

Two-Tier AWS Web Application with Private Database

1. Project Overview

This project demonstrates a secure two-tier architecture on AWS where:

- A public web server hosts a website and backend service.
- A private database server hosts MariaDB with no public access.
- The database can still access the internet only for updates and patching using a NAT Gateway.

The project focuses on real-world AWS networking, security, troubleshooting, and cost awareness.

2. Architecture Summary

High-Level Design

- VPC with a custom IPv4 CIDR
- Public Subnet
 - Web EC2 instance
 - Internet Gateway attached to VPC
- Private Subnet
 - DB EC2 instance (MariaDB)
 - No public IP
- NAT Gateway (Public, Regional)
 - Provides outbound internet access to private subnet
- Security Groups
 - Web SG: allows HTTP (80) and SSH (22 from admin IP)
 - DB SG: allows MySQL (3306) and SSH only from Web SG

Traffic Flow

- User → Web EC2 (HTTP)
- Web EC2 → DB EC2 (MySQL via private IP)
- DB EC2 → Internet (updates via NAT Gateway)

3. AWS Resources Used

Resource	Purpose
VPC	Network isolation
Public Subnet	Hosts web server
Private Subnet	Hosts database server
Internet Gateway	Internet access for public subnet
NAT Gateway (Regional, Public)	Outbound internet for private subnet
EC2 (Web)	Apache + Node.js backend
EC2 (DB)	MariaDB server
Security Groups	Traffic control
Route Tables	Traffic routing

4. Step-by-Step Implementation

4.1 VPC and Networking Setup

- Created a VPC with CIDR block (example: **10.0.0.0/16**).
- Created:
 - **public-subnet**
 - **private-subnet**
- Attached Internet Gateway to the VPC.

Route Tables

- **Public Route Table:**
 - **0.0.0.0/0** → **Internet Gateway**
 - **Private Route Table:**
 - **10.0.0.0/16** → **local**
 - **0.0.0.0/0** → **NAT Gateway**
-

4.2 Web Server EC2 (Public)

Configuration:

- Amazon Linux 2023
- Public IP enabled
- Security Group allows:
 - **HTTP (80)** from **0.0.0.0/0**
 - **SSH (22)** from admin IP

Installed Services:

```
sudo dnf install httpd nodejs -y
```

```
sudo systemctl start httpd
```

```
sudo systemctl enable httpd
```

Website Deployment:

- Copied frontend files to:

```
/var/www/html/
```

- Verified website access via:

`http://<Web-EC2-Public-IP>`

4.3 Private Database EC2

Configuration:

- Amazon Linux 2023
- No public IP
- Placed in private subnet
- Security Group allows:
 - MySQL (3306) only from Web SG
 - SSH (22) only from Web SG

Access Method:

- SSH hop:

Laptop → Web EC2 → DB EC2

4.4 NAT Gateway Setup

Reason:

Private subnet cannot access the internet directly. NAT Gateway enables outbound-only access.

Configuration:

- NAT Gateway type: Public
- Availability mode: Regional
- Elastic IP: Automatic allocation

Troubleshooting:

- Encountered **Blackhole** route status
- Resolved by re-adding the **0.0.0.0/0 → NAT Gateway** route after NAT became **Available**

4.5 MariaDB Installation and Configuration

After NAT was active:

```
sudo dnf install mariadb105-server -y  
sudo systemctl start mariadb  
sudo systemctl enable mariadb
```

Database Security:

```
sudo mysql_secure_installation
```

Database and User Creation:

```
CREATE DATABASE lokha_db;  
CREATE USER 'lokha_user'@'%' IDENTIFIED BY 'StrongPassword';  
GRANT ALL PRIVILEGES ON lokha_db.* TO 'lokha_user'@'%';  
FLUSH PRIVILEGES;
```

4.6 Web → DB Connectivity Test

From Web EC2:

```
mysql -h <DB-PRIVATE-IP> -u lokha_user -p
```

Result:

- Successful connection
- Confirms:
 - DB has no public access

- Only web server can connect
-

4.7 Node.js Backend Integration

- Installed dependencies using:

```
npm install
```

- Backend started with:

```
node server.js
```

Issue Faced:

- Frontend showed connection lost

Root Cause:

- Frontend JavaScript was calling:

```
http://localhost:3000
```

- localhost in browser refers to user's PC, not EC2

Fix:

- Updated frontend API URL to:

```
http://<Web-EC2-Public-IP>:3000
```

5. Common Issues & Troubleshooting

Issue	Cause	Resolution
-------	-------	------------

<code>dnf install</code>	No internet in private subnet	Added NAT Gateway
timeout		
SSH permission denied	Missing key on web EC2	Converted <code>.ppk</code> to <code>.pem</code>
MariaDB access denied	User/host mismatch or password	Recreated DB user
<code>Cannot GET /</code>	No root route in Node.js	Expected behavior
<code>connection lost</code>	Frontend using localhost	Used EC2 public IP

6. Security Best Practices Followed

- Database has no public IP
 - Security group references instead of CIDR for DB access
 - NAT Gateway used instead of public DB exposure
 - Root DB login disabled remotely
 - Separate DB user for application
-

7. Screenshots

The screenshot shows the AWS EC2 Instances page. There are two instances listed: dbserver1 and server1. Both are running t2.micro instances. The instance details for server1 are expanded, showing the following networking information:

Subnet ID	Availability zone
subnet-0466c756f12fed7d4 (public-subnet)	ap-south-1a

IP addresses:

Public IPv4 address	Private IPv4 addresses	IPv6 addresses
65.1.131.25 open address	10.0.1.211	-

Hostname and DNS:

Public DNS	Private IP DNS name (IPv4 only)	IPv4-only IP based name: A record only
-	ip-10-0-1-211.ap-south-1.compute.internal	-

Dualstack - IP based name: A and AAAA record

IPv6-only - IP based name: AAAA record only	Public hostname type
-	-

Activate Windows: Go to Settings to activate Windows.

1.server1

The screenshot shows the AWS EC2 Instances page. There are two instances listed: dbserver1 and server1. Both are running t2.micro instances. The instance details for dbserver1 are expanded, showing the following networking information:

Subnet ID	Availability zone
subnet-0c4ffe6c991221909 (private-subnet)	ap-south-1a

IP addresses:

Public IPv4 address	Private IPv4 addresses	IPv6 addresses
-	10.0.2.232	-

Hostname and DNS:

Public DNS	Private IP DNS name (IPv4 only)	IPv4-only IP based name: A record only
-	ip-10-0-2-232.ap-south-1.compute.internal	-

Dualstack - IP based name: A and AAAA record

IPv6-only - IP based name: AAAA record only	Public hostname type
-	-

Activate Windows: Go to Settings to activate Windows.

2.database server

Route tables (1/9) Info

Name	Route table ID	Explicit subnet associations	Main	VPC	Owner ID	
project-vpc-rtb-public	rtb-0fe88051290276b85	-	No	vpc-00e7880946f770ea6 proj...	581483106531	
-	rtb-0deaa8fbfc46c84	-	Yes	vpc-00e7880946f770ea6 proj...	581483106531	
project-vpc-rtb-private1-ap-south-1a	rtb-04310d858d7f82b32	-	No	vpc-00e7880946f770ea6 proj...	581483106531	
-	rtb-059a7d9b4ca40f09a	-	Yes	vpc-0e6feb31f67cea745	581483106531	
project-vpc-rtb-private2-ap-south-1b	rtb-00a7d031294fa7b25	-	No	vpc-00e7880946f770ea6 proj...	581483106531	
-	rtb-00a7a8011cb259bf3	-	Yes	vpc-05e79e19cf17eb848 proj...	581483106531	
public-rt	rtb-0041dd0e037841ba	subnet-0466c756f12fed7...	No	vpc-05e79e19cf17eb848 proj...	581483106531	
private-rt	rtb-08f785146a2152cc5	subnet-0c4ffe6c991221909	No	vpc-05e79e19cf17eb848 proj...	581483106531	
-	rtb-0e507d0987fb863bb	-	nat-1fc1cd59a81ec...	No	vpc-05e79e19cf17eb848 proj...	581483106531

rtb-08f785146a2152cc5 / private-rt

Details

Route table ID rtb-08f785146a2152cc5	Main No	Explicit subnet associations subnet-0c4ffe6c991221909 / private-subnet	Edge associations
VPC vpc-05e79e19cf17eb848 project-vpc-last25	Owner ID 581483106531		

Activate Windows
Go to Settings to activate Windows.

