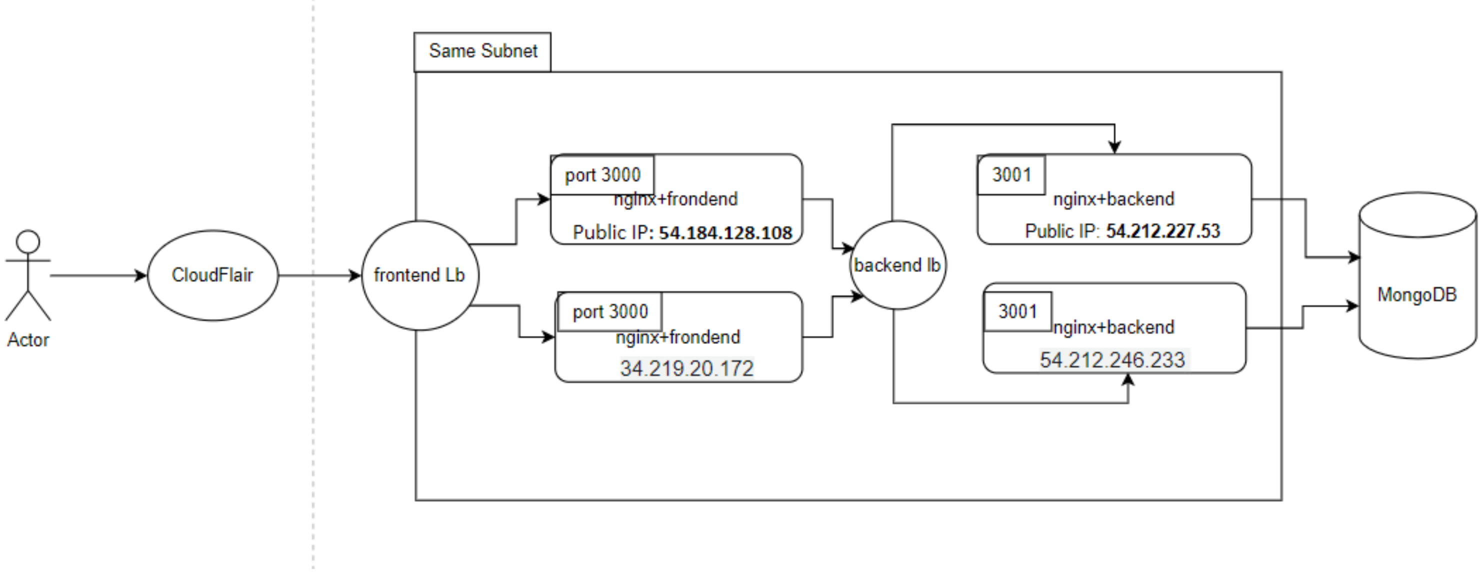
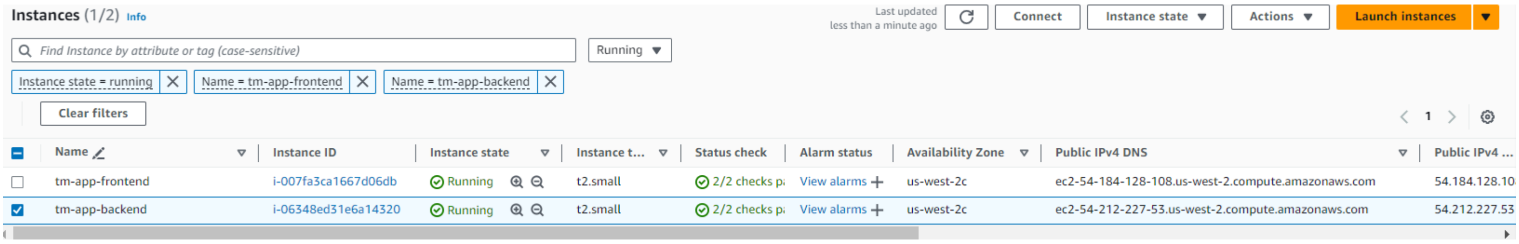
- Diagram Flow



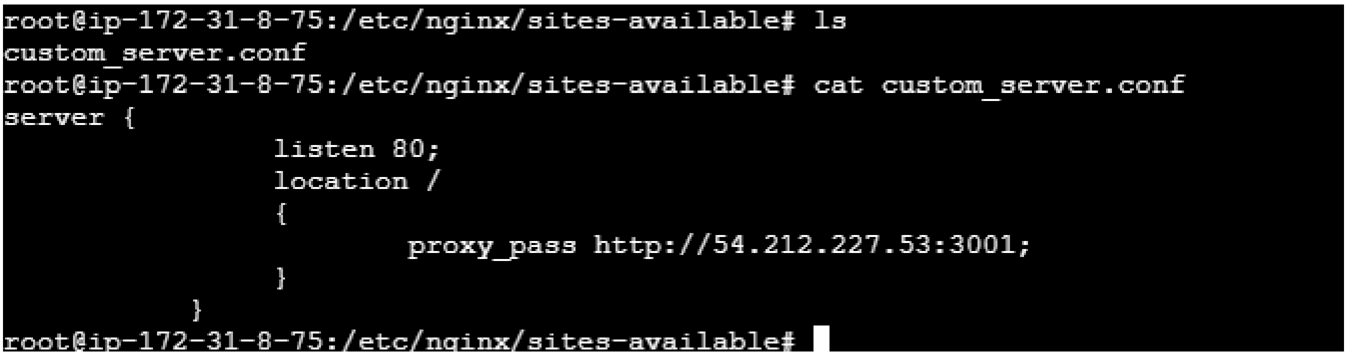
- Creating 2 instance



Instances (1/2) Info

	Name	Instance ID	Instance state	Instance t...	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IPv4 ...
□	tm-app-frontend	i-007fa3ca1667d06db	✓ Running	t2.small	✓ 2/2 checks pa	View alarms +	us-west-2c	ec2-54-184-128-108.us-west-2.compute.amazonaws.com	54.184.128.10
☑	tm-app-backend	i-06348ed31e6a14320	✓ Running	t2.small	✓ 2/2 checks pa	View alarms +	us-west-2c	ec2-54-212-227-53.us-west-2.compute.amazonaws.com	54.212.227.53

