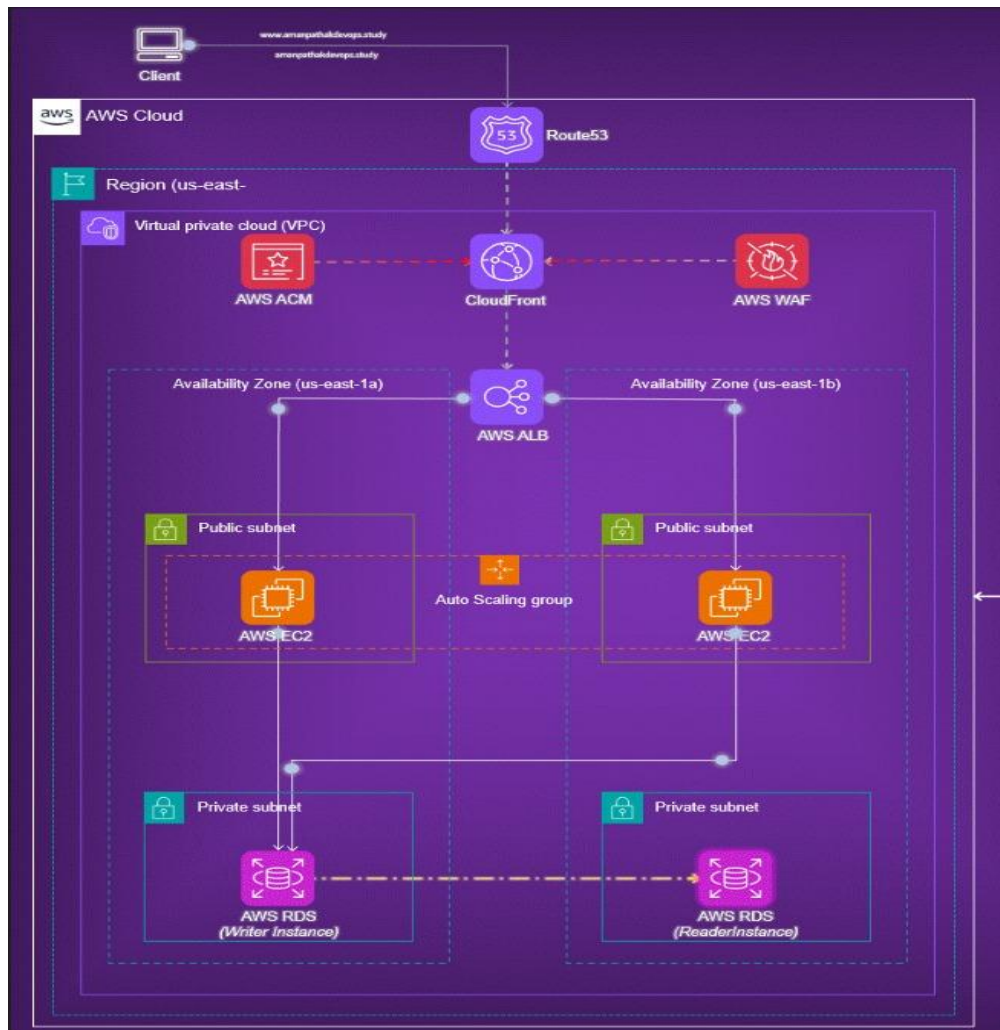


## TWO TIER ARCHITECTURE USING ROUTE53



1. Create 2 Public subnets, 2 Private subnets, Route tables, VPC, Internet Gateway, Nat gateway and subnet associations using VPC and more

VPC > Your VPCs > Create VPC

### Create VPC Info

A VPC is an isolated portion of the AWS Cloud populated by AWS objects, such as Amazon EC2 instances. Mouse over a resource to highlight the related resources.

#### VPC settings

Resources to create Info  
Create only the VPC resource or the VPC and other networking resources.

☐ VPC only ☒ VPC and more

Name tag auto-generation Info  
Enter a value for the Name tag. This value will be used to auto-generate Name tags for all resources in the VPC.

☒ Auto-generate  
project-2

IPv4 CIDR block Info  
Determine the starting IP and the size of your VPC using CIDR notation.

10.0.0.0/16 65,536 IPs  
CIDR block size must be between /16 and /28.

