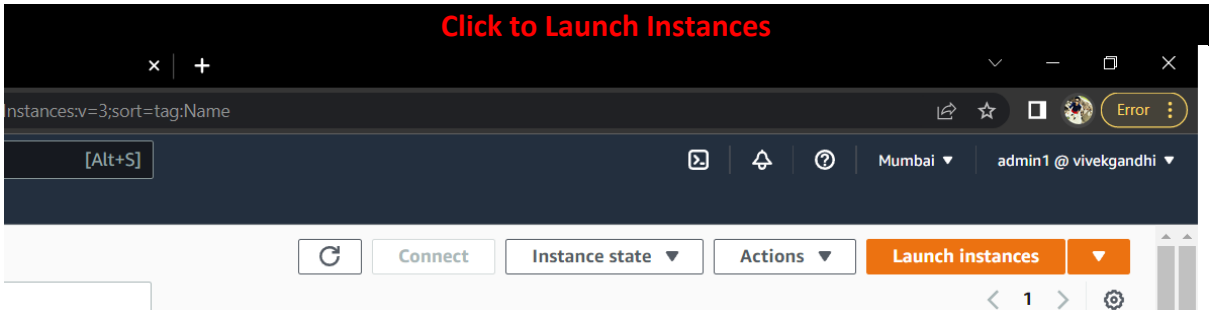


CREATED BY:- VIVEK GANDHI

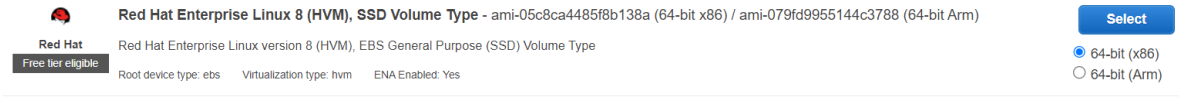
PROJECT-1

3. Create EC2 INSTANCES

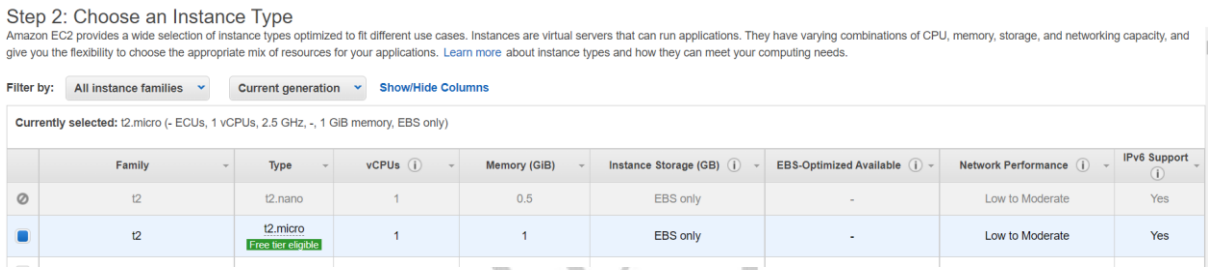
Click to Launch Instances



Select Redhat Linux

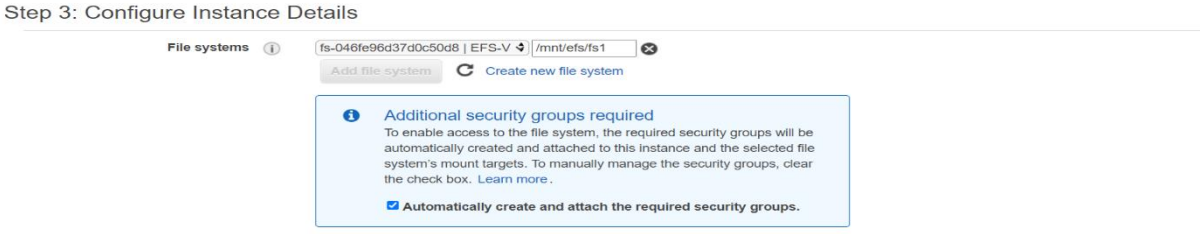


Select instance type T2 micro

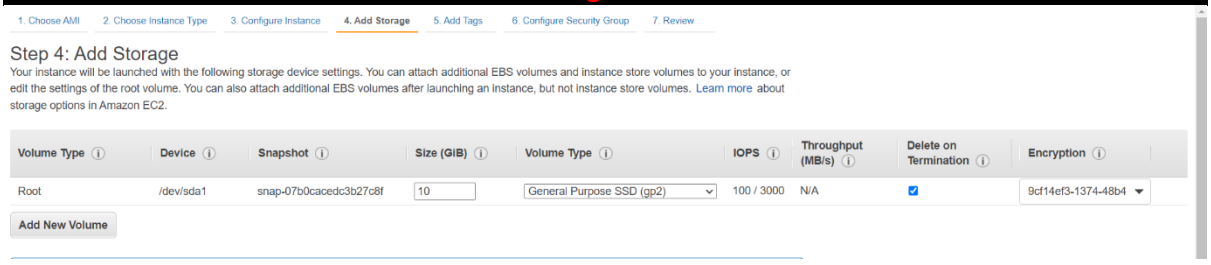


Select subnet and make sure auto-assign is public

Select EFS



Add storage with KMS



PROJECT-1

Add Tags

1. Choose AMI | 2. Choose Instance Type | 3. Configure Instance | 4. Add Storage | **5. Add Tags** | 6. Configure Security Group | 7. Review

Step 5: Add Tags

A tag consists of a case-sensitive key-value pair. For example, you could define a tag with key = Name and value = Webserver.
A copy of a tag can be applied to volumes, instances or both.
Tags will be applied to all instances and volumes. Learn more about tagging your Amazon EC2 resources.

Key (128 characters maximum)	Value (256 characters maximum)	Instances	Volumes	Network Interfaces
Name	Main Copy	☑	☑	☑

Add another tag (Up to 50 tags maximum)

SELECT SG

Step 6: Configure Security Group

A security group is a set of firewall rules that control the traffic for your instance. On this page, you can add rules to allow specific traffic to reach your instance. For example, if you want to set up a web server and allow Internet traffic to reach your instance, add rules that allow unrestricted access to the HTTP and HTTPS ports. You can create a new security group or select from an existing one below. Learn more about Amazon EC2 security groups.

Assign a security group: ◉ Create a **new** security group
○ Select an **existing** security group

Security group name: EC2-Main-SG
Description: launch-wizard-1 created 2022-07-13T14:50:54.387+05:30

Type	Protocol	Port Range	Source	Description
All traffic	All	0 - 65535	Anywhere 0.0.0.0/0, ::/0	e.g. SSH for Admin Desktop

Add Rule

Download Key

Create a new key pair

Key pair type

◉ RSA ○ ED25519

Key pair name

EC2-MAIN

Download Key Pair

Create successfully

Instances (1) Info

Connect | Instance state | Actions | Launch instances

Search

Instance state = running | Clear filters

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DN
Main Copy	i-0bc12764930795c99	Running	t2.micro	2/2 checks passed	No alarms	ap-south-1a	ec2-3-110-190

Open puttygen (Load → Save Private key)

PuTTY Key Generator

File Key Conversions Help

Key

Public key for pasting into OpenSSH authorized_keys file:

```
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABAQCj9YWuBPfLgTDhc9MbkNCYKQ32qw5R0Nqd
+Fc846SDfMe6kX1chXo/u/0AKQ+JyBRPvk
+ZHy8UavmSRzVVtFWGChobdmKjP2ORu4zJaMgKtt02APDTknhJUohOXhCRAWeZVF8kQs6Pu/6dkXdqEUaWWNA6J
ucB9wJDcnXwyL+h1hZYLPAYZJaV
+3YKcNcHZckumuByCfzGexJrVW9/MPC65dev997SdRaXl3CeVgUu0A1cEFLFw7D06SjYfHeHoEXBhgd1JR9iGiQdOa1/8
```

Key fingerprint: ssh-rsa 2048 SHA256:75GRoLgaZ2LTwLOUboYa8wgsUB1I5lqlilvWAV2SJmc
Key comment: imported-openssh-key
Key passphrase:
Confirm passphrase:

Actions

Generate a public/private key pair — Generate
Load an existing private key file — Load
Save the generated key — Save public key | Save private key

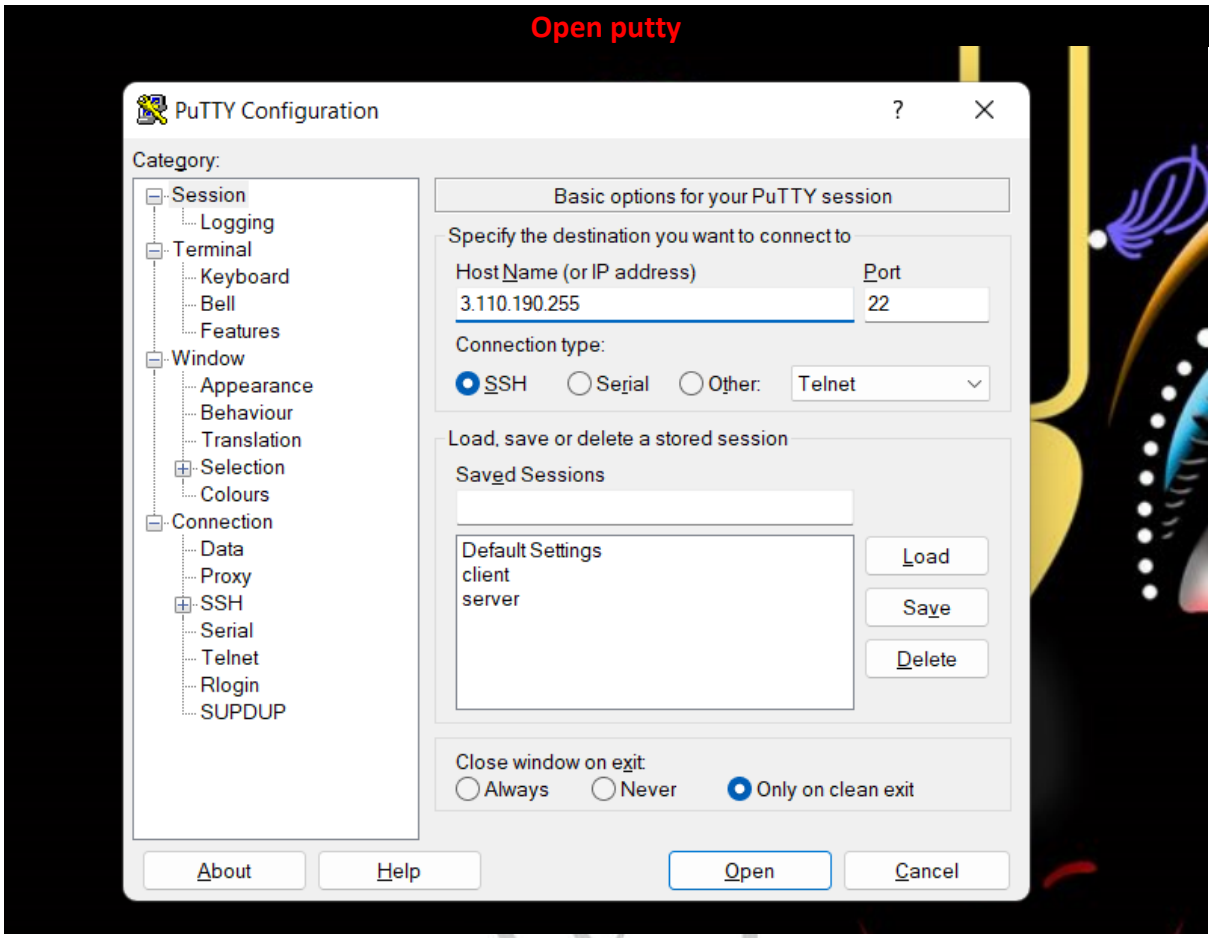
Parameters

Type of key to generate:
◉ RSA ○ DSA ○ ECDSA ○ EdDSA ○ SSH-1 (RSA)
Number of bits in a generated key: 2048

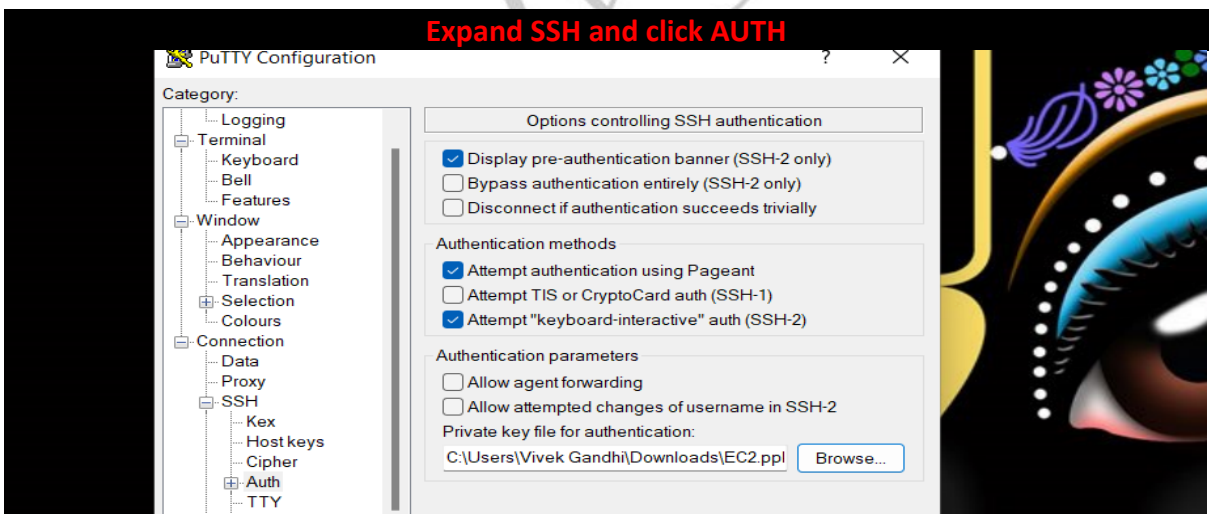
CREATED BY:- VIVEK GANDHI

PROJECT-1

Open putty



Expand SSH and click AUTH



Enter User name

PROJECT-1

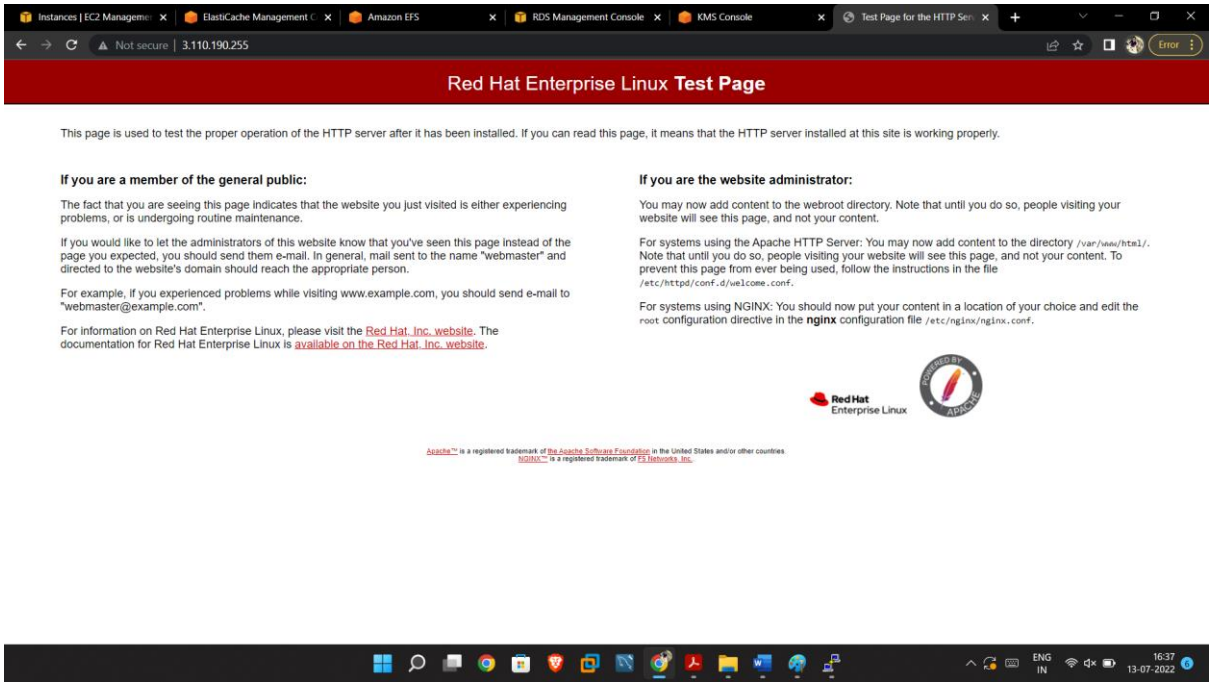
Install httpd

```
[root@ip-172-31-35-217 ec2-user]# yum install httpd
Updating Subscription Management repositories.
Unable to read consumer identity

This system is not registered with an entitlement server. You can use subscription-manager to register.

Last metadata expiration check: 0:26:54 ago on Wed 13 Jul 2022 09:46:13 AM UTC.
Dependencies resolved.
```

Start services

root@ip-172-31-35-217:/home/ec2-user

```
[root@ip-172-31-35-217 ec2-user]# systemctl start httpd
[root@ip-172-31-35-217 ec2-user]# systemctl enable httpd
Created symlink /etc/systemd/system/multi-user.target.wants/httpd.service → /usr/lib/systemd/system/httpd.service.
[root@ip-172-31-35-217 ec2-user]#
```



Install redis service

```
[root@ip-172-31-35-217 ec2-user]# yum install redis
Updating Subscription Management repositories.
Unable to read consumer identity

This system is not registered with an entitlement server. You can use subscription-manager to register.

Last metadata expiration check: 0:30:16 ago on Wed 13 Jul 2022 09:46:13 AM UTC.
Dependencies resolved.
```

SEE EFS

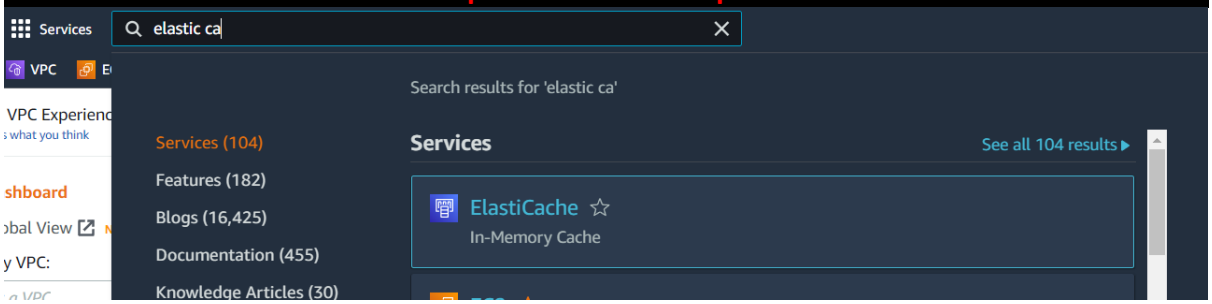
root@ip-172-31-35-217:/home/ec2-user

```
[root@ip-172-31-35-217 ec2-user]# df -hT
Filesystem                                      Type      Size  Used Avail Use% Mounted on
devtmpfs                                        devtmpfs  368M     0  368M   0% /dev
tmpfs                                           tmpfs     401M     0  401M   0% /dev/shm
tmpfs                                           tmpfs     401M   16M  386M   4% /run
tmpfs                                           tmpfs     401M     0  401M   0% /sys/fs/cgroup
/dev/xvda2                                      xfs        10G  2.7G  7.4G  27% /
fs-046fe96d37d0c50d8.efs.ap-south-1.amazonaws.com:/ nfs4      8.0E     0  8.0E   0% /mnt/efs/fs1
tmpfs                                           tmpfs      81M     0   81M   0% /run/user/1000
[root@ip-172-31-35-217 ec2-user]#
```