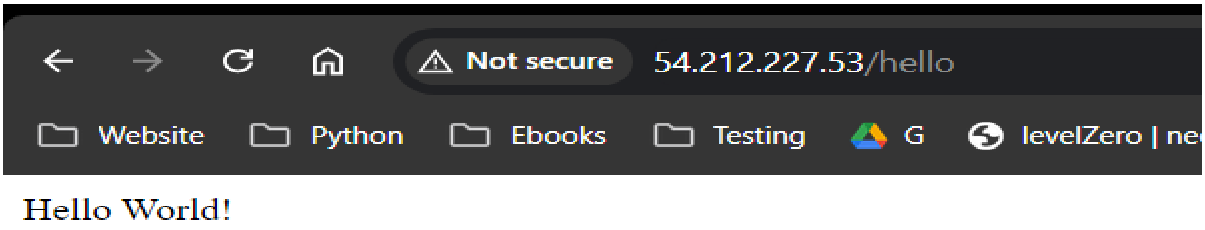
- Backend Configuration
  - Public IP: **54.212.227.53** VPC: **vpc-0321f38a7b594180d** Subnet ID**: subnet-0f30c30418def6379**
  - **Installing nodejs, npm and nginx**
    - # sudo apt update
    - # sudo apt install nodejs -y
    - # sudo apt install npm -y
    - sudo apt install nginx –y
  - **Cloning the repo**
    - # sudo su
    - # cd /
    - # git clone https://github.com/UnpredictablePrashant/TravelMemory
  - **/TravelMemory/backend**
    - Create .env
      - MONGO_URI='mongodb+srv://bharat:<password>@bharatcluster01.xybln.mongodb.net/travelmemory'
      - PORT=3001
    - cd /TravelMemory/backend
    - npm install
    - node index.js & **# & to run the command in background**
  - **Nginx Configuration**
    - # sudo unlink /etc/nginx/sites-enabled/default
    - # cd /etc/nginx/sites-available/
    - # sudo vim custom_server.conf
    - # systemctl restart nginx



```
root@ip-172-31-8-75:/etc/nginx/sites-available# ls
custom_server.conf
root@ip-172-31-8-75:/etc/nginx/sites-available# cat custom_server.conf
server {
                listen 80;
                location /
                {
                        proxy_pass http://54.212.227.53:3001;
                }
        }
root@ip-172-31-8-75:/etc/nginx/sites-available#
```

Note: 1. in proxy_pass give your backend instance IP along with port so that any request landing on nginx port 80 can be redirected to backend port

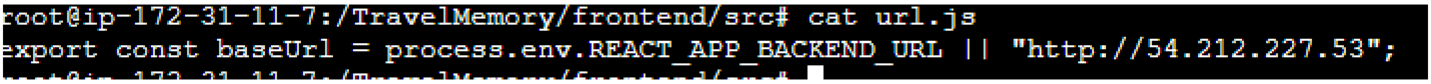
2. Hit the ip of your instance without any port



54.212.227.53/hello

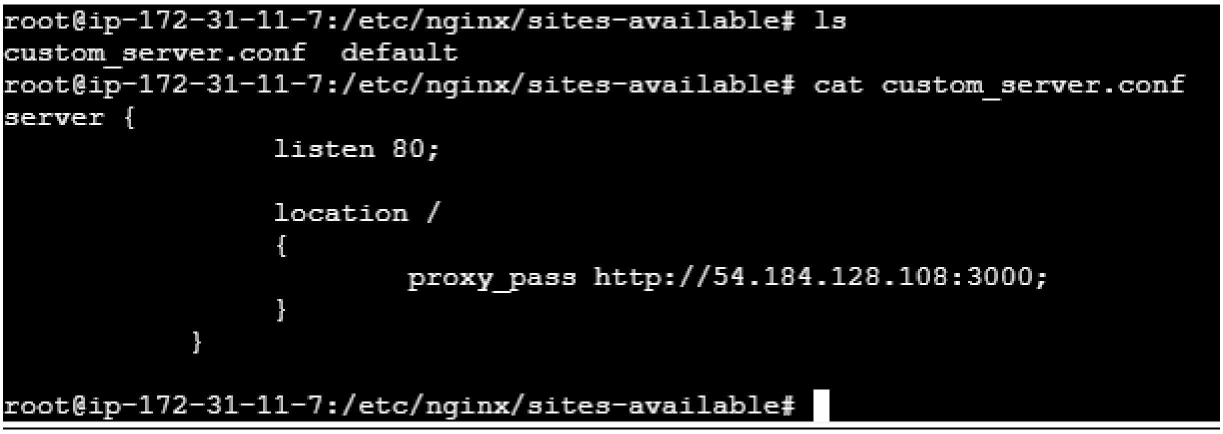
Hello World!

- Frontend Configuration
  - Public IP: **54.184.128.108** VPC: **vpc-0321f38a7b594180d** Subnet ID**: subnet-0f30c30418def6379**
  - **Installing nodejs, npm and nginx**
    - # sudo apt update
    - # sudo apt install nodejs -y
    - # sudo apt install npm -y
    - sudo apt install nginx –y
  - **Cloning the repo**
    - # sudo su
    - # cd /
    - # git clone https://github.com/UnpredictablePrashant/TravelMemory
  - **/TravelMemory/frontend**
    - cd /TravelMemory/frontend/src
    - vim url.js
      - edit url to http://<your Ec2 backend IP>



```
root@ip-172-31-11-7:/TravelMemory/frontend/src# cat url.js
export const baseUrl = process.env.REACT_APP_BACKEND_URL || "http://54.212.227.53";
root@ip-172-31-11-7:/TravelMemory/frontend/src#
```

    - cd /TravelMemory/frontend
    - npm install
    - npm start & **# & to run the command in background**
    -
  - Nginx Configuration
    - # sudo unlink /etc/nginx/sites-enabled/default
    - # cd /etc/nginx/sites-available/
    - # sudo vim custom_server.conf
    - # sudo systemctl restart nginx



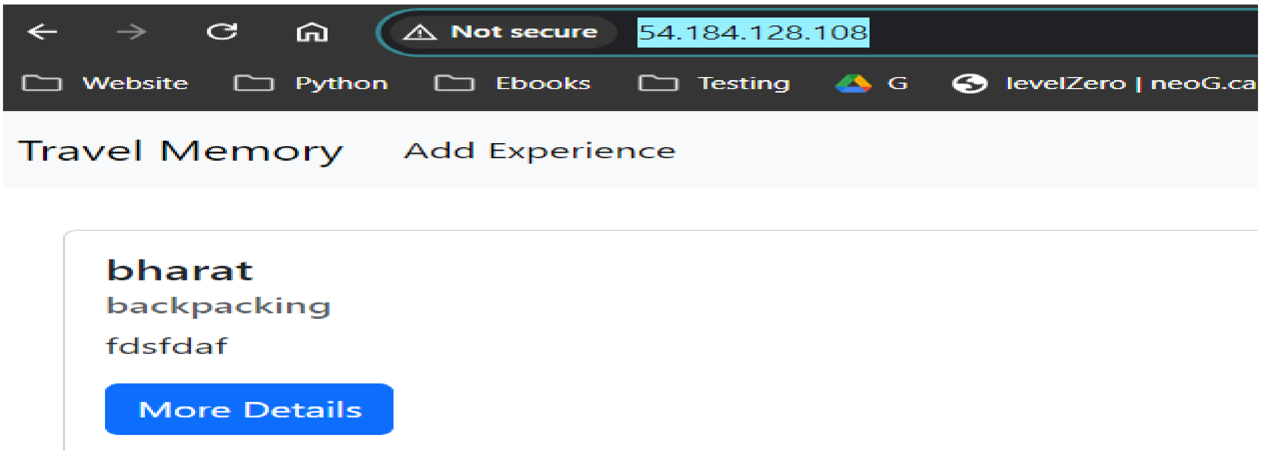
```
root@ip-172-31-11-7:/etc/nginx/sites-available# ls
custom_server.conf  default
root@ip-172-31-11-7:/etc/nginx/sites-available# cat custom_server.conf
server {
                listen 80;

                location /
                {
                        proxy_pass http://54.184.128.108:3000;
                }
           }

root@ip-172-31-11-7:/etc/nginx/sites-available# 
```

