AWS Secure Self-Hosted Collaboration Platform.

Mattermost + MySQL on AWS (Custom VPC Architecture)

By

Esmeralda C. Cabrera Ventura

Project Overview

Many organizations rely on real-time collaboration tools such as Slack or Microsoft Teams. However, regulated environments (government, defense, healthcare, finance) often cannot use third-party managed SaaS platforms due to compliance, data residency, or security policies.

This project demonstrates how to design, deploy, and validate a secure, self-hosted team communication platform on AWS, using open-source software and standard cloud networking best practices. The final solution deploys Mattermost (an open-source Slack alternative) backed by MySQL, hosted entirely inside a custom AWS VPC with proper network isolation, routing, and security controls.

Problem Statement

The challenge was to design an architecture that:

• Allows public user access to a collaboration application

• Keeps the database fully private and inaccessible from the internet

• Supports secure administrative access for setup and maintenance

• Uses AWS-native networking components (VPC, subnets, route tables, gateways)

• Mimics a real-world enterprise deployment, not a simplified demo

Solution Summary

The solution uses a two-tier architecture inside a custom VPC:

• **Public subnet** o Application Server (Mattermost)

- Internet Gateway
- NAT Gateway

• **Private subnet** o Database Server (MySQL)

Administrative access to the private database server is performed indirectly via a bastion host pattern, and outbound internet access from the private subnet is handled securely through a NAT Gateway.

High-Level Architecture

Key Components

• Custom VPC (10.0.0.0/16)

• Public Subnet (10.0.1.0/24)

• Private Subnet (10.0.2.0/24)

• Internet Gateway

• NAT Gateway with Elastic IP

• Public Route Table (IGW routing)

• Private Route Table (NAT routing)

• EC2 Application Server (Mattermost)

• EC2 Database Server (MySQL)

Traffic Flow

1. Users access Mattermost via http://<public-ip>:8065

2. Application server communicates with MySQL over the private subnet

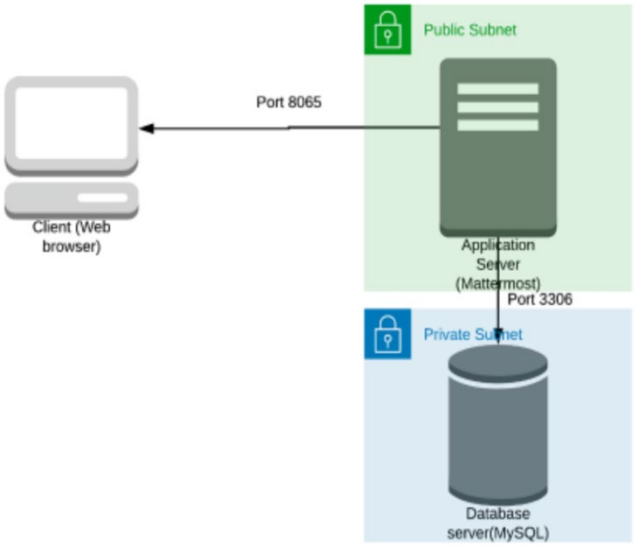
3. Database server has no direct internet exposure

4. Outbound updates from the private subnet flow through the NAT Gateway

Architectural Design

Architecture Design Rationale Design Decision	Reason
Public/Private Subnet Separation	Enforces network isolation and least privilege
NAT Gateway	Allows private resources outbound access without inbound exposure
Bastion Host Access	Prevents direct SSH access to private resources
Custom Route Tables	Explicitly control traffic flow
Security Groups	Enforce port-level access control
Self-Hosted Stack	Full control over data and compliance posture

Architecture Diagram



Implementation

Each section below corresponds directly to the screenshots included in the /screenshots directory and follows the original execution order.

Step 1: Custom VPC and Subnet Creation

Objective: Establish isolated networking boundaries.

• Created a custom VPC with CIDR 10.0.0.0/16

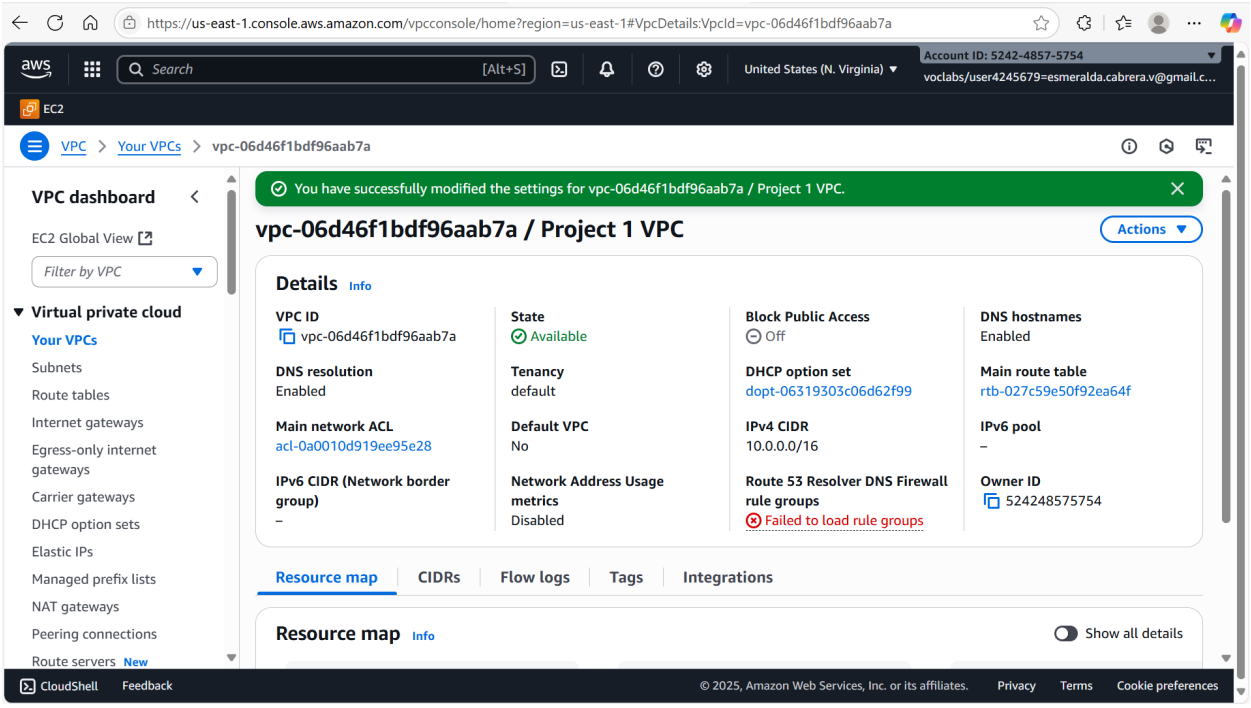
• Enabled DNS hostnames for EC2 resolution

• Created:

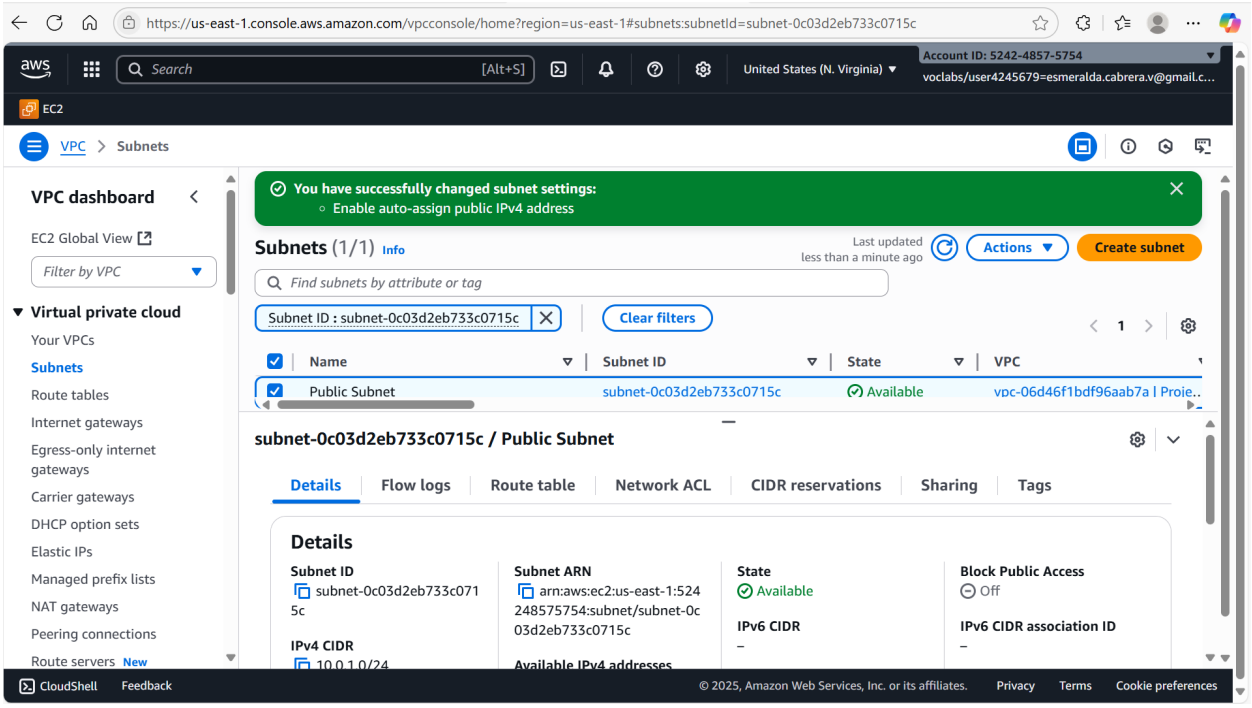
1. Public Subnet (10.0.1.0/24) with auto-assigned public Ips

2. Private Subnet (10.0.2.0/24) with no public IP assignment.

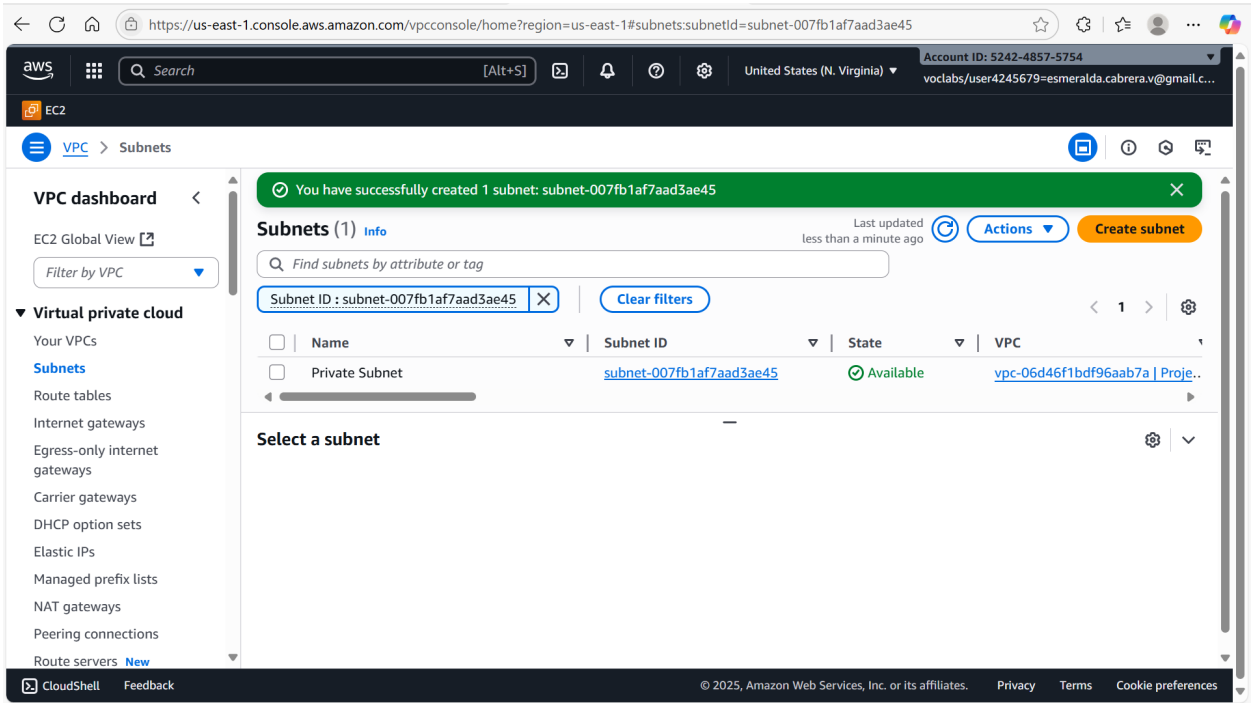
VPC Creation



Public Subnet Creation



Private Subnet Creation



Step 2: Internet Gateway, NAT Gateway, and Routing

Objective: Enable controlled internet connectivity.