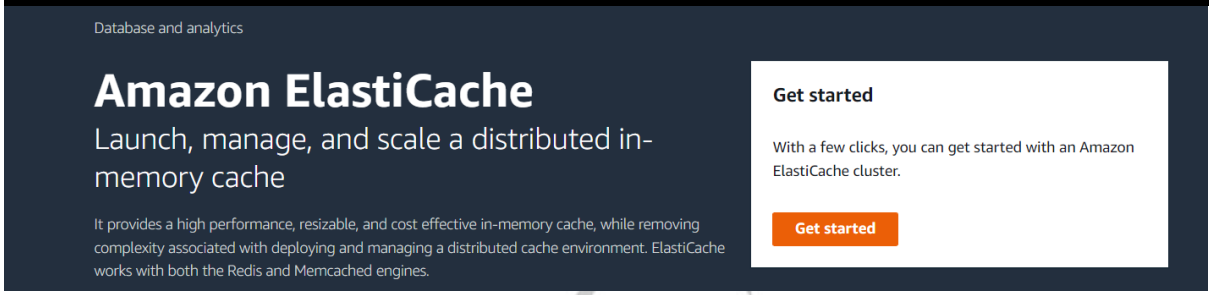
PROJECT-1

4. Create Elastic Cache

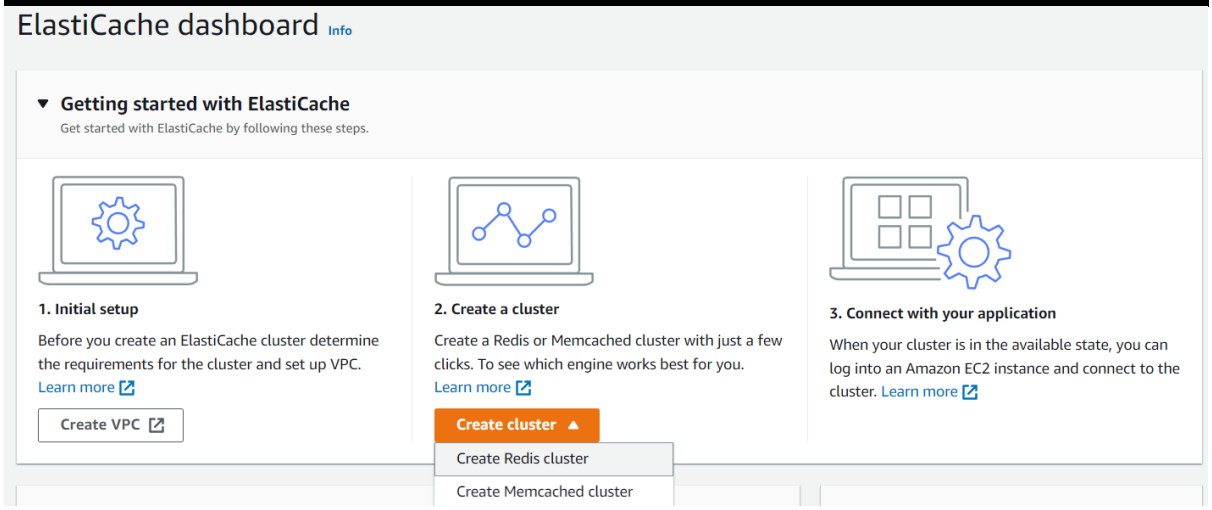
Open Elastic cache open



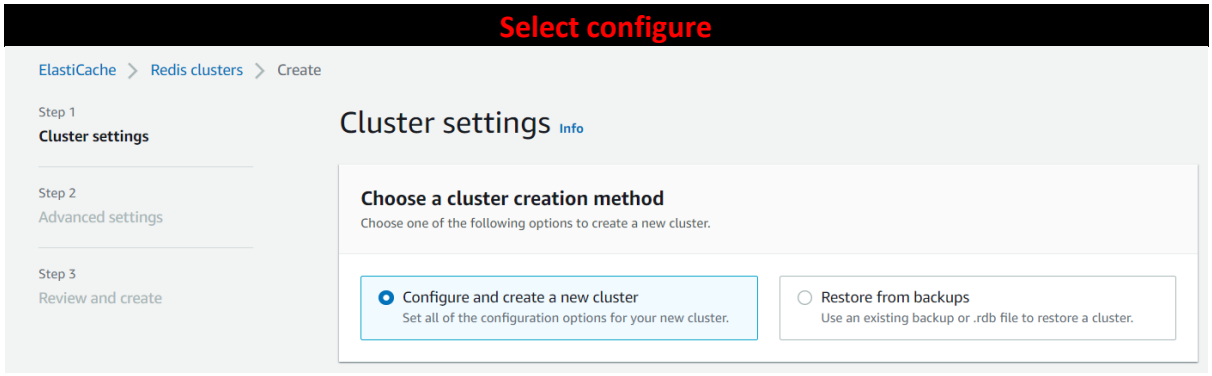
Get started



Select redis



Select configure



PROJECT-1

Enabled

Cluster mode

Scale your cluster dynamically with no downtime.

- (●) **Enabled** — Cluster mode enables replication across multiple shards for enhanced scalability and availability.
- ( ) **Disabled** — The Redis cluster will have a single shard (node group) with one primary node and up to 5 read replica.

Enabling cluster mode supports partitioning your data across up to 500 node groups and improves performance of Redis clusters. Some commands are unavailable in this mode. Learn more

Enter name

Cluster info

Use the following options to configure the cluster.

Name

cache-vivek

The name is required, can have up to 40 characters, and must begin with a letter. It should not end with a hyphen or contain two consecutive hyphens. Valid characters: A-Z, a-z, 0-9, and - (hyphen).

Description - *optional*

cache-vivek

Add location

Location

Choose whether to host the cluster in the AWS Cloud or on premises.

Location

- (●) **AWS Cloud** — Use the AWS Cloud for your ElastiCache instances.
- ( ) **On premises** — Create your ElastiCache instances on an Outpost (through AWS Outposts). You need to create a subnet ID on an Outpost first.

Multi-AZ

- [x] **Enable** — Multi-AZ provides enhanced high availability through automatic failover to a read replica, cross AZs, in case of a primary node failover.

Auto-failover

- [x] **Enable** — ElastiCache Auto Failover provides enhanced high availability through automatic failover to a read replica in case of a primary node failover.

Cluster setting check

Cluster settings

Use the following options to configure the cluster.

Engine version

Version compatibility of the Redis engine that will run on your nodes.

6.2

Port

The port number that nodes accept connections on.

6379

Parameter groups

Parameter groups control the runtime properties of your nodes and clusters.

PROJECT-1

Subnet select

Subnet group settings

A subnet group is a collection of subnets (typically private). Designate a subnet group for your clusters running in an Amazon Virtual Private Cloud (VPC) environment.

Subnet groups

(•) Choose existing subnet group ( ) Create a new subnet group

Subnet groups

A collection of subnets that you can designate for your clusters running in an Amazon VPC.

cache (vpc-0f86a7b4f5b69fdf7) ▼

Select placements

Availability Zone placements

Use the following fields to configure placements for Availability Zones.

Slots and keyspaces

Distribution of the 16,384 Redis cluster keyspace slots across shards.

Equal distribution ▼

Availability Zone placements

HA mode - Globally, distribute AZs to maximize AZ spread across shard masters. At the second level, spread nodes within a shard across AZs for within-shard HA. Low latency mode - For fast writes, put all shard masters in the same AZ.

No preference ▼

< 1 >

Shards	Slots/keyspaces	Primary	Replica 1	Replica 2
Shard 1	Equal distribution	No preference	No preference	No preference
Shard 2	Equal distribution	No preference	No preference	No preference
Shard 3	Equal distribution	No preference	No preference	No preference

Cancel Next

Check file

Step 1
Cluster settings

Step 2
Advanced settings

Step 3
Review and create

Advanced settings Info

Security

Use the following section to configure network security and data security for your cluster.

Encryption at rest

☐ Enable

Enables encryption of data stored on disk.

Encryption in transit

☐ Enable

Enables encryption of data that moves between the service and client.

Selected SG

Selected security groups (1)

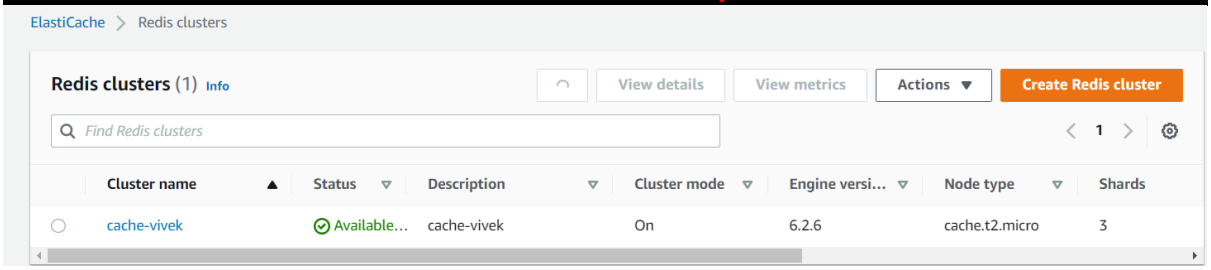
A security group acts like a firewall that controls network access to your clusters.

Manage

Group ID	Name
sg-08b790c8746f05bb3	default

PROJECT-1

Create successfully



Service start

```
[root@ip-172-31-35-217 ec2-user]# systemctl start redis
[root@ip-172-31-35-217 ec2-user]# systemctl enable redis
Created symlink /etc/systemd/system/multi-user.target.wants/redis.service → /usr/lib/systemd/system/redis.service.
[root@ip-172-31-35-217 ec2-user]#
```

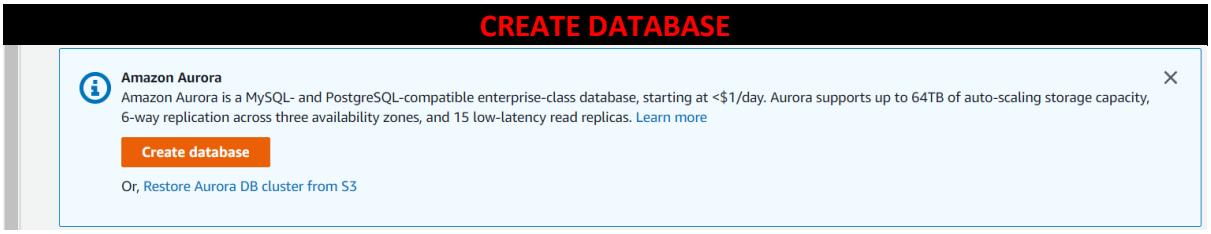
Create and cache create successfully

```
Created symlink /etc/systemd/system/multi-user.target.wants/redis.service → /usr/lib/systemd/system/redis.service.
[root@ip-172-31-35-217 ec2-user]# redis-cli
127.0.0.1:6379> set name vivek
OK
127.0.0.1:6379> get name
"vivek"
127.0.0.1:6379>
```

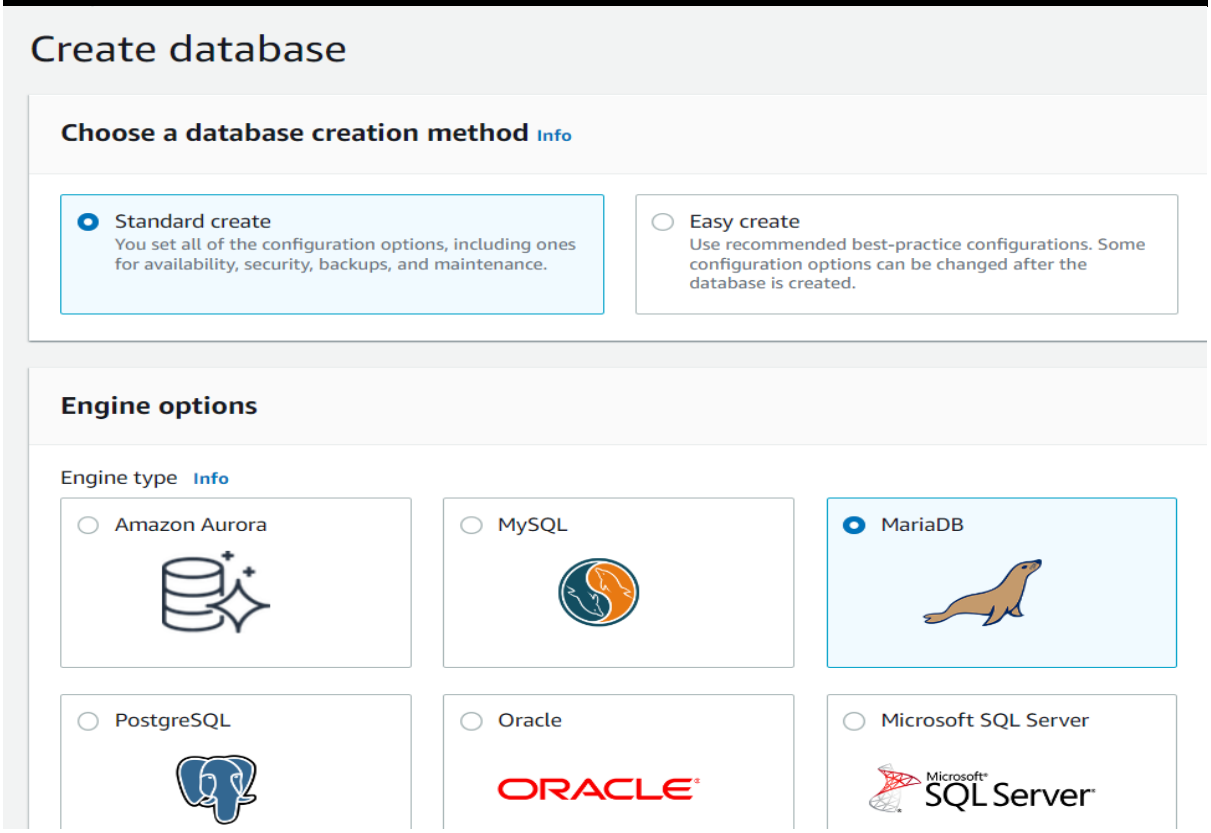
PROJECT-1

5. CREATE RDS

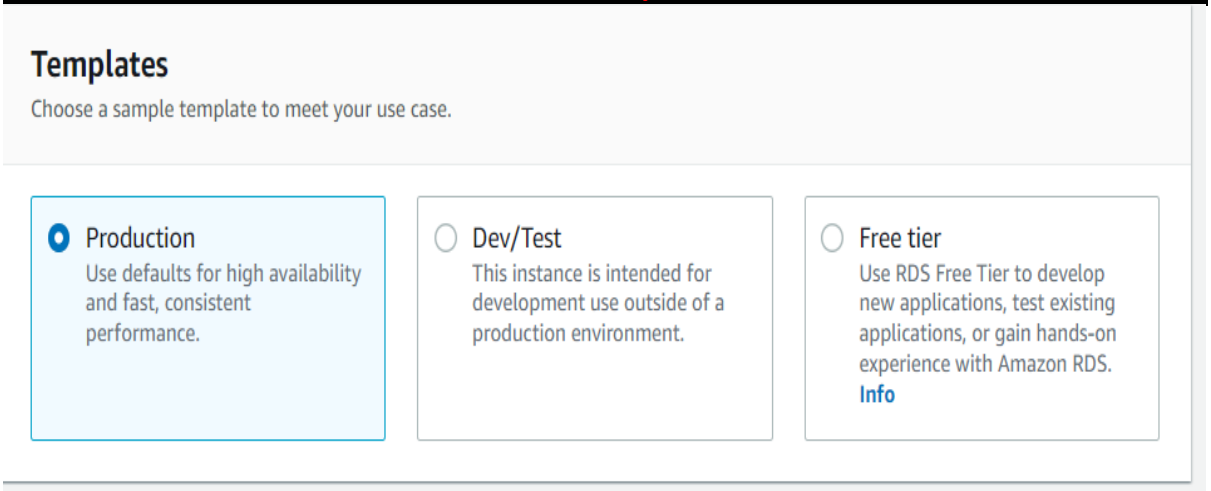
CREATE DATABASE



SELECT DB



SELECT templates



PROJECT-1

Db user name: - vivekdb pw: admin123

DB instance identifier Info
Type a name for your DB instance. The name must be unique across all DB instances owned by your AWS account in the current AWS Region.

vivekdb

The DB instance identifier is case-insensitive, but is stored as all lowercase (as in "mydbinstance"). Constraints: 1 to 60 alphanumeric characters or hyphens. First character must be a letter. Can't contain two consecutive hyphens. Can't end with a hyphen.

▼ Credentials Settings

Master username Info
Type a login ID for the master user of your DB instance.

vivekdb

1 to 16 alphanumeric characters. First character must be a letter.

☐ Auto generate a password
Amazon RDS can generate a password for you, or you can specify your own password.

Master password Info

••••••••

Constraints: At least 8 printable ASCII characters. Can't contain any of the following: / (slash), '(single quote), "(double quote) and @ (at sign).

Confirm password Info

••••••••

Select instance

Instance configuration

The DB instance configuration options below are limited to those supported by the engine that you selected above.

DB instance class Info

○ Standard classes (includes m classes)
○ Memory optimized classes (includes r and x classes)
◉ Burstable classes (includes t classes)

db.t2.micro
1 vCPUs 1 GiB RAM Not EBS Optimized

◉ Include previous generation classes

Storage

Storage

Storage type Info

General Purpose SSD (gp2)
Baseline performance determined by volume size

Allocated storage

20 GiB

(Minimum: 20 GiB. Maximum: 16,384 GiB) Higher allocated storage can improve IOPS performance.

ⓘ Provisioning less than 100 GiB of General Purpose (SSD) storage for high throughput workloads could result in higher latencies upon exhaustion of the initial General Purpose (SSD) IO credit balance. Learn more

Storage autoscaling Info

Provides dynamic scaling support for your database's storage based on your application's needs.

☐ Enable storage autoscaling
Enabling this feature will allow the storage to increase after the specified threshold is exceeded.

PROJECT-1

Select multi-AZ

Availability & durability

Multi-AZ deployment Info

- (●) Create a standby instance (recommended for production usage)
  Creates a standby in a different Availability Zone (AZ) to provide data redundancy, eliminate I/O freezes, and minimize latency spikes during system backups.
- ( ) Do not create a standby instance

SELECT SUBETS

Connectivity

Virtual private cloud (VPC) Info
VPC that defines the virtual networking environment for this DB instance.

Default VPC (vpc-0f86a7b4f5b69fdf7)

Only VPCs with a corresponding DB subnet group are listed.

ⓘ After a database is created, you can't change its VPC.

Subnet group Info
DB subnet group that defines which subnets and IP ranges the DB instance can use in the VPC you selected.

default-vpc-0f86a7b4f5b69fdf7

PUBLIC ACCESS NO AND SELECT SG

Public access Info

- ( ) Yes
  Amazon EC2 instances and devices outside the VPC can connect to your database. Choose one or more VPC security groups that specify which EC2 instances and devices inside the VPC can connect to the database.
- (●) No
  RDS will not assign a public IP address to the database. Only Amazon EC2 instances and devices inside the VPC can connect to your database.

VPC security group
Choose a VPC security group to allow access to your database. Ensure that the security group rules allow the appropriate incoming traffic.

- (●) Choose existing
  Choose existing VPC security groups
- ( ) Create new
  Create new VPC security group

Existing VPC security groups

Choose VPC security groups

default ✕

▶ Additional configuration

CREATED BY:- VIVEK GANDHI

PROJECT-1

Select Database authentication

Database authentication

Database authentication options Info

- (•) Password authentication
  Authenticates using database passwords.
- ( ) Password and IAM database authentication
  Authenticates using the database password and user credentials through AWS IAM users and roles.

Configuration db

▼ Additional configuration

Database options, backup turned off, backtrack turned off, Enhanced Monitoring turned off, maintenance, CloudWatch Logs, delete protection turned off.

Database options

Initial database name Info

vivekdb

If you do not specify a database name, Amazon RDS does not create a database.