Note: 1. in proxy_pass give your frontend instance IP along with port so that any request landing on nginx port 80 can be redirected to frontend port

2. Hit the ip of your instance without any port



Travel Memory Add Experience

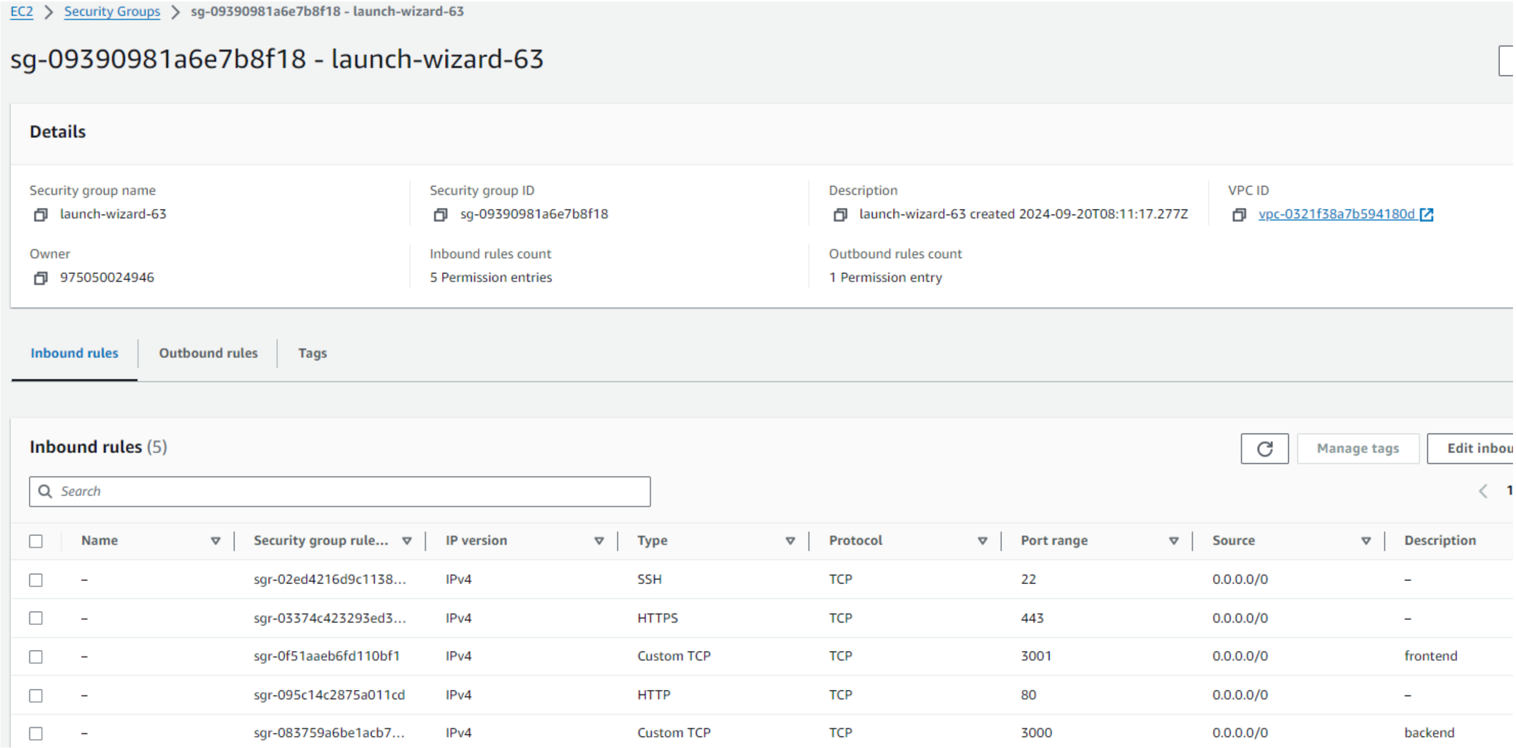
bharat

backpacking

fdsfdaf

More Details

- Create security Group
  - Name: sg-09390981a6e7b8f18 - launch-wizard-63
  - Ingress port:
    - ssh 22
    - http 80
    - port for backend : 3001 frontend port: 3001


EC2 > Security Groups > sg-09390981a6e7b8f18 - launch-wizard-63

sg-09390981a6e7b8f18 - launch-wizard-63

Details

Security group name: launch-wizard-63

Security group ID: sg-09390981a6e7b8f18

Description: launch-wizard-63 created 2024-09-20T08:11:17.277Z

VPC ID: vpc-0321f38a7b594180d

Owner: 975050024946

Inbound rules count: 5 Permission entries

Outbound rules count: 1 Permission entry

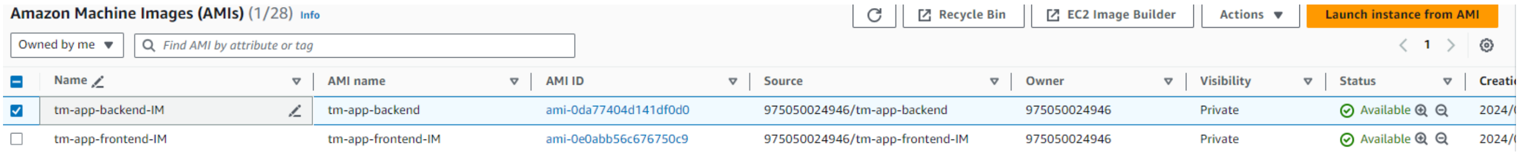
Inbound rules
Outbound rules
Tags

Inbound rules (5)

