

NAME : UDDHAV BAPU GUND | Designation: Jr.AWS Developer

TASK : EC2 as Reverse Proxy

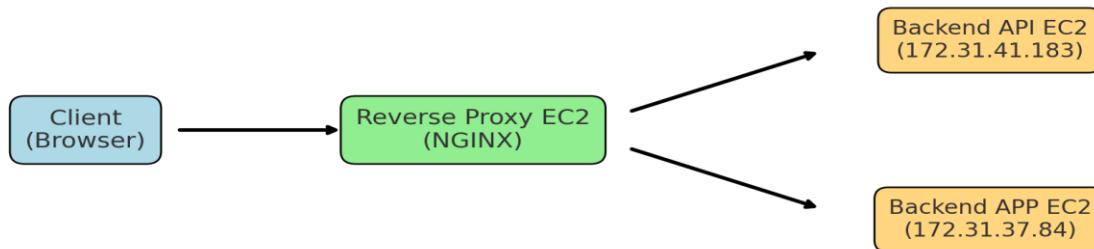
- Launch an EC2 and install NGINX as a reverse proxy.
- Route traffic from /api to one backend EC2, and /app to another EC2.

Objective

The objective of this project is to configure an EC2 instance with NGINX as a reverse proxy, which routes traffic:

- /api → Backend API EC2
- /app → Backend App EC2

Architecture



- Reverse Proxy EC2 → NGINX routes requests.
- Backend API EC2 → Hosts simple Apache HTTP server for /api.
- Backend App EC2 → Hosts simple Apache HTTP server for /app.

Step 1: Launch EC2 Instances

- Created 3 EC2 instances (Amazon Linux 2023, t2.micro).
- ReverseProxy, Backend-API-server, Backend-App.
- Configured security groups: SSH (22), HTTP (80).

Step 2: Configure Backend API Server

Automation with User Data:

To make the backend servers (API and App) self-configuring, I used **EC2 User Data scripts** during instance launch. This ensures that whenever the instances start, Apache HTTP server is automatically installed and configured with the correct response page.

- **Backend API Server (User Data Script):**

```
#!/bin/bash
yum update -y
yum install -y httpd
echo "Hello from API Backend" > /var/www/html/index.html
systemctl start httpd
systemctl enable httpd
```

Step 3: Configure Backend App Server

- **Backend APP Server (User Data Script):**

```
#!/bin/bash
yum update -y
yum install -y httpd
echo "Hello from APP Backend" > /var/www/html/index.html
systemctl start httpd
systemctl enable httpd
```

(1) Chat | AWS Developer | Microservices > Launch an instance | EC2 | us-east-1 > 98.80.246.7

us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#LaunchInstances:

EC2 Instances Launch an Instance

Launch an instance Info

Amazon EC2 allows you to create virtual machines, or instances, that run on the AWS Cloud. Quickly get started by following the simple steps below.

Name and tags Info

Name
Backend-App Add additional tags

Application and OS Images (Amazon Machine Image) Info

An AMI contains the operating system, application server, and applications for your instance. If you don't see a suitable AMI below, use the search field or choose [Browse more AMIs](#).

Search our full catalog including 1000s of application and OS images

Recents **Quick Start**

Amazon Linux macOS Ubuntu Windows Red Hat SUSE Linux Debian

Search Including AMIs from AWS, Marketplace and the Community

Amazon Machine Image (AMI)

Amazon Linux 2023 kernel-6.1 AMI
ami-00ca32bbc84273381 (64-bit (x86), uefi-preferred) / ami-0aa7db6294d00216f (64-bit (Arm), uefi)
Virtualization: hvm ENA enabled: true Root device type: ebs Free tier eligible

Description

CloudShell Feedback

Summary

Number of instances Info
1

Software Image (AMI)
Amazon Linux 2023 AMI 2023.8.2... [read more](#)
ami-00ca32bbc84273381

Virtual server type (instance type)
t2.micro

Firewall (security group)
reverse-proxysg

Storage (volumes)
1 volume(s) - 8 GiB

Free tier: In your first year of opening an AWS account, you get 750 hours per month of t2.micro instance usage (or t3.micro where t2.micro isn't available) when used with free tier AMIs, 750 hours per month of public IPv4 address usage, 30 GiB of EBS storage, 2 million I/Os, 1 GB of snapshots, and 100 GB of bandwidth to the internet. Data transfer charges are not included as part of the free tier allowance.

