

# Step by step guide

## Phase 1

## 1. Create a VPC

Go to: **VPC Dashboard** → **Your VPCs** → **Create VPC**

Field	Value
Name tag	ThreeTier-VPC
IPv4 CIDR block	10.0.0.0/16
Tenancy	Default
IPv6 CIDR block	No IPv6 for now
Enable DNS Hostnames	Yes (important for public DNS resolution)

The screenshot shows the AWS VPC console interface. The main content area displays a table of VPCs. The table has columns for Name, VPC ID, State, Block Public..., and IPv4 CIDR. Two VPCs are listed: 'default' and 'ThreeTier-VPC'. The 'ThreeTier-VPC' is highlighted. The left sidebar shows the navigation menu with 'Virtual private cloud' expanded. The top bar shows the AWS logo and the current region 'us-east-2'.

Name	VPC ID	State	Block Public...	IPv4 CIDR
default	vpc-043960a894aa64bf2	Available	Off	172.31.0.0/16
ThreeTier-VPC	vpc-0401b7102745ad69a	Available	Off	10.0.0.0/16

## 2. Create Subnets

1. Go to **VPC Dashboard** → **Subnets** → **Create subnet**
2. Select your VPC: ThreeTier-VPC
3. Create **one at a time**:
  - **Name Tag**: Use names from table above (e.g., Web-Pub-1)
  - **Availability Zone**: Choose us-east-1a, us-east-1b, etc.
  - **IPv4 CIDR**: e.g., 10.0.1.0/24

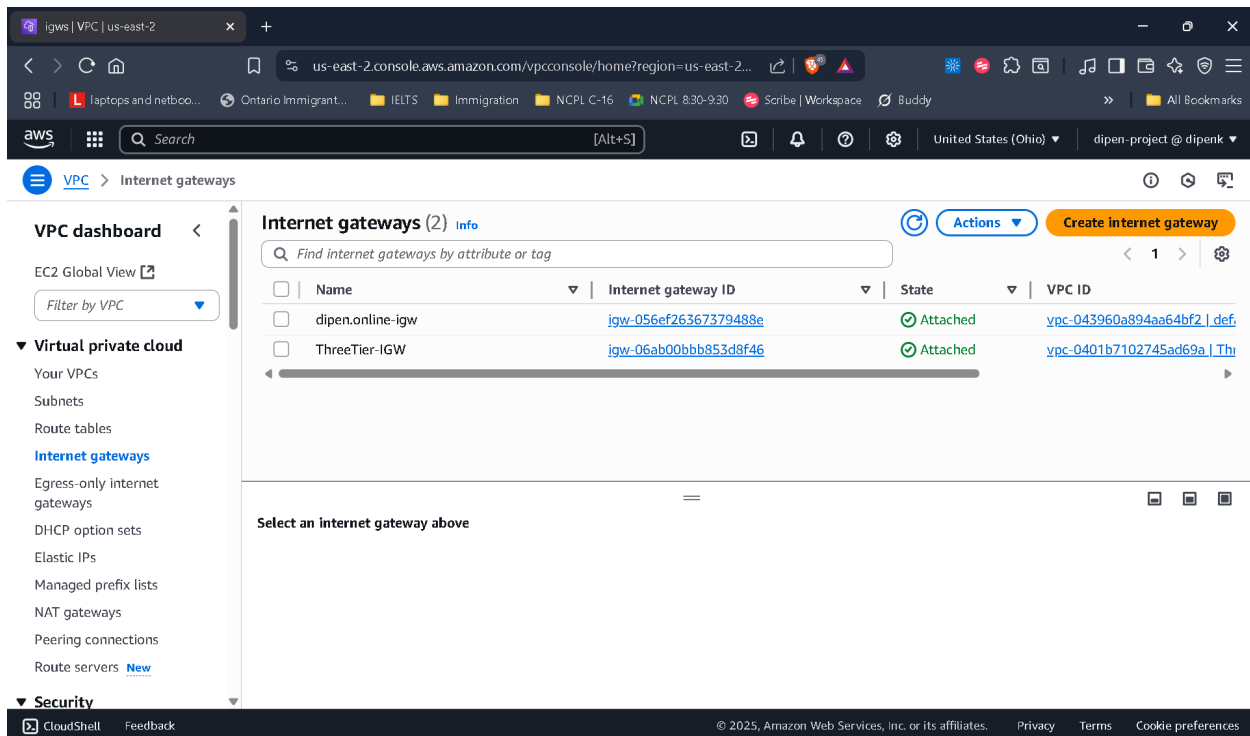
Tier	AZ	Subnet Name	CIDR Block	Type
Web Tier	AZ1	Web-Pub-1	10.0.1.0/24	Public
Web Tier	AZ2	Web-Pub-2	10.0.2.0/24	Public
App Tier	AZ1	App-Priv-1	10.0.3.0/24	Private
App Tier	AZ2	App-Priv-2	10.0.4.0/24	Private
DB Tier	AZ1	DB-Priv-1	10.0.5.0/24	Private
DB Tier	AZ2	DB-Priv-2	10.0.6.0/24	Private