Manage tags
Edit inbou
Search

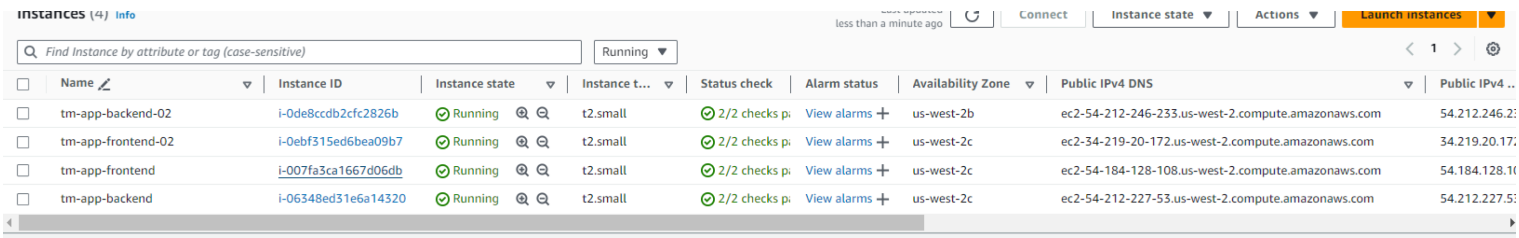
□	Name	Security group rule...	IP version	Type	Protocol	Port range	Source	Description
□	–	sgr-02ed4216d9c1138...	IPv4	SSH	TCP	22	0.0.0.0/0	–
□	–	sgr-03374c423293ed3...	IPv4	HTTPS	TCP	443	0.0.0.0/0	–
□	–	sgr-0f51aaeb6fd110bf1	IPv4	Custom TCP	TCP	3001	0.0.0.0/0	frontend
□	–	sgr-095c14c2875a011cd	IPv4	HTTP	TCP	80	0.0.0.0/0	–
□	–	sgr-083759a6be1acb7...	IPv4	Custom TCP	TCP	3000	0.0.0.0/0	backend

- Create AMI image for frontend and backend and launch 1 more backend and frontend using AMI



Amazon Machine Images (AMIs) (1/28) Info

☑	Name	AMI name	AMI ID	Source	Owner	Visibility	Status	Creati…
☑	tm-app-backend-IM	tm-app-backend	ami-0da77404d141df0d0	975050024946/tm-app-backend	975050024946	Private	✓ Available	2024/(
□	tm-app-frontend-IM	tm-app-frontend-IM	ami-0e0abb56c676750c9	975050024946/tm-app-frontend-IM	975050024946	Private	✓ Available	2024/(

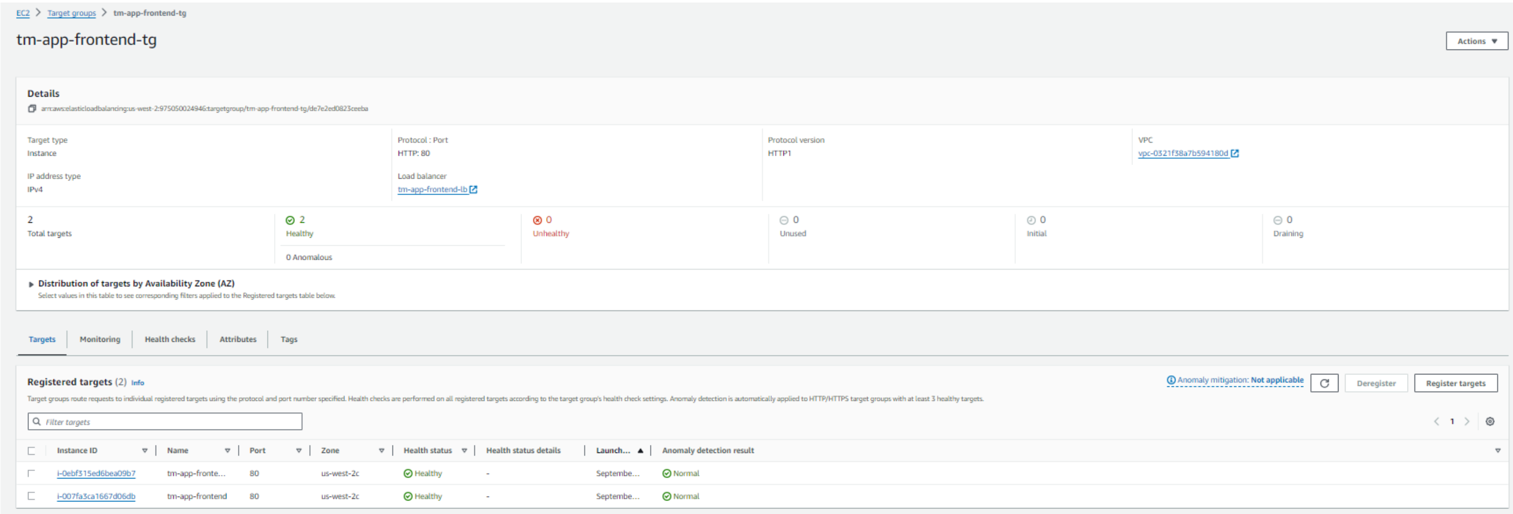


Instances (4) Info

less than a minute ago

□	Name	Instance ID	Instance state	Instance t…	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IPv4 …
□	tm-app-backend-02	i-0de8ccdb2cfc2826b	✓ Running	t2.small	✓ 2/2 checks pa	View alarms +	us-west-2b	ec2-54-212-246-233.us-west-2.compute.amazonaws.com	54.212.246.2:
□	tm-app-frontend-02	i-0ebf315ed6bea09b7	✓ Running	t2.small	✓ 2/2 checks pa	View alarms +	us-west-2c	ec2-34-219-20-172.us-west-2.compute.amazonaws.com	34.219.20.17:
□	tm-app-frontend	i-007fa3ca1667d06db	✓ Running	t2.small	✓ 2/2 checks pa	View alarms +	us-west-2c	ec2-54-184-128-108.us-west-2.compute.amazonaws.com	54.184.128.1(
□	tm-app-backend	i-06348ed31e6a14320	✓ Running	t2.small	✓ 2/2 checks pa	View alarms +	us-west-2c	ec2-54-212-227-53.us-west-2.compute.amazonaws.com	54.212.227.5:

- Create target group for backend and frontend instances
  - **TG: tm-app-frontend-tg**



EC2 > Target groups > tm-app-frontend-tg

tm-app-frontend-tg

Details

arn:aws:elasticloadbalancing:us-west-2:975050024946:targetgroup/tm-app-frontend-tg/de7e2ed0823ceeba

Target type: Instance
IP address type: IPv4
Protocol : Port: HTTP: 80
Load balancer: tm-app-frontend-lb
Protocol version: HTTP1
VPC: vpc-0321f38a7b594180d