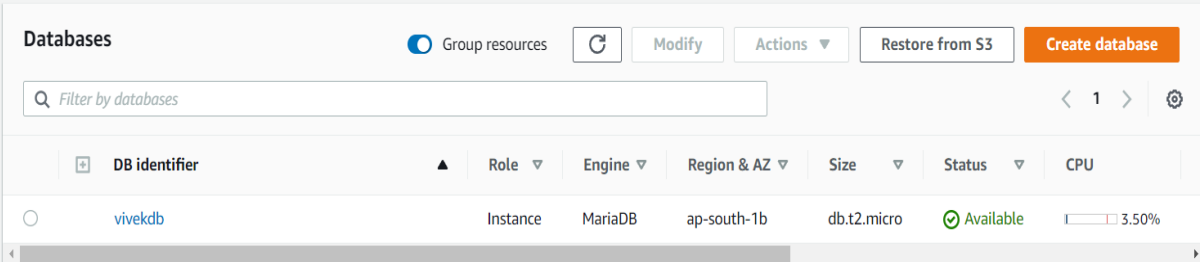
DB parameter group Info

default.mariadb10.6

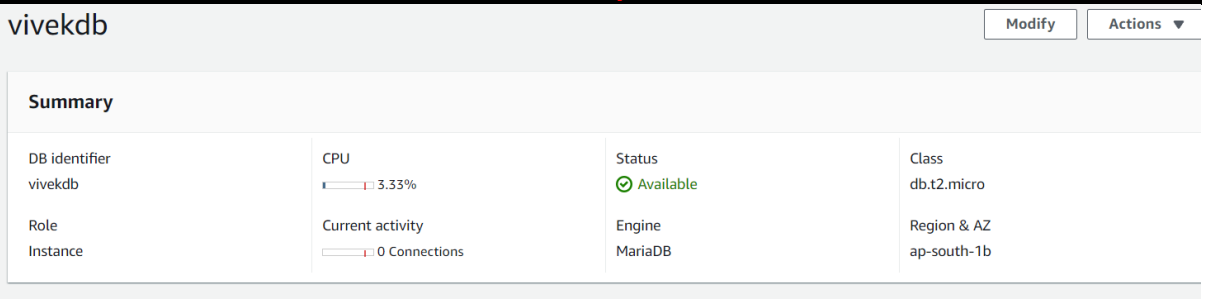
Option group Info

default:mariadb-10-6

Create successfully



Primary DB



PROJECT-1

Secondary DB

Amazon Resource Name (ARN)
arn:aws:rds:ap-south-1:659202326636:db:vivekdb

Resource ID
db-DS5XRC7OOPSUWD3A4F7TWJNEAY

Created time
July 13, 2022, 04:38 (UTC+4:38)

Not enabled

Multi-AZ
Yes

Secondary Zone
ap-south-1a

Install MariaDB

root@ip-172-31-35-217:/home/ec2-user

```
[root@ip-172-31-35-217 ec2-user]# yum install mariadb
```

Connected successfully

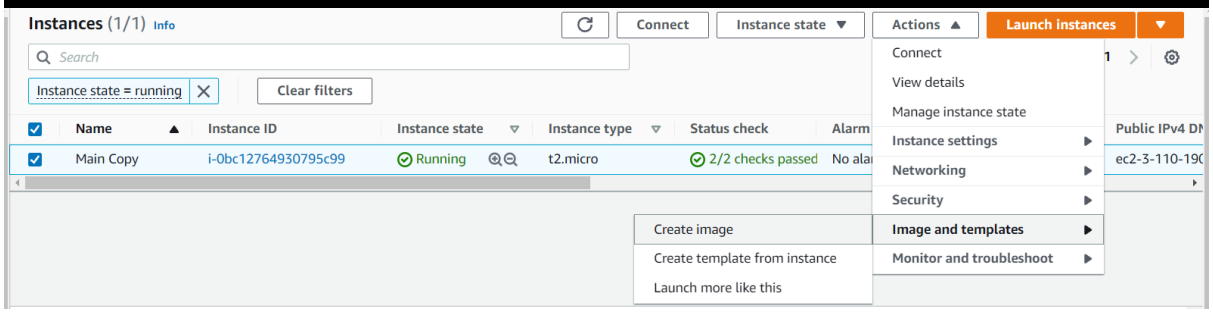
root@ip-172-31-35-217:/home/ec2-user

```
[root@ip-172-31-35-217 ec2-user]# mysql -h vivekdb.c003vi2z6ea3.ap-south-1.rds.amazonaws.com  -u vivekdb -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 41
Server version: 10.6.7-MariaDB managed by https://aws.amazon.com/rds/

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]> show databases;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| innodb             |
| mysql              |
| performance_schema |
| sys                |
| vivekdb            |
+--------------------+
6 rows in set (0.001 sec)

MariaDB [(none)]>
```

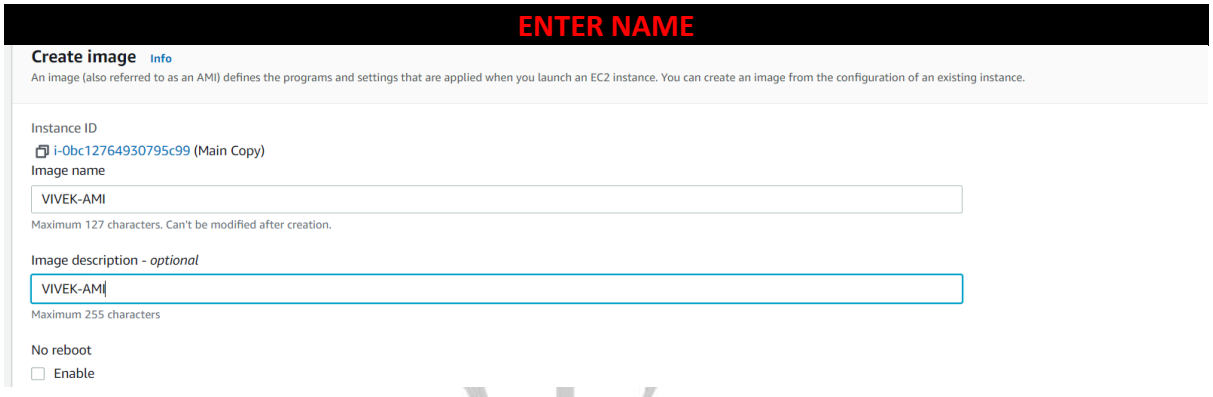
PROJECT-1

6. Create AMI

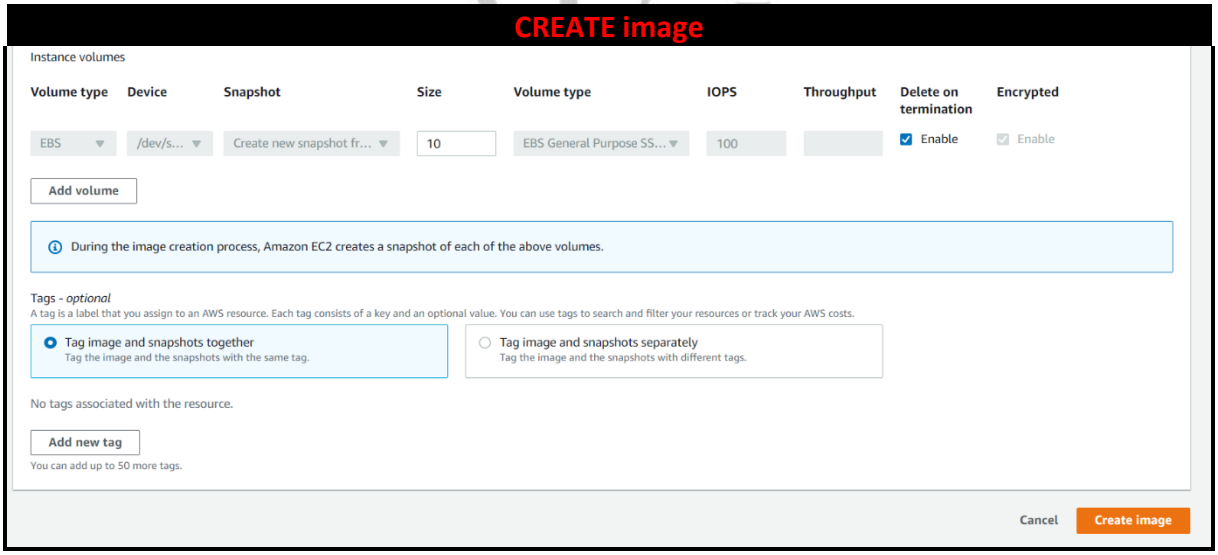
CREATE IMAGE OPTION



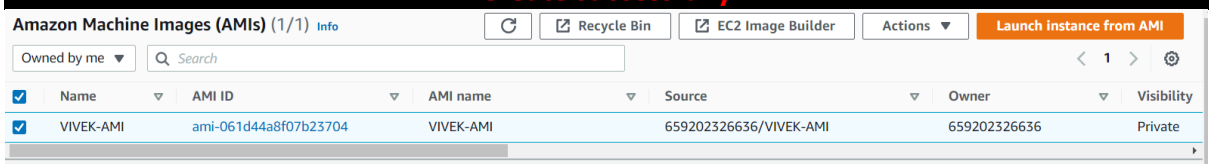
ENTER NAME



CREATE image



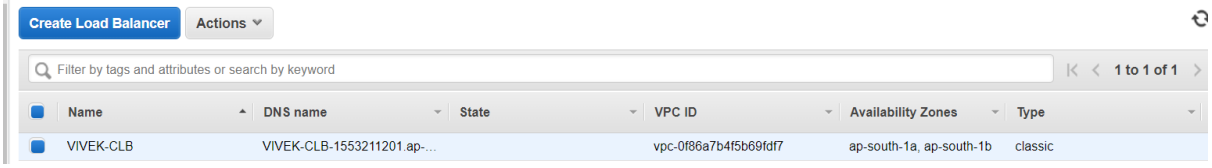
Create successfully



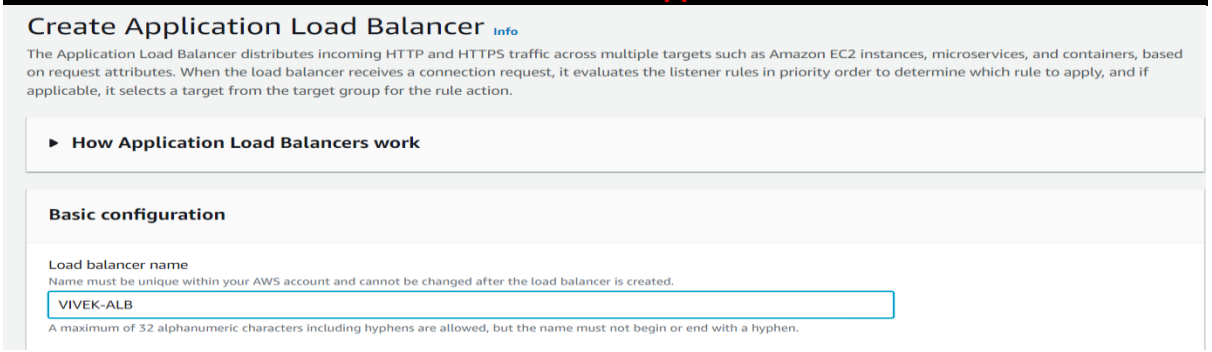
PROJECT-1

7. Create Application LB for Global Accelerator

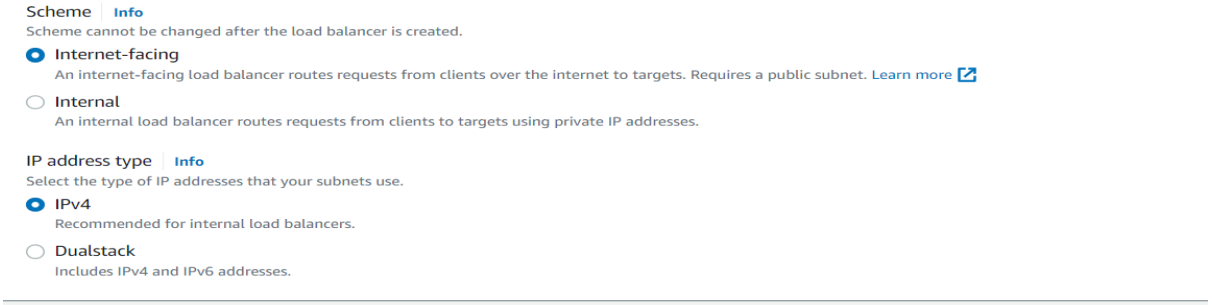
Open LB and select create



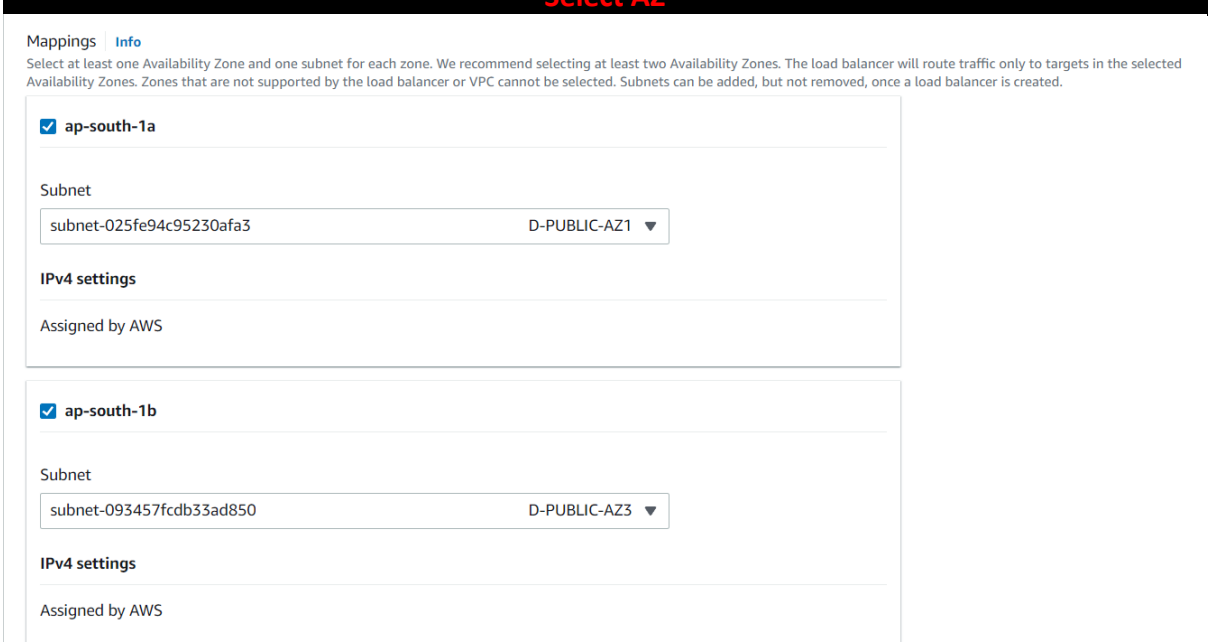
ENTER NAME select Application LB



SELECT INTERNET FACING



Select AZ



PROJECT-1

SELECT SG

Security groups Info

A security group is a set of firewall rules that control the traffic to your load balancer.

Security groups

Select up to 5 security groups

Create new security group

default sg-08b790c8746f05bb3

VPC: vpc-0f86a7b4f5b69fdf7

Create target group

Listeners and routing Info

A listener is a process that checks for connection requests, using the protocol and port you configure. Traffic received by the listener is then routed per your specification. You can specify multiple rules and multiple certificates per listener after the load balancer is created.

Listener HTTP:80 Remove

Protocol	Port	Default action Info
HTTP	80 (1-65535)	Forward to *Select a target group*

Create target group

Select instance

Specify group details

Your load balancer routes requests to the targets in a target group and performs health checks on the targets.

Basic configuration

Settings in this section cannot be changed after the target group is created.

Choose a target type

- Instances
  - Supports load balancing to instances within a specific VPC.
  - Facilitates the use of Amazon EC2 Auto Scaling to manage and scale your EC2 capacity.

Enter name and select configure

Target group name

vivek-alb

A maximum of 32 alphanumeric characters including hyphens are allowed, but the name must not begin or end with a hyphen.

Protocol: HTTP : Port: 80

VPC

Select the VPC with the instances that you want to include in the target group.

- vpc-0f86a7b4f5b69fdf7
IPv4: 172.31.0.0/16

Protocol version

- HTTP1
  Send requests to targets using HTTP/1.1. Supported when the request protocol is HTTP/1.1 or HTTP/2.
- HTTP2
  Send requests to targets using HTTP/2. Supported when the request protocol is HTTP/2 or gRPC, but gRPC-specific features are not available.
- gRPC
  Send requests to targets using gRPC. Supported when the request protocol is gRPC.

PROJECT-1

Enter path

Health checks

The associated load balancer periodically sends requests, per the settings below, to the registered targets to test their status.

Health check protocol

HTTP

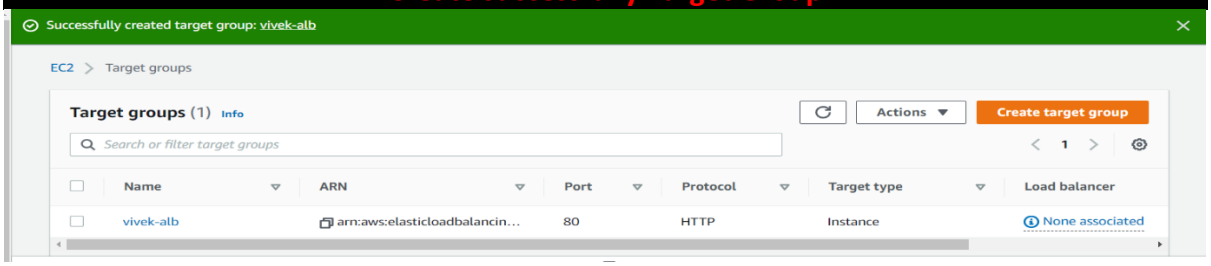
Health check path

Use the default path of "/" to ping the root, or specify a custom path if preferred.

/index.html

Up to 1024 characters allowed.

Create successfully Target Group



Select Target group

Listeners and routing Info

A listener is a process that checks for connection requests, using the protocol and port you configure. Traffic received by the listener is then routed per your specification. You can specify multiple rules and multiple certificates per listener after the load balancer is created.