Cancel **Launch instance** Preview code

© 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

(1) Chat | AWS Developer | Microservices > Launch an instance | EC2 | us-east-1 > 98.80.246.7

us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#LaunchInstances:

EC2 Instances Launch an instance

Amazon Linux 2023 kernel-6.1 AMI
ami-00ca32bbc84273381 (64-bit (x86), uefi-preferred) / ami-0aa7db6294d00216f (64-bit (Arm), uefi)
Virtualization: hvm ENA enabled: true Root device type: ebs Free tier eligible

Description
Amazon Linux 2023 (kernel-6.1) is a modern, general purpose Linux-based OS that comes with 5 years of long term support. It is optimized for AWS and designed to provide a secure, stable and high-performance execution environment to develop and run your cloud applications.

Amazon Linux 2023 AMI 2023.8.20250818.0 x86_64 HVM kernel-6.1

Architecture	Boot mode	AMI ID	Publish Date	Username	Verified provider
64-bit (x86)	uefi-preferred	ami-00ca32bbc84273381	2025-08-13	ec2-user	

Instance type Info | Get advice

t2.micro
Family: t2 1 vCPU 1 GiB Memory Current generation: true On-Demand Windows base pricing: 0.0162 USD per Hour
On-Demand Ubuntu Pro base pricing: 0.0134 USD per Hour On-Demand SUSE base pricing: 0.0116 USD per Hour
On-Demand RHEL base pricing: 0.026 USD per Hour On-Demand Linux base pricing: 0.0116 USD per Hour Free tier eligible

All generations [Compare instance types](#)

Additional costs apply for AMIs with pre-installed software

Key pair (login) Info

You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.

Key pair name - required
webkey Create new key pair

Summary

Number of instances Info
1

Software Image (AMI)
Amazon Linux 2023 AMI 2023.8.2... [read more](#)
ami-00ca32bbc84273381

Virtual server type (instance type)
t2.micro

Firewall (security group)
reverse-proxysg

Storage (volumes)
1 volume(s) - 8 GiB

Free tier: In your first year of opening an AWS account, you get 750 hours per month of t2.micro instance usage (or t3.micro where t2.micro isn't available) when used with free tier AMIs, 750 hours per month of public IPv4 address usage, 30 GiB of EBS storage, 2 million I/Os, 1 GB of snapshots, and 100 GB of bandwidth to the internet. Data transfer charges are not included as part of the free tier allowance.

Cancel **Launch instance** Preview code

© 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

(1) Chat | AWS Developer | Micro X Launch an instance | EC2 | us-east-1 X 98.80.246.7 X +

us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#LaunchInstances:

SWS Search [Alt+S] All Bookmarks United States (N. Virginia) Account ID: 2534-9079-5695 Uddhav Gund

EC2 Instances Launch an instance

Network settings

- VPC - required [Info](#)
 - vpc-0610fd9fedfd36a33 (default)
 - 172.31.0.0/16
- Subnet [Info](#)
 - subnet-056c26216cd0f429 VPC: vpc-0610fd9fedfd36a33 Owner: 253490795695 Availability Zone: us-east-1a (use1-az6) Zone type: Availability Zone IP addresses available: 4090 CIDR: 172.31.32.0/20
- Auto-assign public IP [Info](#)
 - Enable Additional charges apply when outside of free tier allowance
- Firewall (security groups) [Info](#)
 - A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.
 - Create security group Select existing security group
- Common security groups [Info](#)
 - Select security groups
 - reverse-proxysg sg-03a5c5eb5fe53bf4 VPC: vpc-0610fd9fedfd36a33
- Advanced network configuration

Configure storage [Info](#) Advanced

- 1x 8 GiB gp3 Root volume, 3000 IOPS, Not encrypted

Summary

Number of instances [Info](#)

1

Software Image (AMI) Amazon Linux 2023 AMI 2023.8.2...[read more](#) ami-00ca52bbc84273381

Virtual server type (instance type) t2.micro

Firewall (security group) reverse-proxysg

Storage (volumes) 1 volume(s) - 8 GiB

Free tier: In your first year of opening an AWS account, you get 750 hours per month of t2.micro instance usage (or t3.micro where t2.micro isn't available) when used with free tier AMIs, 750 hours per month of public IPv4 address usage, 30 GiB of EBS storage, 2 million I/Os, 1 GB of snapshots, and 100 GB of bandwidth to the internet. Data transfer charges are not included as part of the free tier allowance.