• Attached an Internet Gateway to the VPC

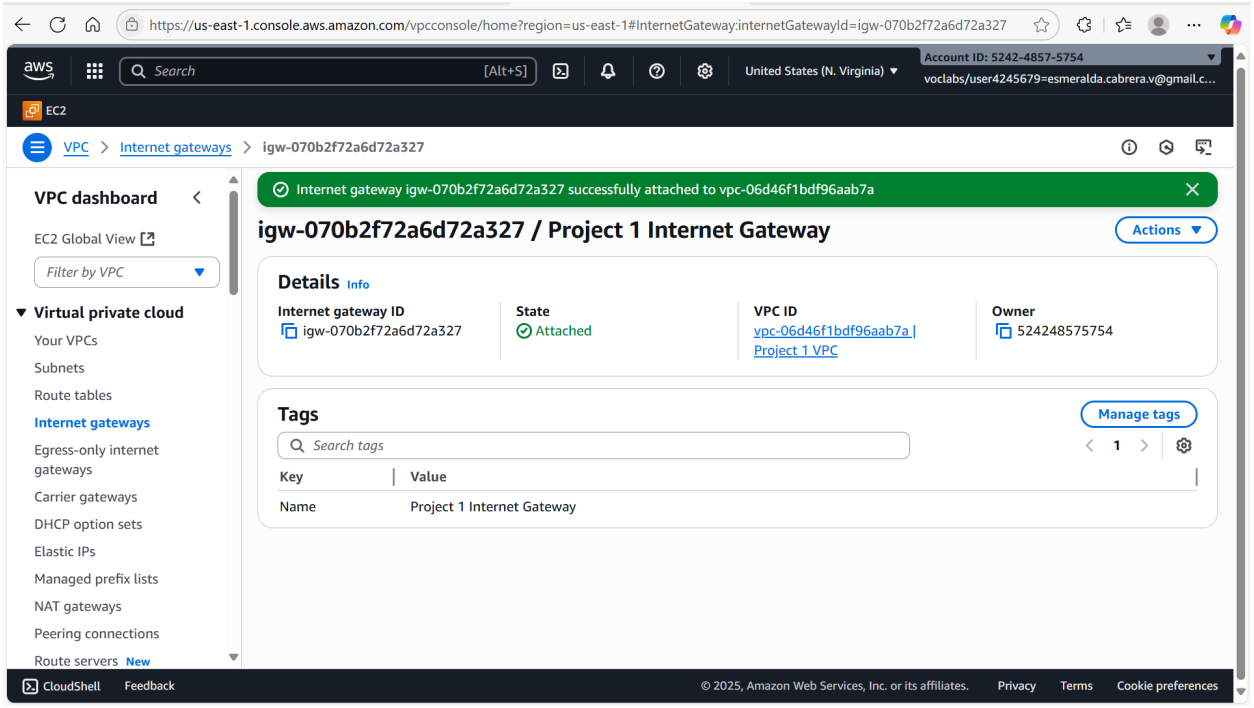
• Created:

1. Public Route Table → 0.0.0.0/0 → Internet Gateway
2. Private Route Table → 0.0.0.0/0 → NAT Gateway

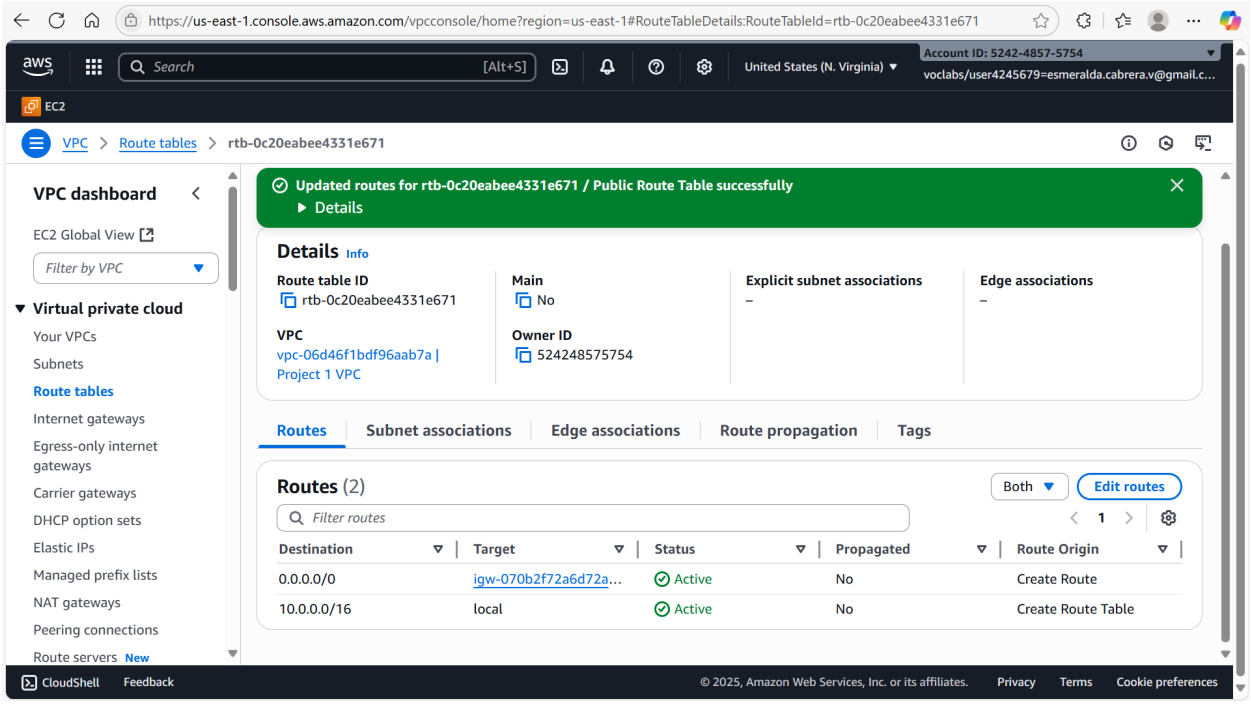
• Deployed a NAT Gateway in the public subnet with an Elastic IP

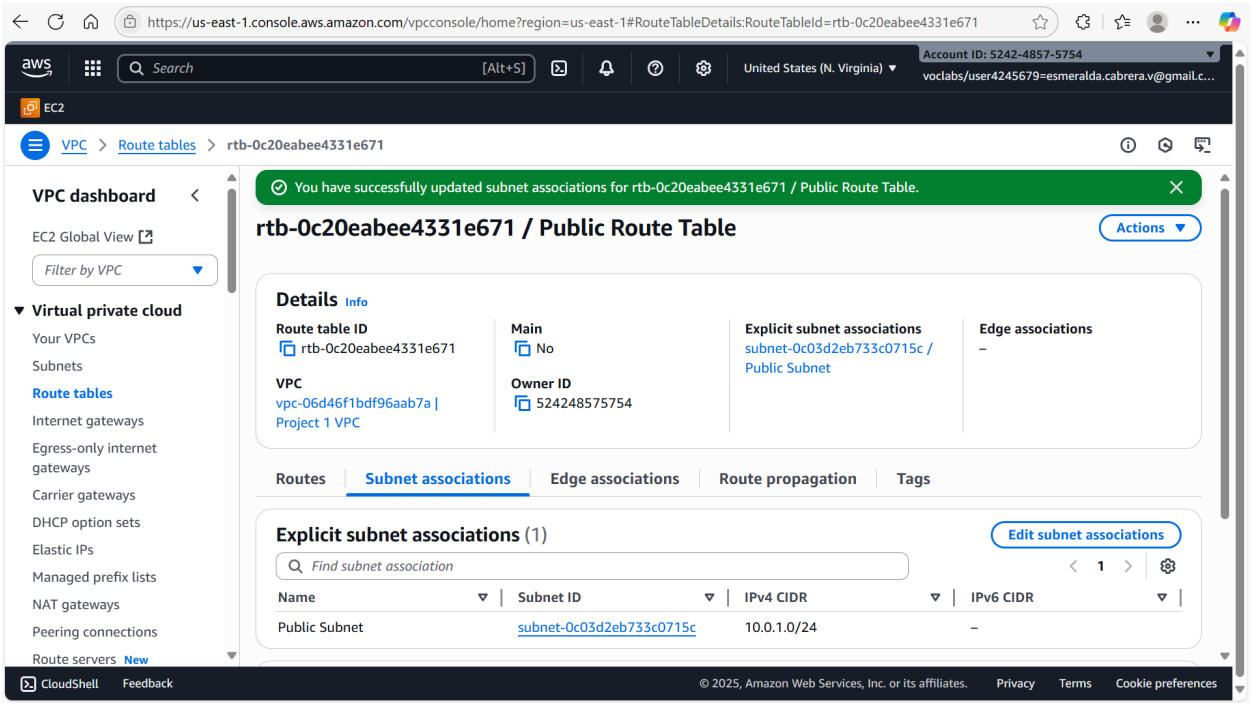
• Associated route tables with their respective subnets