3.Routetable-private

Route tables (1/9) Info

Name	Route table ID	Explicit subnet associations	Main	VPC	Owner ID	
project-vpc-rtb-public	rtb-0fe88051290276b85	-	No	vpc-00e7880946f770ea6 proj...	581483106531	
-	rtb-0deaa8fbfc46c84	-	Yes	vpc-00e7880946f770ea6 proj...	581483106531	
project-vpc-rtb-private1-ap-south-1a	rtb-04310d858d7f82b32	-	No	vpc-00e7880946f770ea6 proj...	581483106531	
-	rtb-059a7d9b4ca40f09a	-	Yes	vpc-0e6feb31f67cea745	581483106531	
project-vpc-rtb-private2-ap-south-1b	rtb-00a7d031294fa7b25	-	No	vpc-00e7880946f770ea6 proj...	581483106531	
-	rtb-00a7a8011cb259bf3	-	Yes	vpc-05e79e19cf17eb848 proj...	581483106531	
public-rt	rtb-0b41dd0e037841ba	subnet-0466c756f12fed7...	No	vpc-05e79e19cf17eb848 proj...	581483106531	
-	rtb-0e507d0987fb863bb	-	nat-1fc1cd59a81ec...	No	vpc-05e79e19cf17eb848 proj...	581483106531

rtb-0b41dd0e037841ba / public-rt

Details

Route table ID rtb-0b41dd0e037841ba	Main No	Explicit subnet associations subnet-0466c756f12fed7d4 / public-subnet	Edge associations
VPC vpc-05e79e19cf17eb848 project-vpc-last25	Owner ID 581483106531		

Activate Windows
Go to Settings to activate Windows.

4.Routetable-public

NAT gateways (1/1) Info

Name	NAT gateway ID	Connectivity...	State	State message	Availability ...	Route table ID	Primary public IP...	Primary private I...	Primary network...
project-nat	nat-1fc1cd59a81ecf504	Public	Available	-	Regional	rtb-0e507d09b...	13.205.99.145	-	-

nat-1fc1cd59a81ecf504 / project-nat

Details **IP addresses** **Monitoring** **Flow logs** **Tags**

NAT gateway ID nat-1fc1cd59a81ecf504	Availability mode Regional	State Available	State message -
NAT gateway ARN arn:aws:ec2:ap-south-1:581483106531:natgateway/nat-1fc1cd59a81ecf504	Connectivity type Public	Created Wednesday, December 31, 2025 at 17:45:14 GMT+5:30	Deleted -
VPC vpc-05e79e19c17eb848 / project-vpc-last25	Method of EIP allocation Automatic		

Activate Windows
Go to Settings to activate Windows.

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

5.Nat gateways

Elastic IP addresses (1/1) Info

Name	Allocated IPv4 addr...	Type	Allocation ID	Reverse DNS record	Associated instance ID	Private IP address	Association ID
13.205.99.145	13.205.99.145	Public IP	eipalloc-02b80afac31145921	-	-	13.205.99.145	eipassoc-0015b52

13.205.99.145

Summary **Tags**

Allocated IPv4 address 13.205.99.145	Type Public IP	Allocation ID eipalloc-02b80afac31145921	Reverse DNS record -
Association ID eipassoc-0015b52a9a1631a08	Scope VPC	Associated instance ID -	Private IP address 13.205.99.145
Network interface ID -	Network interface owner account ID -	Public DNS -	NAT Gateway ID nat-1fc1cd59a81ecf504 (project-nat)
Address pool Amazon	Network border group ap-south-1	Service managed nat	

Activate Windows
Go to Settings to activate Windows.

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

6.Elastic IP

The screenshot shows the AWS VPC Subnets console. On the left, there's a navigation sidebar with sections like VPC dashboard, Virtual private cloud, Security, and PrivateLink and Lattice. The main area displays a table of subnets with columns for Name, Subnet ID, State, VPC, Block Public Access, IPv4 CIDR, IPv6 CIDR, and IPv6 CIDR association ID. One subnet, 'private-subnet' (subnet-0c4ffe6c991221909), is selected. Below the table, a detailed view for 'private-subnet' is shown, including its subnet ID, ARN, state, VPC, and various configuration details like IPv4 CIDR (10.0.2.0/24), Availability Zone (ap-s1-az1), and Network border group (ap-south-1). A note at the bottom right says 'Activate Windows'.

7.Private subnet

This screenshot is similar to the one above but shows a different subnet. The selected subnet is 'public-subnet' (subnet-0466c756f12fed7d4). The detailed view shows its subnet ID, ARN, state, VPC, and configuration details. The IPv4 CIDR is 10.0.1.0/24, and the Availability Zone is ap-s1-az1. A note at the bottom right says 'Activate Windows'.