Cancel [Launch instance](#) [Preview code](#)

© 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

(1) Chat | AWS Developer | Micro X Launch an instance | EC2 | us-east-1 X 98.80.246.7 X +

us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#LaunchInstances:

SWS Search [Alt+S] All Bookmarks United States (N. Virginia) Account ID: 2534-9079-5695 Uddhav Gund

EC2 Instances Launch an instance

Metadata response hop limit [Info](#)

2

Allow tags in metadata [Info](#)

Select

User data - optional [Info](#)

Upload a file with your user data or enter it in the field.

```
#!/bin/bash
sudo yum update -y
sudo yum install -y httpd
echo "<h1>Hello from APP Backend</h1>" | sudo tee /var/www/html/index.html
sudo systemctl start httpd
sudo systemctl enable httpd
```

User data has already been base64 encoded

Summary

Number of instances [Info](#)

1

Software Image (AMI) Amazon Linux 2023 AMI 2023.8.2...[read more](#) ami-00ca52bbc84273381

Virtual server type (instance type) t2.micro

Firewall (security group) reverse-proxysg

Storage (volumes) 1 volume(s) - 8 GiB

Free tier: In your first year of opening an AWS account, you get 750 hours per month of t2.micro instance usage (or t3.micro where t2.micro isn't available) when used with free tier AMIs, 750 hours per month of public IPv4 address usage, 30 GiB of EBS storage, 2 million I/Os, 1 GB of snapshots, and 100 GB of bandwidth to the internet. Data transfer charges are not included as part of the free tier allowance.

Cancel [Launch instance](#) [Preview code](#)

© 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Security Group Specification :

The screenshot shows the AWS EC2 Security Groups page. On the left, a sidebar lists various EC2 services: Dashboard, EC2 Global View, Events, Instances (selected), 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, Network & Security (selected), Security Groups, Elastic IPs, Placement Groups. The main content area displays the details for a security group named 'sg-03a5c5eb3fef53bf4 - reverse-proxysg'. The 'Details' section includes the security group name ('reverse-proxysg'), ID ('sg-03a5c5eb3fef53bf4'), owner ('253490795695'), inbound rules count (3), and outbound rules count (1). The 'Inbound rules' tab is selected, showing three entries:

Name	Security group rule ID	IP version	Type	Protocol	Port range	Source	Description
-	sgr-07f382f01c46a9bcf	IPv4	HTTP	TCP	80	0.0.0.0/0	-
-	sgr-0acc05f43f2aa657f	IPv4	HTTPS	TCP	443	0.0.0.0/0	-
-	sgr-07153753672d53c3b	IPv4	SSH	TCP	22	0.0.0.0/0	-

Three instances:

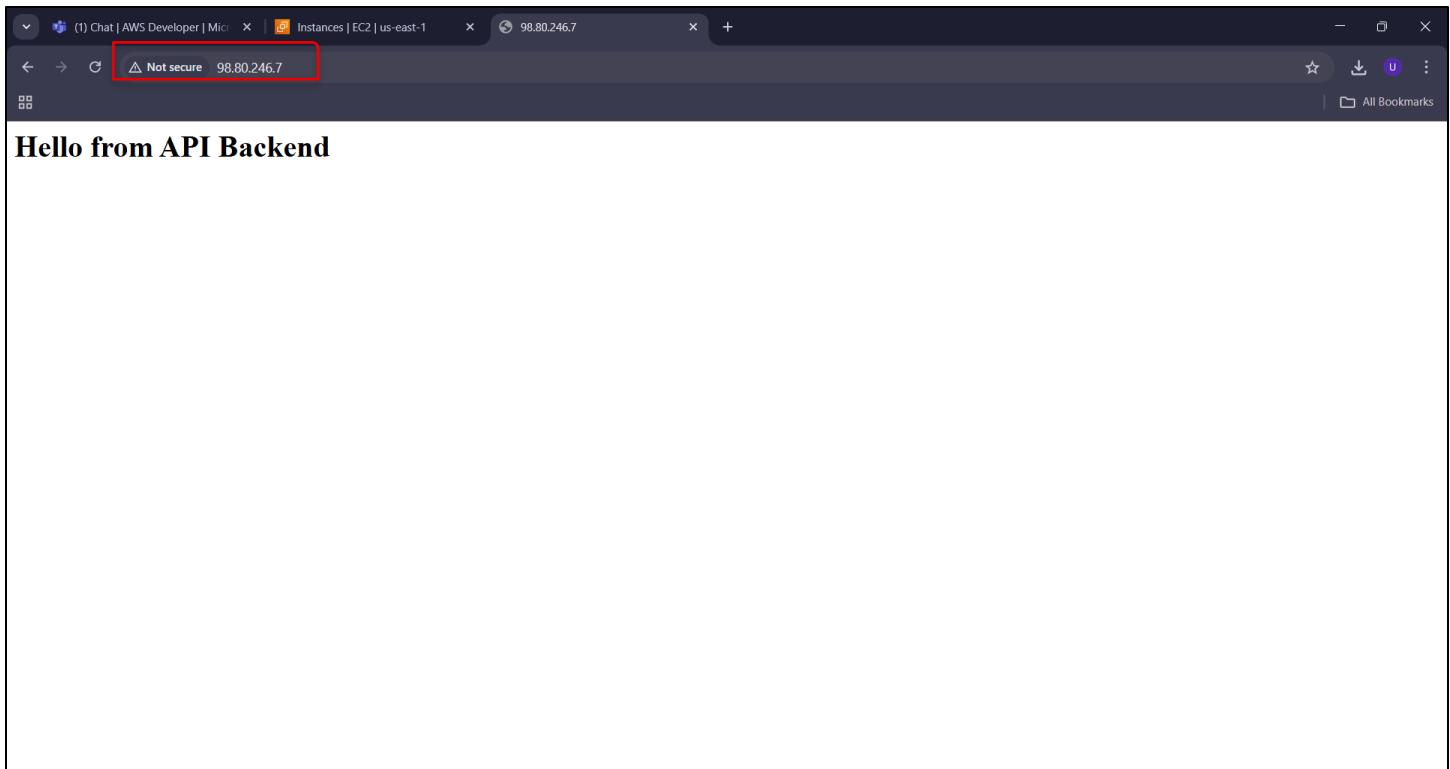
The screenshot shows the AWS EC2 Instances page. The sidebar is identical to the previous screenshot. The main content area displays a table of instances:

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IPv4 IP
Backend-API-server	i-0dc40201d78d141cf	Running	t2.micro	2/2 checks passed	View alarms +	us-east-1a	ec2-98-80-246-7.comp...	98.80.246.7
ReverseProxy	i-0c32d55f051a7c7cc	Running	t2.micro	2/2 checks passed	View alarms +	us-east-1a	ec2-13-216-242-110.co...	13.216.242.110
Backend-App	i-0655e8767b51adb30	Running	t2.micro	2/2 checks passed	View alarms +	us-east-1a	ec2-13-220-216-100.co...	13.220.216.100