2 Total targets | 2 Healthy | 0 Anomalous | 0 Unhealthy | 0 Unused | 0 Initial | 0 Draining

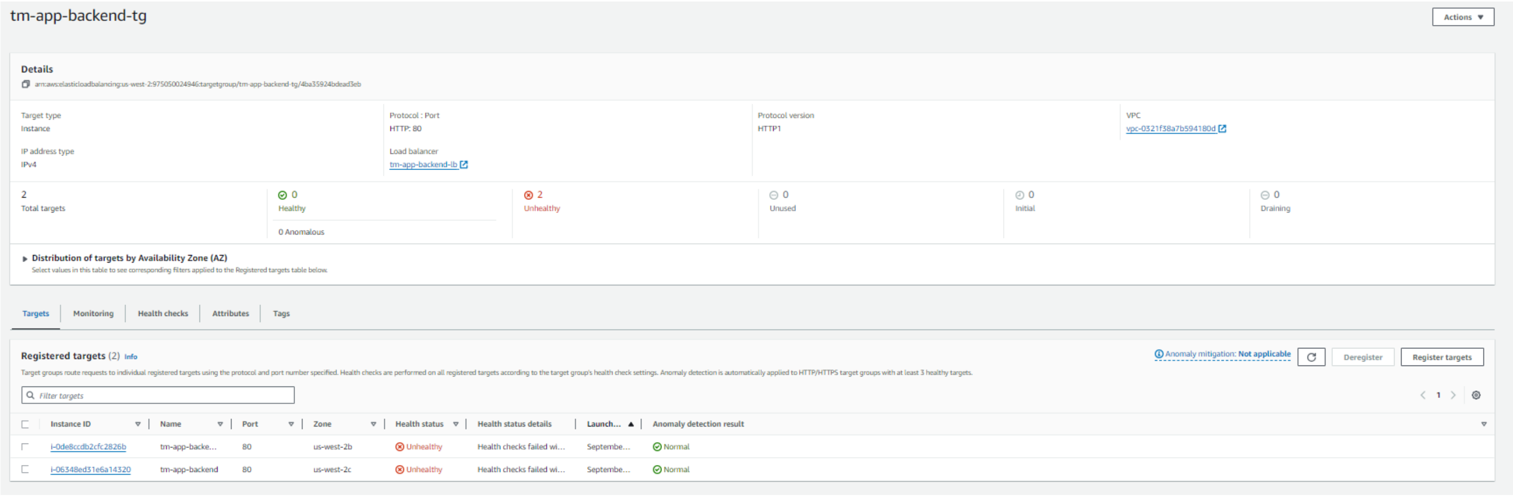
Distribution of targets by Availability Zone (AZ)
Select values in this table to see corresponding filters applied to the Registered targets table below.

Registered targets (2) Info

Target groups route requests to individual registered targets using the protocol and port number specified. Health checks are performed on all registered targets according to the target group's health check settings. Anomaly detection is automatically applied to HTTP/HTTPS target groups with at least 3 healthy targets.

□	Instance ID	Name	Port	Zone	Health status	Health status details	Launch… ▲	Anomaly detection result
□	i-0ebf315ed6bea09b7	tm-app-fronte…	80	us-west-2c	✓ Healthy	-	Septembe…	✓ Normal
□	i-007fa3ca1667d06db	tm-app-frontend	80	us-west-2c	✓ Healthy	-	Septembe…	✓ Normal

  - **TG: tm-app-backend-tg**



tm-app-backend-tg

Details

arn:aws:elasticloadbalancing:us-west-2:975050024946:targetgroup/tm-app-backend-tg/4ba35924bdead3eb

Target type: Instance
IP address type: IPv4
Protocol : Port: HTTP: 80
Load balancer: tm-app-backend-lb
Protocol version: HTTP1
VPC: vpc-0321f38a7b594180d

2 Total targets | 0 Healthy | 0 Anomalous | 2 Unhealthy | 0 Unused | 0 Initial | 0 Draining

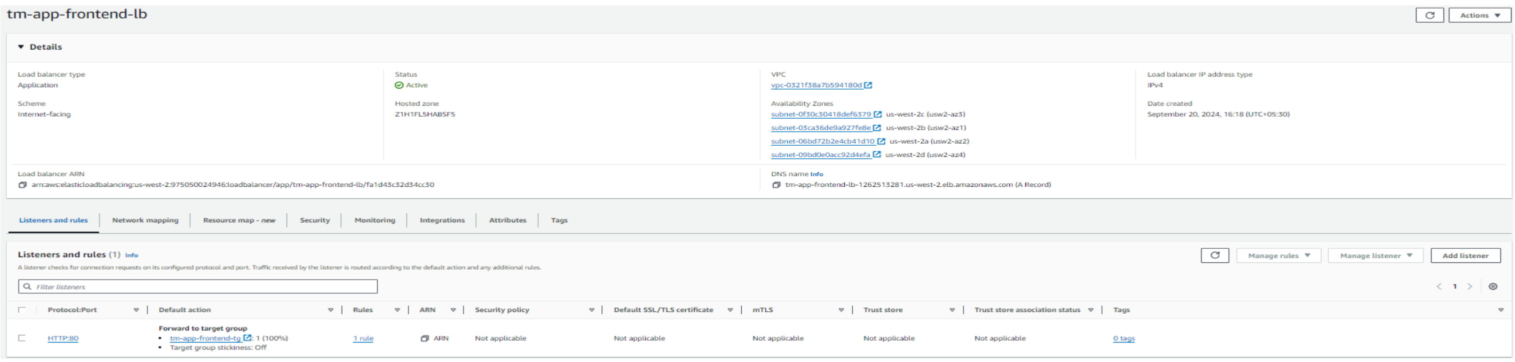
Distribution of targets by Availability Zone (AZ)
Select values in this table to see corresponding filters applied to the Registered targets table below.

Registered targets (2) Info

Target groups route requests to individual registered targets using the protocol and port number specified. Health checks are performed on all registered targets according to the target group's health check settings. Anomaly detection is automatically applied to HTTP/HTTPS target groups with at least 3 healthy targets.

□	Instance ID	Name	Port	Zone	Health status	Health status details	Launch… ▲	Anomaly detection result
□	i-0de8ccdb2cfc2826b	tm-app-backe…	80	us-west-2b	⊗ Unhealthy	Health checks failed wi…	Septembe…	✓ Normal
□	i-06348ed31e6a14320	tm-app-backend	80	us-west-2c	⊗ Unhealthy	Health checks failed wi…	Septembe…	✓ Normal