The screenshot shows the AWS VPC Subnets console for the us-east-2 region. The left sidebar contains the VPC dashboard menu with options like EC2 Global View, Filter by VPC, Virtual private cloud, Your VPCs, Subnets, Route tables, Internet gateways, Egress-only internet gateways, DHCP option sets, Elastic IPs, Managed prefix lists, NAT gateways, Peering connections, and Route servers. The main content area displays a table of 8 subnets, all in an 'Available' state. The table columns are Name, Subnet ID, State, and VPC. The subnets listed are App-Priv-1, App-Priv-2, DB-Priv-1, DB-Priv-2, default, default, Web-Pub-1, and Web-Pub-2. At the bottom, there is a 'Select a subnet' button.

Name	Subnet ID	State	VPC
App-Priv-1	subnet-07eb8462b470d979a	Available	vpc-0401b7102745ad69a   Thre...
App-Priv-2	subnet-06efb4847f30888fc	Available	vpc-0401b7102745ad69a   Thre...
DB-Priv-1	subnet-04096aba77919a02f	Available	vpc-0401b7102745ad69a   Thre...
DB-Priv-2	subnet-0e1b8ad794e329876	Available	vpc-0401b7102745ad69a   Thre...
default	subnet-01ac3f419e579bfad	Available	vpc-043960a894aa64bf2   default
default	subnet-04fc9f8dd406566f0	Available	vpc-043960a894aa64bf2   default
Web-Pub-1	subnet-0c2b69a1713d53c31	Available	vpc-0401b7102745ad69a   Thre...
Web-Pub-2	subnet-05258f856060d7ed9	Available	vpc-0401b7102745ad69a   Thre...

### 3. Create and Attach IGW

1. Go to **VPC Dashboard** → **Internet Gateways** → **Create Internet Gateway**
2. **Name:** ThreeTier-IGW
3. Click **Create Internet Gateway**
4. After creation, **select it** → **Actions** → **Attach to VPC**
5. Choose your VPC: ThreeTier-VPC



The screenshot shows the AWS VPC console interface. The left sidebar contains the 'VPC dashboard' with a search bar and a list of resources under 'Virtual private cloud' and 'Security'. The main content area is titled 'Internet gateways (2)' and includes a search bar and a table of existing gateways. The table lists two gateways: 'dipen.online-igw' and 'ThreeTier-IGW', both with a state of 'Attached'. Below the table, there is a prompt to 'Select an internet gateway above'.

<input type="checkbox"/>	Name	Internet gateway ID	State	VPC ID
<input type="checkbox"/>	dipen.online-igw	<a href="#">igw-056ef26367379488e</a>	Attached	<a href="#">vpc-043960a894aa64bf2</a>   <a href="#">def</a>
<input type="checkbox"/>	ThreeTier-IGW	<a href="#">igw-06ab00bbb853d8f46</a>	Attached	<a href="#">vpc-0401b7102745ad69a</a>   <a href="#">Th</a>

## 4. Public Route Table

1. **VPC Dashboard** → **Route Tables** → **Create Route Table**
2. **Name:** Public-RT
3. **VPC:** Select ThreeTier-VPC
4. Click **Create**
5. Select Public-RT → **Routes tab** → **Edit routes**
6. Click **Add route:**
  - **Destination:** 0.0.0.0/0
  - **Target:** Choose the Internet Gateway (ThreeTier-IGW)
7. Click **Save changes**
8. Now go to **Subnet Associations** → **Edit subnet associations**
  - Select: Web-Pub-1 and Web-Pub-2
  - Click **Save**