8.Public subnet

Internet gateways (1/1) info

Name	Internet gateway ID	State	VPC ID	Owner
project-igw	igw-035e118a882b215cc	Attached	vpc-05e79e19cf17eb848 project-vpc-l...	581483106531

igw-035e118a882b215cc / project-igw

Details **Tags**

Internet gateway ID	State	VPC ID	Owner
igw-035e118a882b215cc	Attached	vpc-05e79e19cf17eb848 project-vpc-l...	581483106531

Activate Windows
Go to Settings to activate Windows.

9. Internet Gateway

Your VPCs (1/3) info

Name	VPC ID	State	Encryption controls	Block Public Access	IPv4 CIDR	IPv6 CIDR	DHCP option set
vpc-0e6feb31f67cea745	vpc-0e6feb31f67cea745	Available	-	Off	172.31.0.0/16	-	dopt-05ab4604
project-vpc-vc	vpc-00e78809467f70ea6	Available	-	Off	10.0.0.0/16	-	dopt-05ab4604
project-vpc-last25	vpc-05e79e19cf17eb848	Available	-	Off	10.0.0.0/16	-	dopt-05ab4604

vpc-05e79e19cf17eb848 / project-vpc-last25

Details **Resource map** **CIDRs** **Flow logs** **Tags** **Integrations**

Details		Block Public Access		DNS hostnames	
VPC ID	vpc-05e79e19cf17eb848	State	Off	Disabled	
DNS resolution	Enabled	Tenancy	default	Main route table	rtb-007a8011bcd259bf5
Main network ACL	acl-03630087892cf3727	Default VPC	No	IPv6 pool	-
IPv6 CIDR (Network border group)	-	Network Address Usage metrics	Disabled	Route 53 Resolver DNS Firewall rule groups	-
Encryption control ID	-	Encryption control mode	-	Owner ID	581483106531

Activate Windows
Go to Settings to activate Windows.

10. VPC

The screenshot shows the AWS EC2 Security Groups page. On the left, there's a navigation sidebar with links like Instance Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Capacity Reservations, Capacity Manager, Images, Elastic Block Store, Network & Security (Security Groups selected), Load Balancing, Auto Scaling, and Settings. The main area has a title "Security Groups (8) info" and a search bar "Find security groups by attribute or tag". Below is a table with columns: Name, Security group ID, Security group name, VPC ID, Description, Owner, and Inbound. The table lists the following security groups:

Name	Security group ID	Security group name	VPC ID	Description	Owner	Inbound
-	sg-0122edf575e9a19e	launch-wizard-2	vpc-0e6feb31f67cea745	launch-wizard-2 created 2025-12-19T0...	581483106531	3 Perr
-	sg-0975657e26ccb75d	wordpress	vpc-0e6feb31f67cea745	wordpress created 2025-11-25T12:48:0...	581483106531	3 Perr
-	sg-092caeef08e340e39	default	vpc-0e6feb31f67cea745	default VPC security group	581483106531	1 Perr
-	sg-0ba8e498hbfb5c8	db-sg	vpc-05e79e19cf17eb848	db-sg 2025-12-31T09:02:29.667Z	581483106531	2 Perr
-	sg-0a84e25b4fdab4c4d	web-sg	vpc-05e79e19cf17eb848	Web Server (Public)31/12/25	581483106531	2 Perr
-	sg-0b3b16f52e19b325	launch-wizard-1	vpc-0e6feb31f67cea745	launch-wizard-1 created 2025-12-19T0...	581483106531	3 Perr
-	sg-00220895c58bbcd3	default	vpc-05e79e19cf17eb848	default VPC security group	581483106531	1 Perr
-	sg-0a03a5e52dd79a636	default	vpc-00e788094677f0eab	default VPC security group	581483106531	1 Perr

At the bottom, there's a "Select a security group" dropdown and a note "Activate Windows Go to Settings to activate Windows." The footer includes links for CloudShell, Feedback, Console Mobile App, and various AWS services.