Listener HTTP:80

Remove

Protocol: HTTP

Port: 80

1-65535

Default action Info

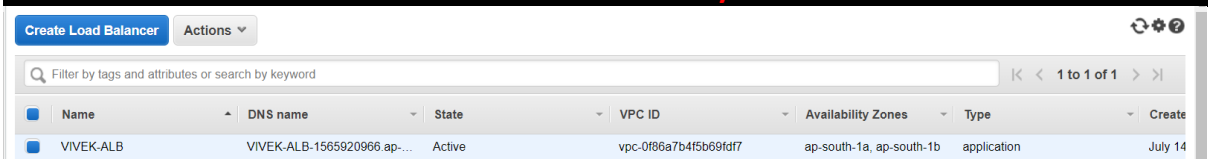
Forward to: vivek-alb

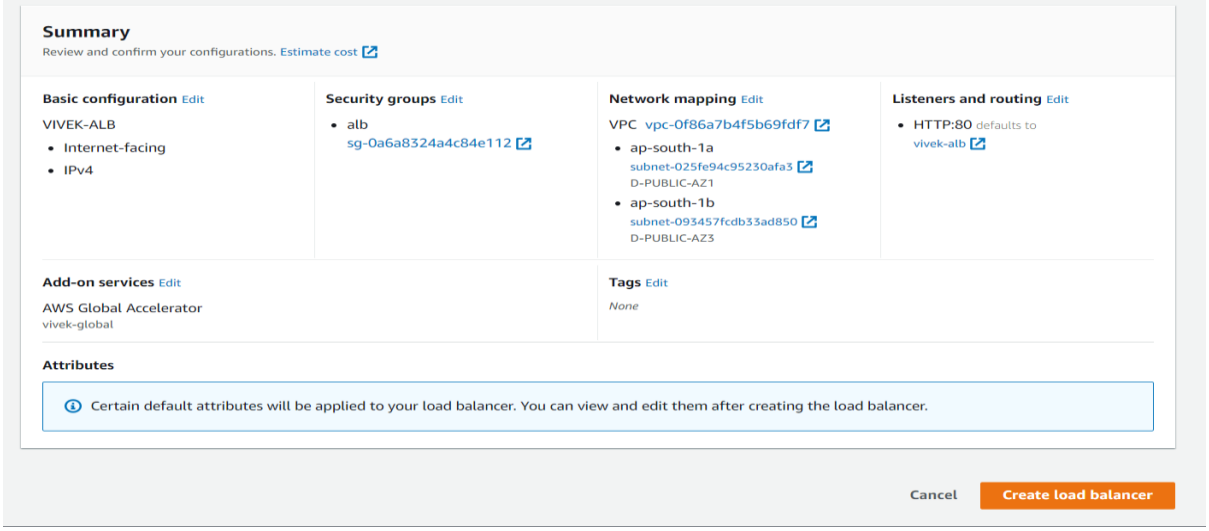
Target type: Instance, IPv4

HTTP

Create target group

Create successfully

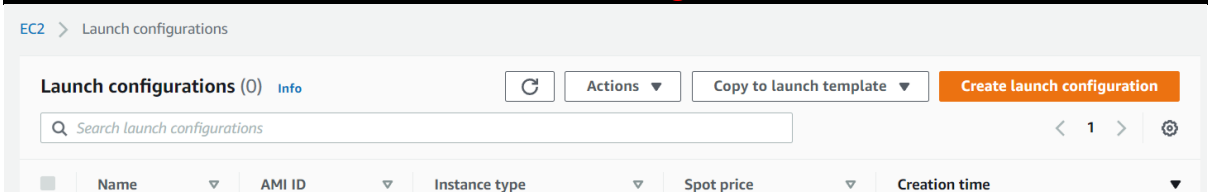




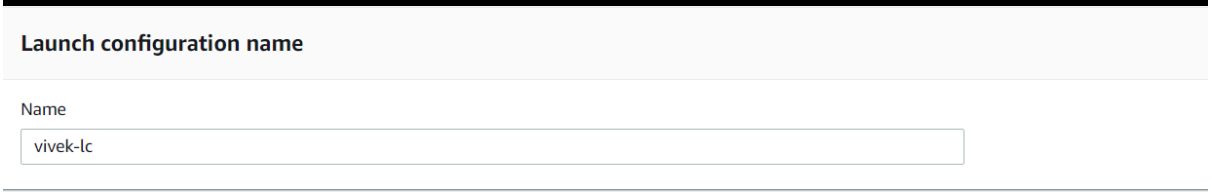
PROJECT-1

8. Create Auto scaling

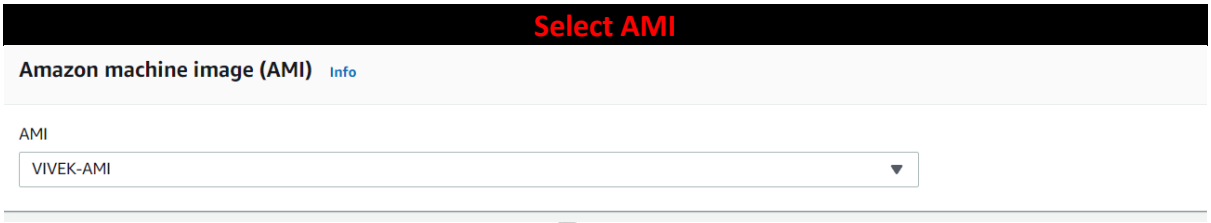
OPEN launch configuration



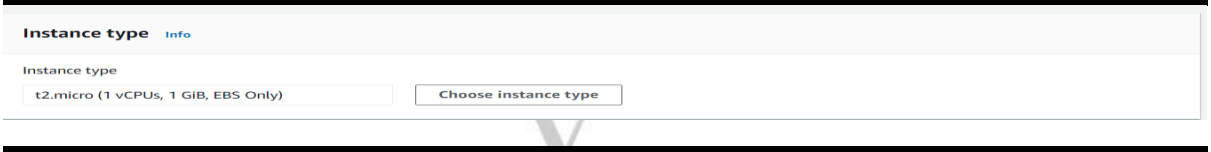
Enter name



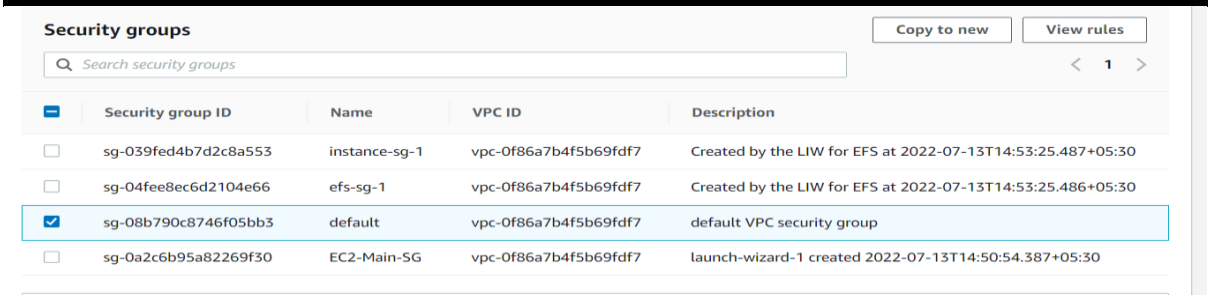
Select AMI



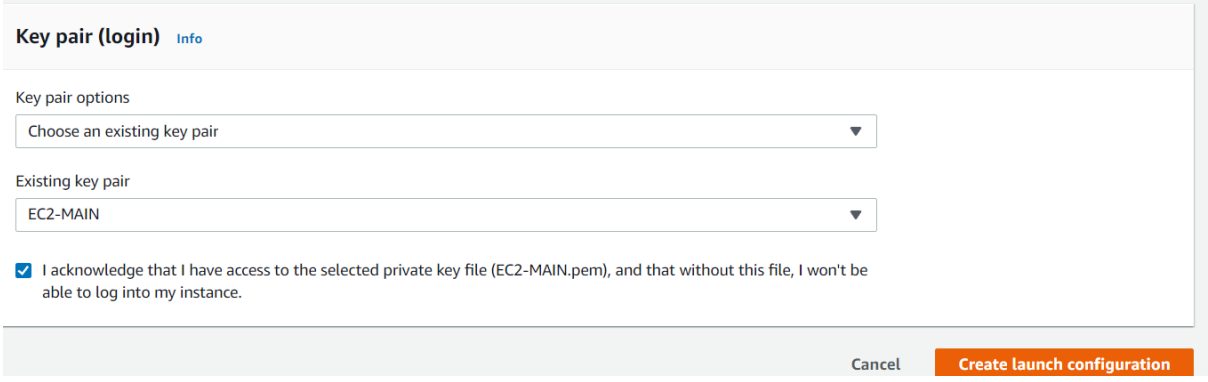
Select instance



Create or select SG

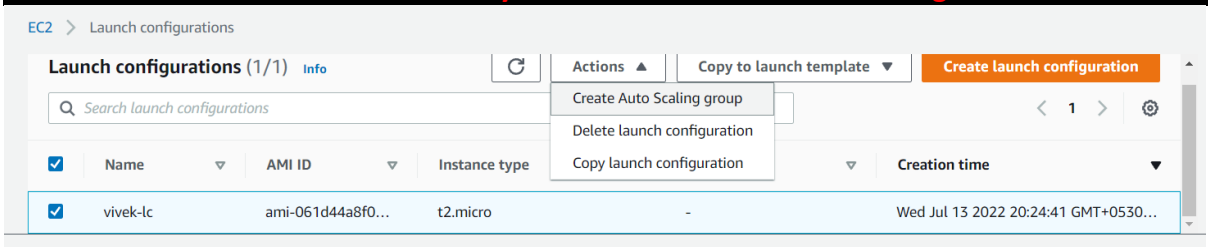


Key pair select

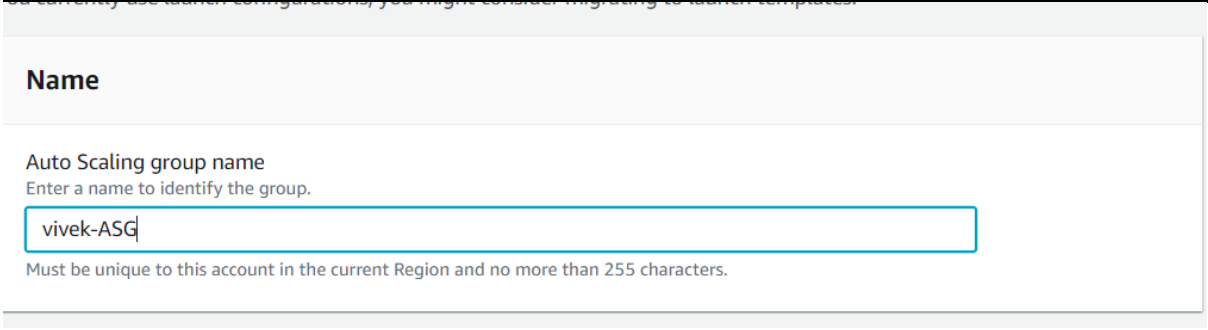


PROJECT-1

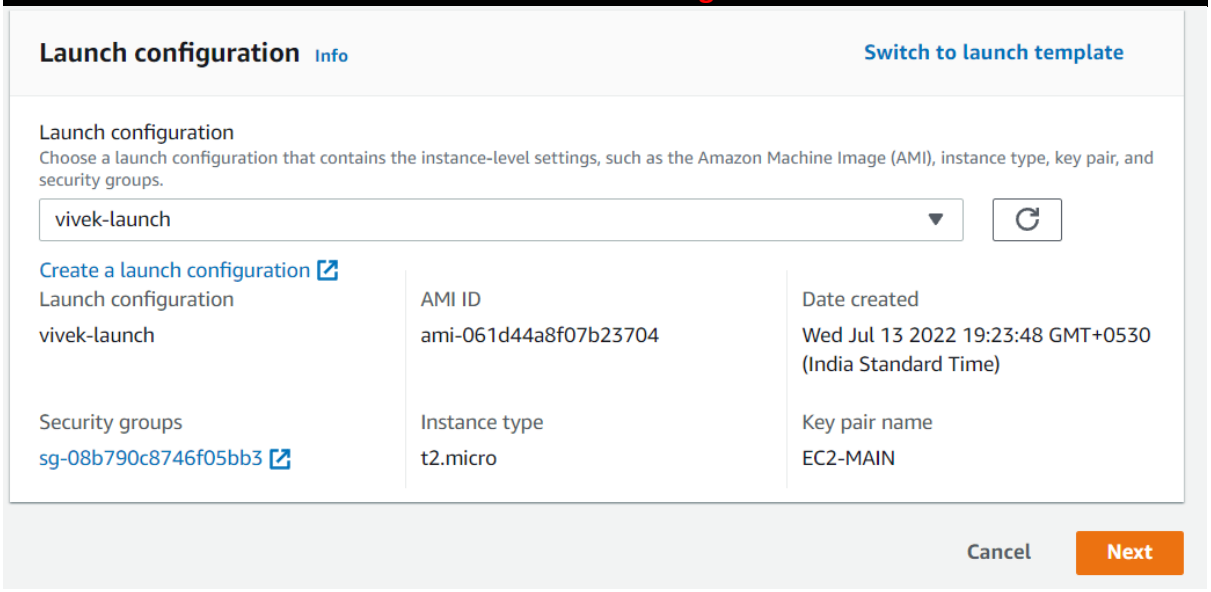
Create successfully and select to create auto scaling