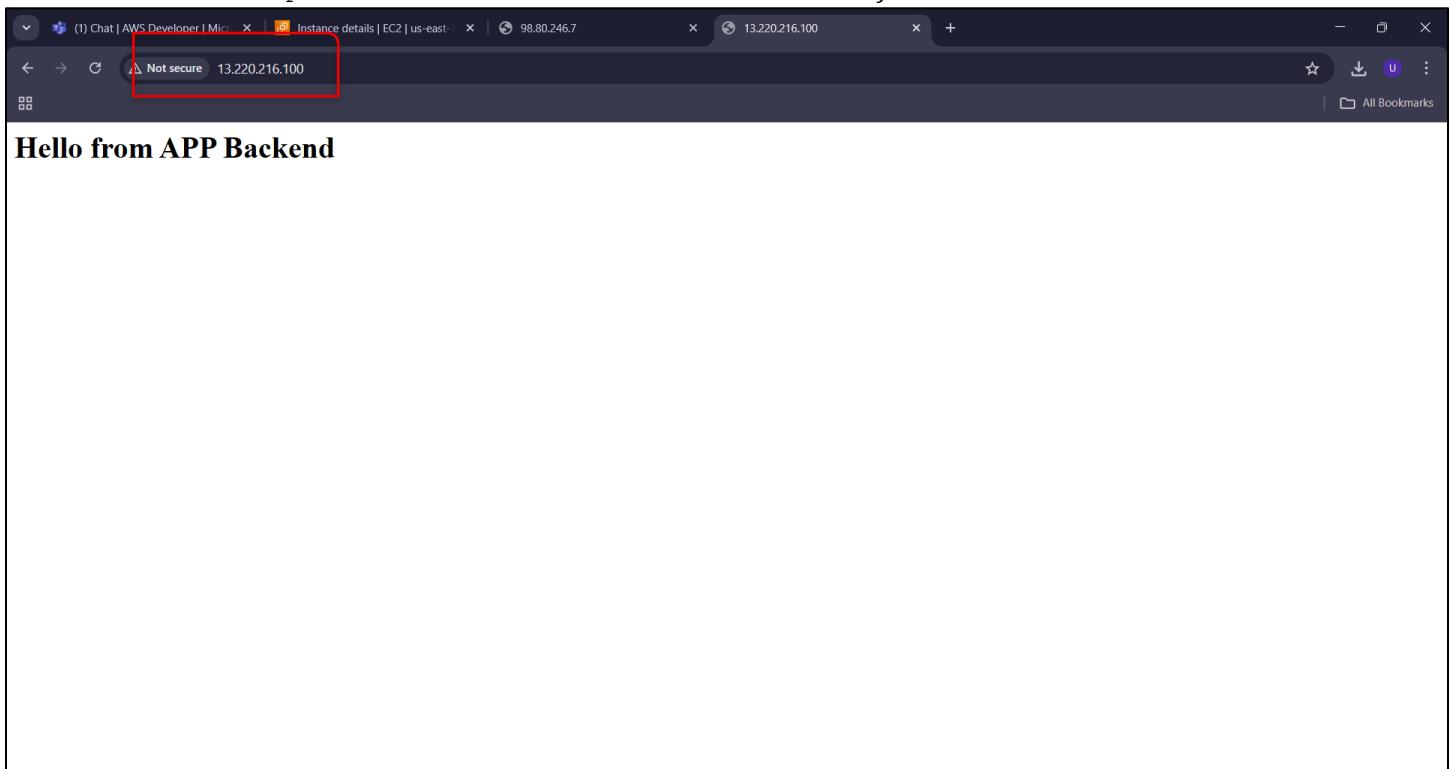
A red box highlights the first three rows of the table. Below the table, it says '3 instances selected'. At the bottom, there are tabs for Monitoring, CloudWatch Metrics, CloudWatch Logs, and CloudWatch Metrics Insights.

Testing for Backend-API-server instance and Backend-App EC2:

Test in browser → `http://<Backend-API-server-Public-IP>` shows *Hello from API Backend.*



Test in browser → `http://<Backend-APP-Public-IP>` shows *Hello from API Backend*.