Creation and Configuration of Internet Gateway

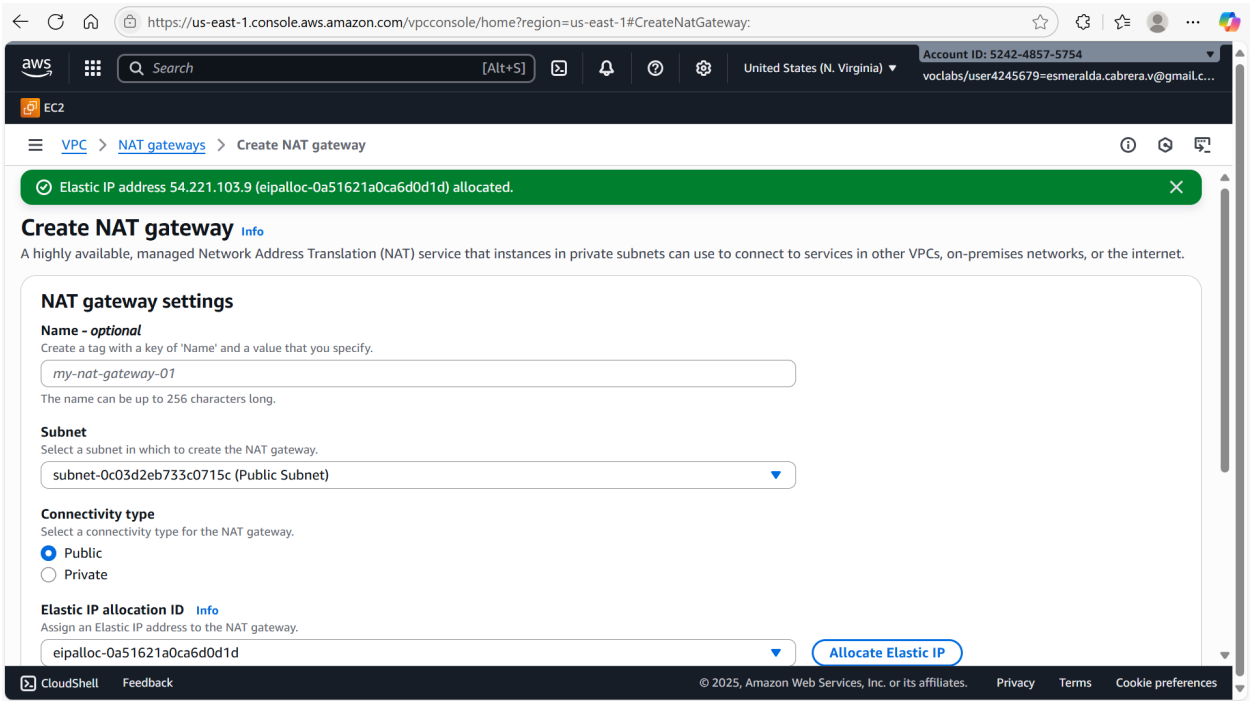


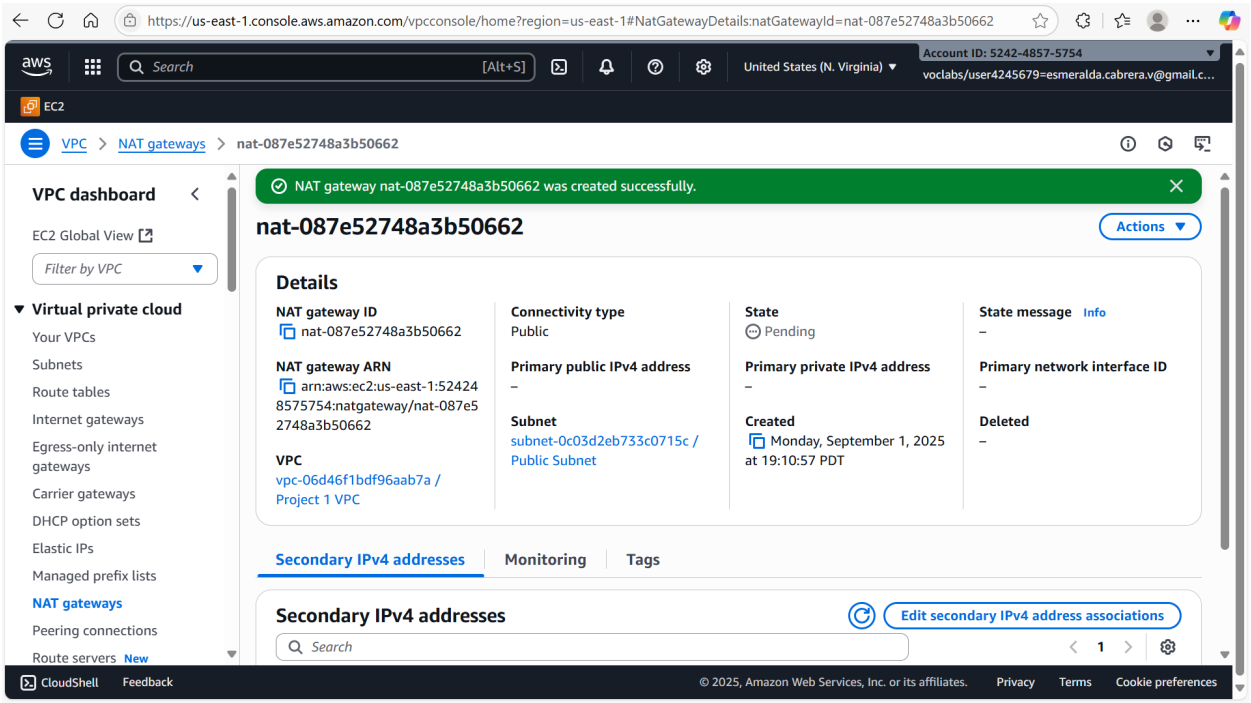
Creation of public route table





Creation of NAT gateway





Step 3: EC2 Instance Deployment

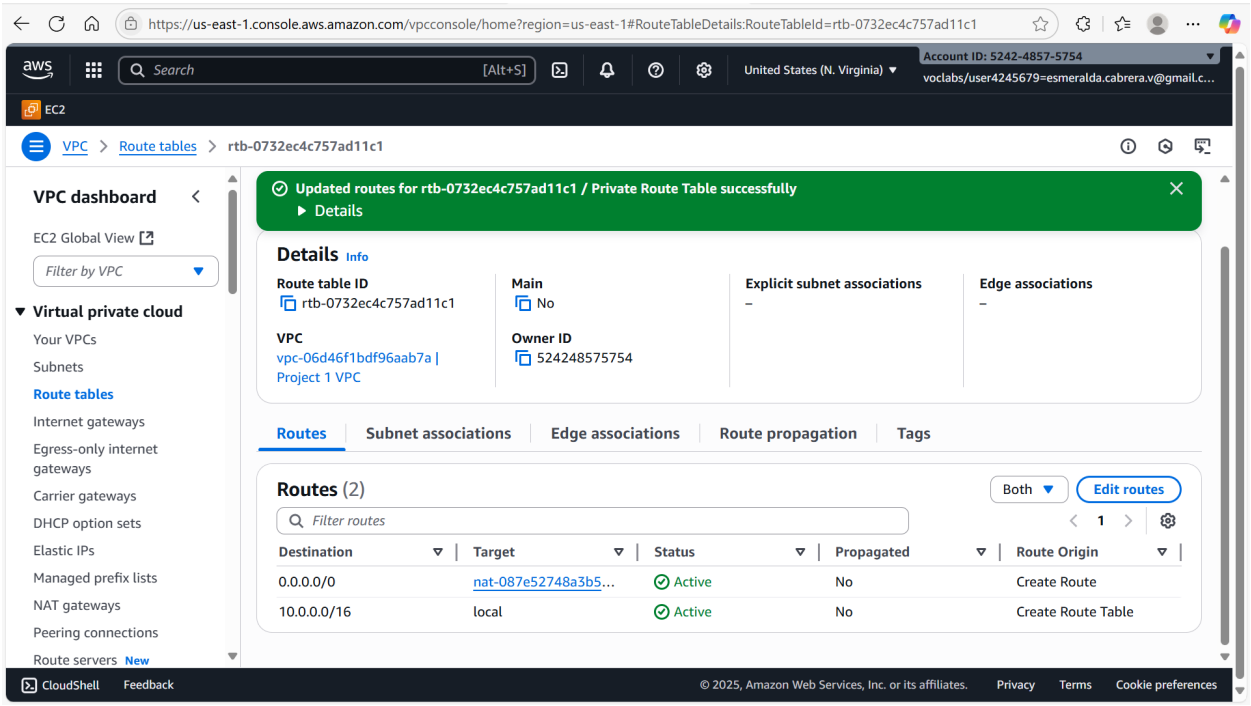
Application Server (Public Subnet)

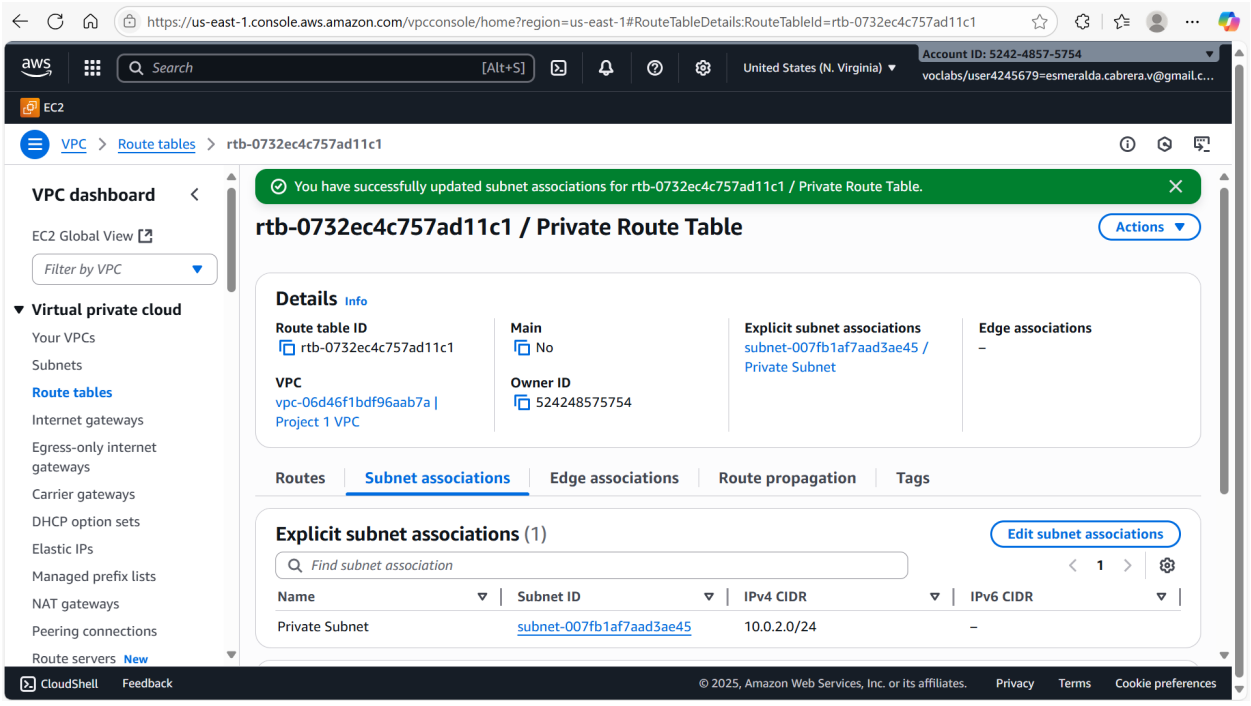
- Amazon Linux 2 AMI
- Instance type: t3.micro
- Security Group:
  - 22 (SSH)
  - 80 / 443 (future web access)
  - 8065 (Mattermost)
- Assigned public IP

Database Server (Private Subnet)