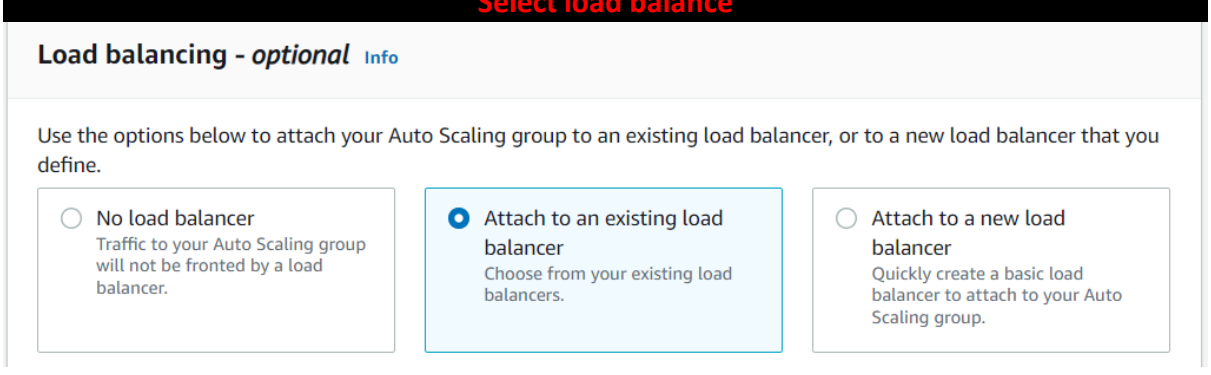
Enter name



Select launch configure

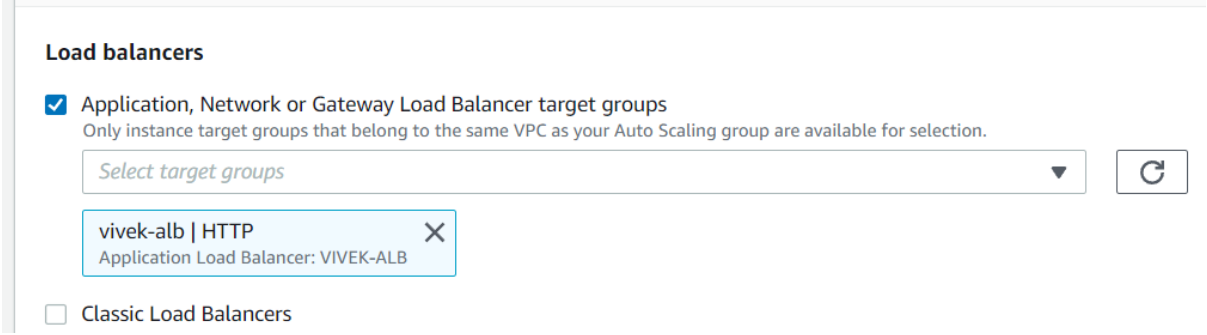


Select load balance

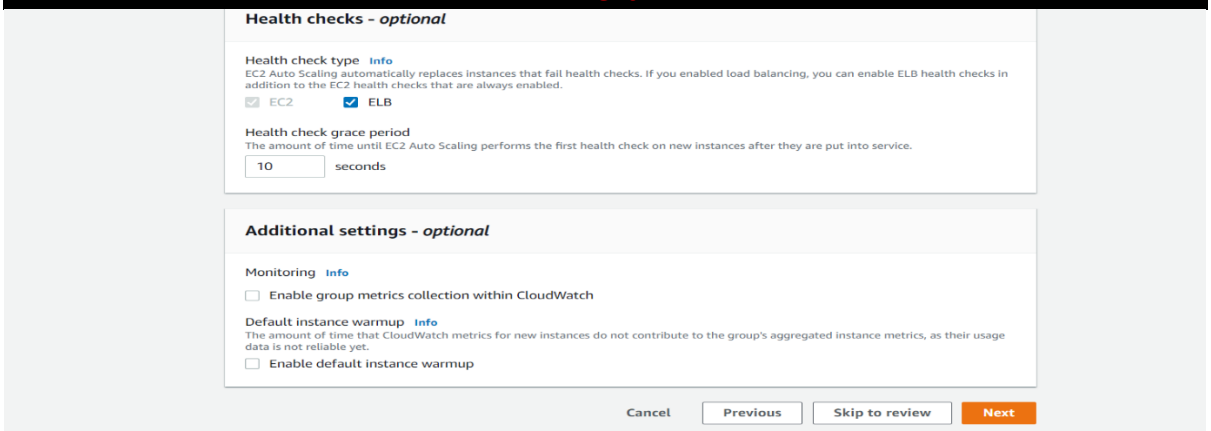


PROJECT-1

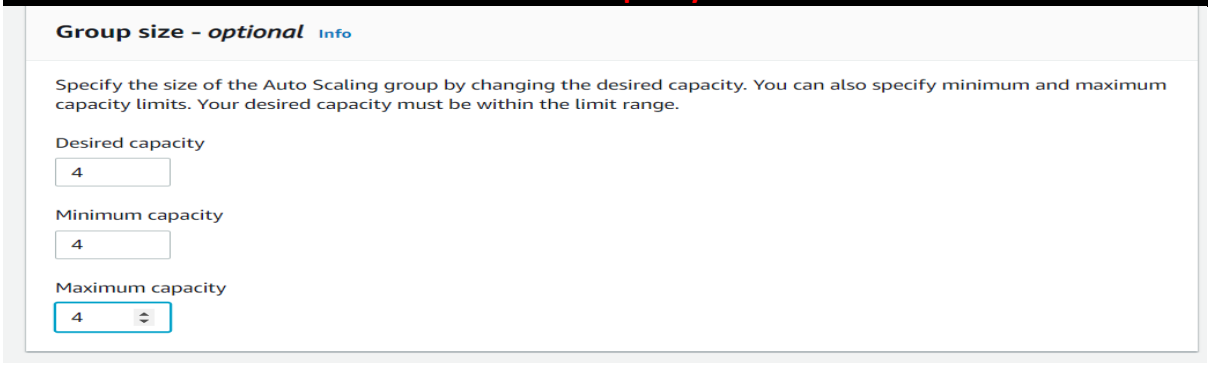
Select target application



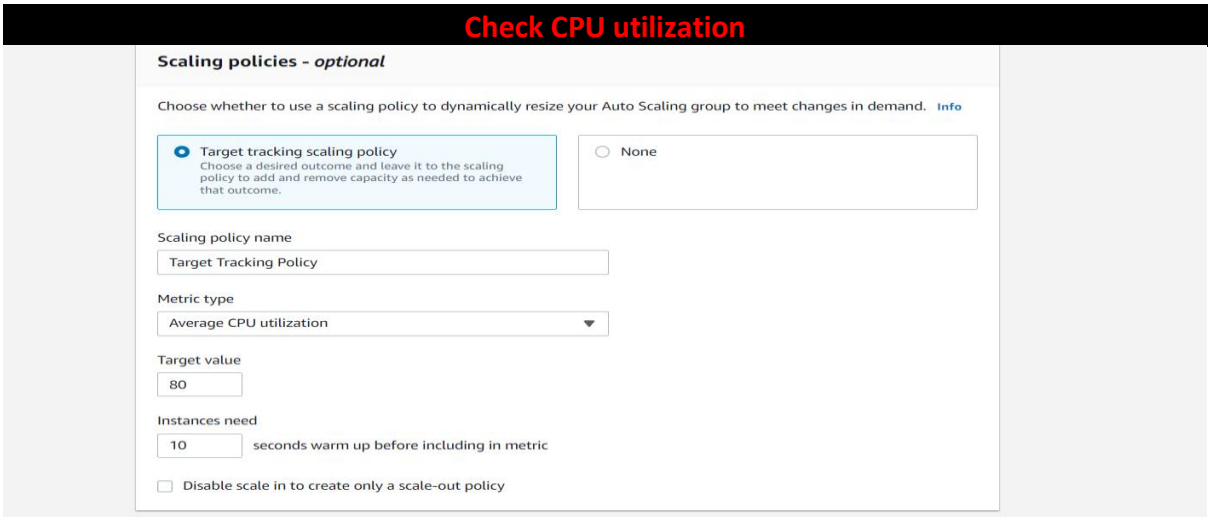
Next



Select capacity

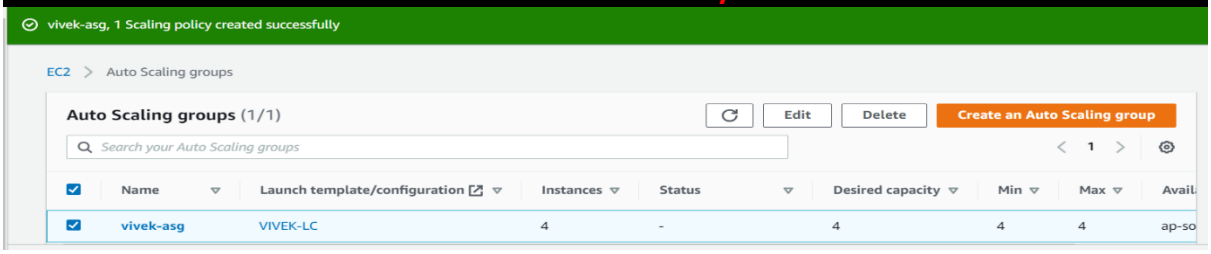


Check CPU utilization



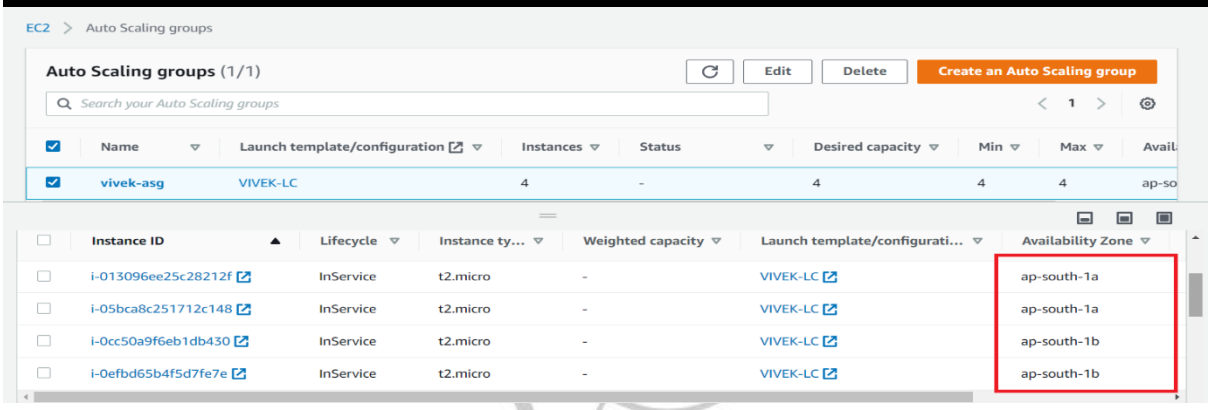
PROJECT-1

Create successfully

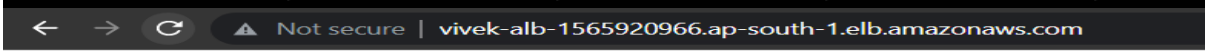


See activity

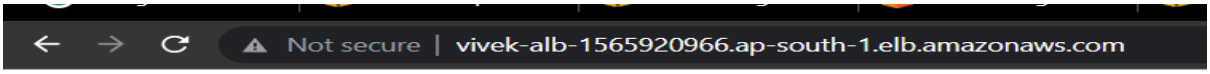
See instance in different AZ



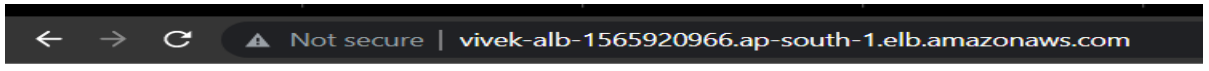
Also working LB



Automatic create Instances EC1-AZa-ASG for testing

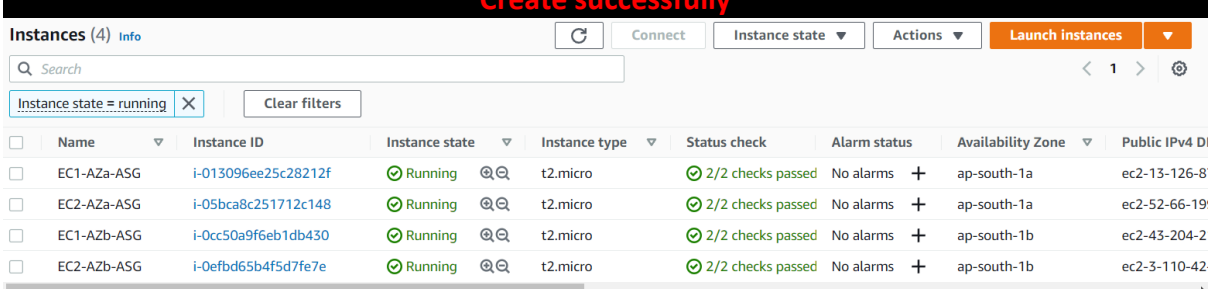


Automatic create Instance EC2-AZb-ASG for testing



Automatic create Instance EC2-AZa-ASG for testing

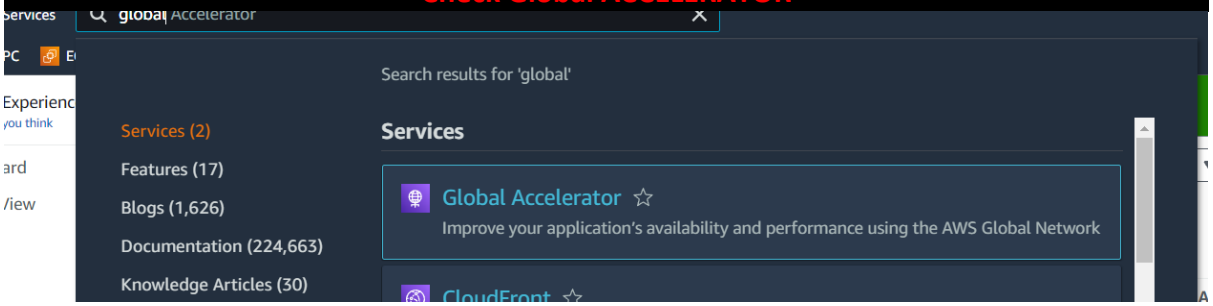
Create successfully



PROJECT-1

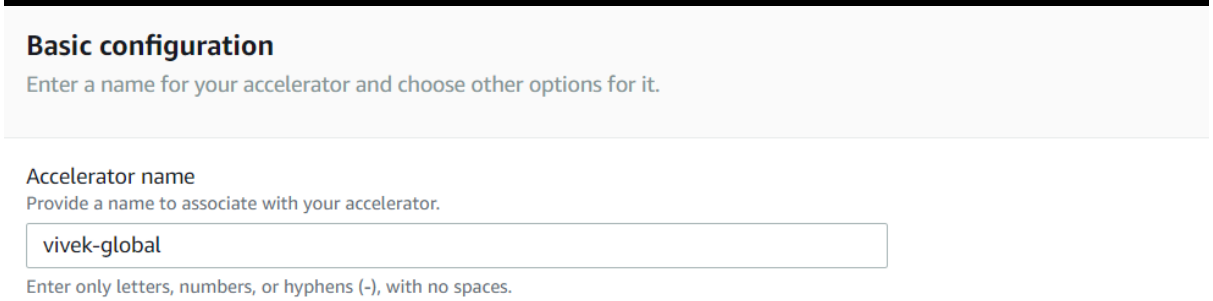
9. Create Global Accelerator

Check Global ACCELERATOR

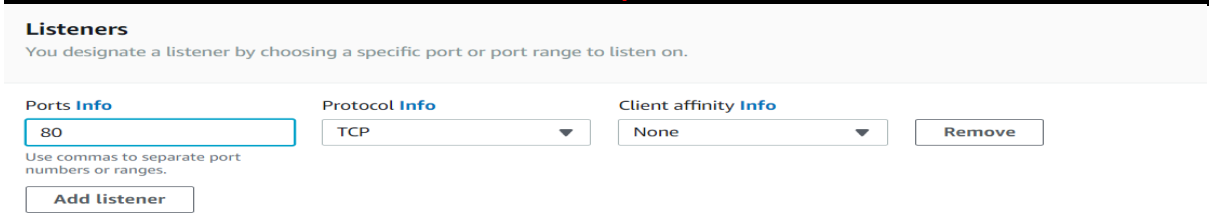


Click to Accelerator

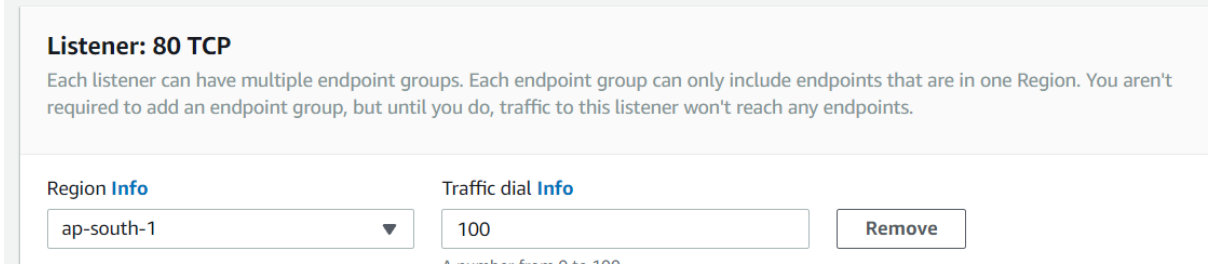
Enter name



Select port

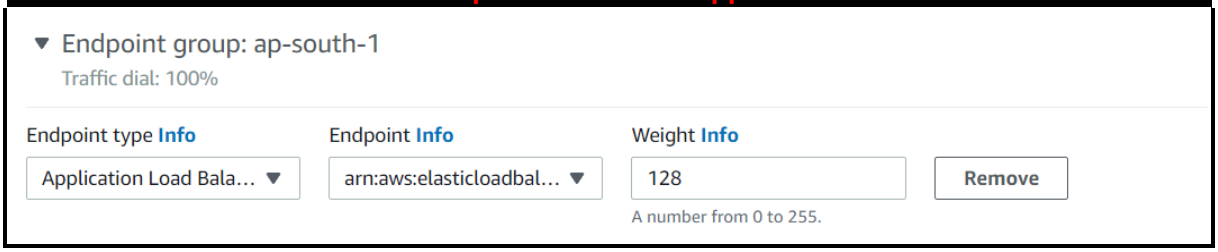


Select zone

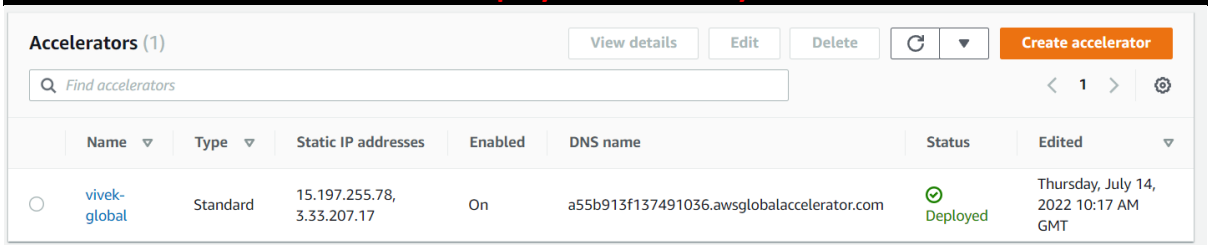


PROJECT-1

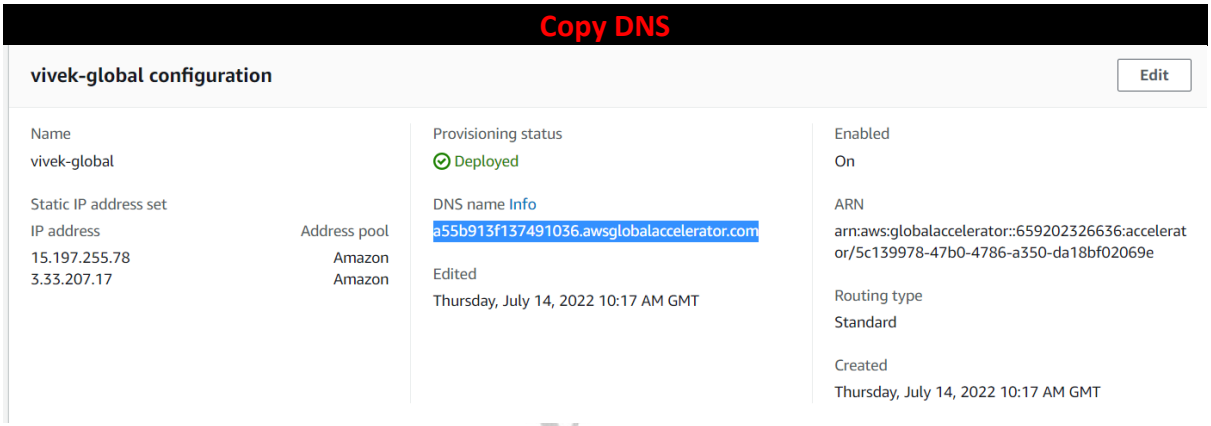
Select Endpoint and select Application LB



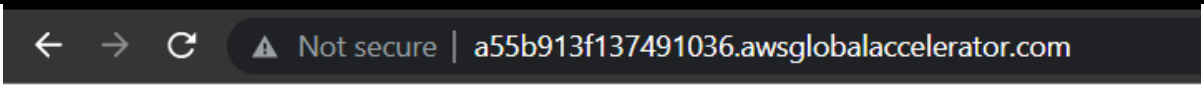
Deployed successfully



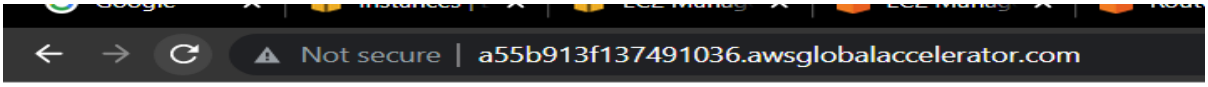
Copy DNS



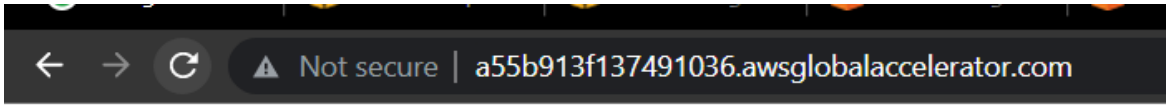
Browse and check



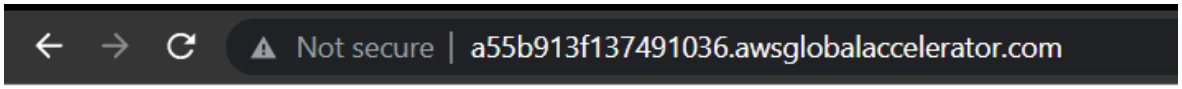
Automatic create Instance EC2-AZb-ASG for testing



Automatic create Instance EC2-AZa-ASG for testing



Automatic create Instance EC1-AZb-ASG for testing

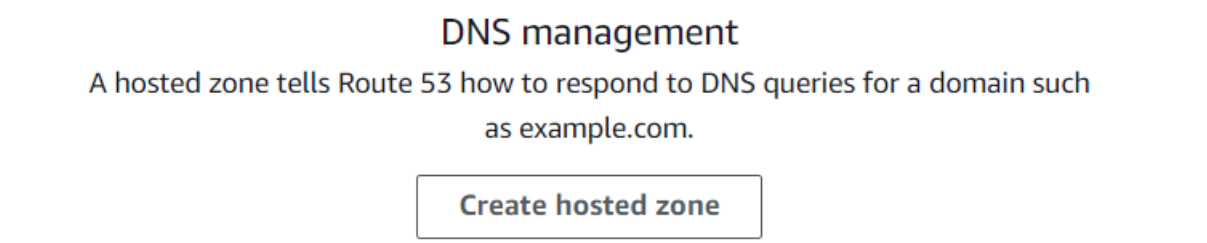


Automatic create Instances EC1-AZa-ASG for testing

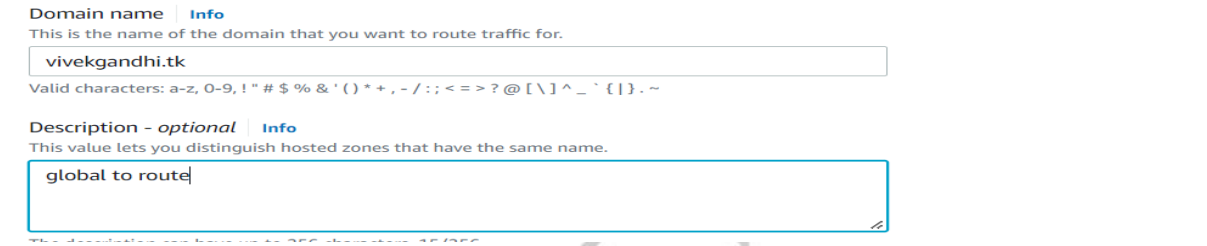
PROJECT-1

10. Create ROUTE53

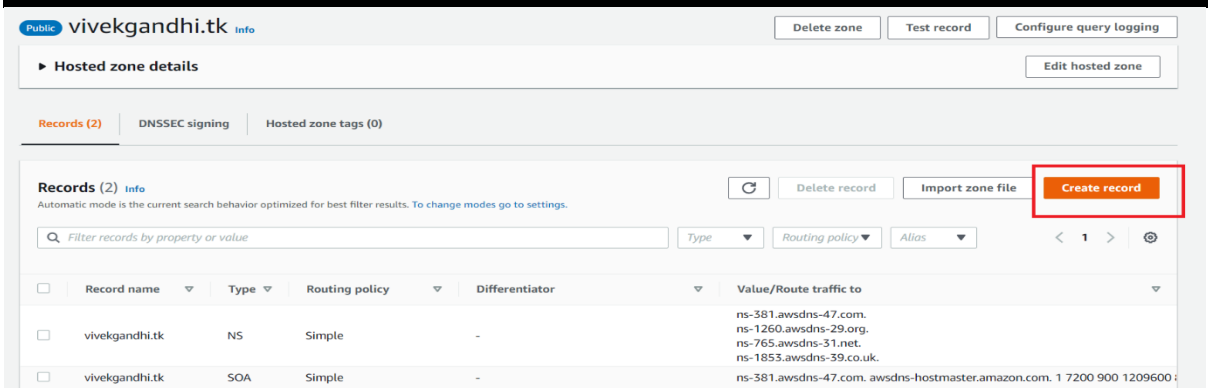
Open Route53



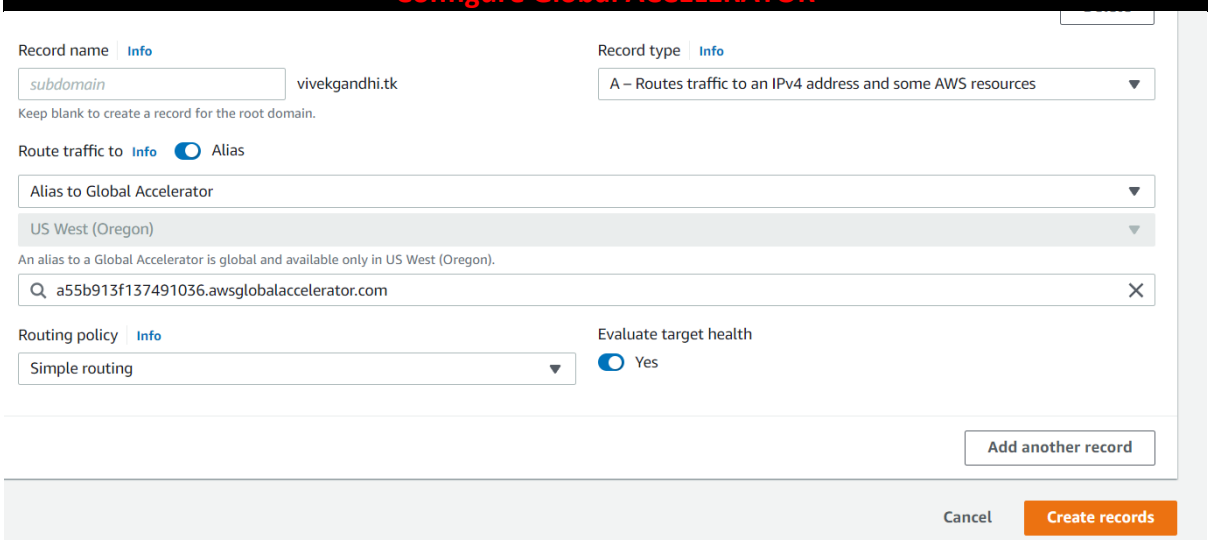
Enter domain name



Create record select



Configure Global ACCELERATOR



PROJECT-1

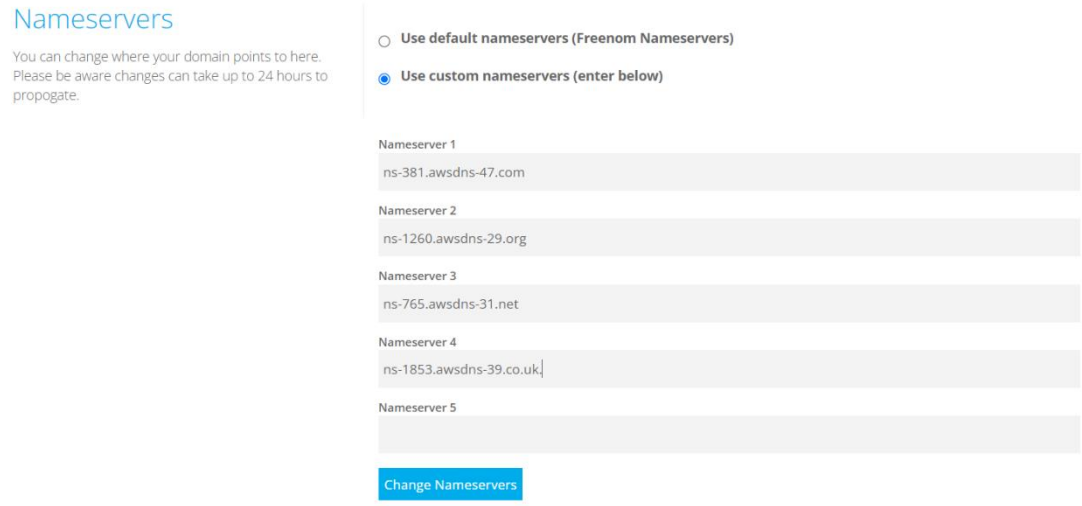
Create successfully

Record name	Type	Routing policy	Differentiator	Value/Route traffic to
vivekgandhi.tk	A	Simple	-	a55b913f137491036.awsglobalaccelerator.com.
				ns-381.awsdns-47.com.