Step 4: Install NGINX on Reverse Proxy

Manual Configuration of Reverse Proxy

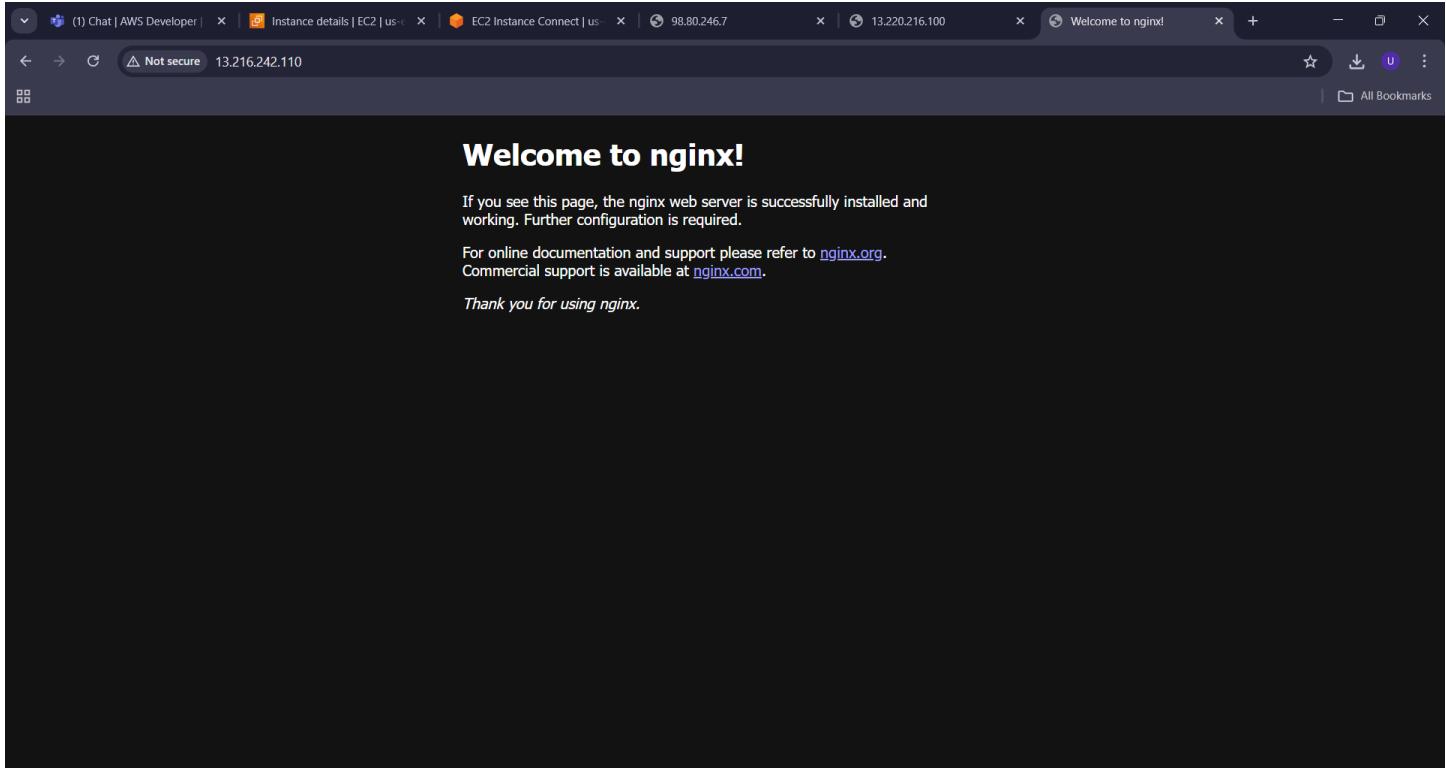
For the Reverse Proxy EC2, I connected via **EC2 Instance Connect** and manually installed and configured **NGINX**.

- Installed NGINX using dnf.
- Edited /etc/nginx/nginx.conf to add reverse proxy rules:

Commands:

```
sudo dnf install -y nginx
sudo systemctl enable nginx
sudo systemctl start nginx
```

Test in browser → `http://<ReverseProxy-Public-IP>` shows *Welcome to NGINX*.



Step 5: Configure NGINX as Reverse Proxy

Edited config file:

```
sudo nano /etc/nginx/nginx.conf
```

- Edited `/etc/nginx/nginx.conf` to add reverse proxy rules:

```
location /api/ {
    proxy_pass http://172.31.41.183;
}

location /app/ {
    proxy_pass http://172.31.37.84;
}
```

Test config and restart:

```
sudo nginx -t
sudo systemctl restart nginx
```

All commands history which used in this task for ngnix:

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

Last login: Wed Sep 3 09:21:29 2025 from 18.206.107.29
[ec2-user@ip-172-31-44-102 ~]$ HISTORY
[hash: HISTORY: command not found
[ec2-user@ip-172-31-44-102 ~]$ history
1 sudo yum update -y
2 sudo amazon-linux-extras enable nginx1
3 sudo dnf install -y nginx
4 sudo systemctl enable nginx
5 sudo systemctl start nginx
6 systemctl status nginx
7 sudo nano /etc/nginx/nginx.conf
8 sudo nginx
9 sudo nano /etc/nginx/nginx.conf
10 sudo systemctl restart nginx
11 sudo nginx -t
12 sudo systemctl restart nginx
13 sudo systemctl status nginx
14 HISTORY
15 history
[ec2-user@ip-172-31-44-102 ~]$
```

i-0c32d55f051a7c7cc (ReverseProxy)

PublicIPs: 13.216.242.110 PrivateIPs: 172.31.44.102

© 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

```
GNU nano 8.3
/etc/nginx/nginx.conf

tcp_nopush      on;
keepalive_timeout 65;
types_hash_max_size 4096;

include          /etc/nginx/mime.types;
default_type     application/octet-stream;