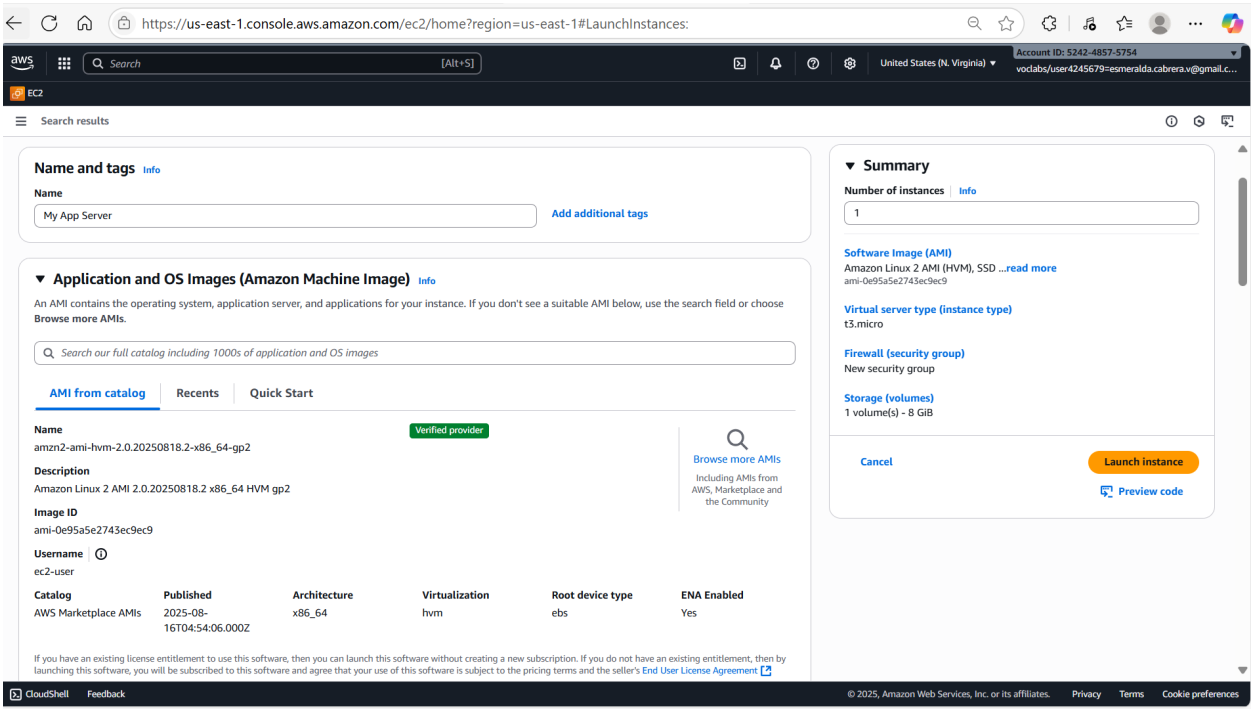
- Amazon Linux 2 AMI
- Instance type: t3.micro
- No public IP
- Security Group:
  - 22 (SSH via bastion)
  - 3306 (MySQL)
  - 80 / 443 (installation dependencies)