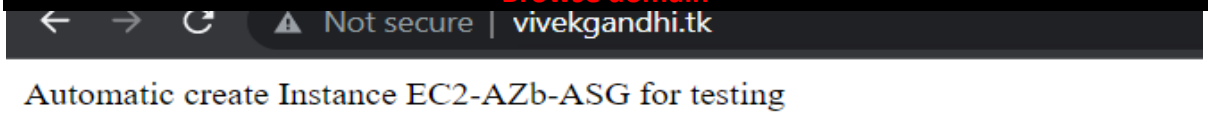
By default, create nameserver

vivekgandhi.tk	NS	Simple	-	ns-381.awsdns-47.com. ns-1260.awsdns-29.org. ns-765.awsdns-31.net. ns-1853.awsdns-39.co.uk.
vivekgandhi.tk	SOA	Simple	-	ns-381.awsdns-47.com. awsdns-hostmaster.amazon.com. 1 7200 900 12096

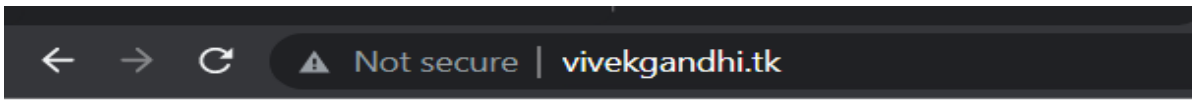
Add nameserver entry



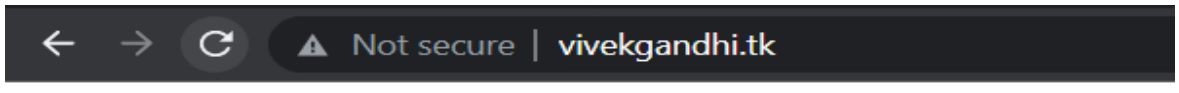
Browse domain



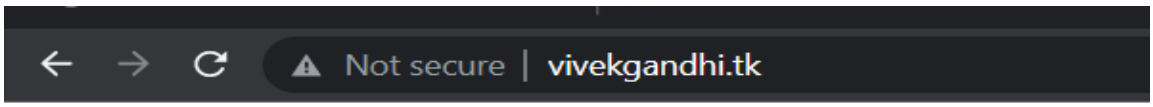
Automatic create Instance EC2-AZb-ASG for testing