The screenshot shows the AWS Management Console interface for Route Tables in the us-east-2 region. The left sidebar displays the VPC dashboard with a filter by VPC. The main content area shows a list of route tables, including 'default', '-', and 'Public-RT'. The 'Public-RT' route table is selected, and its details are shown below the list. The details include the route table ID, explicit subnet associations, edge associations, and a main status. The 'Public-RT' route table has 2 subnets associated with it.

Name	Route table ID	Explicit subnet associ...	Edge associations	Main
default	<a href="#">rtb-050ee44538e7afde1</a>	-	-	Yes
-	<a href="#">rtb-00328f8bd35317a13</a>	-	-	Yes
Public-RT	<a href="#">rtb-009d503d7029ac683</a>	2 subnets	-	No

Select a route table

VPC | us-east-2

us-east-2.console.aws.amazon.com/vpcconsole/home?region=us-east-2...

Search [Alt+S]

United States (Ohio) | dipen-project @ dipenk

VPC > Route tables > rtb-009d503d7029ac683

rtb-009d503d7029ac683 / Public-RT

Actions

VPC dashboard

EC2 Global View

Filter by VPC

Virtual private cloud

Your VPCs

Subnets

Route tables

Internet gateways

Egress-only internet gateways

DHCP option sets

Elastic IPs

Managed prefix lists

NAT gateways

Peering connections

Route servers

Details

Route table ID

rtb-009d503d7029ac683

VPC

vpc-0401b7102745ad69a | ThreeTier-VPC

Main

No

Owner ID

167611893883

Explicit subnet associations

2 subnets

Edge associations

-

Routes

Subnet associations

Edge associations

Route propagation

Tags

Routes (2)

Filter routes

Both

Edit routes

1

Destination

Target

Status

Propagated

0.0.0.0/0

igw-06ab00bbb853d8f46

Active

No

10.0.0.0/16

local

Active

No

CloudShell

Feedback

© 2025, Amazon Web Services, Inc. or its affiliates.

Privacy

Terms

Cookie preferences

VPC | us-east-2

us-east-2.console.aws.amazon.com/vpcconsole/home?region=us-east-2...

Search [Alt+S]

United States (Ohio) | dipen-project @ dipenk

VPC > Route tables > rtb-009d503d7029ac683

rtb-009d503d7029ac683 / Public-RT

Actions

VPC dashboard

EC2 Global View

Filter by VPC

Virtual private cloud

Your VPCs

Subnets

Route tables

Internet gateways

Egress-only internet gateways

DHCP option sets

Elastic IPs

Managed prefix lists

NAT gateways

Peering connections

Route servers

Details

Route table ID

rtb-009d503d7029ac683

VPC

vpc-0401b7102745ad69a | ThreeTier-VPC

Main

No

Owner ID

167611893883

Explicit subnet associations

2 subnets

Edge associations

-

Routes

Subnet associations

Edge associations

Route propagation

Tags

Explicit subnet associations (2)

Find subnet association

Edit subnet associations

1

Name

Subnet ID

IPv4 CIDR

IPv6 CIDR

Web-Pub-2

subnet-05258f856060d7ed9

10.0.2.0/24

-

Web-Pub-1

subnet-0c2b69a1713d53c31

10.0.1.0/24

-

CloudShell

Feedback

© 2025, Amazon Web Services, Inc. or its affiliates.

Privacy

Terms

Cookie preferences

## 5. Private Route Table

1. Back in **Route Tables** → **Create Route Table**
2. **Name:** Private-RT
3. **VPC:** Select ThreeTier-VPC
4. Click **Create**
5. **No need to add route to IGW**
6. Go to **Subnet Associations** → **Edit subnet associations**
  - Select: App-Priv-1, App-Priv-2, DB-Priv-1, DB-Priv-2
  - Click **Save**

The screenshot shows the AWS Management Console interface for Route Tables in the us-east-2 region. The left sidebar contains navigation links for VPC dashboard, EC2 Global View, and Virtual private cloud resources. The main content area displays a list of route tables. The 'Private-RT' route table is selected, and its subnet associations are shown in a detailed view below.