IPv6 CIDR block Info  
☒ No IPv6 CIDR block  
☐ Amazon-provided IPv6 CIDR block

#### Preview

VPC Show details  
Your AWS virtual network

project-2-vpc

Subnets (4)  
Subnets within this VPC

**us-east-1a**

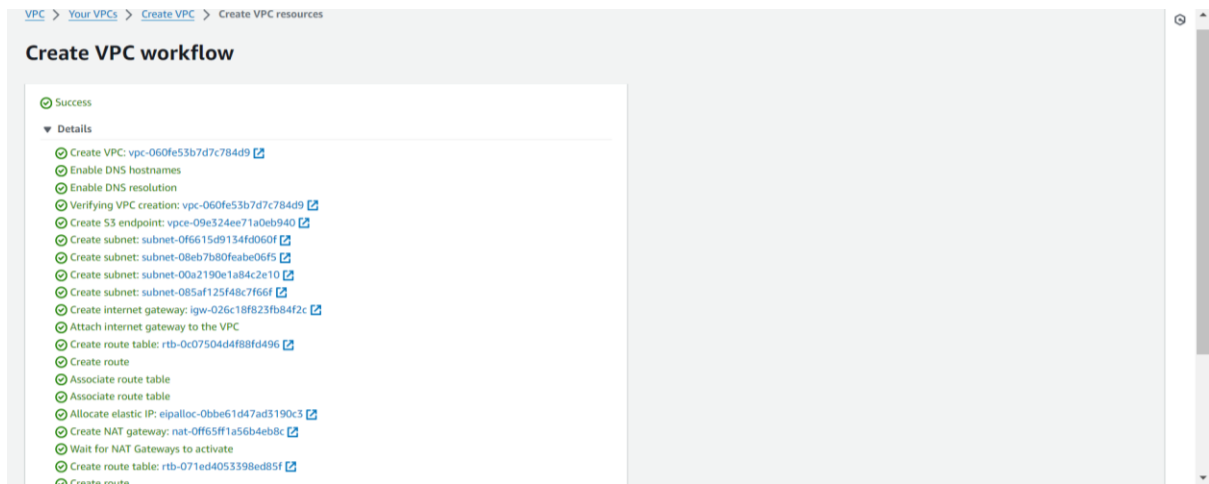
- project-2-subnet-public1-us-east-1a
- project-2-subnet-private1-us-east-

**us-east-1b**

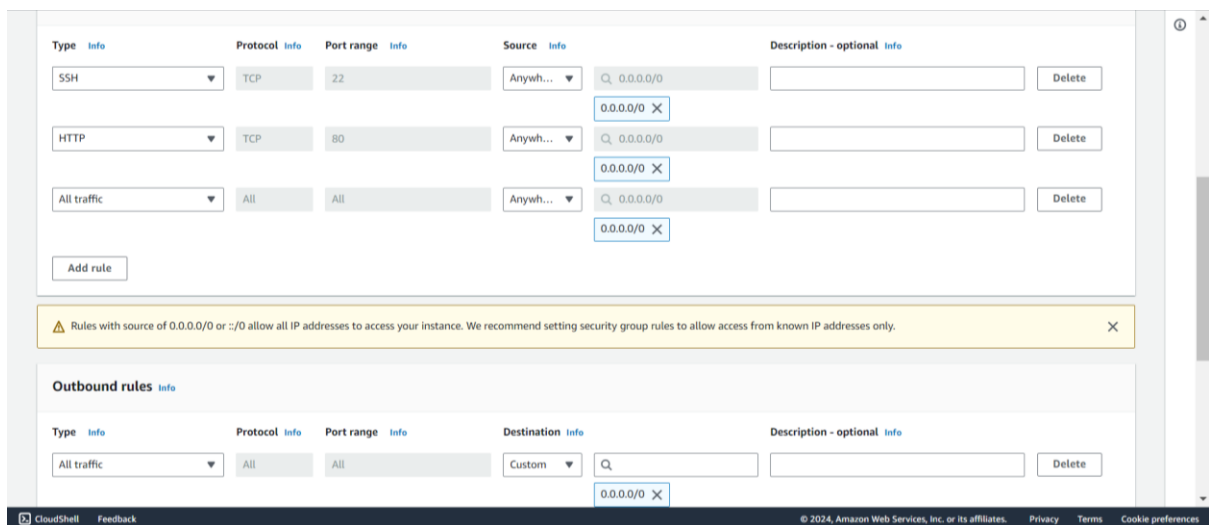
- project-2-subnet-public2-us-east-1b
- project-2-subnet-private2-us-east-

Route tables (3)  
Route network traffic to resources

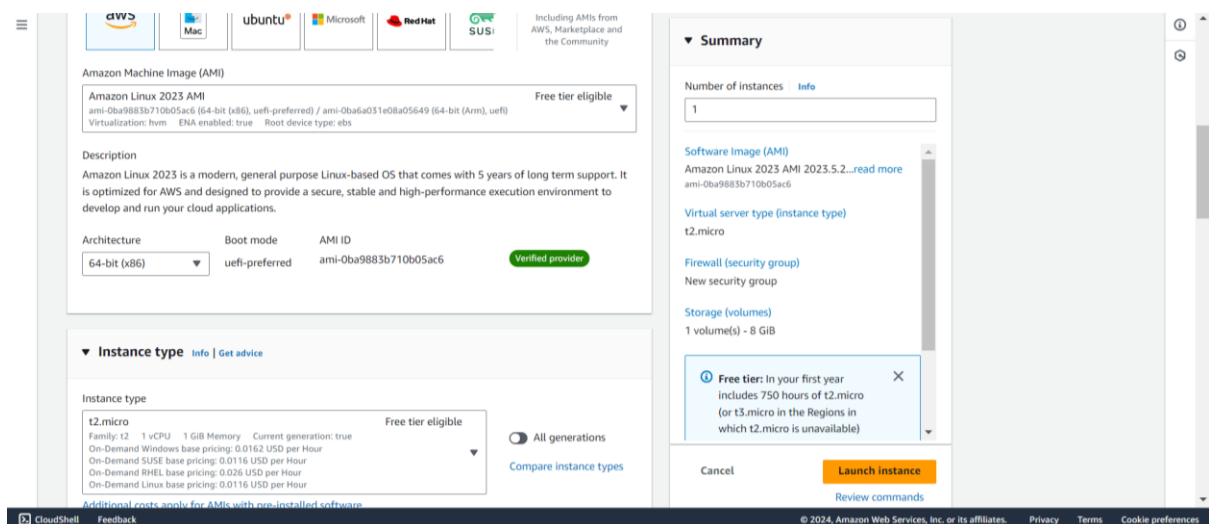
- project-2-rtb-public
- project-2-rtb-private1-us-east-
- project-2-rtb-private2-us-east-