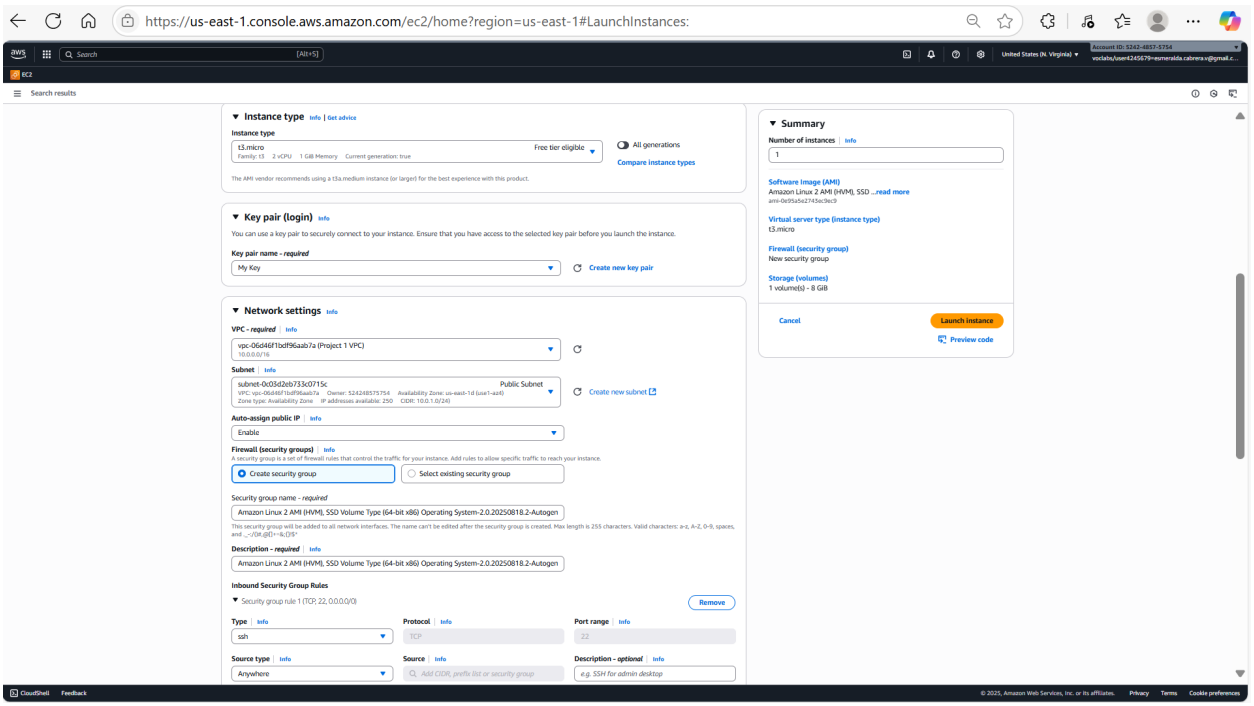
Creation of private route tables

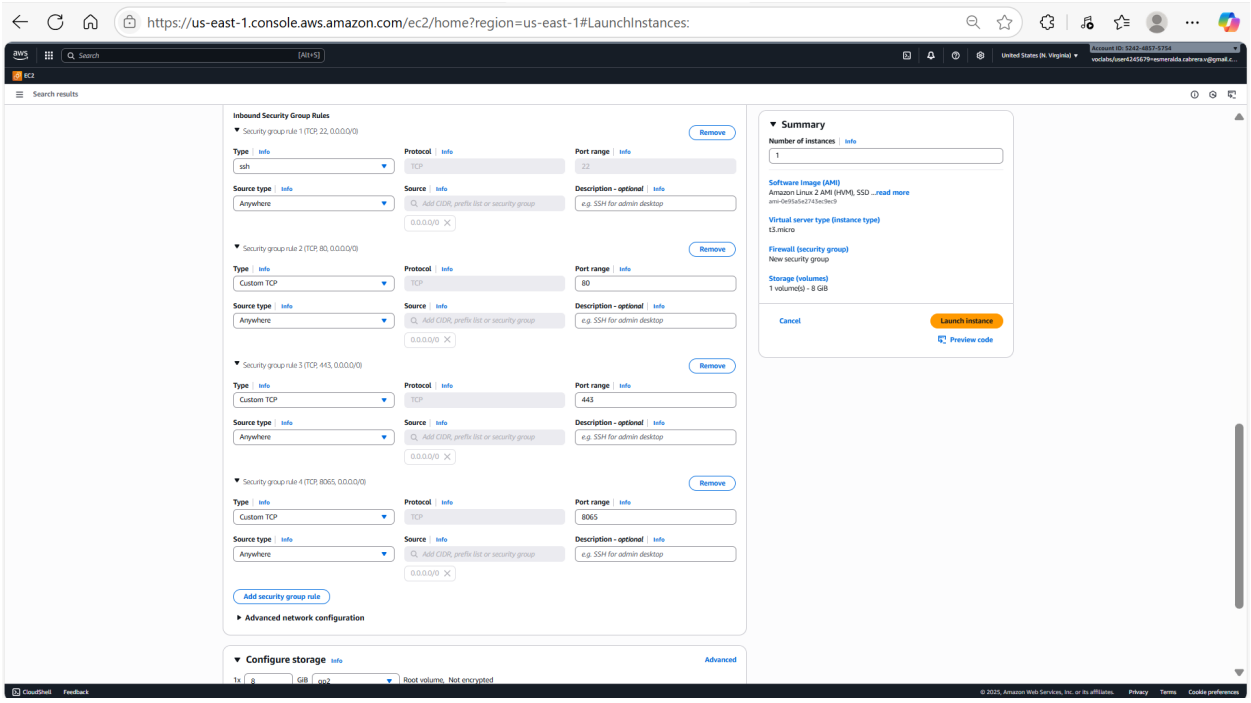


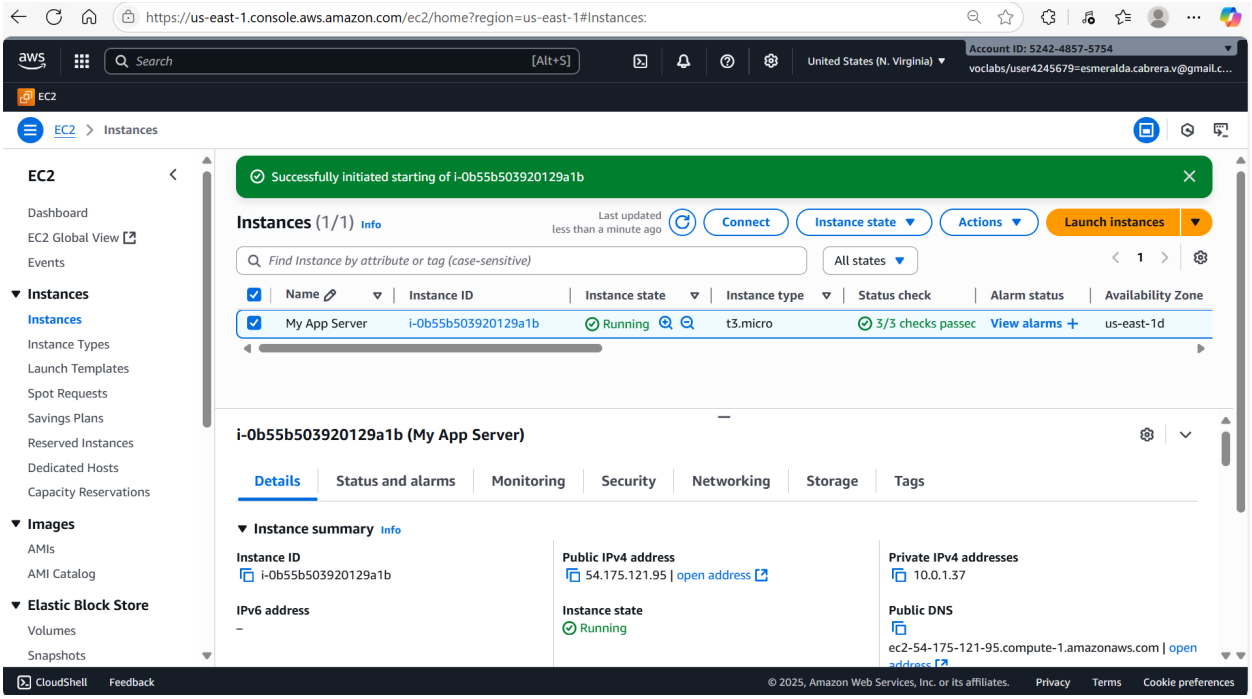


Creation of application server

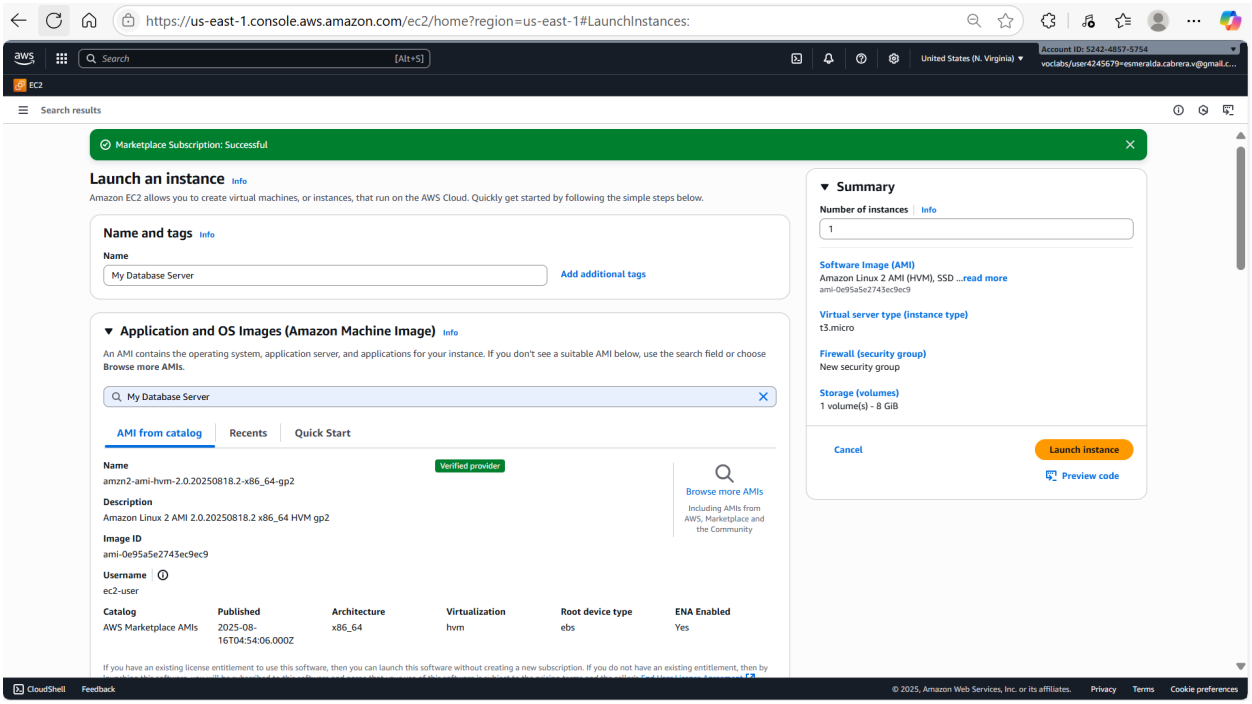


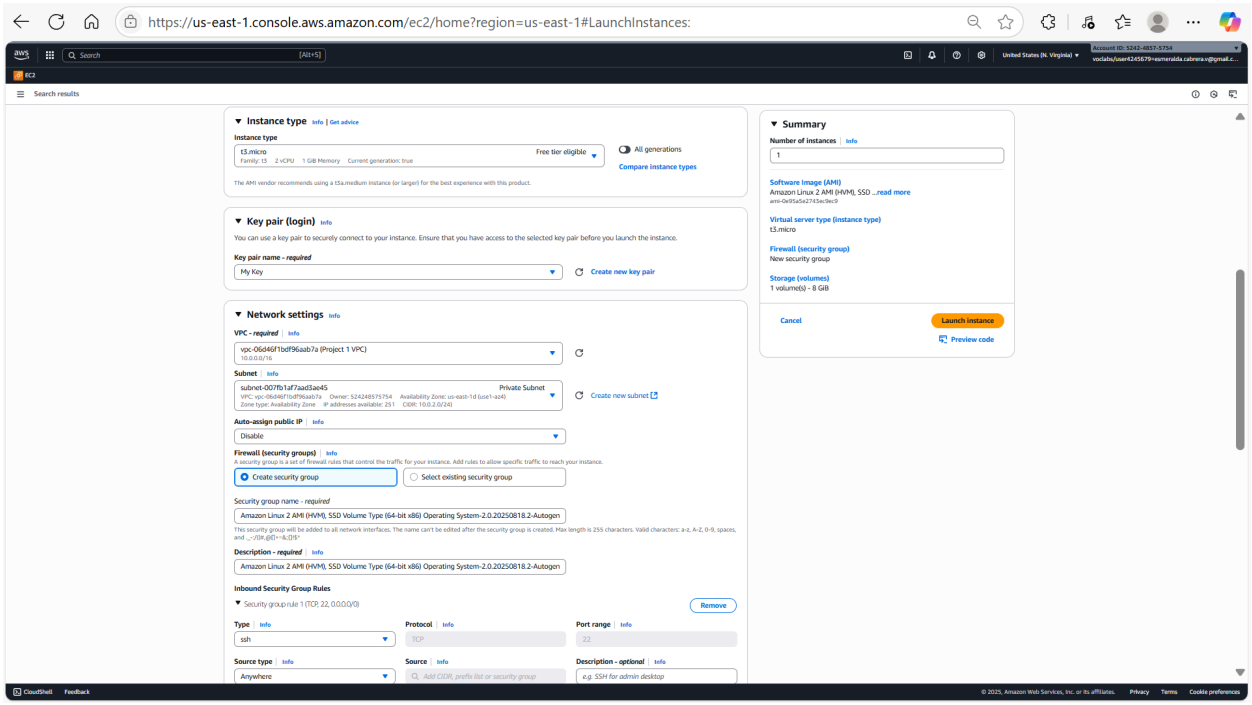


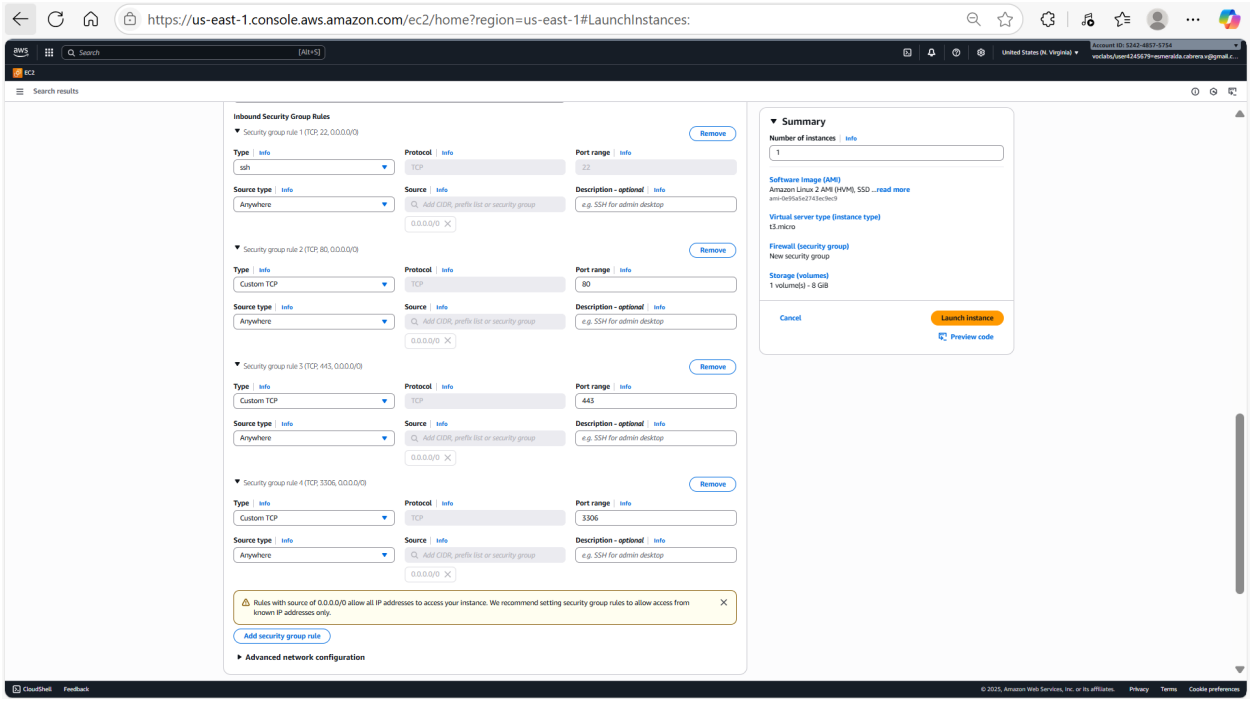

https://us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#LaunchInstances:
Inbound Security Group Rules
Security group rule 1 (TCP, 22, 0.0.0.0/0)
Security group rule 2 (TCP, 80, 0.0.0.0/0)
Security group rule 3 (TCP, 443, 0.0.0.0/0)
Security group rule 4 (TCP, 8065, 0.0.0.0/0)
Add security group rule
Advanced network configuration
Configure storage
Summary
Number of instances
Software Image (AMI)
Amazon Linux 2 AMI (HVM), SSD ...read more
Virtual server type (instance type)
t3.micro
Firewall (security group)
New security group
Storage (volumes)
1 volume(s) - 8 GiB
Cancel
Launch instance
Preview code

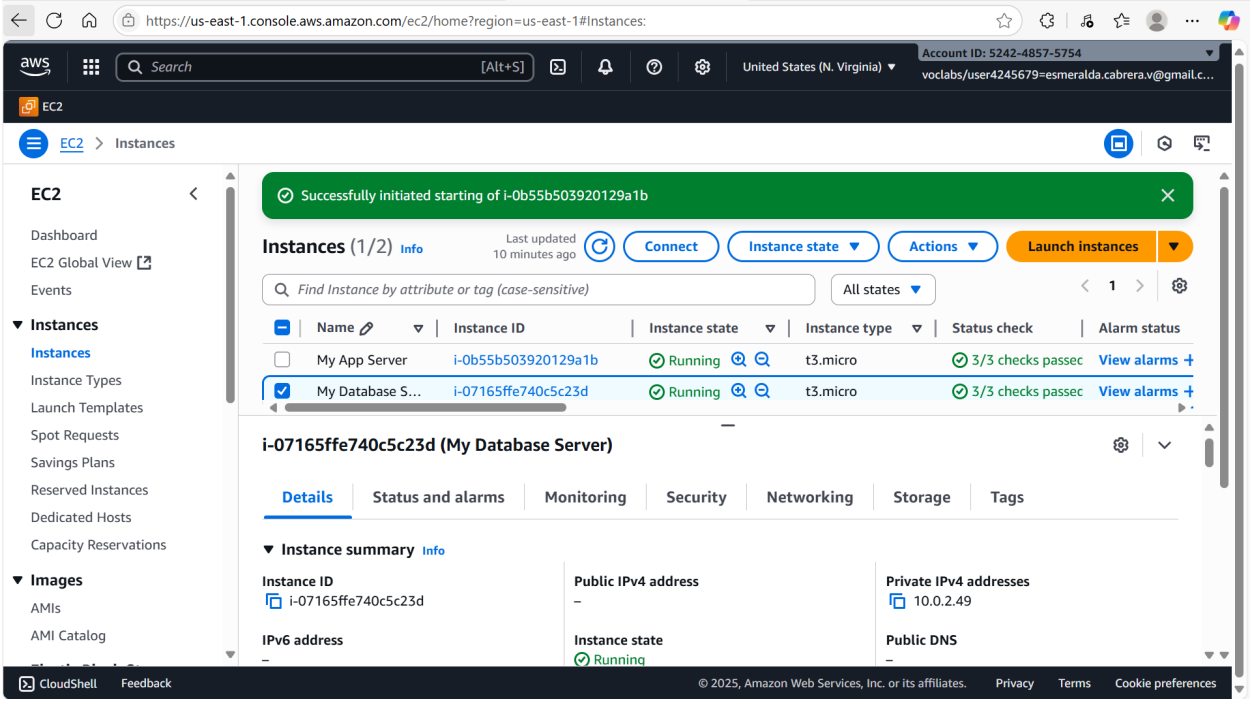

https://us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#Instances:
Search
[Alt+S]
United States (N. Virginia)
Account ID: 5242-4857-5754
voclabs/user4245679=esmeralda.cabrera.v@gmail.c...
EC2
EC2 > Instances
Successfully initiated starting of i-0b55b503920129a1b
Instances (1/1) Info
Last updated less than a minute ago
Connect
Instance state
Actions
Launch instances
Find Instance by attribute or tag (case-sensitive)
All states
Name
Instance ID
Instance state
Instance type
Status check
Alarm status
Availability Zone
My App Server
i-0b55b503920129a1b
Running
t3.micro
3/3 checks passec
View alarms
us-east-1d
Dashboard
EC2 Global View
Events
Instances
Instances
Instance Types
Launch Templates
Spot Requests
Savings Plans
Reserved Instances
Dedicated Hosts
Capacity Reservations
Images
AMIs
AMI Catalog
Elastic Block Store
Volumes
Snapshots
i-0b55b503920129a1b (My App Server)
Details
Status and alarms
Monitoring
Security
Networking
Storage
Tags
Instance summary Info
Instance ID
i-0b55b503920129a1b
IPv6 address
–
Public IPv4 address
54.175.121.95 | open address
Instance state
Running
Private IPv4 addresses
10.0.1.37
Public DNS
ec2-54-175-121-95.compute-1.amazonaws.com | open address
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