- Create 2 LB – one for frontend and frontend target group and via-versa for backend
  - LB: tm-app-frontend-lb



tm-app-frontend-lb

Details

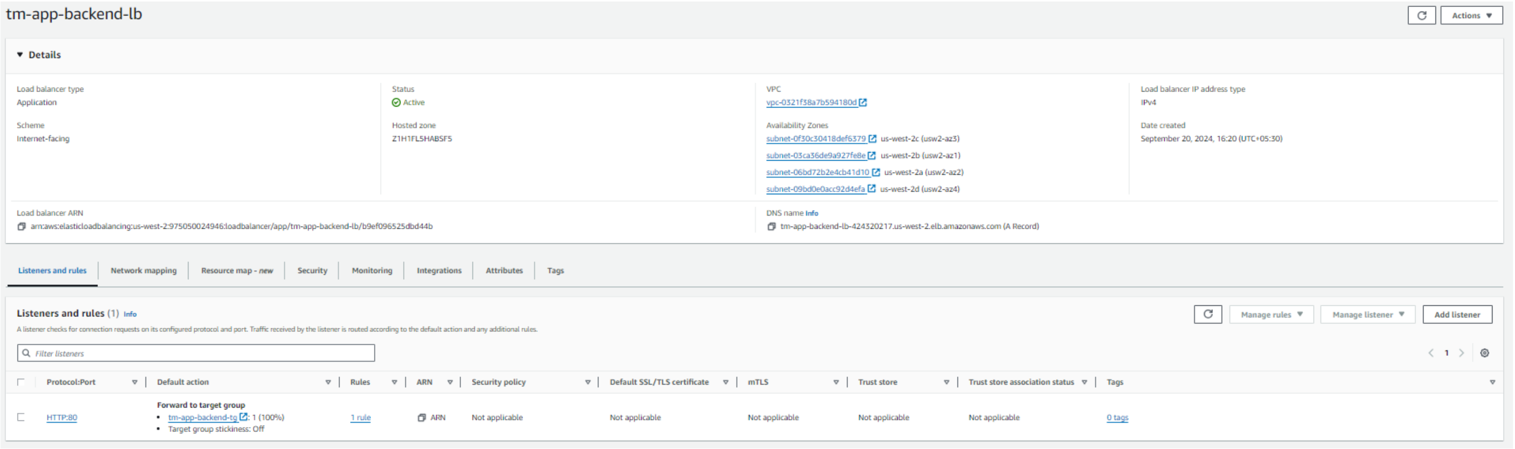
Load balancer type: Application
Scheme: Internet-facing
Status: ✓ Active
Hosted zone: Z1H1FL5HABSF5
VPC: vpc-0321f38a7b594180d
Availability Zones:
subnet-0f30c30418def6379 us-west-2c (usw2-az3)
subnet-03ca36de9a927fe8e us-west-2b (usw2-az1)
subnet-06bd72b2e4cb41d10 us-west-2a (usw2-az2)
subnet-09bd0e0acc92d4efa us-west-2d (usw2-az4)
Load balancer IP address type: IPv4
Date created: September 20, 2024, 16:18 (UTC+05:30)
Load balancer ARN: arn:aws:elasticloadbalancing:us-west-2:975050024946:loadbalancer/app/tm-app-frontend-lb/fa1d43c32d34cc30
DNS name Info: tm-app-frontend-lb-1262513281.us-west-2.elb.amazonaws.com (A Record)

Listeners and rules (1) Info

A listener checks for connection requests on its configured protocol and port. Traffic received by the listener is routed according to the default action and any additional rules.

□	Protocol:Port	Default action	Rules	ARN	Security policy	Default SSL/TLS certificate	mTLS	Trust store	Trust store association status	Tags
□	HTTP:80	Forward to target group • tm-app-frontend-tg: 1 (100%) • Target group stickiness: Off	1 rule	ARN	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable	0 tags

- ○ LB: tm-app-backend-lb



tm-app-backend-lb

▼ Details

Load balancer type Application	Status Active	VPC vpc-0321f38a7b594180d	Load balancer IP address type IPv4
Scheme Internet-facing	Hosted zone Z1H1FL5HABSF5	Availability Zones subnet-0f30c30418def6379 us-west-2c (usw2-az3) subnet-03ca36de9a927fe8e us-west-2b (usw2-az1) subnet-06bd72b2e4cb41d10 us-west-2a (usw2-az2) subnet-09bd0e0acc92d4efa us-west-2d (usw2-az4)	Date created September 20, 2024, 16:20 (UTC+05:30)

Load balancer ARN
arn:aws:elasticloadbalancing:us-west-2:975050024946:loadbalancer/app/tm-app-backend-lb/b9ef096525dbd44b

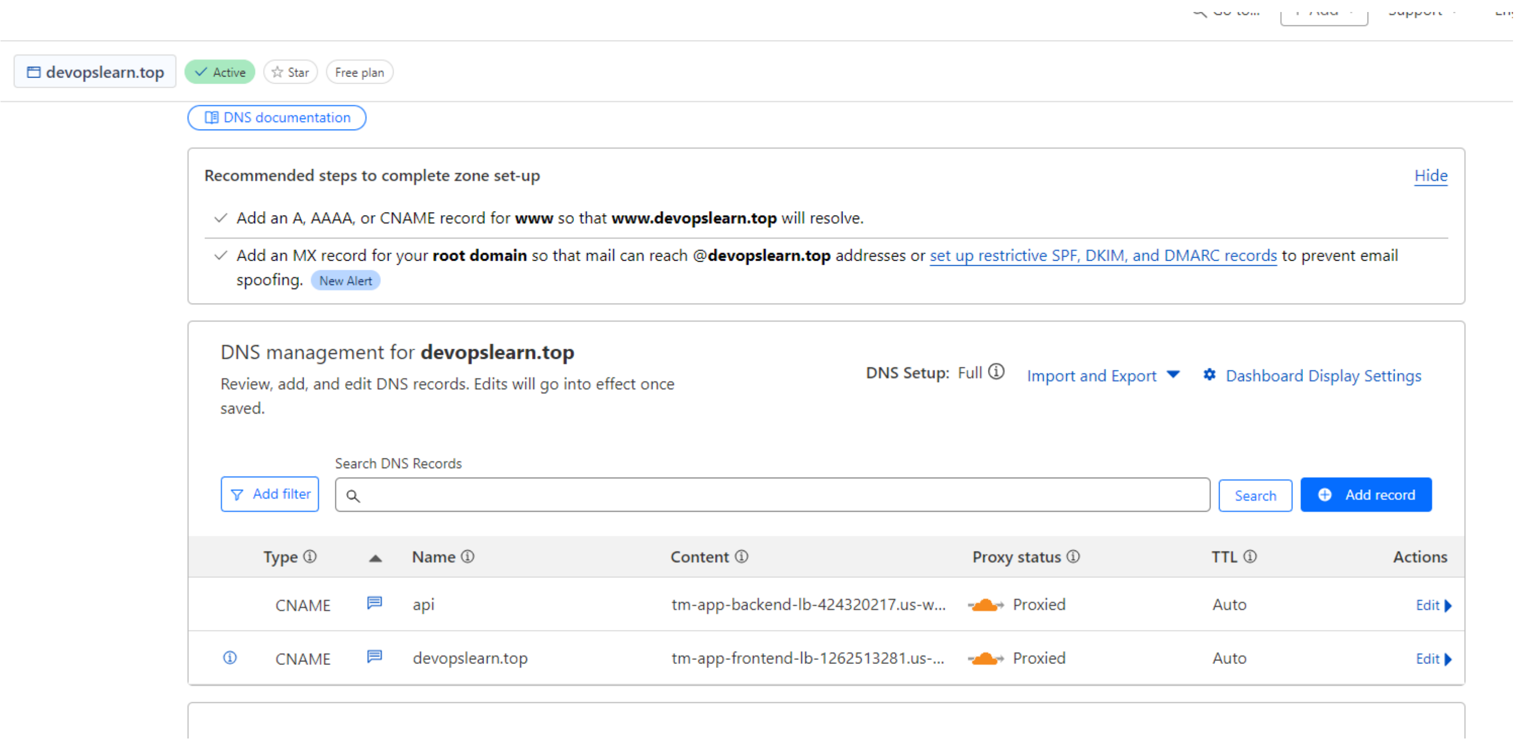
DNS name Info
tm-app-backend-lb-424320217.us-west-2.elb.amazonaws.com (A Record)

Listeners and rules (1) Info

A listener checks for connection requests on its configured protocol and port. Traffic received by the listener is routed according to the default action and any additional rules.