## 2. Create security group



## 3. Launch two instances, one public instance and one private instance and connect them.



EC2 Dashboard

EC2 Global View

Events

Console-to-Code

Instances

Instance Types

Launch Templates

Spot Requests

Savings Plans

Reserved Instances

Dedicated Hosts

Capacity Reservations

Images

AMIs

AMI Catalog

Elastic Block Store

Volumes

Snapshots

Lifecycle Manager

Instances (2) Info

Find Instance by attribute or tag (case-sensitive)

All states

Connect

Instance state

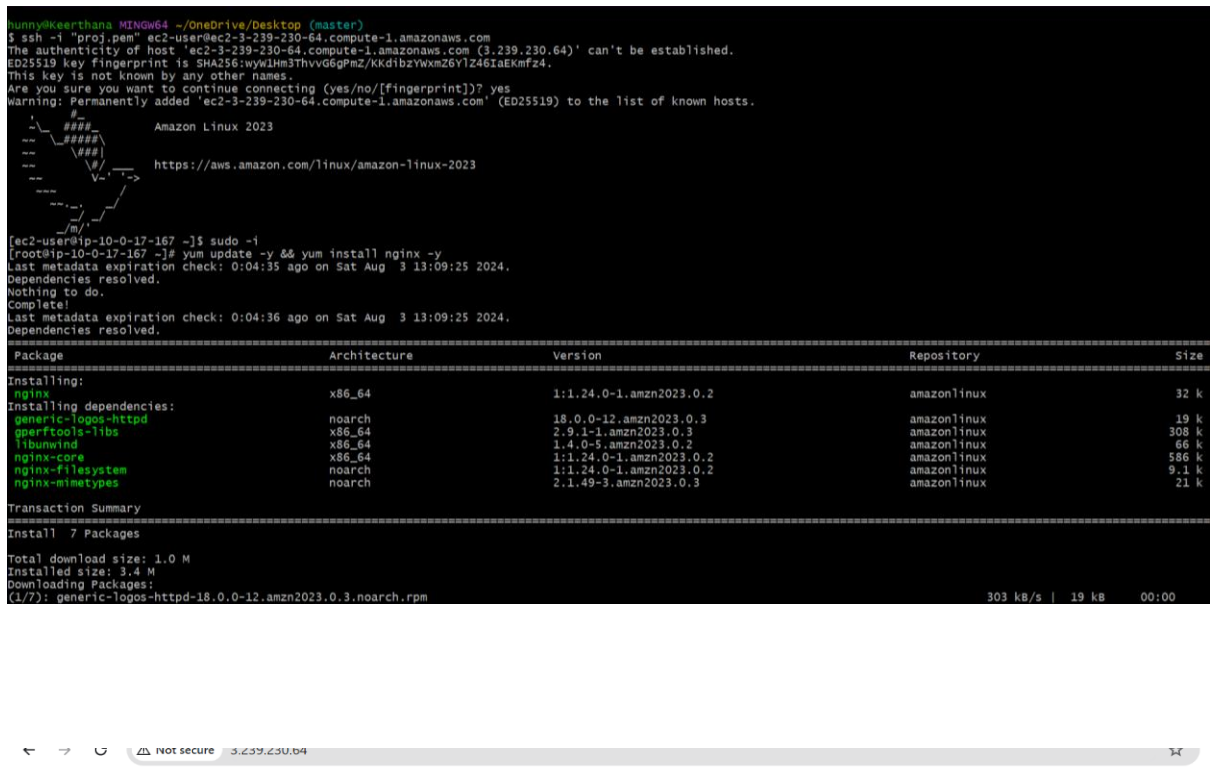
Actions

Launch instances

	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS
<input type="checkbox"/>	private-ec2	i-0550d20f5e090e3e5	Running	t2.micro	Initializing	View alarms	us-east-1a	-
<input type="checkbox"/>	public-ec2	i-0ce5c6052fb6dde3e	Running	t2.micro	2/2 checks passed	View alarms	us-east-1b	-

Select an instance

```
hanny@Keerthana-MINGW64 ~/OneDrive/Desktop (master)
$ ssh -i "proj.pem" ec2-user@ec2-3-239-230-64.compute-1.amazonaws.com
The authenticity of host 'ec2-3-239-230-64.compute-1.amazonaws.com (3.239.230.64)' can't be established.
ED25519 key fingerprint is SHA256:wywIHm3ThvvG6gPmZ/KKdibzyWxmZ6Y1Z46IaEkmfz4.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'ec2-3-239-230-64.compute-1.amazonaws.com' (ED25519) to the list of known hosts.
```



```
Amazon Linux 2023
https://aws.amazon.com/linux/amazon-linux-2023

[ec2-user@ip-10-0-17-167 ~]$ sudo -i
[root@ip-10-0-17-167 ~]# yum update -y && yum install nginx -y
Last metadata expiration check: 0:04:35 ago on Sat Aug 3 13:09:25 2024.
Dependencies resolved.
Nothing to do.
Complete!
Last metadata expiration check: 0:04:36 ago on Sat Aug 3 13:09:25 2024.
Dependencies resolved.

Package Architecture Version Repository Size
Installing:
nginx x86_64 1:1.24.0-1.amzn2023.0.2 amazonlinux 32 k
Installing dependencies:
generic-logos-httpd noarch 18.0.0-12.amzn2023.0.3 amazonlinux 19 k
gperftools-libs x86_64 2.9.1-1.amzn2023.0.3 amazonlinux 308 k
libunwind x86_64 1.4.0-5.amzn2023.0.2 amazonlinux 66 k
nginx-core x86_64 1:1.24.0-1.amzn2023.0.2 amazonlinux 586 k
nginxfilesystem noarch 1:1.24.0-1.amzn2023.0.2 amazonlinux 9.1 k
nginx-mimetypes noarch 2.1.49-3.amzn2023.0.3 amazonlinux 21 k

Transaction Summary
Install 7 Packages

Total download size: 1.0 M
Installed size: 3.4 M
Downloading Packages:
(1/7): generic-logos-httpd-18.0.0-12.amzn2023.0.3.noarch.rpm 303 kB/s | 19 kB 00:00
```