11.security Group

```
[ec2-user@ip-10-0-2-232:~] Cleaning up...
[ec2-user@ip-10-0-1-211 ~]$ mysql -h 10.0.2.232 -u lokha_user -p
Enter password:
Welcome to the MariaDB monitor. Commands end with ; or \g.
Your MariaDB connection id is 19
Server version: 10.5.29-MariaDB MariaDB Server

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)]> EXIT
Bye
[ec2-user@ip-10-0-1-211 ~]$ ssh -i ~/key191225.pem ec2-user@10.0.2.232
      #_
      _\###_          Amazon Linux 2023
      _\###_\_
      \###|
      \#/   https://aws.amazon.com/linux/amazon-linux-2023
      V~'`->
      /_
      /`_/
      /m/'_
Last login: Wed Dec 31 12:32:01 2025 from 10.0.1.211
[ec2-user@ip-10-0-2-232 ~]$ ssh -i ~/key191225.pem ec2-user@10.0.2.232
^C
[ec2-user@ip-10-0-2-232 ~]$ mysql -u root -p
Enter password:
Welcome to the MariaDB monitor. Commands end with ; or \g.
Your MariaDB connection id is 20
Server version: 10.5.29-MariaDB MariaDB Server

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)]> SELECT user, host FROM mysql.user;
+-----+-----+
| User      | Host      |
+-----+-----+
| lokha_user | %        |
| mariadb.sys | localhost |
| mysql      | localhost |
| root       | localhost |
+-----+-----+
4 rows in set (0.001 sec)

MariaDB [(none)]> 
```

12.ssh to db server

```

ec2-user@ip-10-0-2-232:~$ Cleaning up...
[ec2-user@ip-10-0-1-211 ~]$ mysql -h 10.0.2.232 -u lokha_user -p
Enter password:
Welcome to the MariaDB monitor. Commands end with ; or \g.
Your MariaDB connection id is 19
Server version: 10.5.29-MariaDB MariaDB Server

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)]> EXIT
Bye
[ec2-user@ip-10-0-1-211 ~]$ ssh -i ~/key191225.pem ec2-user@10.0.2.232
      #
      # Amazon Linux 2023
      #\###\#
      \###\#
      \###\#
      \#/ https://aws.amazon.com/linux/amazon-linux-2023
      V~'-'-
      /-
      /-
      /m/-
Last login: Wed Dec 31 12:32:01 2025 from 10.0.1.211
[ec2-user@ip-10-0-2-232 ~]$ ssh -i ~/key191225.pem ec2-user@10.0.2.232
^C
[ec2-user@ip-10-0-2-232 ~]$ mysql -u root -p
Enter password:
Welcome to the MariaDB monitor. Commands end with ; or \g.
Your MariaDB connection id is 20
Server version: 10.5.29-MariaDB MariaDB Server

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)]> SELECT user, host FROM mysql.user;
+-----+-----+
| User      | Host     |
+-----+-----+
| lokha_user | %       |
| mariadb.sys | localhost |
| mysql      | localhost |
| root       | localhost |
+-----+-----+
4 rows in set (0.001 sec)

MariaDB [(none)]> 
```

13.checking user exits

```
ec2-user@ip-10-0-1-211:~  
Enter current password for root (enter for none):  
OK, successfully used password, moving on...  
  
Setting the root password or using the unix_socket ensures that nobody  
can log into the MariaDB root user without the proper authorisation.  
  
You already have your root account protected, so you can safely answer 'n'.  
  
Switch to unix_socket authentication [Y/n] ^C  
Aborting!  
  
Cleaning up...  
[ec2-user@ip-10-0-1-211 ~]$ mysql -h 10.0.2.232 -u lokha_user -p  
Enter password:  
Welcome to the MariaDB monitor. Commands end with ; or \g.  
Your MariaDB connection id is 19  
Server version: 10.5.29-MariaDB MariaDB Server  
  
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)]> 
```

14.connecting to db server via server1

7. Cost Management & Cleanup

Charged Resources

- EC2 instances
- NAT Gateway
- Elastic IP

Cleanup Actions

- Terminated EC2 instances
- Deleted NAT Gateway
- Released Elastic IP

Free Resources (no cost):

- VPC
- Subnets
- Route Tables
- Security Groups

8. Final Outcome

This project successfully demonstrates:

- Secure AWS networking
 - Real-world two-tier architecture
 - Controlled database access
 - Outbound-only internet from private subnet
 - Practical troubleshooting skills
-

9. Interview-Ready Summary

"I deployed a two-tier AWS architecture with a public web server and a private database server. The database has no public IP and is accessible only from the web server using security group references. A NAT Gateway provides outbound internet access for patching while maintaining isolation."

10. Conclusion

This project provides a strong foundation in AWS networking, security, and application deployment, closely matching real production environments and interview expectations.