Name	Route table ID	Explicit subnet associ...	Edge associations	Main
<input type="checkbox"/> default	<a href="#">rtb-050ee44538e7afde1</a>	-	-	Yes
<input type="checkbox"/> -	<a href="#">rtb-00328f8bd35317a13</a>	-	-	Yes
<input type="checkbox"/> Public-RT	<a href="#">rtb-009d503d7029ac683</a>	2 subnets	-	No
<input checked="" type="checkbox"/> Private-RT	<a href="#">rtb-0fb93d7615ee05371</a>	4 subnets	-	No

Name	Subnet ID	IPv4 CIDR	IPv6 CIDR
App-Priv-1	<a href="#">subnet-07eb8462b470d979a</a>	10.0.3.0/24	-
DB-Priv-2	<a href="#">subnet-0e1b8ad794e329876</a>	10.0.6.0/24	-
DB-Priv-1	<a href="#">subnet-04096aba77919a02f</a>	10.0.5.0/24	-
App-Priv-2	<a href="#">subnet-06efb4847f30888fc</a>	10.0.4.0/24	-

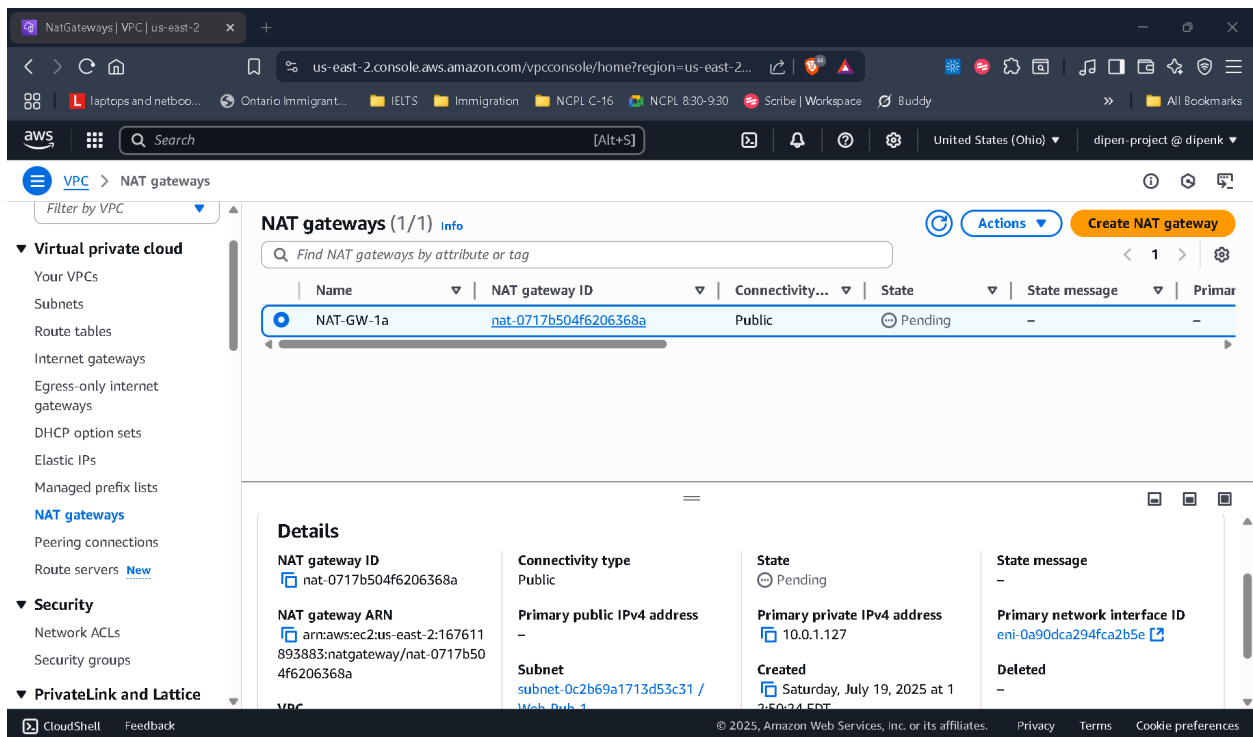
[ Web-Pub-1, Web-Pub-2 ] --> Public-RT --> IGW --> Internet

[ App-Priv-1..DB-Priv-2 ] --> Private-RT --> (internal only)

## Create a NAT Gateway

1. Go to **VPC Dashboard** → **NAT Gateways** → **Create NAT Gateway**
2. **Name:** NAT-GW-AZ1
3. **Subnet:** Select Web-Pub-1 (public subnet)
4. **Elastic IP:** Click “Allocate Elastic IP” → choose it
5. Click **Create NAT Gateway**

Wait for status to show as Available.