Creation of database server






https://us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#LaunchInstances:
Inbound Security Group Rules
Security group rule 1 (TCP, 22, 0.0.0.0/0)
Security group rule 2 (TCP, 80, 0.0.0.0/0)
Security group rule 3 (TCP, 443, 0.0.0.0/0)
Security group rule 4 (TCP, 3306, 0.0.0.0/0)
Rules with source of 0.0.0.0/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.
Add security group rule
Advanced network configuration
Summary
Number of instances
Software Image (AMI)
Amazon Linux 2 AMI (HVM), SSD ...read more
Virtual server type (instance type)
t3.micro
Firewall (security group)
New security group
Storage (volumes)
1 volume(s) - 8 GiB
Cancel
Launch instance
Preview code


https://us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#Instances:
Search
[Alt+S]
United States (N. Virginia)
Account ID: 5242-4857-5754
voclabs/user4245679=esmeralda.cabrera.v@gmail.c...
EC2
EC2 > Instances
Successfully initiated starting of i-0b55b503920129a1b
Dashboard
EC2 Global View
Events
Instances
Instances
Instance Types
Launch Templates
Spot Requests
Savings Plans
Reserved Instances
Dedicated Hosts
Capacity Reservations
Images
AMIs
AMI Catalog
Instances (1/2) Info
Last updated 10 minutes ago
Connect
Instance state
Actions
Launch instances
Find Instance by attribute or tag (case-sensitive)
All states
Name
Instance ID
Instance state
Instance type
Status check
Alarm status
My App Server
i-0b55b503920129a1b
Running
t3.micro
3/3 checks passec
View alarms
My Database S...
i-07165ffe740c5c23d
Running
t3.micro
3/3 checks passec
View alarms
i-07165ffe740c5c23d (My Database Server)
Details
Status and alarms
Monitoring
Security
Networking
Storage
Tags
Instance summary Info
Instance ID
i-07165ffe740c5c23d
Public IPv4 address
–
Private IPv4 addresses
10.0.2.49
IPv6 address
–
Instance state
Running
Public DNS
–
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

Step 4: MySQL Installation and Configuration (Private Server)

Objective: Prepare the database backend securely.