Not secure 3.239.230.64

this is public instance

#### 4. Create autoscaling group, for that first create ami image and launch template

EC2 > Instances > i-Oe594a4ba551f2c07 > Create image

### Create image Info

An image (also referred to as an AMI) defines the programs and settings that are applied when you launch an EC2 instance. You can create an image from the configuration of an existing instance.

Instance ID  
i-Oe594a4ba551f2c07 (public-ec2)

Image name  
proj-2-public  
Maximum 127 characters. Can't be modified after creation.

Image description - optional  
nothing  
Maximum 255 characters

No reboot  
☐ Enable

Instance volumes

Storage type	Device	Snapshot	Size	Volume type	IOPS	Throughput	Delete on termination	Encrypted
--------------	--------	----------	------	-------------	------	------------	-----------------------	-----------

EC2 > Auto scaling groups > Create Auto scaling group

Step 1  
[Choose launch template](#)

Step 2  
[Choose instance launch options](#)

Step 3 - optional  
**Configure advanced options**

Step 4 - optional  
[Configure group size and scaling](#)

Step 5 - optional  
[Add notifications](#)

Step 6 - optional  
[Add tags](#)

Step 7  
[Review](#)

### Configure advanced options - optional Info

Integrate your Auto Scaling group with other services to distribute network traffic across multiple servers using a load balancer or to establish service-to-service communications using VPC Lattice. You can also set options that give you more control over health check replacements and monitoring.

#### Load balancing Info

Use the options below to attach your Auto Scaling group to an existing load balancer, or to a new load balancer that you define.

☐ No load balancer  
Traffic to your Auto Scaling group will not be fronted by a load balancer.

☒ Attach to an existing load balancer  
Choose from your existing load balancers.

☐ Attach to a new load balancer  
Quickly create a basic load balancer to attach to your Auto Scaling group.

#### Attach to an existing load balancer

Select the load balancers that you want to attach to your Auto Scaling group.

☒ Choose from your load balancer target groups  
This option allows you to attach Application, Network, or Gateway Load Balancers.

☐ Choose from Classic Load Balancers

Existing load balancer target groups

Select target groups

private-tg | HTTP  
Application Load Balancer: proj-lb

#### VPC Lattice integration options Info

To improve networking capabilities and scalability, integrate your Auto Scaling group with VPC Lattice. VPC Lattice facilitates communications between AWS services and helps you connect and manage your applications across compute services in AWS.

Select VPC Lattice service to attach

☒ No VPC Lattice service  
VPC Lattice will not manage your Auto Scaling group's network access and connectivity with other services.

☐ Attach to VPC Lattice service  
Incoming requests associated with specified VPC Lattice target groups will be routed to your Auto Scaling group.

[Create new VPC Lattice service](#)

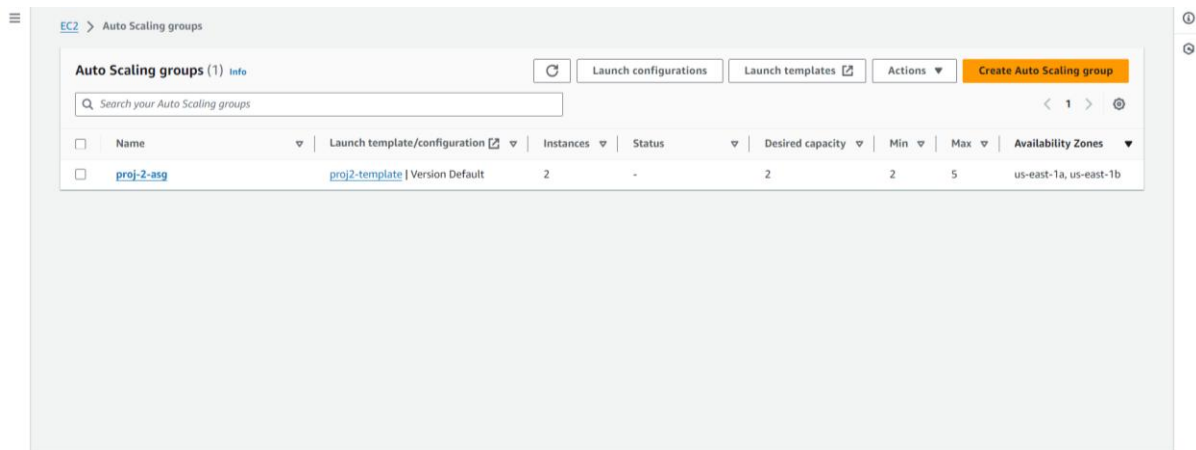
#### Health checks

Health checks increase availability by replacing unhealthy instances. When you use multiple health checks, all are evaluated, and if at least one fails, instance replacement occurs.