Automatic create Instances EC1-AZa-ASG for testing



Automatic create Instance EC2-AZa-ASG for testing

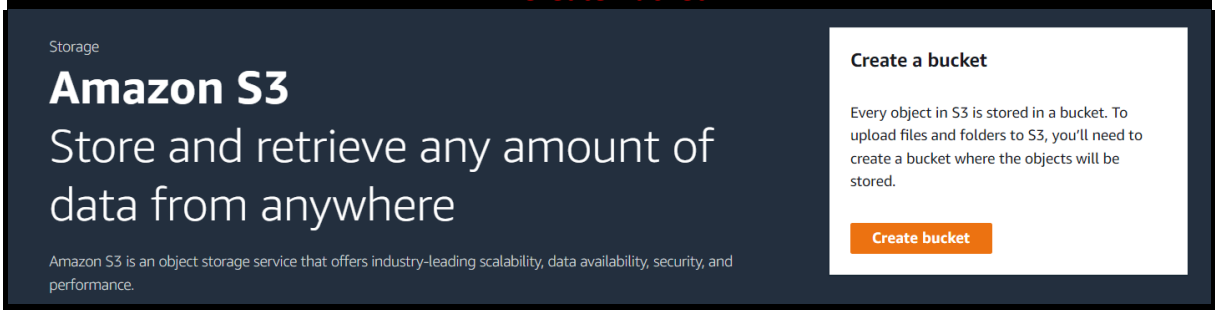


Automatic create Instance EC1-AZb-ASG for testing

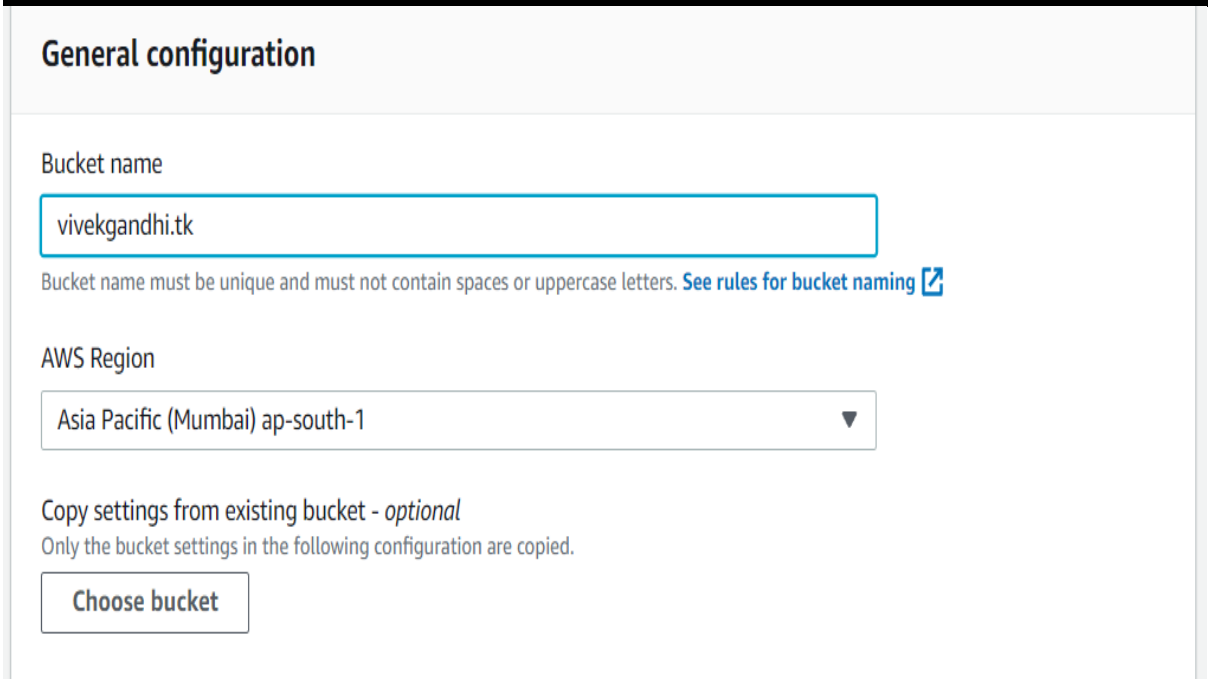
PROJECT-1

11. Open S3 in EC2 and put html file

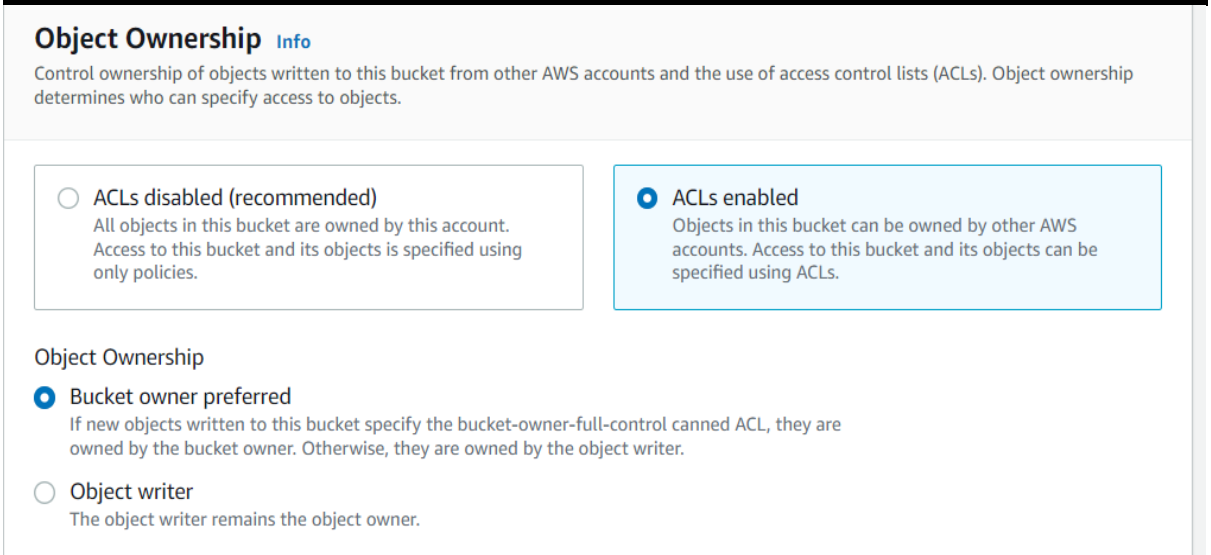
Create Bucket



Enter bucket name

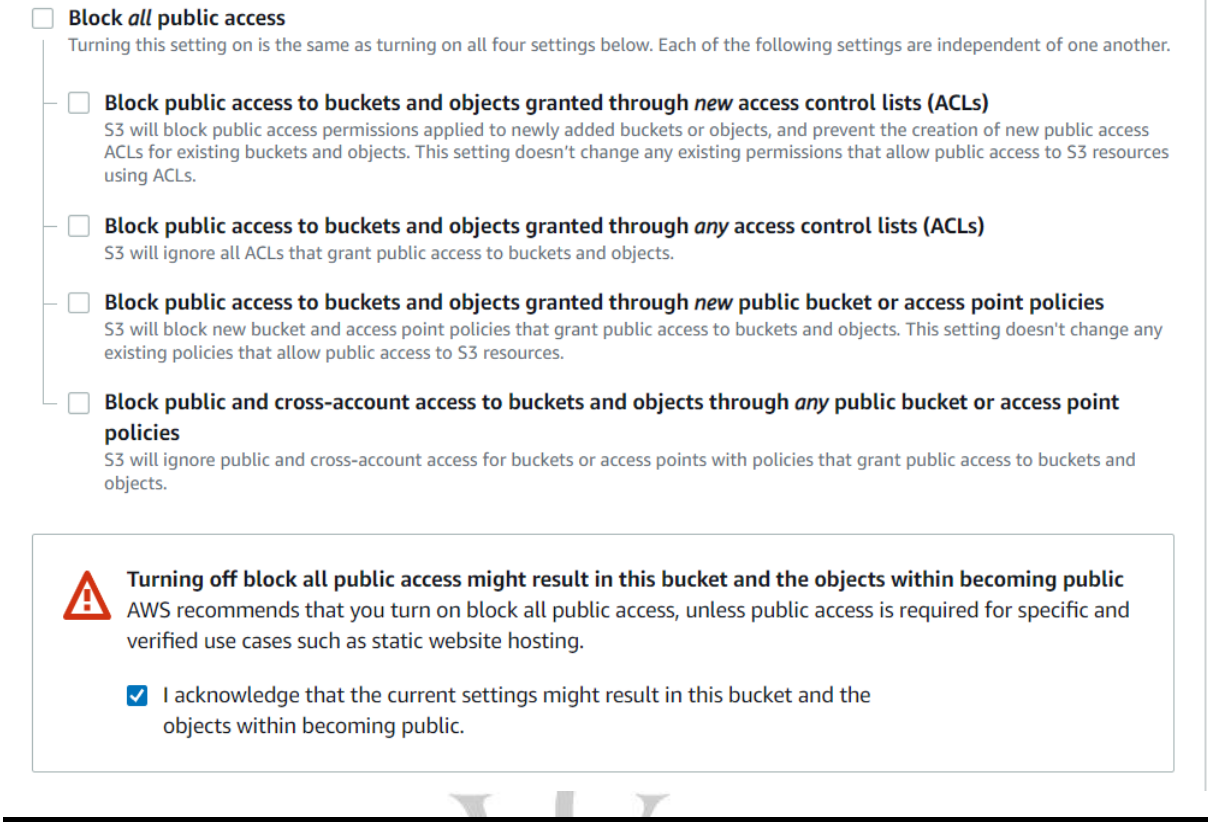


ACLS enbled

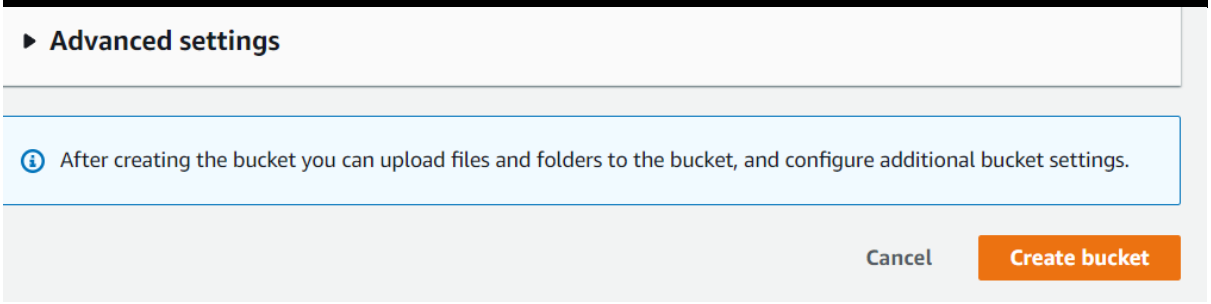


PROJECT-1

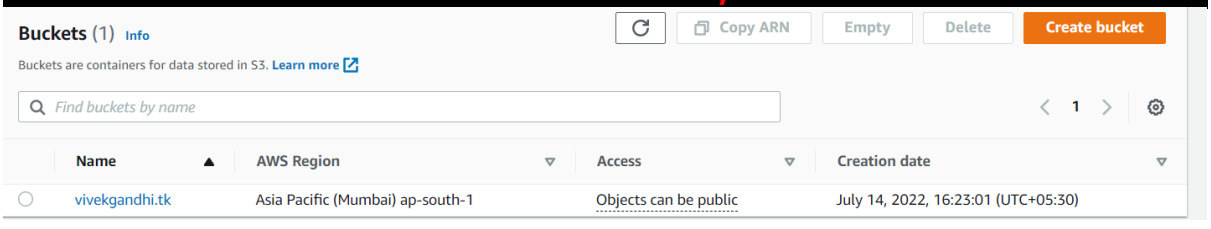
Access all



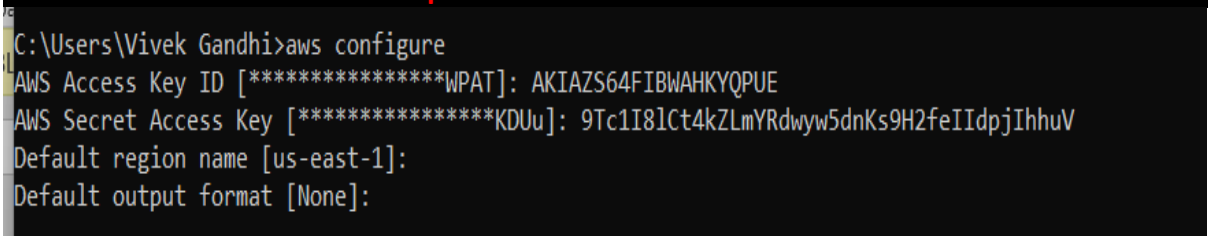
Create Bucket



Create successfully

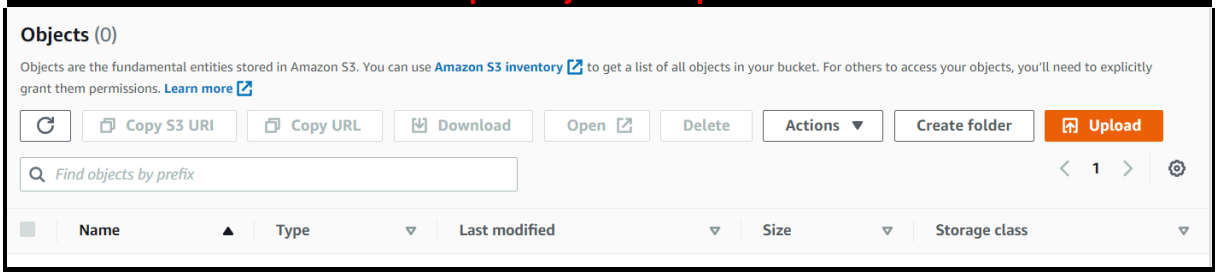


Open cmd and connected aws

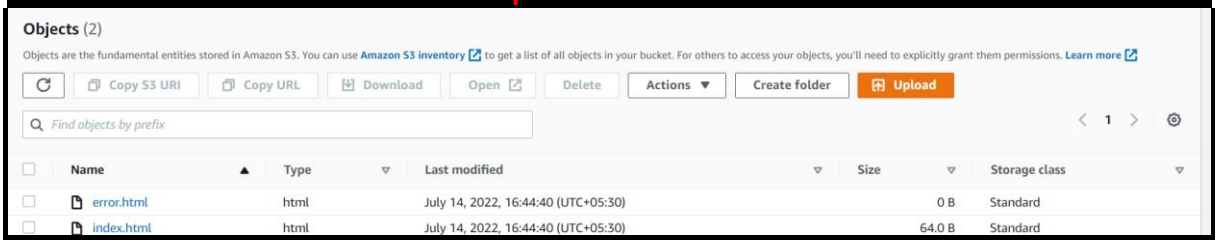


PROJECT-1

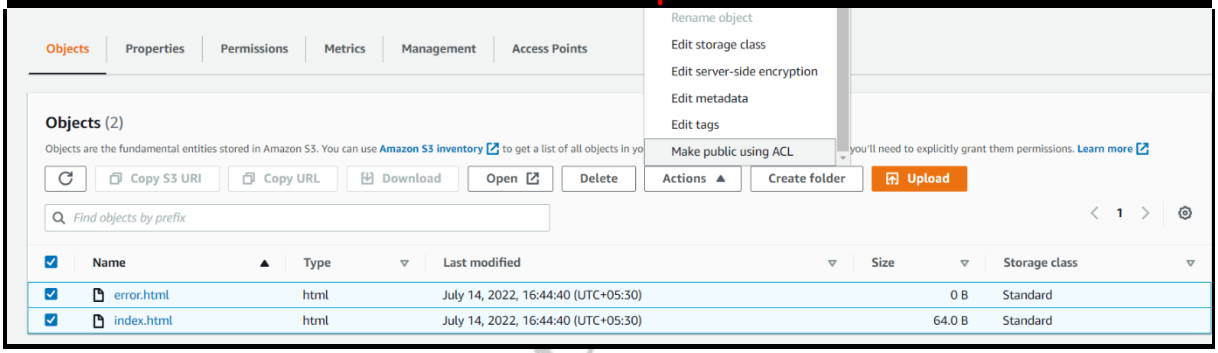
Open Object and upload



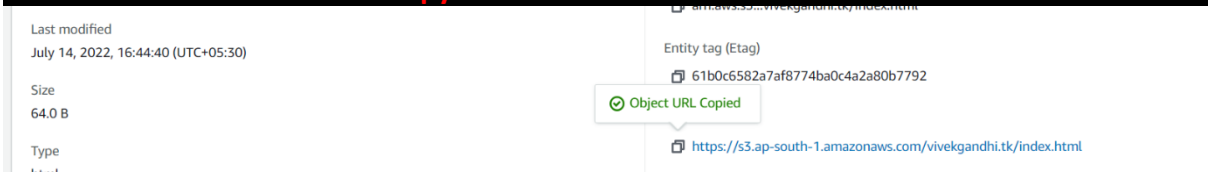
Upload successful



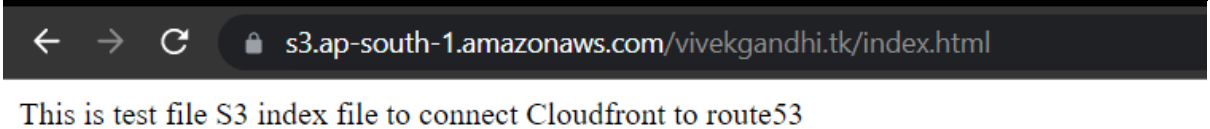
Create make public



Copy URL to check Browse or not

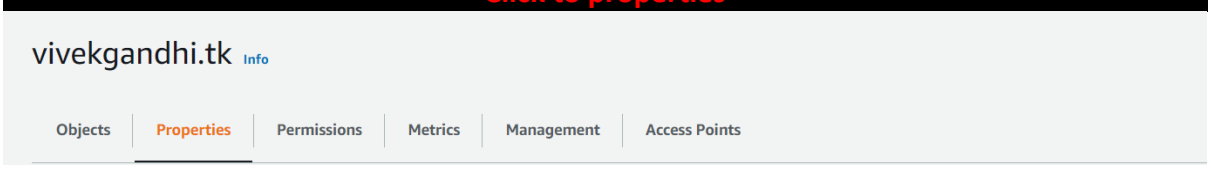


Open HTML file



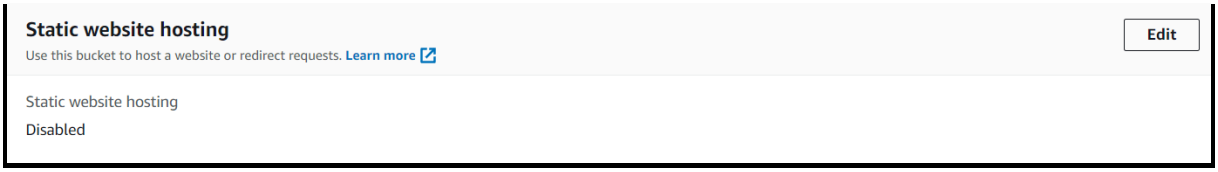
Create static website

Click to properties

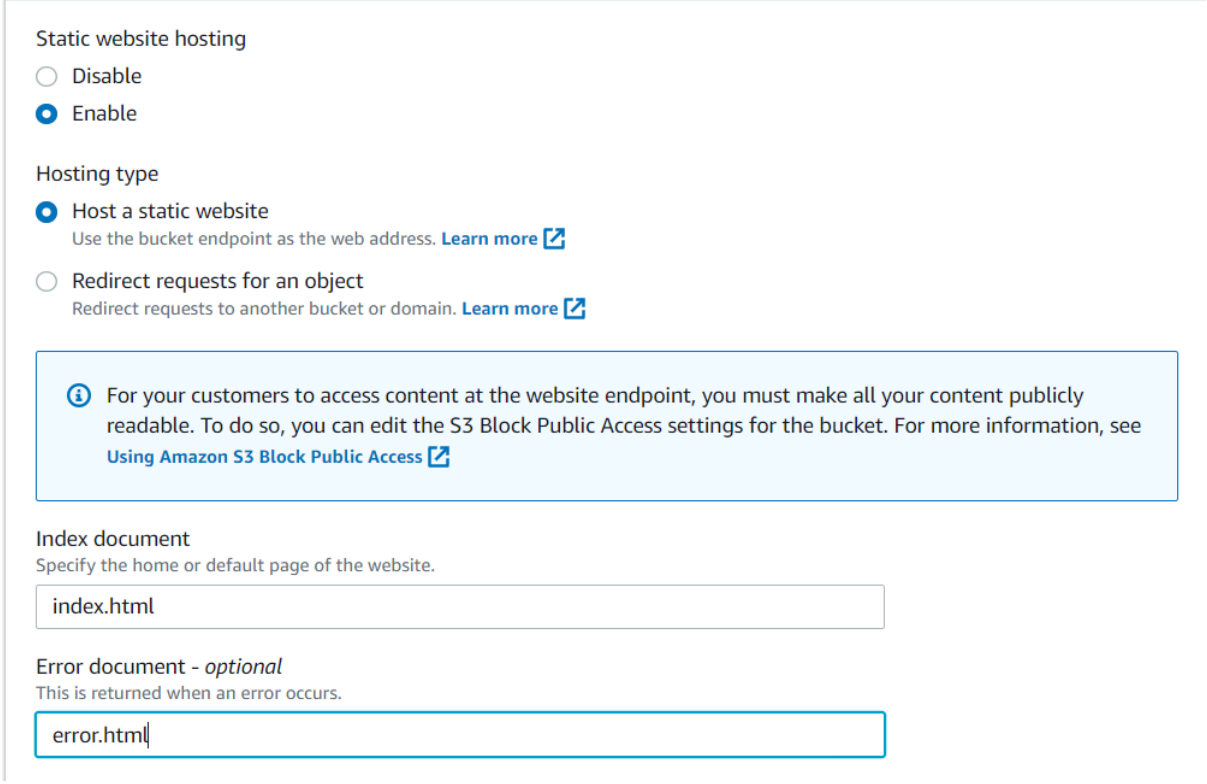


PROJECT-1

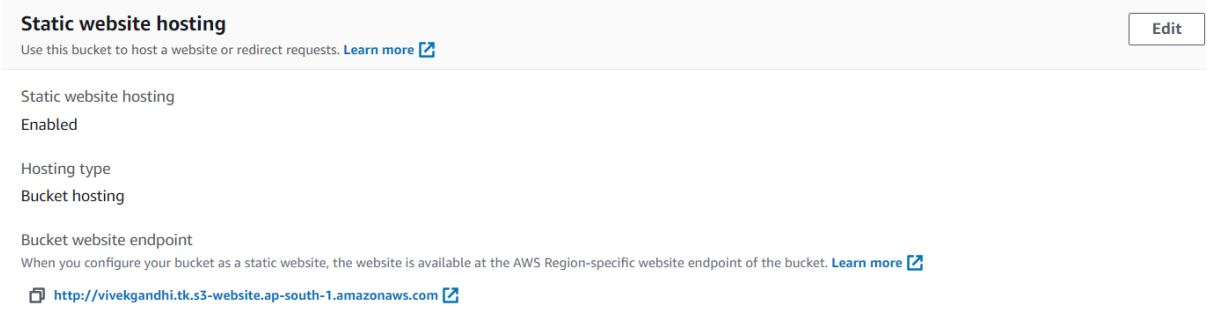
Edit static web



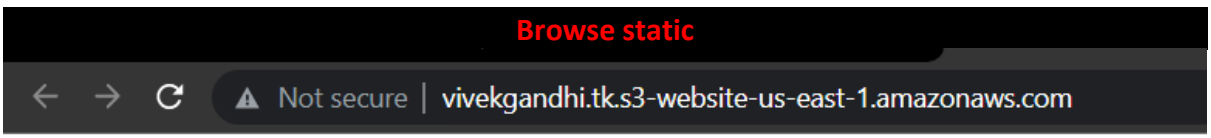
configure



Create successfully



Browse static

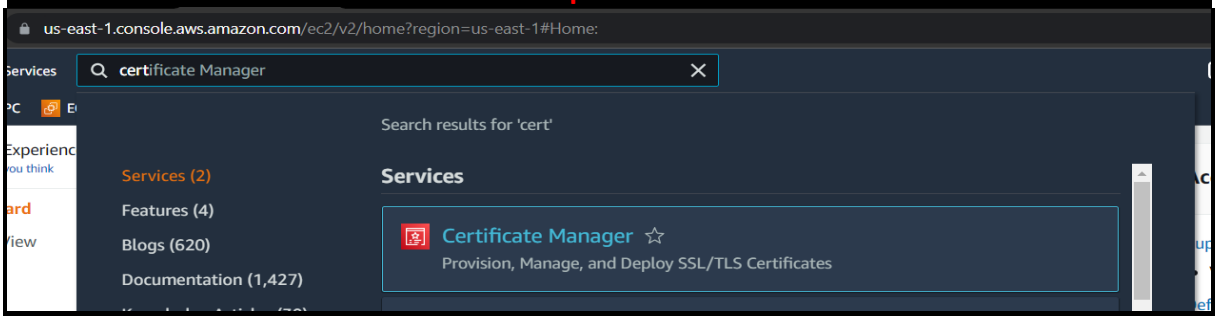


This is test file S3 index file to connect Cloudfront to route53

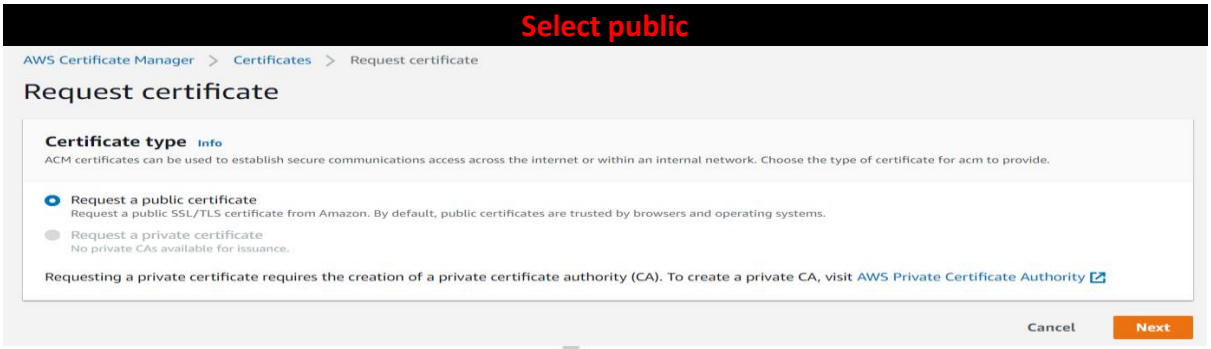
PROJECT-1

12. Configure SSL

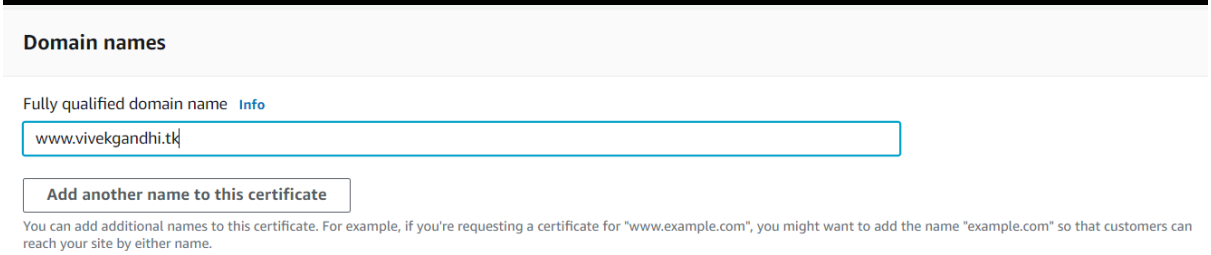
Open SSL



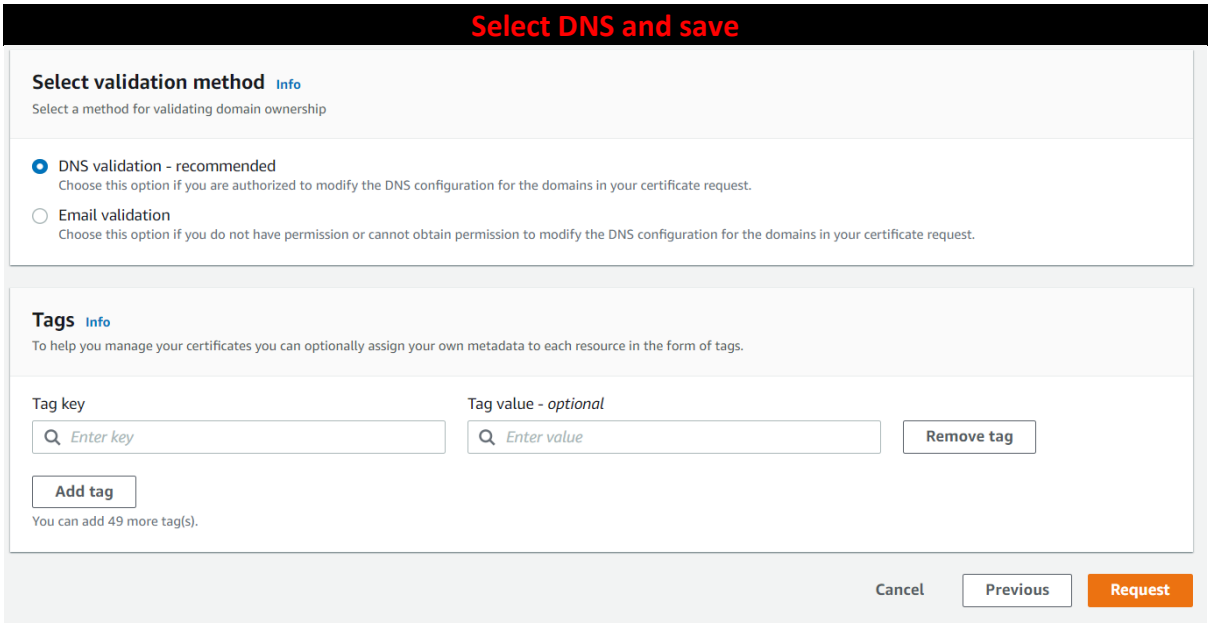
Select public



Enter domain name

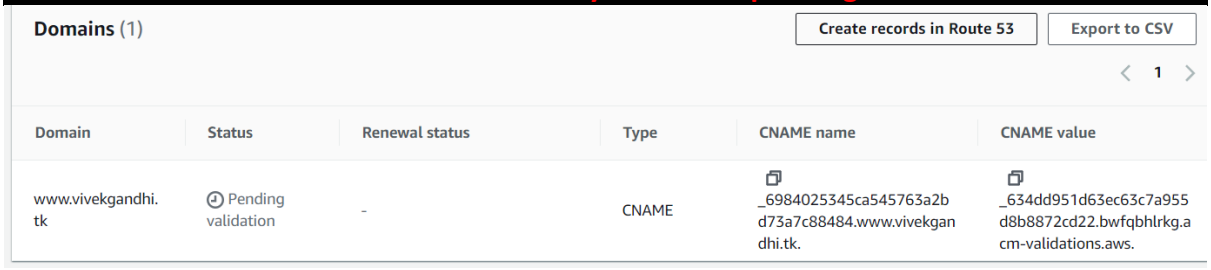


Select DNS and save

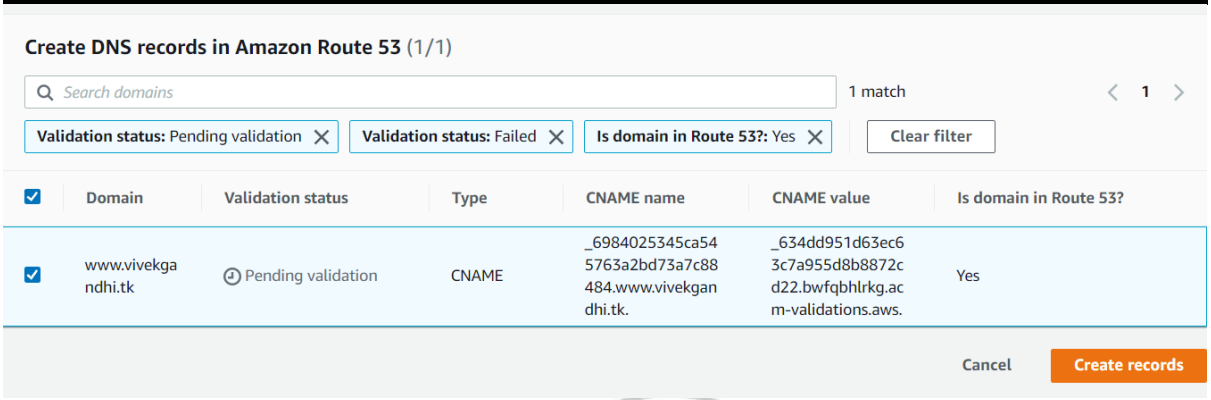


PROJECT-1

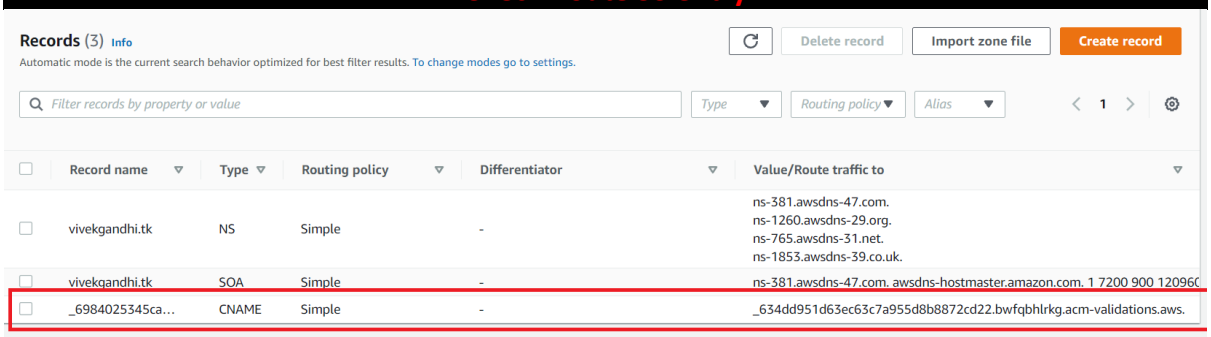
Create successfully but status pending



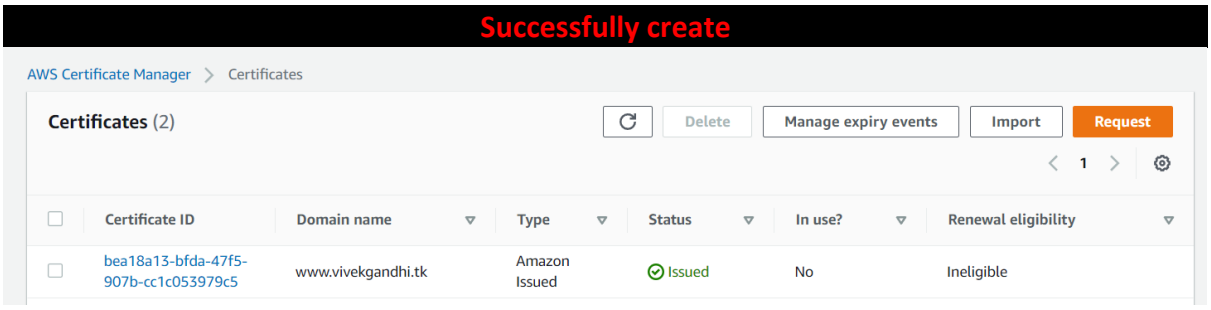
Enter Certificate and click to add route 53