The screenshot shows the AWS VPC console for the us-east-2 region. The left sidebar lists various VPC resources, with 'NAT gateways' selected. The main panel displays a table of NAT gateways with one entry, 'NAT-GW-1a', which is in a 'Pending' state. Below the table, the 'Details' section provides specific information about the gateway.

Name	NAT gateway ID	Connectivity...	State	State message	Primary
NAT-GW-1a	nat-0717b504f6206368a	Public	Pending	-	-

Details	
<b>NAT gateway ID</b> nat-0717b504f6206368a	<b>Connectivity type</b> Public
<b>NAT gateway ARN</b> arn:aws:ec2:us-east-2:167611893883:natgateway/nat-0717b504f6206368a	<b>Primary public IPv4 address</b> -
<b>Subnet</b> subnet-0c2b69a1713d53c31 / Web-Pub-1	<b>Primary private IPv4 address</b> 10.0.1.127
<b>Created</b> Saturday, July 19, 2025 at 1:50:24 EDT	<b>Primary network interface ID</b> eni-0a90dca294fca2b5e
<b>Deleted</b> -	



## Update Private Route Table to Use NAT

1. Go to **Route Tables** → **Select Private-RT**
2. Click **Routes tab** → **Edit routes**
3. **Add route:**
  - **Destination:** 0.0.0.0/0
  - **Target:** Your newly created **NAT Gateway**

Click **Save Changes**

The screenshot shows the AWS Management Console interface for the 'Route tables' section. The left sidebar lists various VPC resources, with 'Route tables' selected. The main content area displays a list of route tables. The 'Private-RT' (rtb-0fb93d7615ee05371) is selected, and its 'Routes' tab is active. The 'Routes' table shows two entries: a default route (0.0.0.0/0) pointing to a NAT gateway (nat-0717b504f6206368a) and a local route (10.0.0.0/16) pointing to 'local'. Both routes are 'Active' and 'Propagated'.

**Route tables (1/4)** Info

Last updated 1 minute ago

Find route tables by attribute or tag

Name	Route table ID	Explicit subnet associ...	Edge associations	Main
default	rtb-050ee44538e7afde1	-	-	Yes
-	rtb-00328f8bd35317a13	-	-	Yes
Public-RT	rtb-009d503d7029ac683	2 subnets	-	No
<input checked="" type="checkbox"/> Private-RT	rtb-0fb93d7615ee05371	4 subnets	-	No

**rtb-0fb93d7615ee05371 / Private-RT**

Routes (2)

Filter routes

Destination	Target	Status	Propagated
0.0.0.0/0	nat-0717b504f6206368a	Active	No
10.0.0.0/16	local	Active	No

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

Phase 2

## SG Design for 3-Tier Architecture

SG Name	Attached To	Inbound Rules From	Outbound
SG-Web	Web EC2 / ALB	0.0.0.0/0: HTTP (80), SSH (22)	All traffic
SG-App	App EC2	SG-Web: TCP 3000 (custom app port)	All traffic
SG-DB	RDS DB	SG-App: TCP 3306 (MySQL/Aurora)	All traffic

## Create Security Groups

### SG-Web (Web Tier)

1. Go to **EC2** → **Security Groups** → **Create Security Group**
2. **Name:** SG-Web
3. **VPC:** ThreeTier-VPC
4. **Inbound Rules:**
  - Type: HTTP | Port: 80 | Source: 0.0.0.0/0
  - Type: SSH | Port: 22 | Source: your IP (My IP) (*or restrict it later*)
5. **Outbound:** Allow all (default)