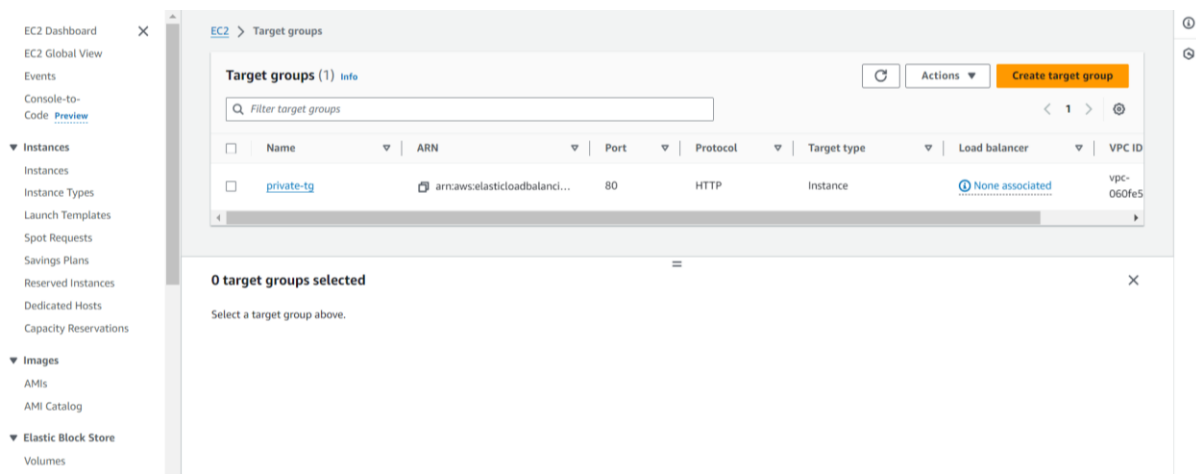
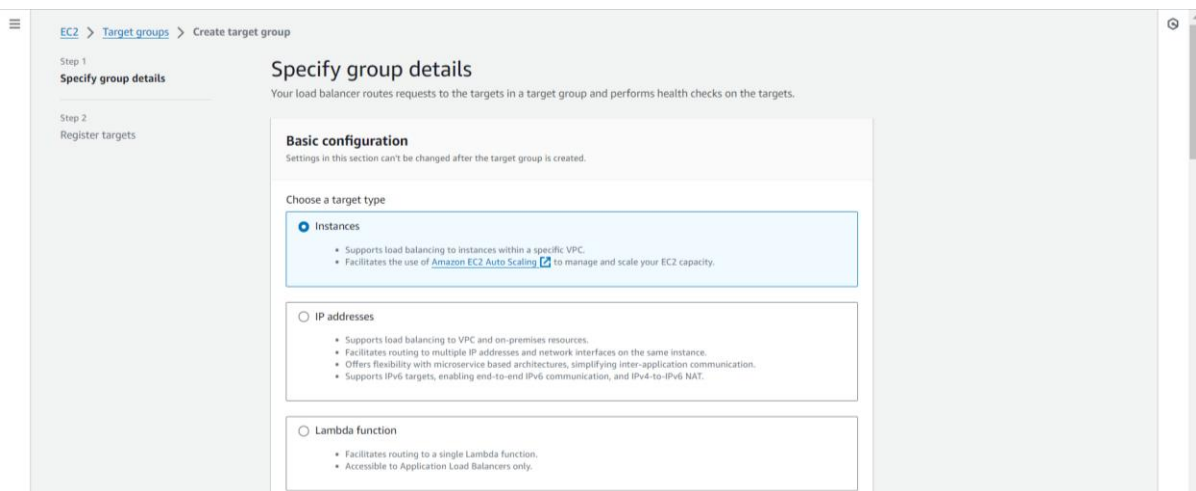
EC2 health checks  
[Always enabled](#)

Additional health check types - optional Info

☒ Turn on Elastic Load Balancing health checks **Recommended**



## 5. Create target group and load balancer



## Create load balancer

Security groups

Select up to 5 security groups

proj-ig-01a517e7566eed1b2 VPC: vpc-060fe53b7d7c784d9

Listeners and routing

A listener is a process that checks for connection requests using the port and protocol you configure. The rules that you define for a listener determine how the load balancer routes requests to its registered targets.

Listener HTTP:80

Remove

Protocol HTTP

Port 80

Default action

Forward to private-tg

HTTP

1-65535

Target type: Instance, IPv4

Create target group

Listener tags - optional

Consider adding tags to your listener. Tags enable you to categorize your AWS resources so you can more easily manage them.

Add listener tag

You can add up to 50 more tags.

EC2 Dashboard

EC2 Global View

Events

Console-to-Code

Instances

Instance Types

Launch Templates

Spot Requests

Savings Plans

Reserved Instances

Dedicated Hosts

Capacity Reservations

Images

AMIs

AMI Catalog

Elastic Block Store

Volumes

Snapshots

EC2 > Load balancers

Load balancers (1)

Elastic Load Balancing scales your load balancer capacity automatically in response to changes in incoming traffic.

Filter load balancers

1

	Name	DNS name	State	VPC ID	Availability Zones	Type	Date created
<input type="checkbox"/>	proj-lb	proj-lb-1997404611.us-east-1.elb.amazonaws.com	Active	vpc-060fe53b7d7c784d9	2 Availability Zones	application	August 3, 2019

0 load balancers selected

Select a load balancer above.

Copy DNS from above image and keep refreshing to get both the instances output

← → ↺

Not secure

proj-lb-1997404611.us-east-1.elb.amazonaws.com

☆

this is public instance

← → ↺

Not secure

proj-lb-1997404611.us-east-1.elb.amazonaws.com

☆

this is private instance

## 6. Create RDS

The screenshot shows the 'Create database' page in the AWS Management Console. The page is titled 'Create database' and has a breadcrumb trail 'RDS > Create database'. It features two main sections: 'Choose a database creation method' and 'Engine options'.