# Load modular configuration files from the /etc/nginx/conf.d directory.
# See http://nginx.org/en/docs/ngx_core_module.html#include
# for more information.

server {
    listen        80;
    listen        [::]:80;
    server_name  _;

    # Reverse proxy rules
    location /api/ {
        proxy_pass http://172.31.41.183/;
    }

    location /app/ {
        proxy_pass http://172.31.37.84/;
    }

    # Default root
    root          /usr/share/nginx/html;

    # Load configuration files for the default server block.
    include /etc/nginx/default.d/*.conf;

    error_page 404 /404.html;
    location = /404.html {
```

i-0c32d55f051a7c7cc (ReverseProxy)

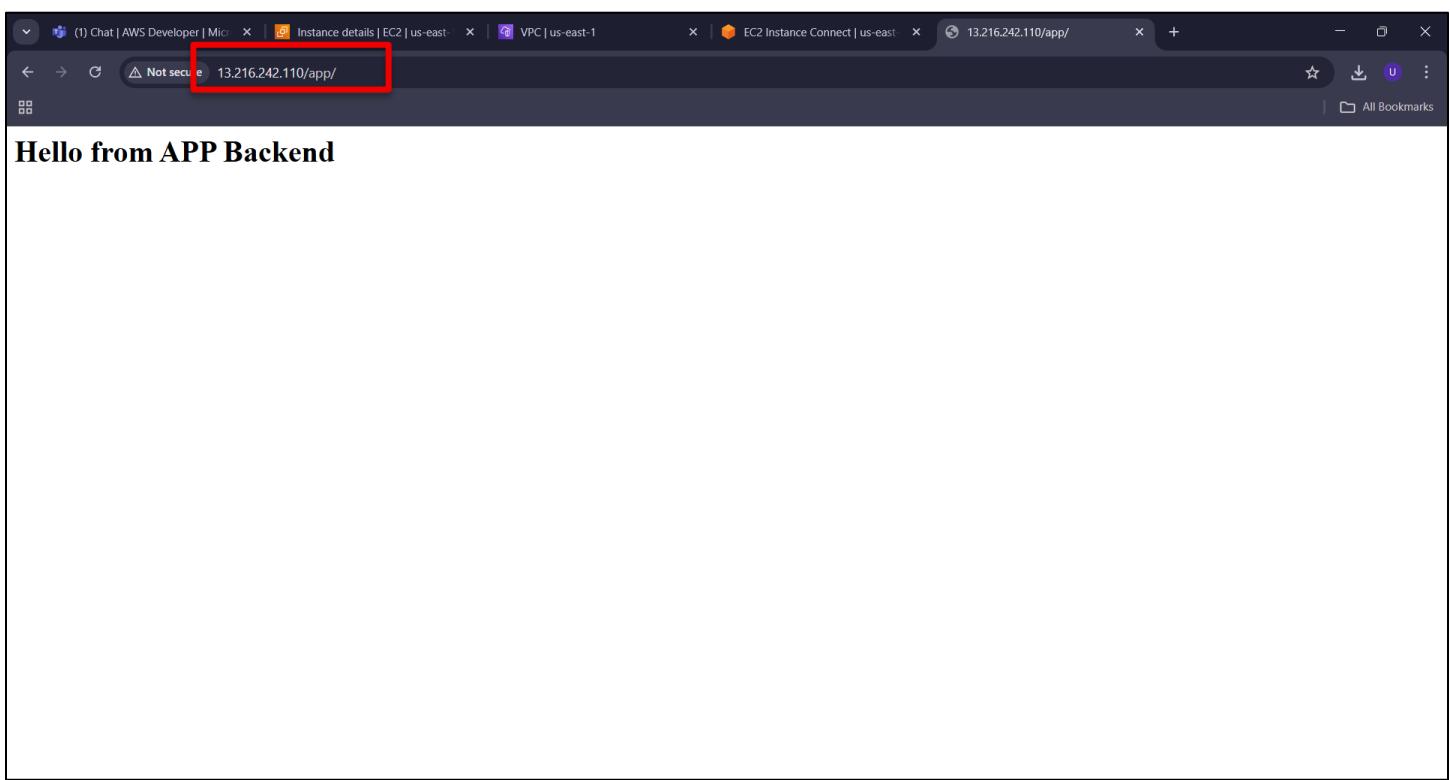
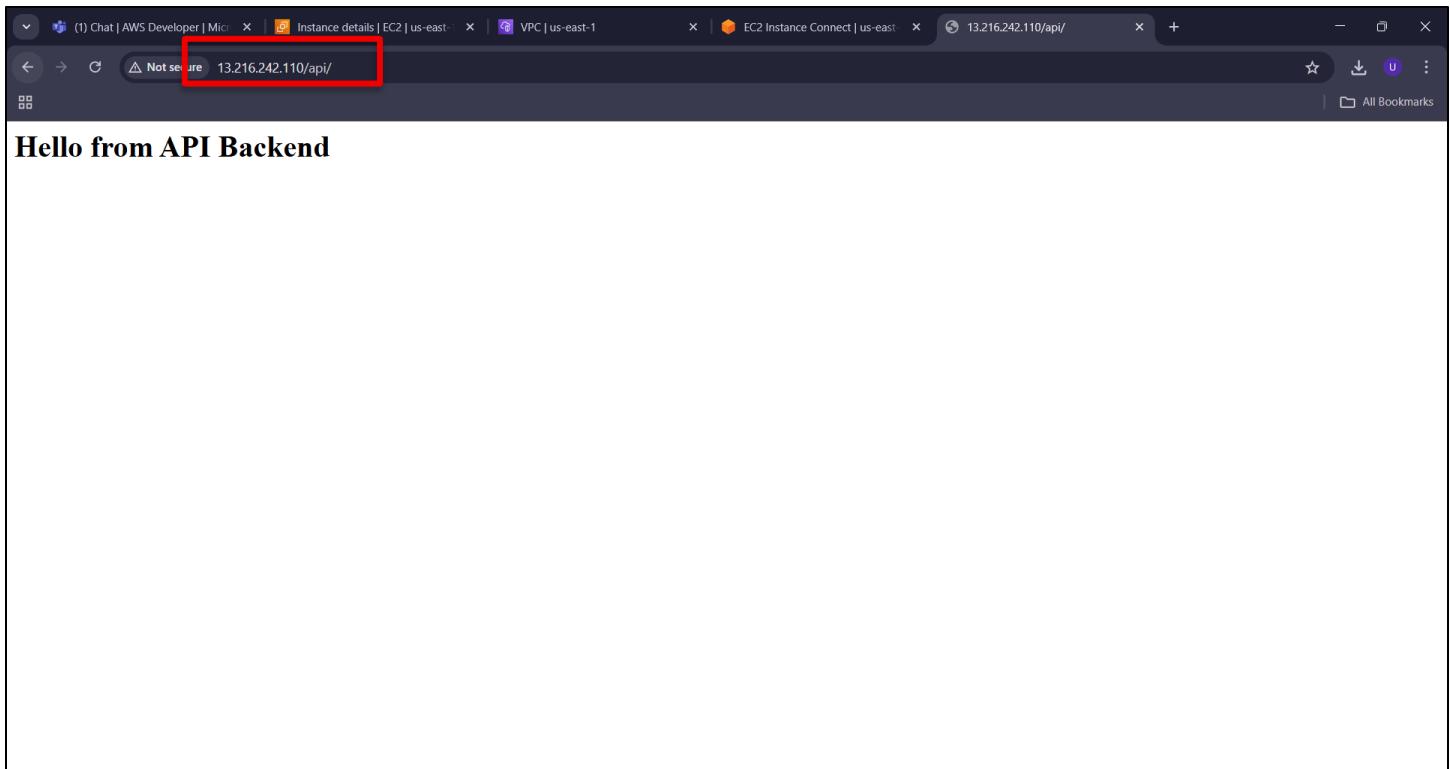
PublicIPs: 13.216.242.110 PrivateIPs: 172.31.44.102

© 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Step 6: Testing

From browser:

- <http://<ReverseProxy-Public-IP>/api> → Hello from API Backend
- <http://<ReverseProxy-Public-IP>/app> → Hello from APP Backend



Results

- Successfully routed /api to Backend-API EC2.
- Successfully routed /app to Backend-App EC2.

Conclusion

This project demonstrates how AWS EC2 and NGINX can be used to build a reverse proxy solution to manage traffic routing to multiple backend servers.