Check Route 53 entry



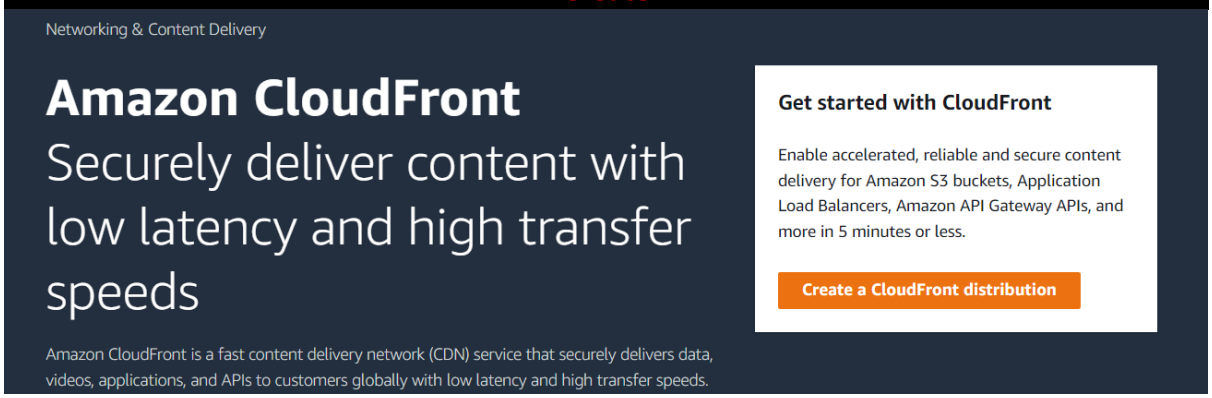
Successfully create



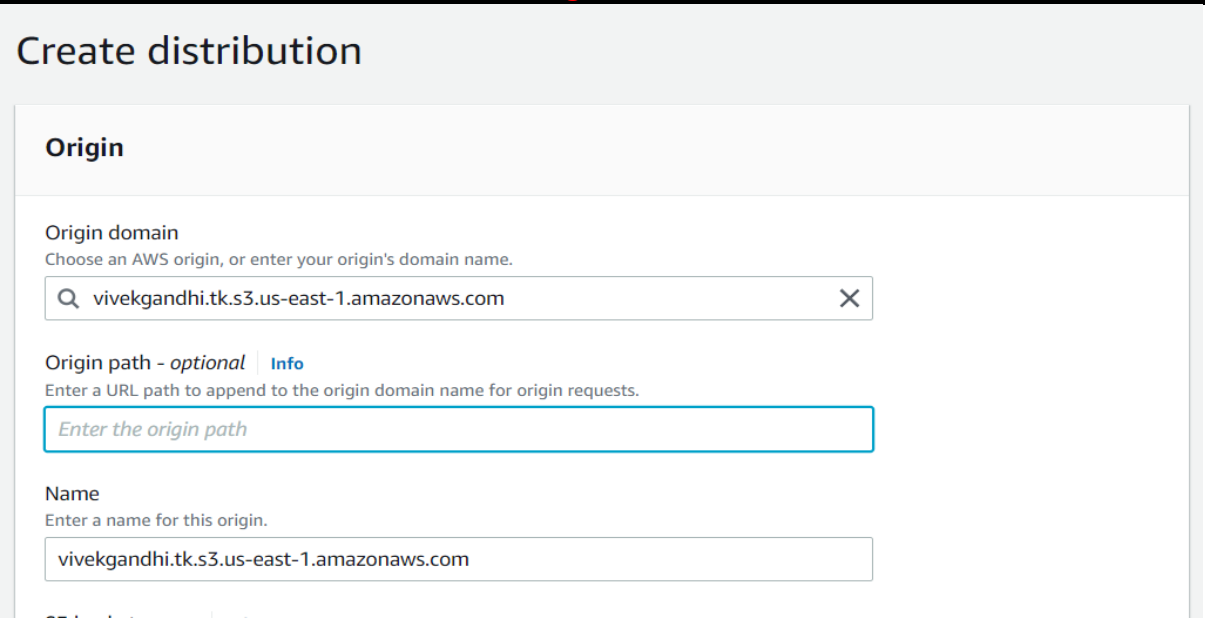
PROJECT-1

13.Configure Cloud front

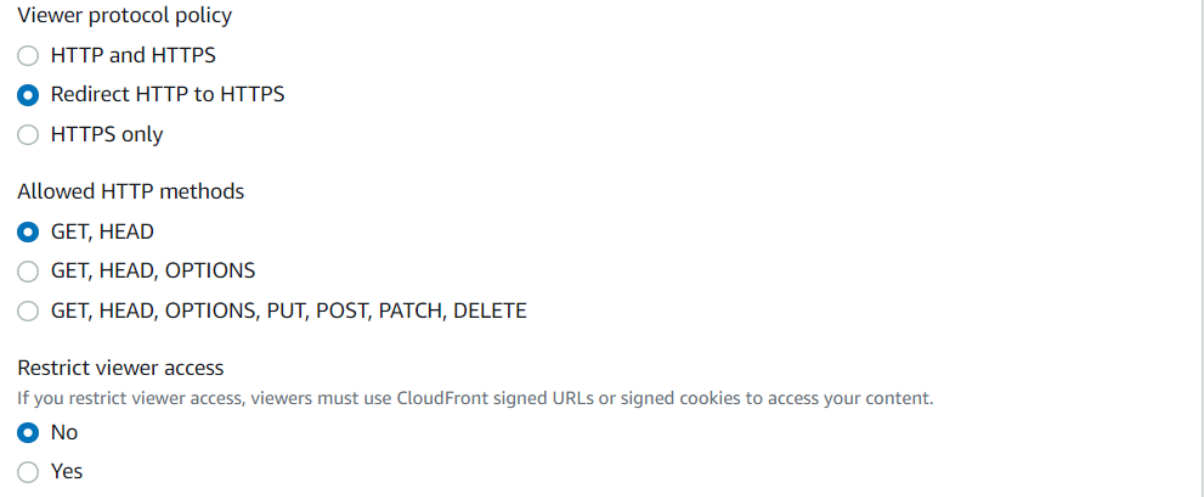
Create



Enter origin S3 domain



Redirect secure http



Configure as it is

Cache key and origin requests

We recommend using a cache policy and origin request policy to control the cache key and origin requests.

- (•) Cache policy and origin request policy (recommended)
- ( ) Legacy cache settings

Cache policy

Choose an existing cache policy or create a new one.

CachingOptimized — Recommended for S3 origins
Default policy when CF compression is enabled

Create policy | View policy

Origin request policy - *optional*

Choose an existing origin request policy or create a new one.

Select origin policy

Create policy

Response headers policy - *optional*

Choose an existing response headers policy or create a new one.

Select response headers

Create policy

▶ **Additional settings**

Enter domain name and add ssl

Alternate domain name (CNAME) - *optional*

Add the custom domain names that you use in URLs for the files served by this distribution.

www.vivekgandhi.tk — Remove

Add item

To add a list of alternative domain names, use the bulk editor.

Custom SSL certificate - *optional*

Associate a certificate from AWS Certificate Manager. The certificate must be in the US East (N. Virginia) Region (us-east-1).

www.vivekgandhi.tk (bea18a13-bfda-47f5-907b-cc1c053979c5)

www.vivekgandhi.tk | Request certificate

Legacy clients support - $600/month prorated charge applies. Most customers do not need this.

CloudFront allocates dedicated IP addresses at each CloudFront edge location to serve your content over HTTPS.

- [ ] Enabled

Security policy

The security policy determines the SSL or TLS protocol and the specific ciphers that CloudFront uses for HTTPS connections with viewers (clients).

- (•) TLSv1.2_2021 (recommended)
- ( ) TLSv1.2_2019
- ( ) TLSv1.2_2018
- ( ) TLSv1.1_2016
- ( ) TLSv1_2016
- ( ) TLSv1

PROJECT-1

Enter html name

Default root object - *optional*

The object (file name) to return when a viewer requests the root URL (/) instead of a specific object.

index.html

Standard logging

Get logs of viewer requests delivered to an Amazon S3 bucket.

- (•) Off
- ( ) On

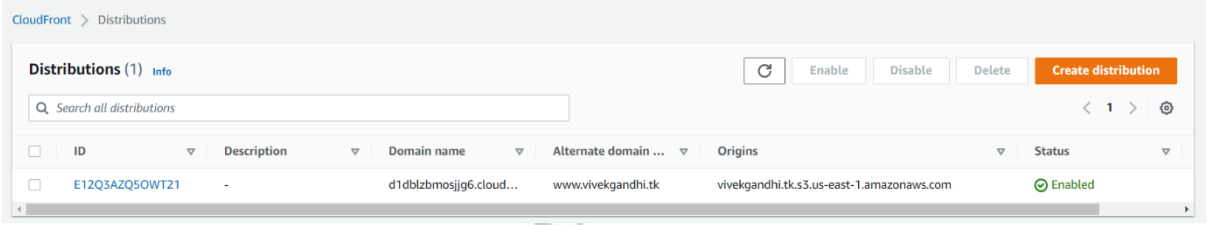
IPv6

- ( ) Off
- (•) On

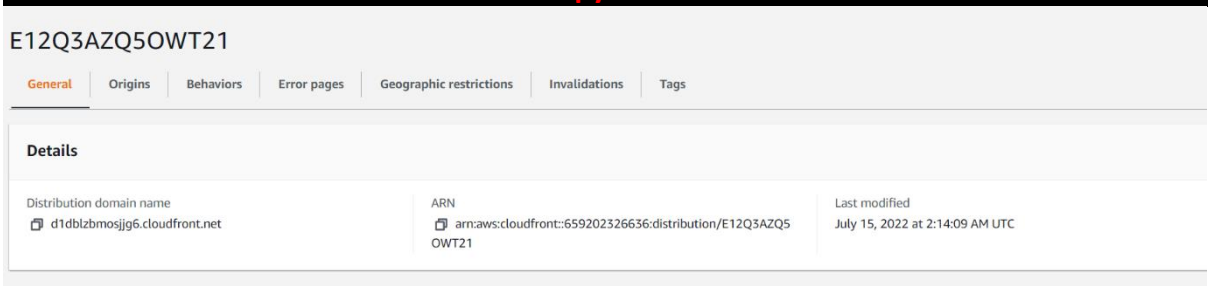
Description - *optional*

Cancel | Create distribution

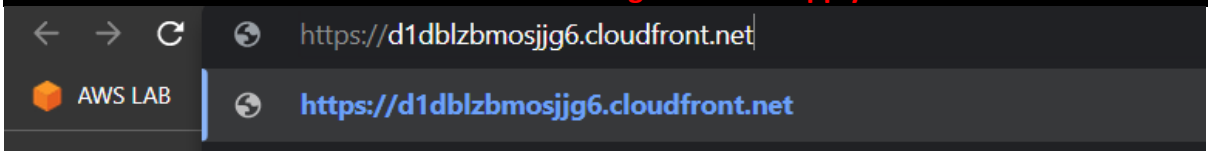
Create successfully

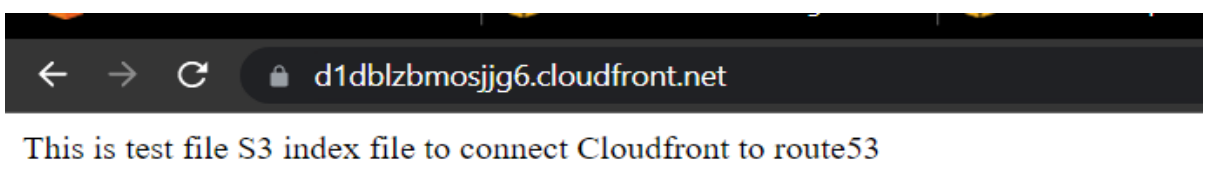


Copy DNS



BROWSE DNS working or not SSL apply

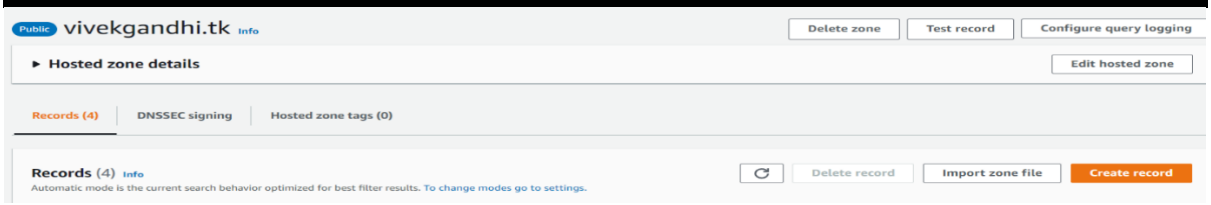




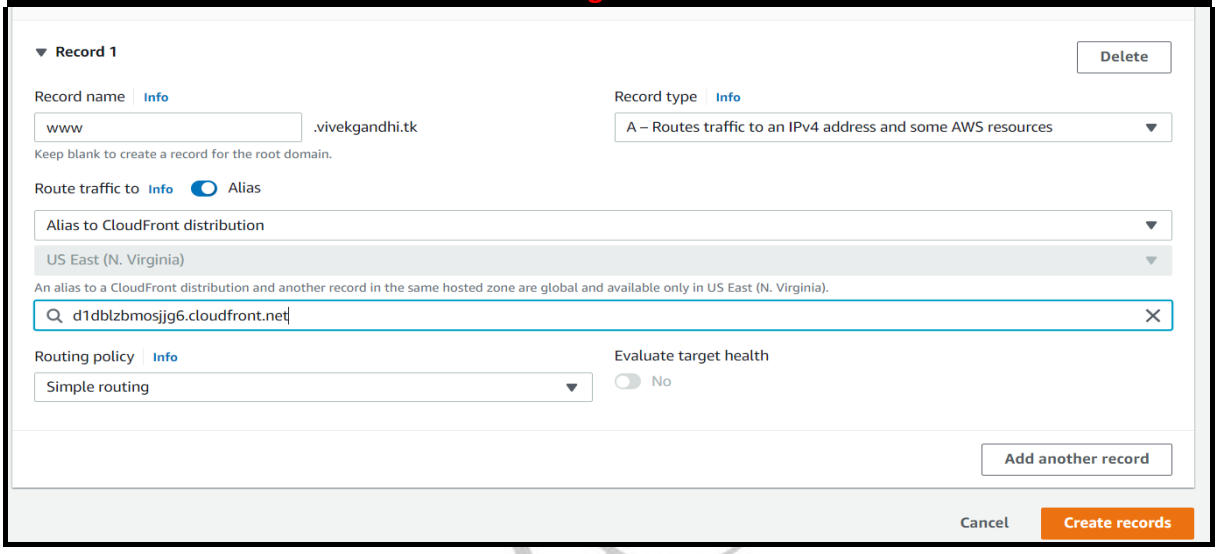
PROJECT-1

14. Integrate Cloud front to route 53

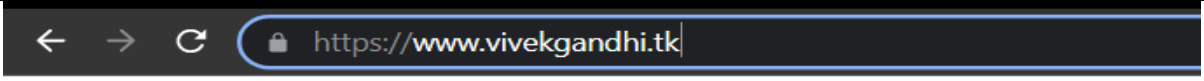
Create Record



Configuration

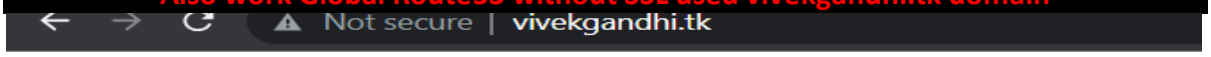


Check to browse Domain with SSL used www.vivekgandhi.tk domain



This is test file S3 index file to connect Cloudfront to route53

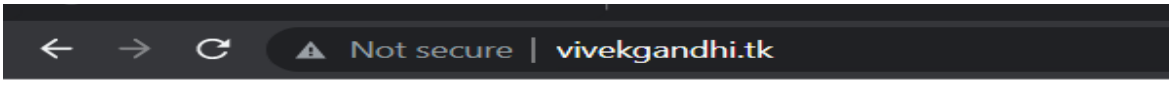
Also work Global Route53 without SSL used vivekgandhi.tk domain



Automatic create Instance EC2-AZb-ASG for testing

Automatic create Instances EC1-AZa-ASG for testing

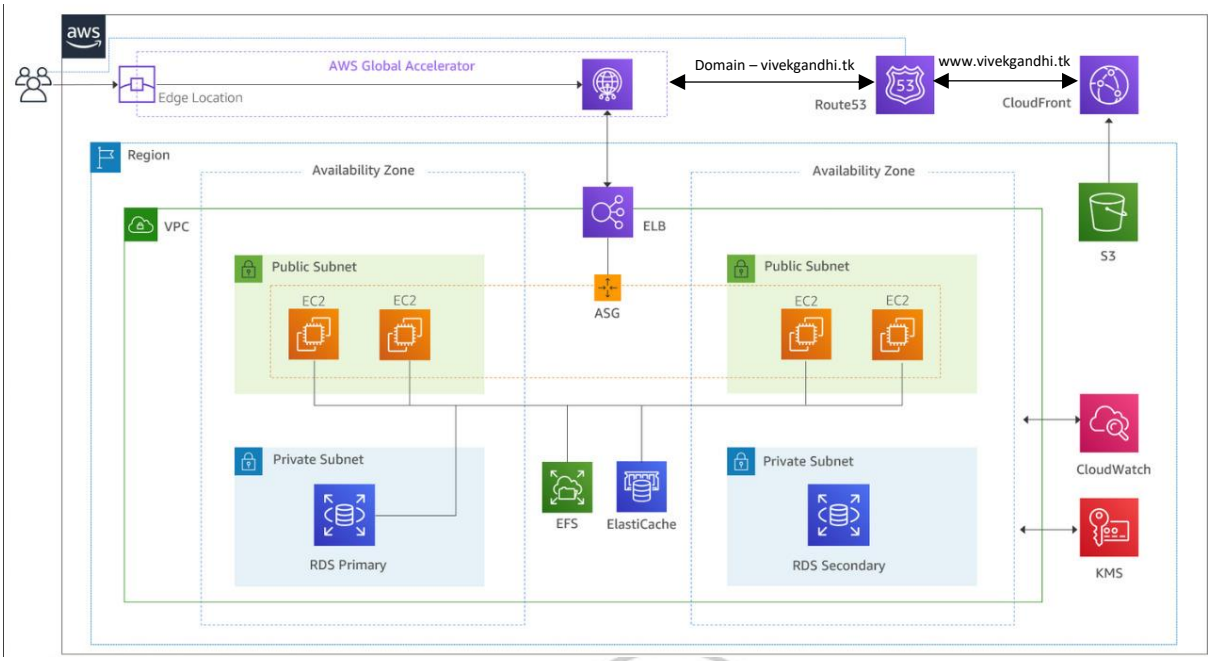
Automatic create Instance EC2-AZa-ASG for testing



Automatic create Instance EC1-AZb-ASG for testing

PROJECT-1

Final Project Summary



All Main Configure Points

1. Create EFS
2. Create KMS (if configure Autoscaling not create instance)
3. Create EC2 INSTANCES with Attached EFS for create Main AMI
4. Create Elastic Cache and configure EC2 instance connectivity checked
5. CREATE RDS and configure EC2 instance connectivity checked
6. Create AMI (Installation service → Httpd, Redis, MariaDB)
7. Create Application LB for global Accelerator
8. Create Auto scaling
9. Create Global Accelerator WITH Integrate Application lb
10. Create ROUTE53 with Integrate Global Accelerator
11. Open S3 in EC2 and put html file
12. Configure SSL attached route53
13. Configure Cloud front attached S3 DNS and SSL certificate
14. Create Route53 Integrate to Cloud front

DOMAIN USED

1. **vivekgandhi.tk → Redirect Global Accelerator DNS → Redirect ALB → Redirect ASG**
2. **www.vivekgandhi.tk → Redirect Cloud Front → Redirect static website → Redirect S3**