**Choose a database creation method**

- Standard create** (selected): You set all of the configuration options, including ones for availability, security, backups, and maintenance.
- Easy create**: Use recommended best-practice configurations. Some configuration options can be changed after the database is created.

**Engine options**

Engine type [Info](#)

- Aurora (MySQL Compatible)** (selected)
- Aurora (PostgreSQL Compatible)**
- MySQL** (selected)
- MariaDB**

**MySQL**

MySQL is the most popular open source database in the world. MySQL on RDS offers the rich features of the MySQL community edition with the flexibility to easily scale compute resources or storage capacity for your database.

- Supports database size up to 64 TiB.
- Supports General Purpose, Memory Optimized, and Burstable Performance instance classes.
- Supports automated backup and point-in-time recovery.
- Supports up to 15 Read Replicas per instance, within a single Region or 5 read replicas cross-region.

The screenshot shows the 'database-1' page in the AWS Management Console. The page has a breadcrumb trail 'RDS > Databases > database-1' and buttons for 'Refresh', 'Modify', and 'Actions'. It features a 'Summary' section and a 'Connectivity & security' section.

**Summary**

DB identifier	Status	Role	Engine	Recommendations
database-1	Available	Instance	MySQL Community	
CPU	Class	Current activity	Region & AZ	
4.98%	db.t3.micro	0 Connections	us-east-1b	

**Connectivity & security**

Endpoint & port

Endpoint: database-1.cnqmi2qm4xy.us-east-1.rds.amazonaws.com

Port: 3306

Networking

Availability Zone: us-east-1b

VPC: project-2-vpc (vpc-0a517e7566eed1b2)

Security

VPC security groups: proj (sg-01a517e7566eed1b2)

rds-ec2-1 (sg-0521cdf90dd65769c)

Active

The screenshot shows the 'database-1' page in the AWS Management Console, specifically the 'Connected compute resources' and 'Proxies' sections.

**Connected compute resources (1)**

Connections to compute resources that were created automatically by RDS are shown here. Connections to compute resources that were created manually aren't shown.

Filter by compute resources

Resource identifier	Resource type	Availability Zone	VPC security group	Compute resource security group	Connected proxy
i-0e594a4ba551f2c07	EC2 instance	us-east-1b	rds-ec2-1	ec2-rds-1	-

**Proxies (0)**

Filter by proxies

Proxy identifier	Status	Engine family
------------------	--------	---------------

No proxies

## 7. The output will be shown as follows

```
---
  #/
  V-/-> https://aws.amazon.com/linux/amazon-linux-2023
  ---

Last login: Sat Aug 3 13:11:58 2024 from 175.101.32.98
[ec2-user@ip-10-0-17-167 ~]$ sudo -i
[root@ip-10-0-17-167 ~]# yum update -y && yum install nginx -y
Last metadata expiration check: 0:58:45 ago on Sat Aug 3 13:09:25 2024.
Dependencies resolved.
Nothing to do.
Complete!
Last metadata expiration check: 0:58:46 ago on Sat Aug 3 13:09:25 2024.
Package nginx-1:1.24.0-1.amzn2023.0.2.x86_64 is already installed.
Dependencies resolved.
Nothing to do.
Complete!
[root@ip-10-0-17-167 ~]# echo "Saving value to MySQL RDS"
Saving value to MySQL RDS
[root@ip-10-0-17-167 ~]# mysql -h $DB_HOST -u $DB_USER -p $DB_PASSWORD -D $DB_NAME -e "INSERT INTO versionReact (version) VALUES ('$REACT_APP_VERSION');"
-bash: mysql: command not found
[root@ip-10-0-17-167 ~]# wget https://repo.mysql.com/mysql80-community-release-el9-3.noarch.rpm
--2024-08-03 14:10:05-- https://repo.mysql.com/mysql80-community-release-el9-3.noarch.rpm
Resolving repo.mysql.com (repo.mysql.com)... 23.33.204.253, 2600:1408:ec00:88f::1d68, 2600:1408:ec00:888::1d68
Connecting to repo.mysql.com (repo.mysql.com)[23.33.204.253]:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 10715 (10K) [application/x-redhat-package-manager]
Saving to: 'mysql80-community-release-el9-3.noarch.rpm'

mysql80-community-release-el9-3.noarch.rpm 100%[=====] 10.46K --.-KB/s in 0s

2024-08-03 14:10:05 (239 MB/s) - 'mysql80-community-release-el9-3.noarch.rpm' saved [10715/10715]

[root@ip-10-0-17-167 ~]# sudo ls
mysql80-community-release-el9-3.noarch.rpm
[root@ip-10-0-17-167 ~]# sudo yum install mysql80-community-release-el9-1.noarch.rpm
Last metadata expiration check: 1:01:20 ago on Sat Aug 3 13:09:25 2024.
Can not load RPM file: mysql80-community-release-el9-1.noarch.rpm.
Could not open: mysql80-community-release-el9-1.noarch.rpm
[root@ip-10-0-17-167 ~]# sudo dnf update -y
Last metadata expiration check: 1:10:26 ago on Sat Aug 3 13:09:25 2024.
Dependencies resolved.
Nothing to do.
Complete!
[root@ip-10-0-17-167 ~]# sudo dnf install mariadb105
Last metadata expiration check: 1:10:42 ago on Sat Aug 3 13:09:25 2024.
```

```
| Database
+-----+
| information_schema |
| mysql              |
| performance_schema |
| sys                |
+-----+
4 rows in set (0.001 sec)

MySQL [(none)]> create database keerthana;
Query OK, 1 row affected (0.006 sec)

MySQL [(none)]> show databases;
+-----+
| Database |
+-----+
| information_schema |
| keerthana          |
| mysql              |
| performance_schema |
| sys                |
+-----+
5 rows in set (0.001 sec)

MySQL [(none)]> use keerthana;
Database changed
MySQL [keerthana]> CREATE TABLE Persons (
  -> ID int NOT NULL,
  -> LastName varchar(255) NOT NULL,
  -> FirstName varchar(255),
  -> Age int,
  -> PRIMARY KEY (ID)
  -> );
Query OK, 0 rows affected (0.027 sec)

MySQL [keerthana]> INSERT INTO Persons (ID, LastName, FirstName, Age)
  -> VALUES (101, 'Keerthana', 'Dasari', 22);
Query OK, 1 row affected (0.010 sec)

MySQL [keerthana]> select * from Persons;
+-----+
| ID | LastName | FirstName | Age |
+-----+
| 101 | Keerthana | Dasari    | 22  |
+-----+
1 row in set (0.001 sec)
```