□	Protocol:Port	Default action	Rules	ARN	Security policy	Default SSL/TLS certificate	mTLS	Trust store	Trust store association status	Tags
□	HTTP:80	Forward to target group • tm-app-backend-tg: 1 (100%) • Target group stickiness: Off	1 rule	ARN	Not applicable	Not applicable	Not applicable	Not applicable	Not applicable	0 tags

- ➢ Login into cloudflare www.cloudflare.com and go to DNS records
  - ○ Add CNAME for frontend LB
  - ○ Add CNAME for Backend LB



Recommended steps to complete zone set-up

✓ Add an A, AAAA, or CNAME record for **www** so that **www.devopslearn.top** will resolve.

✓ Add an MX record for your **root domain** so that mail can reach **@devopslearn.top** addresses or set up restrictive SPF, DKIM, and DMARC records to prevent email spoofing. New Alert

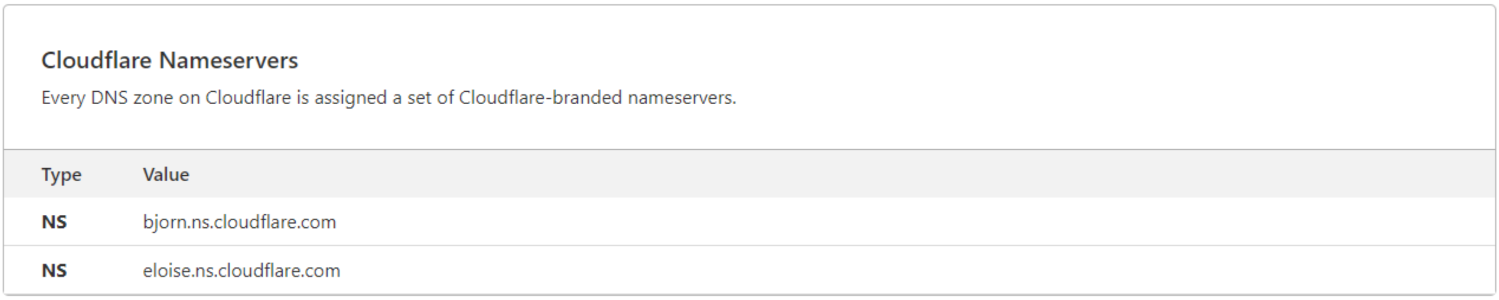
DNS management for **devopslearn.top**

Review, add, and edit DNS records. Edits will go into effect once saved.

DNS Setup: Full

Type	Name	Content	Proxy status	TTL	Actions
CNAME	api	tm-app-backend-lb-424320217.us-w...	Proxied	Auto	Edit ▸
CNAME	devopslearn.top	tm-app-frontend-lb-1262513281.us-...	Proxied	Auto	Edit ▸

- ○ Take Cloudflare Nameservers and map to your domain name servers

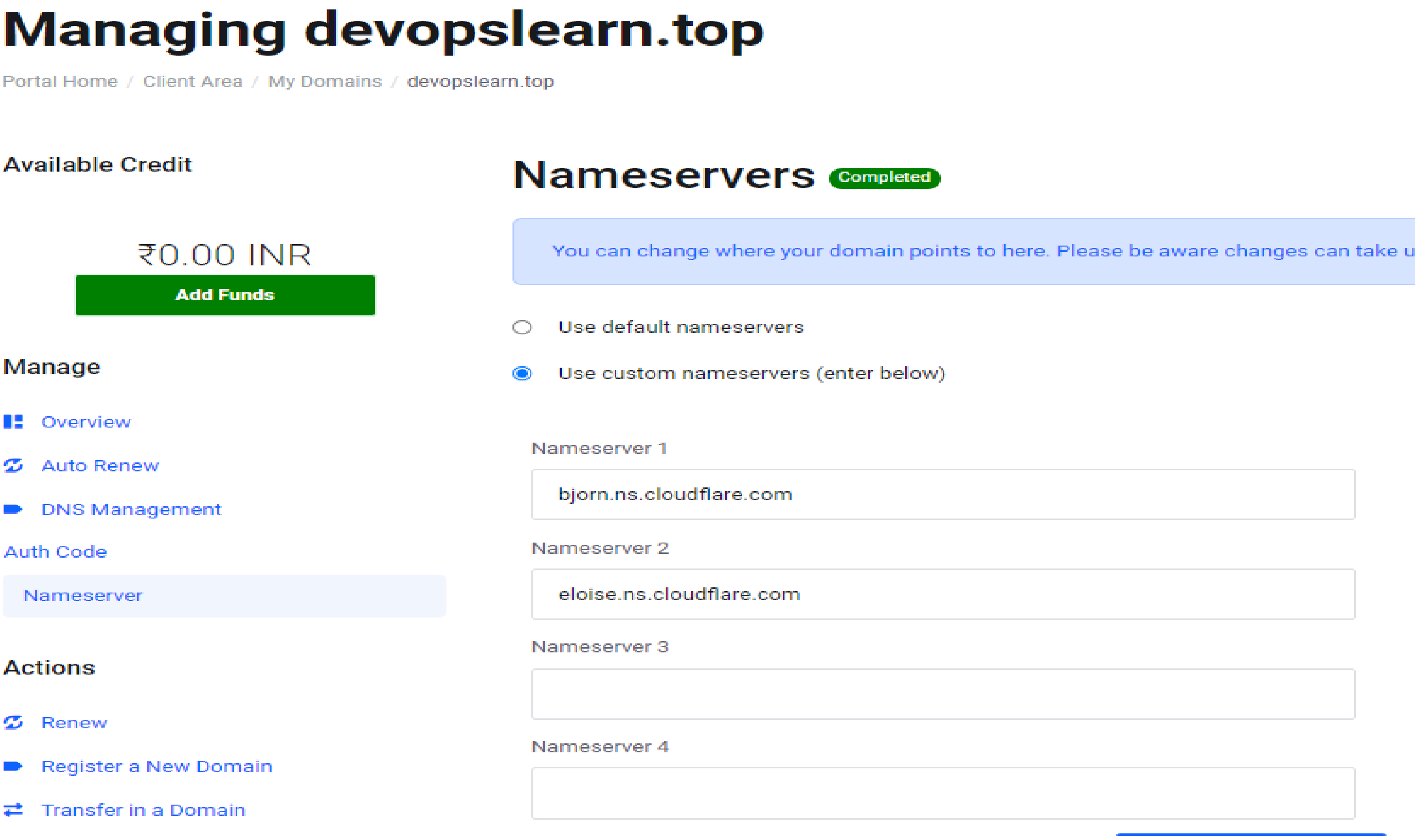


Cloudflare Nameservers

Every DNS zone on Cloudflare is assigned a set of Cloudflare-branded nameservers.