The screenshot shows the AWS Management Console for the 'us-east-2' region, specifically the 'Security Groups' page. The selected security group is 'sg-0383516e36434d0c0 - SG-Web'. The console displays the following details:

- Details:**
  - Security group name: SG-Web
  - Security group ID: sg-0383516e36434d0c0
  - Description: SG-Web
  - VPC ID: vpc-0401b7102745ad69a
  - Owner: 167611893883
  - Inbound rules count: 2 Permission entries
  - Outbound rules count: 1 Permission entry
- Inbound rules (2):**
  - Rule 1: Name '-', Security group rule ID 'sgr-0d3e97d2281dd51dc', IP version 'IPv4', Type 'SSH', Protocol 'TCP', Port range '22'.
  - Rule 2: Name '-', Security group rule ID 'sgr-0c246fc5ff387dc58', IP version 'IPv4', Type 'HTTP', Protocol 'TCP', Port range '80'.

The console also shows tabs for 'Outbound rules', 'Sharing - new', 'VPC associations - new', and 'Tags'. At the bottom, there are links for 'CloudShell', 'Feedback', and '© 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences'.

## SG-App (App Tier)

1. **Name:** SG-App
2. **Inbound Rules:**
  - Type: Custom TCP | Port: 3000 (*or your backend port*) | Source: SG-Web
3. **Outbound:** Allow all

The screenshot displays the AWS Management Console interface for a Security Group. The breadcrumb navigation shows the path: EC2 > Security Groups > sg-07c08ed628e388e7d - SG-App. The main heading is "sg-07c08ed628e388e7d - SG-App" with an "Actions" button. Below this, a "Details" section provides key information:

<b>Security group name</b> SG-App	<b>Security group ID</b> sg-07c08ed628e388e7d	<b>Description</b> SG-App	<b>VPC ID</b> vpc-0401b7102745ad69a
<b>Owner</b> 167611893883	<b>Inbound rules count</b> 1 Permission entry	<b>Outbound rules count</b> 1 Permission entry	

Below the details, there are tabs for "Inbound rules", "Outbound rules", "Sharing - new", "VPC associations - new", and "Tags". The "Inbound rules" tab is active, showing a table with one rule:

IP version	Type	Protocol	Port range	Source	Description
5	Custom TCP	TCP	3000	sg-0383516e36434d0c0...	-

At the bottom of the console, there is a footer with "CloudShell", "Feedback", and copyright information for Amazon Web Services, Inc. or its affiliates, along with links for "Privacy", "Terms", and "Cookie preferences".

SG-DB (Database Tier)

- 1. Name: SG-DB
- 2. Inbound Rules:
  - Type: MySQL/Aurora | Port: 3306 | Source: SG-App
- 3. Outbound: Allow all

RouteTables | VPC | us-east-2

SecurityGroup | EC2 | us-east-2

us-east-2.console.aws.amazon.com/ec2/home?region=us-east-2#SecurityGro...

laptops and netboo... Ontario Immigrant... IELTS Immigration NCPL C-16 NCPL 8:30-9:30 Scribe | Workspace Buddy All Bookmarks

aws Search [Alt+S] United States (Ohio) dipen-project @ dipenk

EC2 > Security Groups > sg-01d15615a2a964e42 - SG-DB

sg-01d15615a2a964e42 - SG-DB

Actions

Details

Security group name  
SG-DB

Owner  
167611893883

Security group ID  
sg-01d15615a2a964e42

Inbound rules count  
1 Permission entry

Description  
SG-DB

Outbound rules count  
1 Permission entry

VPC ID  
vpc-0401b7102745ad69a

Inbound rules Outbound rules Sharing - new VPC associations - new Tags

Inbound rules (1)

Manage tags Edit inbound rules

Search

IP version	Type	Protocol	Port range	Source	Description
-	MySQL/Aurora	TCP	3306	sg-07c08ed628e388e7d...	-

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

RouteTables | VPC | us-east-2

SecurityGroups | EC2 | us-east-2

us-east-2.console.aws.amazon.com/ec2/home?region=us-east-2#SecurityGro...

laptops and netboo... Ontario Immigrant... IELTS Immigration NCPL C-16 NCPL 8:30-9:30 Scribe | Workspace Buddy All Bookmarks

aws Search [Alt+S] United States (Ohio) dipen-project @ dipenk

EC2 > Security Groups

Security Groups (9)