## 8. Create Route53 → Hosted Zones → Create hosted zone

Route 53 > Hosted zones > Create hosted zone

Create hosted zone [info](#)

Hosted zone configuration

A hosted zone is a container that holds information about how you want to route traffic for a domain, such as example.com, and its subdomains.

Domain name [info](#)

This is the name of the domain that you want to route traffic for.

sankirthana.xyz

Valid characters: a-z, 0-9, ! \* # \$ % & ' { } + , - . / : ; < = > ? @ [ \ ] ^ \_ ` { } . ~

Description - optional [info](#)

This value lets you distinguish hosted zones that have the same name.

The hosted zone is used for...

The description can have up to 256 characters. 0/256

Type [info](#)

The type indicates whether you want to route traffic on the internet or in an Amazon VPC.

☒ Public hosted zone

A public hosted zone determines how traffic is routed on the internet.

☐ Private hosted zone

A private hosted zone determines how traffic is routed within an Amazon VPC.

CloudShell

Feedback

© 2024, Amazon Web Services, Inc. or its affiliates. [Privacy](#) [Terms](#) [Cookie preferences](#)



## 9. Create record

Route 53 > Hosted zones > sankirthana.xyz > Create record

### Create record Info

**Quick create record** [Switch to wizard](#)

**Record 1** Delete

Record name Info  sankirthana.xyz Record type Info

Keep blank to create a record for the root domain.

☒ Alias

Route traffic to Info

Alias hosted zone ID: Z55XDOTRQ7X7K

Routing policy Info  Evaluate target health ☒ Yes

Add another record

Routing policy Info  Evaluate target health ☐ No

Add another record

Cancel Create records

**View existing records**

The following table lists the existing records in sankirthana.xyz.

**Existing records (2) Info**

Automatic mode is the current search behavior optimized for best filter results. [To change modes go to settings.](#)

<input type="checkbox"/>	Record ...	Type	Routin...	Differ...	Alias	Value/Route traffic to	TTL (s...	Health ...
<input type="checkbox"/>	sankirtha...	NS	Simple	-	No	ns-1222.awsdns-24.org. ns-1920.awsdns-48.co.uk. ns-274.awsdns-34.com. ns-977.awsdns-58.net.	172800	-
<input type="checkbox"/>	sankirtha...	SOA	Simple	-	No	ns-1222.awsdns-24.org. aws...	900	-

## 10. Create ACM certificate

AWS Certificate Manager > Certificates > Request certificate > Request public certificate

### Request public certificate

**Domain names**

Provide one or more domain names for your certificate.

Fully qualified domain name Info

Add another name to this certificate

You can add additional names to this certificate. For example, if you're requesting a certificate for "www.example.com", you might want to add the name "example.com" so that customers can reach your site by either name.

**Validation method Info**

Select a method for validating domain ownership.

☒ DNS validation - recommended

Choose this option if you are authorized to modify the DNS configuration for the domains in your certificate request.

☐ Email validation

Choose this option if you do not have permission or cannot obtain permission to modify the DNS configuration for the domains in your certificate request.

**Key algorithm Info**

List certificates

Request certificate

Import certificate

AWS Private CA

AWS Certificate Manager > Certificates > 694e77d6-803d-4919-b2ed-e7b1b3257d91

694e77d6-803d-4919-b2ed-e7b1b3257d91

Delete

Certificate status

Identifier

694e77d6-803d-4919-b2ed-e7b1b3257d91

Status

Pending validation

Info

ARN

arn:aws:acm:us-east-1:891376940033:certificate/694e77d6-803d-4919-b2ed-e7b1b3257d91

Type

Amazon issued

Domains (1)

Create records in Route 53

Export to CSV

< 1 >

Request certificate

Import certificate

AWS Private CA

Certificates (1)

Refresh

Delete

Manage expiry events

Import

Request

< 1 >

<input type="checkbox"/>	Certificate ID	Domain name	Type	Status	In use
<input type="checkbox"/>	b5afdec0-7c56-471d-b5be-e46f9fb35b7	sankirthana.xyz	Amazon issued	Issued	Yes

11. Create records in Route 53

AWS Certificate Manager (ACM)

List certificates

Request certificate

Import certificate

AWS Private CA

ARN

arn:aws:acm:us-east-1:891376940033:certificate/694e77d6-803d-4919-b2ed-e7b1b3257d91