• Copied database PEM key to the application server

• Used SSH from the app server to reach the private database server

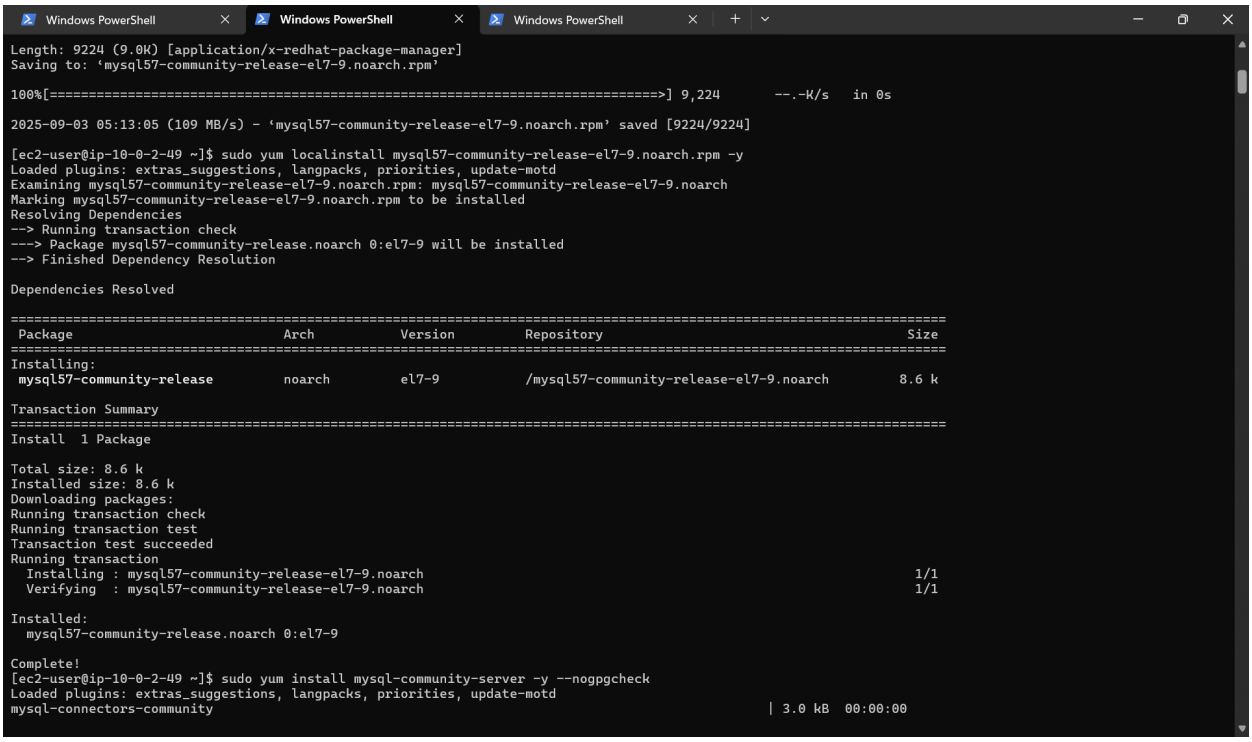
• Installed MySQL 5.7

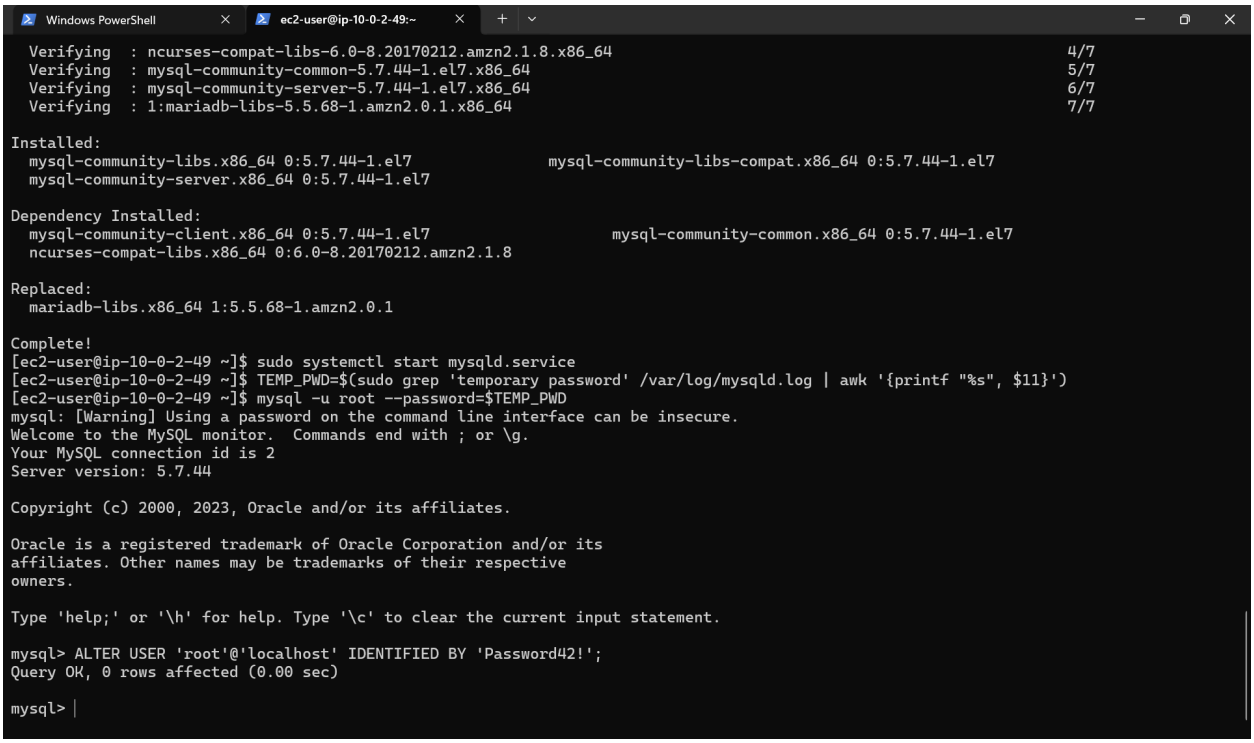
• Retrieved temporary root password

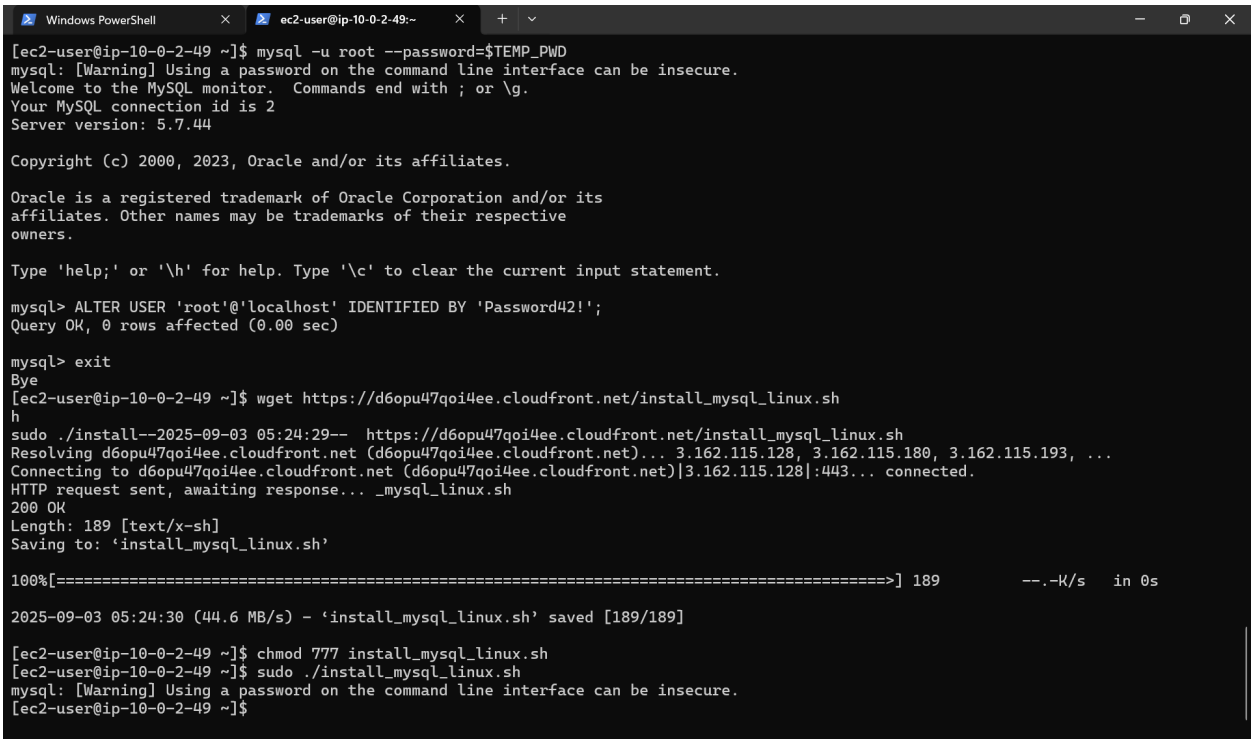
• Set a permanent root password

• Executed provided initialization script

Installation and configuration of MySQL




Windows PowerShell
ec2-user@ip-10-0-2-49:~
Verifying : ncurses-compat-libs-6.0-8.20170212.amzn2.1.8.x86_64 4/7
Verifying : mysql-community-common-5.7.44-1.el7.x86_64 5/7
Verifying : mysql-community-server-5.7.44-1.el7.x86_64 6/7
Verifying : 1:mariadb-libs-5.5.68-1.amzn2.0.1.x86_64 7/7
Installed:
mysql-community-libs.x86_64 0:5.7.44-1.el7
mysql-community-libs-compat.x86_64 0:5.7.44-1.el7
mysql-community-server.x86_64 0:5.7.44-1.el7
Dependency Installed:
mysql-community-client.x86_64 0:5.7.44-1.el7
mysql-community-common.x86_64 0:5.7.44-1.el7
ncurses-compat-libs.x86_64 0:6.0-8.20170212.amzn2.1.8
Replaced:
mariadb-libs.x86_64 1:5.5.68-1.amzn2.0.1
Complete!
[ec2-user@ip-10-0-2-49 ~]$ sudo systemctl start mysqld.service
[ec2-user@ip-10-0-2-49 ~]$ TEMP_PWD=$(sudo grep 'temporary password' /var/log/mysqld.log | awk '{printf "%s", $11}')
[ec2-user@ip-10-0-2-49 ~]$ mysql -u root --password=$TEMP_PWD
mysql: [Warning] Using a password on the command line interface can be insecure.
Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 2
Server version: 5.7.44
Copyright (c) 2000, 2023, Oracle and/or its affiliates.
Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.
mysql> ALTER USER 'root'@'localhost' IDENTIFIED BY 'Password42!';
Query OK, 0 rows affected (0.00 sec)
mysql>


Windows PowerShell
ec2-user@ip-10-0-2-49:~
[ec2-user@ip-10-0-2-49 ~]$ mysql -u root --password=$TEMP_PWD
mysql: [Warning] Using a password on the command line interface can be insecure.
Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 2
Server version: 5.7.44
Copyright (c) 2000, 2023, Oracle and/or its affiliates.
Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.
mysql> ALTER USER 'root'@'localhost' IDENTIFIED BY 'Password42!';
Query OK, 0 rows affected (0.00 sec)
mysql> exit
Bye
[ec2-user@ip-10-0-2-49 ~]$ wget https://d6opu47qoi4ee.cloudfront.net/install_mysql_linux.sh
h
sudo ./install--2025-09-03 05:24:29-- https://d6opu47qoi4ee.cloudfront.net/install_mysql_linux.sh
Resolving d6opu47qoi4ee.cloudfront.net (d6opu47qoi4ee.cloudfront.net)... 3.162.115.128, 3.162.115.180, 3.162.115.193, ...
Connecting to d6opu47qoi4ee.cloudfront.net (d6opu47qoi4ee.cloudfront.net)|3.162.115.128|:443... connected.
HTTP request sent, awaiting response... _mysql_linux.sh
200 OK
Length: 189 [text/x-sh]
Saving to: ‘install_mysql_linux.sh’
100%[==========>] 189 --.-K/s in 0s
2025-09-03 05:24:30 (44.6 MB/s) - ‘install_mysql_linux.sh’ saved [189/189]
[ec2-user@ip-10-0-2-49 ~]$ chmod 777 install_mysql_linux.sh
[ec2-user@ip-10-0-2-49 ~]$ sudo ./install_mysql_linux.sh
mysql: [Warning] Using a password on the command line interface can be insecure.
[ec2-user@ip-10-0-2-49 ~]$

Step 5: Mattermost Installation and Validation

Objective: Deploy and validate the collaboration platform.

• Installed Mattermost on the application server

• Configured it to connect to the private MySQL instance

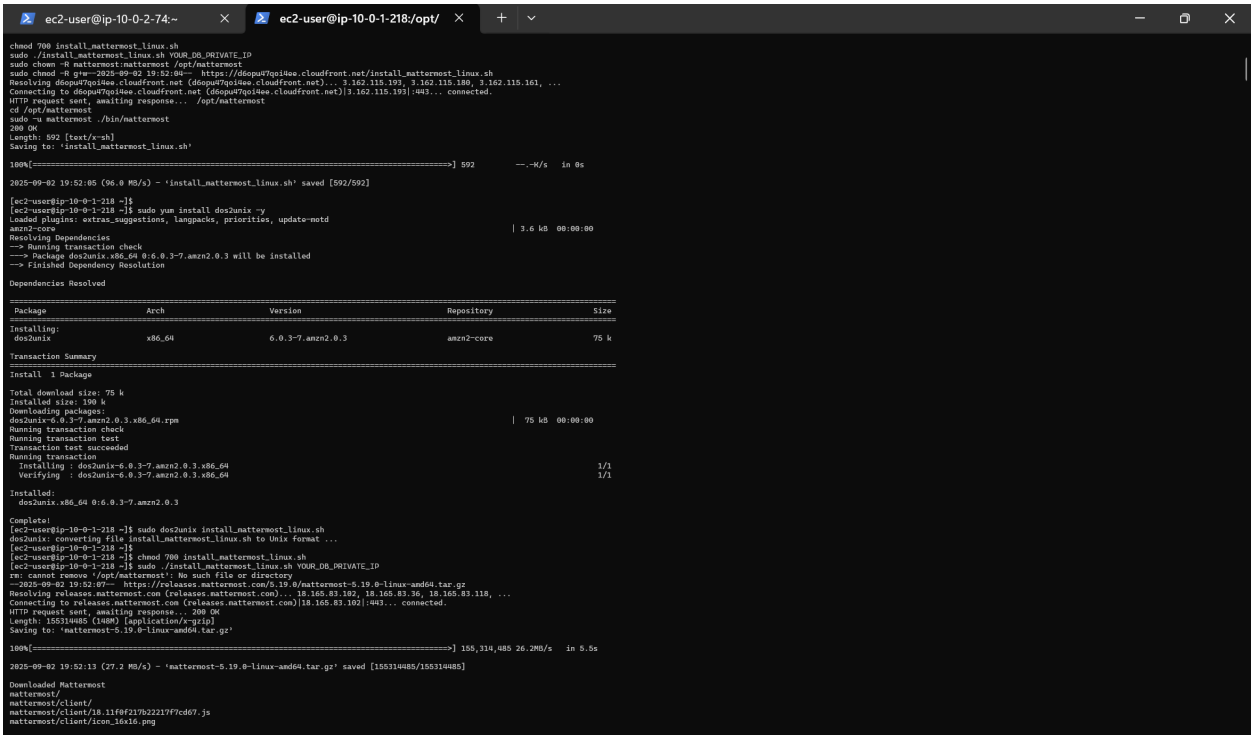
• Fixed permissions and ownership

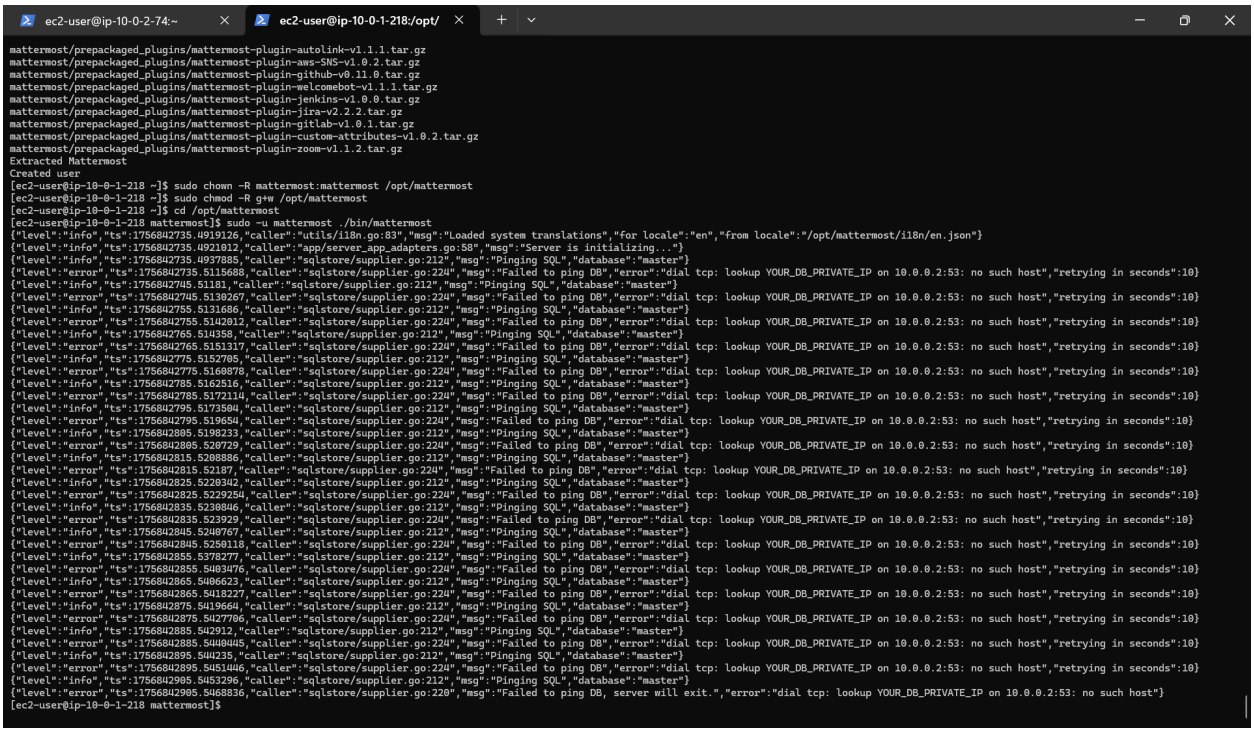
• Started Mattermost service

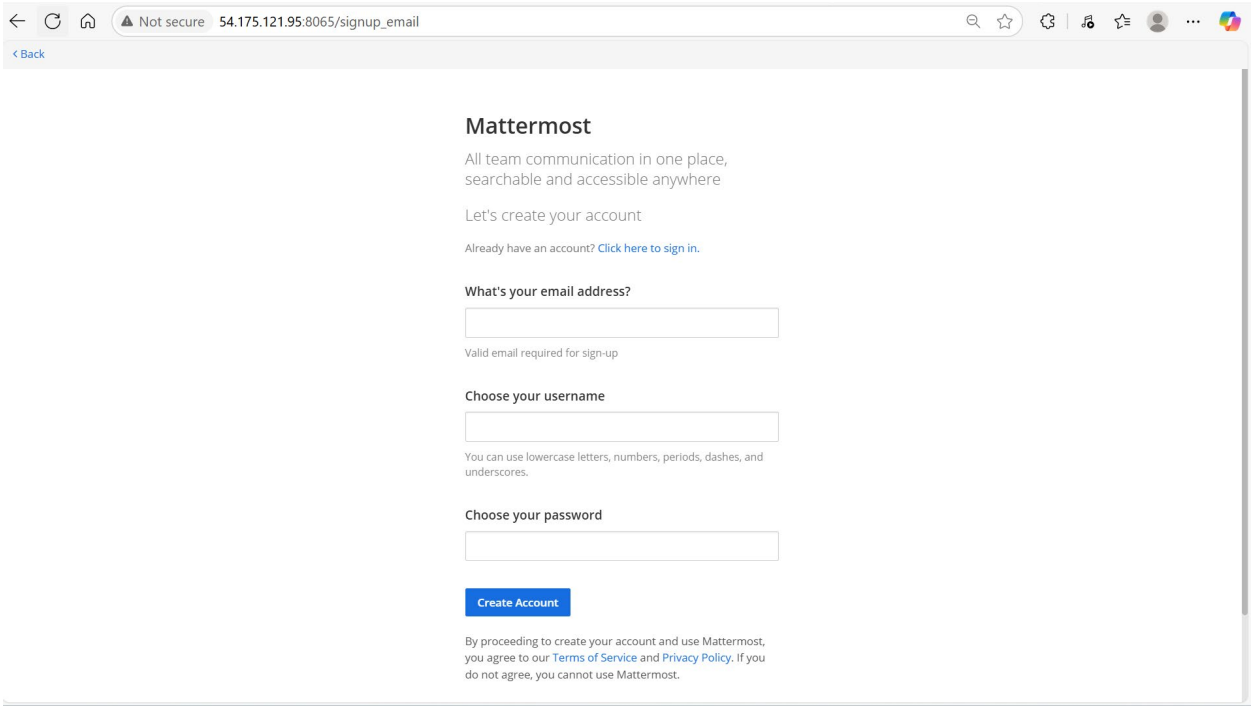
• Verified successful deployment via browser