Type	Value
NS	bjorn.ns.cloudflare.com
NS	eloise.ns.cloudflare.com

- ➢ Login to the portal from where you have purchased the domain (in my case https://www.ambitionhost.in/)



Managing devopslearn.top

Portal Home / Client Area / My Domains / devopslearn.top

Available Credit

₹0.00 INR

Add Funds

Manage

- Overview
- Auto Renew
- DNS Management
- Auth Code
- Nameserver

Actions

- Renew
- Register a New Domain
- Transfer in a Domain

Nameservers Completed

You can change where your domain points to here. Please be aware changes can take u

○ Use default nameservers

● Use custom nameservers (enter below)

Nameserver 1

bjorn.ns.cloudflare.com

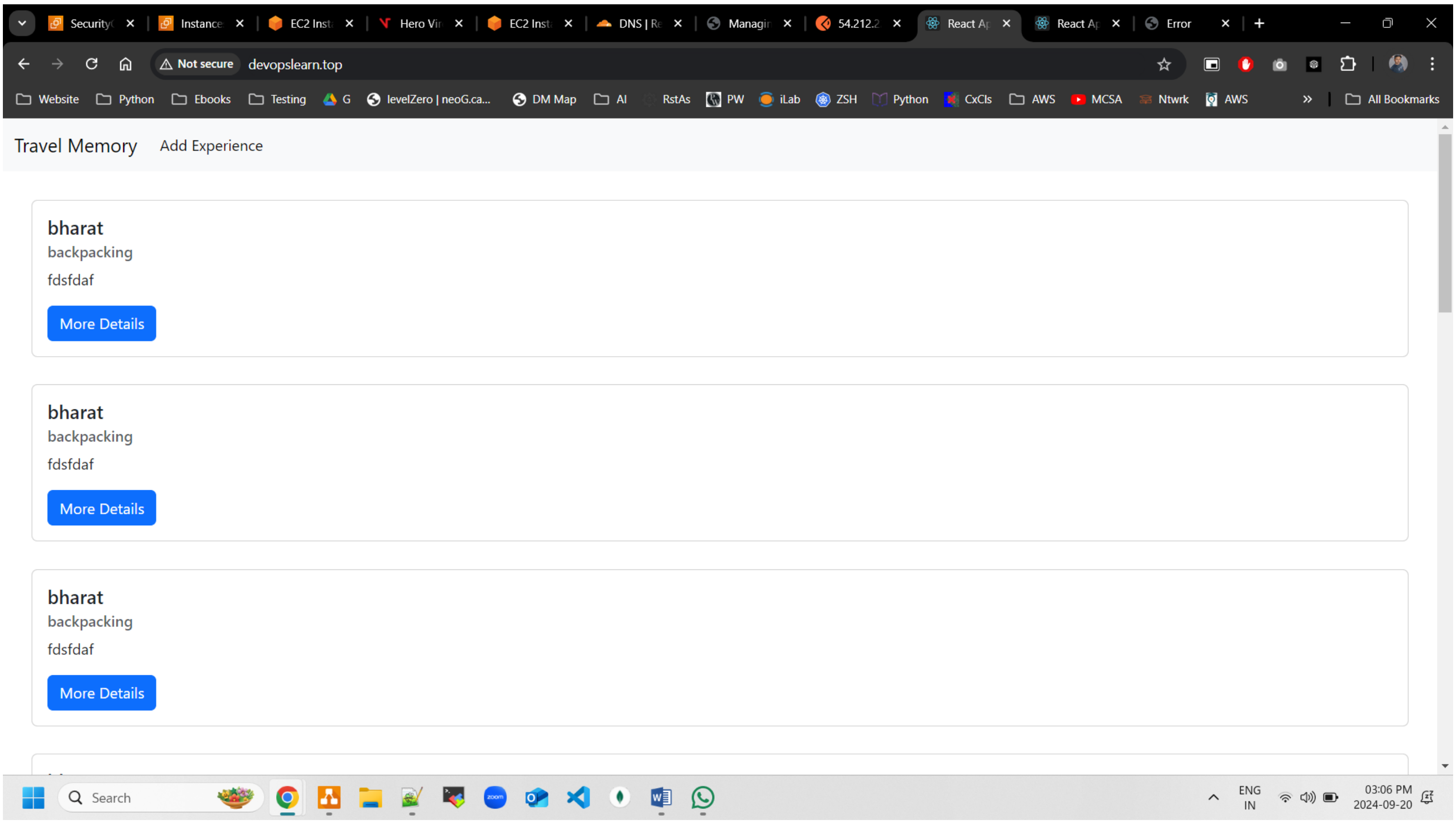
Nameserver 2

eloise.ns.cloudflare.com

Nameserver 3

Nameserver 4

- ➢ Once done then hit your domain name (in my case domain: devopslearn.top)



devopslearn.top

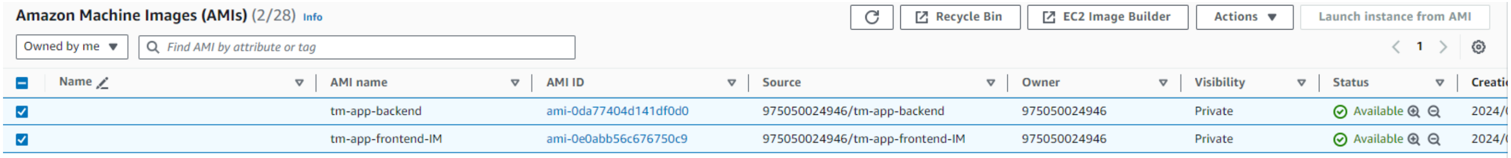
Travel Memory   Add Experience

bharat
backpacking
fdsfdaf
More Details

bharat
backpacking
fdsfdaf
More Details

bharat
backpacking
fdsfdaf
More Details

- Create AIM images for your frontend and backend



	Name	AMI name	AMI ID	Source	Owner	Visibility	Status	Creati
☑		tm-app-backend	ami-0da77404d141df0d0	975050024946/tm-app-backend	975050024946	Private	✓ Available	2024/
☑		tm-app-frontend-IM	ami-0e0abb56c676750c9	975050024946/tm-app-frontend-IM	975050024946	Private	✓ Available	2024/