Type

Amazon issued

Domains (1)

Create records in Route 53

Export to CSV

< 1 >

Domain	Status	Renewal status	Type	CNAME name
sankirthana.xyz	Pending validation	-	CNAME	_97c3fe70cb5020b0e24ac4b4ff8bb25e.sankirthana.xyz

Details

In use	Serial number	Requested at	Renewal eligibility
No	N/A	August 03, 2024, 20:50:21 (UTC+05:30)	Ineligible

Domain name	Public key info	Issued at
sankirthana.xyz	RSA 2048	N/A

List certificates

Request certificate

Import certificate

AWS Private CA

AWS Certificate Manager > Certificates > 694e77d6-803d-4919-b2ed-e7b1b3257d91 >

Create DNS records in Amazon Route 53

Create DNS records in Amazon Route 53 (1/1)

Search domains

< 1 >

<input checked="" type="checkbox"/>	Domain	Validation status	Is domain in Route 53?
<input checked="" type="checkbox"/>	sankirthana.xyz	Pending validation	Yes

Cancel Create records

## 12. Create Cloud Front → Distributions

The screenshot displays the AWS CloudFront console. The top section shows the 'Create distribution' form with the following details:

- Origin domain:** proj-lb-1997404611.us-east-1.elb.amazonaws.com
- Protocol:** HTTPS only (selected)
- HTTPS port:** 443
- Minimum Origin SSL protocol:** TLSv1.2 (selected)

The bottom section shows the details for the distribution 'E1LPF4AVN1FMOR':

- Distribution domain name:** d1563gmn94etxp.cloudfront.net
- ARN:** arn:aws:cloudfront:891376940033:distribution/E1LPF4AVN1FMOR
- Last modified:** Deploying
- Settings:** Description, Price class (Use all edge locations), Supported HTTP versions (HTTP/2, HTTP/1.1, HTTP/1.0), Alternate domain names (Custom SSL certificate: sankirthana.xyz), Security policy (TLSv1.2\_2021), Standard logging (Off), Cookie logging (Off), Default root object (-).

The bottom part of the screenshot shows the 'Distributions (1)' table with the following data:

ID	Description	Type	Domain name	Alternate ...	Origins	Status
E1LPF4AVN1FMOR	-	Production	d1563gmn94etxp.cloudfront.net	-	proj-lb-19974046	Enabled

## 12. Create Cloud Front → Distributions

The screenshot displays the AWS EC2 console, specifically the 'HTTP:80' listener details for the 'proj-lb' load balancer. The details section shows:

- Protocol:Port:** HTTP:80
- Load balancer:** proj-lb
- Default actions:** Forward to target group (private-to: 1 (100%), Target group stickiness: Off)
- Listener ARN:** arn:aws:elasticloadbalancing:us-east-1:891376940033:listener/app/proj-lb/bbbe370edbd75eb/9e942ab364a2f85d

The bottom section shows the 'Listener rules (1)' table with the following data:

Listener rules (1)
Filter rules

**EDIT LISTENER**  
Edit the protocol, port or default actions of your Application Load Balancer (ALB) listener.

► Load balancer details: proj-lb

**Listener details**  
A listener checks for connection requests using the protocol and port that you configure. The default action and any additional rules that you create determine how the Application Load Balancer routes requests to its registered targets.

Listener ARN  
arn:aws:elasticloadbalancing:us-east-1:891376940033:listener/app/proj-lb/bbbe370edbd75eb/9e942ab364a2f85d

**Listener configuration**  
The listener will be identified by the protocol and port.

Protocol  
Used for connections from clients to the load balancer.  
HTTP

Port  
The port on which the load balancer is listening for connections.  
80  
1-65535

**Default actions** [Info](#)  
The default action is used if no other rules apply. Choose the default action for traffic on this listener.

**Routing actions**

☒ Forward to target groups ☐ Redirect to URL ☐ Return fixed response

**Hosted zones details** [Edit hosted zone](#)

[Records \(4\)](#) [DNSSEC signing](#) [Hosted zone tags \(0\)](#)

**Records (4)** [Info](#) [Delete record](#) [Import zone file](#) [Create record](#)

Automatic mode is the current search behavior optimized for best filter results. To change modes go to settings.

Filter records by property or value

<input type="checkbox"/>	Record ...	Type	Routin...	Differ...	Alias	Value/Route traffic to	TTL (s...)	Health ...	Evaluat
<input type="checkbox"/>	sankirtha...	A	Simple	-	Yes	dualstack-proj-lb-199740461...	-	-	Yes
<input type="checkbox"/>	sankirtha...	NS	Simple	-	No	ns-468.awsdns-58.com. ns-634.awsdns-15.net. ns-1788.awsdns-31.co.uk. ns-1241.awsdns-27.org.	172800	-	-
<input type="checkbox"/>	sankirtha...	SOA	Simple	-	No	ns-468.awsdns-58.com. awsd...	900	-	-
<input type="checkbox"/>	_97c3fe7...	CNAME	Simple	-	No	_5fc6e11b7d478239765f35f...	300	-	-

15. Go to godaddy website and change name servers.

**Edit nameservers**  
Choose nameservers for sankirthana.xyz

☐ GoDaddy Nameservers (recommended)

☒ I'll use my own nameservers

ns-468.awsdns-58.com

ns-634.awsdns-15.net

ns-1788.awsdns-31.co.uk

ns-1241.awsdns-27.org

[Add Nameserver](#)

[Save](#) [Cancel](#)

[Change Nameservers](#)

16. Copy domain name and paste it in google to get the output