Access URL: http://<Application_Server_Public_IP>:8065

Installation and configuration of Mattermost




ec2-user@ip-10-0-2-74:~
ec2-user@ip-10-0-1-218:/opt/
mattermost/prepackaged_plugins/mattermost-plugin-autolink-v1.1.1.tar.gz
mattermost/prepackaged_plugins/mattermost-plugin-aws-SNS-v1.0.2.tar.gz
mattermost/prepackaged_plugins/mattermost-plugin-github-v0.11.0.tar.gz
mattermost/prepackaged_plugins/mattermost-plugin-welcomebot-v1.1.1.tar.gz
mattermost/prepackaged_plugins/mattermost-plugin-jenkins-v1.0.0.tar.gz
mattermost/prepackaged_plugins/mattermost-plugin-jira-v2.2.2.tar.gz
mattermost/prepackaged_plugins/mattermost-plugin-gitlab-v1.0.1.tar.gz
mattermost/prepackaged_plugins/mattermost-plugin-custom-attributes-v1.0.2.tar.gz
mattermost/prepackaged_plugins/mattermost-plugin-zoom-v1.1.2.tar.gz
Extracted Mattermost
Created user
[ec2-user@ip-10-0-1-218 ~]$ sudo chown -R mattermost:mattermost /opt/mattermost
[ec2-user@ip-10-0-1-218 ~]$ sudo chmod -R g+w /opt/mattermost
[ec2-user@ip-10-0-1-218 ~]$ cd /opt/mattermost
[ec2-user@ip-10-0-1-218 mattermost]$ sudo -u mattermost ./bin/mattermost
{"level":"info","ts":1756842735.4919126,"caller":"utils/i18n.go:83","msg":"Loaded system translations","for locale":"en","from locale":"/opt/mattermost/i18n/en.json"}
{"level":"info","ts":1756842735.4921012,"caller":"app/server_app_adapters.go:58","msg":"Server is initializing..."}
{"level":"info","ts":1756842735.4937885,"caller":"sqlstore/supplier.go:212","msg":"Pinging SQL","database":"master"}
{"level":"error","ts":1756842735.5115688,"caller":"sqlstore/supplier.go:224","msg":"Failed to ping DB","error":"dial tcp: lookup YOUR_DB_PRIVATE_IP on 10.0.0.2:53: no such host","retrying in seconds":10}
{"level":"info","ts":1756842745.51181,"caller":"sqlstore/supplier.go:212","msg":"Pinging SQL","database":"master"}
{"level":"error","ts":1756842745.5130267,"caller":"sqlstore/supplier.go:224","msg":"Failed to ping DB","error":"dial tcp: lookup YOUR_DB_PRIVATE_IP on 10.0.0.2:53: no such host","retrying in seconds":10}
{"level":"info","ts":1756842755.5131686,"caller":"sqlstore/supplier.go:212","msg":"Pinging SQL","database":"master"}
{"level":"error","ts":1756842755.5142012,"caller":"sqlstore/supplier.go:224","msg":"Failed to ping DB","error":"dial tcp: lookup YOUR_DB_PRIVATE_IP on 10.0.0.2:53: no such host","retrying in seconds":10}
{"level":"info","ts":1756842765.514358,"caller":"sqlstore/supplier.go:212","msg":"Pinging SQL","database":"master"}
{"level":"error","ts":1756842765.5151317,"caller":"sqlstore/supplier.go:224","msg":"Failed to ping DB","error":"dial tcp: lookup YOUR_DB_PRIVATE_IP on 10.0.0.2:53: no such host","retrying in seconds":10}
{"level":"info","ts":1756842775.5152705,"caller":"sqlstore/supplier.go:212","msg":"Pinging SQL","database":"master"}
{"level":"error","ts":1756842775.5160878,"caller":"sqlstore/supplier.go:224","msg":"Failed to ping DB","error":"dial tcp: lookup YOUR_DB_PRIVATE_IP on 10.0.0.2:53: no such host","retrying in seconds":10}
{"level":"info","ts":1756842785.5162516,"caller":"sqlstore/supplier.go:212","msg":"Pinging SQL","database":"master"}
{"level":"error","ts":1756842785.5172114,"caller":"sqlstore/supplier.go:224","msg":"Failed to ping DB","error":"dial tcp: lookup YOUR_DB_PRIVATE_IP on 10.0.0.2:53: no such host","retrying in seconds":10}
{"level":"info","ts":1756842795.5173504,"caller":"sqlstore/supplier.go:212","msg":"Pinging SQL","database":"master"}
{"level":"error","ts":1756842795.519654,"caller":"sqlstore/supplier.go:224","msg":"Failed to ping DB","error":"dial tcp: lookup YOUR_DB_PRIVATE_IP on 10.0.0.2:53: no such host","retrying in seconds":10}
{"level":"info","ts":1756842805.5198233,"caller":"sqlstore/supplier.go:212","msg":"Pinging SQL","database":"master"}
{"level":"error","ts":1756842805.520729,"caller":"sqlstore/supplier.go:224","msg":"Failed to ping DB","error":"dial tcp: lookup YOUR_DB_PRIVATE_IP on 10.0.0.2:53: no such host","retrying in seconds":10}
{"level":"info","ts":1756842815.5208886,"caller":"sqlstore/supplier.go:212","msg":"Pinging SQL","database":"master"}
{"level":"error","ts":1756842815.52187,"caller":"sqlstore/supplier.go:224","msg":"Failed to ping DB","error":"dial tcp: lookup YOUR_DB_PRIVATE_IP on 10.0.0.2:53: no such host","retrying in seconds":10}
{"level":"info","ts":1756842825.5220342,"caller":"sqlstore/supplier.go:212","msg":"Pinging SQL","database":"master"}
{"level":"error","ts":1756842825.5229254,"caller":"sqlstore/supplier.go:224","msg":"Failed to ping DB","error":"dial tcp: lookup YOUR_DB_PRIVATE_IP on 10.0.0.2:53: no such host","retrying in seconds":10}
{"level":"info","ts":1756842835.5230846,"caller":"sqlstore/supplier.go:212","msg":"Pinging SQL","database":"master"}
{"level":"error","ts":1756842835.523929,"caller":"sqlstore/supplier.go:224","msg":"Failed to ping DB","error":"dial tcp: lookup YOUR_DB_PRIVATE_IP on 10.0.0.2:53: no such host","retrying in seconds":10}
{"level":"info","ts":1756842845.5240767,"caller":"sqlstore/supplier.go:212","msg":"Pinging SQL","database":"master"}
{"level":"error","ts":1756842845.5250118,"caller":"sqlstore/supplier.go:224","msg":"Failed to ping DB","error":"dial tcp: lookup YOUR_DB_PRIVATE_IP on 10.0.0.2:53: no such host","retrying in seconds":10}
{"level":"info","ts":1756842855.5378277,"caller":"sqlstore/supplier.go:212","msg":"Pinging SQL","database":"master"}
{"level":"error","ts":1756842855.5403476,"caller":"sqlstore/supplier.go:224","msg":"Failed to ping DB","error":"dial tcp: lookup YOUR_DB_PRIVATE_IP on 10.0.0.2:53: no such host","retrying in seconds":10}
{"level":"info","ts":1756842865.5406623,"caller":"sqlstore/supplier.go:212","msg":"Pinging SQL","database":"master"}
{"level":"error","ts":1756842865.5418227,"caller":"sqlstore/supplier.go:224","msg":"Failed to ping DB","error":"dial tcp: lookup YOUR_DB_PRIVATE_IP on 10.0.0.2:53: no such host","retrying in seconds":10}
{"level":"info","ts":1756842875.5419664,"caller":"sqlstore/supplier.go:212","msg":"Pinging SQL","database":"master"}
{"level":"error","ts":1756842875.5427706,"caller":"sqlstore/supplier.go:224","msg":"Failed to ping DB","error":"dial tcp: lookup YOUR_DB_PRIVATE_IP on 10.0.0.2:53: no such host","retrying in seconds":10}
{"level":"info","ts":1756842885.542912,"caller":"sqlstore/supplier.go:212","msg":"Pinging SQL","database":"master"}
{"level":"error","ts":1756842885.5440445,"caller":"sqlstore/supplier.go:224","msg":"Failed to ping DB","error":"dial tcp: lookup YOUR_DB_PRIVATE_IP on 10.0.0.2:53: no such host","retrying in seconds":10}
{"level":"info","ts":1756842895.544235,"caller":"sqlstore/supplier.go:212","msg":"Pinging SQL","database":"master"}
{"level":"error","ts":1756842895.5451446,"caller":"sqlstore/supplier.go:224","msg":"Failed to ping DB","error":"dial tcp: lookup YOUR_DB_PRIVATE_IP on 10.0.0.2:53: no such host","retrying in seconds":10}
{"level":"info","ts":1756842905.5453296,"caller":"sqlstore/supplier.go:212","msg":"Pinging SQL","database":"master"}
{"level":"error","ts":1756842905.5468836,"caller":"sqlstore/supplier.go:220","msg":"Failed to ping DB, server will exit.","error":"dial tcp: lookup YOUR_DB_PRIVATE_IP on 10.0.0.2:53: no such host"}
[ec2-user@ip-10-0-1-218 mattermost]$


Not secure
54.175.121.95:8065/signup_email
Back
Mattermost
All team communication in one place, searchable and accessible anywhere
Let's create your account
Already have an account? Click here to sign in.
What's your email address?
Valid email required for sign-up
Choose your username
You can use lowercase letters, numbers, periods, dashes, and underscores.
Choose your password
Create Account
By proceeding to create your account and use Mattermost, you agree to our Terms of Service and Privacy Policy. If you do not agree, you cannot use Mattermost.

Final Outcome

• Fully functional self-hosted collaboration platform

• Secure database isolation

• Controlled network access

• Realistic enterprise cloud architecture

• Clean separation of concerns

Security & Scalability Considerations

Security Enhancements

• Restrict SSH to known IPs or migrate to AWS SSM

• Lock MySQL access to application SG only

• Add HTTPS via ALB + ACM

• Disable open sign-ups in Mattermost

Scalability Options

• Move MySQL to Amazon RDS

• Store uploads on S3 or EFS

• Add Auto Scaling Group for application servers

• Front with Application Load Balancer

Skills Demonstrated

• AWS VPC design

• Subnetting and CIDR planning

- Internet & NAT Gateways
- Route table configuration
- Bastion host pattern
- EC2 provisioning
- Linux server administration
- Secure application deployment
- Cloud troubleshooting