Find security groups by attribute or tag

Actions

Export security groups to CSV

Create security group

	Name	Security group ID	Security group name	VPC ID	Description
<input type="checkbox"/>	SG-Web	sg-0383516e3643d0c0	SG-Web	vpc-0401b7102745ad69a	SG-Web
<input type="checkbox"/>	SG-DB	sg-01d15615a2a964e42	SG-DB	vpc-0401b7102745ad69a	SG-DB
<input type="checkbox"/>	SG-App	sg-07c08ed628e388e7d	SG-App	vpc-0401b7102745ad69a	SG-App
<input type="checkbox"/>	-	sg-0beff7802da47532a	default	vpc-043960a894aa64bf2	default VPC sec
<input type="checkbox"/>	-	sg-002e8a8938fa1a8ac	WebServSecurityGroup	vpc-043960a894aa64bf2	WebServSG
<input type="checkbox"/>	-	sg-0ce926b21f2192656	default	vpc-0401b7102745ad69a	default VPC sec
<input type="checkbox"/>	-	sg-0db5868aa8ad47998	NLBSecurityGroup	vpc-043960a894aa64bf2	NLBSecurityGrc

Select a security group

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

## Public NACL

1. Go to **VPC Dashboard** → **Network ACLs** → **Create NACL**
2. **Name:** Public-NACL
3. **VPC:** ThreeTier-VPC
4. Click **Create**
5. Click on the new NACL → **Subnet Associations**
  - Associate: Web-Pub-1, Web-Pub-2

Rule #	Type	Protocol	Port Range	Source	Allow/Deny
100	HTTP	TCP	80	0.0.0.0/0	ALLOW
110	SSH	TCP	22	your IP CIDR	ALLOW
120	Ephemeral	TCP	1024–65535	0.0.0.0/0	ALLOW
*	ALL	ALL	ALL	0.0.0.0/0	DENY

Rule #	Type	Protocol	Port Range	Destination	Allow/Deny
100	HTTP	TCP	80	0.0.0.0/0	ALLOW
110	Ephemeral	TCP	1024–65535	0.0.0.0/0	ALLOW
*	ALL	ALL	ALL	0.0.0.0/0	DENY

VPC | us-east-2

SecurityGroups | EC2 | us-east-2

us-east-2.console.aws.amazon.com/vpcconsole/home?region=us-east-2#Net...

laptops and netboo...

Ontario Immigrant...

IELTS

Immigration

NCPL C-16

NCPL 8:30-9:30

Scribe | Workspace

Buddy

All Bookmarks

aws

Search

[Alt+S]

United States (Ohio)

dipen-project @ dipenk

VPC

Network ACLs

acl-0a52a1d8641ffa3fc / Public-NACL

Actions

Details

info

Network ACL ID

acl-0a52a1d8641ffa3fc

Associated with

2 Subnets

Default

No

VPC ID

vpc-0401b7102745ad69a / ThreeTier-VPC

Owner

167611893883

Inbound rules

Outbound rules

Subnet associations

Tags

Subnet associations (2)

Edit subnet associations

Filter subnet associations

< 1 >

Name	Subnet ID	Associated with	Availability Zone	IPv4 CIDR	IPv6 CIDR
Web-Pub-1	subnet-0c2b69a1713d5...	acl-0a52a1d8641ffa3fc / Public-NACL	us-east-2a	10.0.1.0/24	–
Web-Pub-2	subnet-05258f856060d...	acl-0a52a1d8641ffa3fc / Public-NACL	us-east-2b	10.0.2.0/24	–

CloudShell

Feedback

© 2025, Amazon Web Services, Inc. or its affiliates.

Privacy

Terms

Cookie preferences



## Private NACL

Repeat above process for Private-NACL and associate:

- App-Priv-1
- App-Priv-2
- DB-Priv-1
- DB-Priv-2

### Inbound rules

Rule #	Type	Port Range	Source	Allow/Deny
100	Custom TCP	3000	10.0.1.0/24	ALLOW
110	MySQL	3306	10.0.3.0/24	ALLOW
120	Ephemeral	1024–65535	10.0.1.0/24	ALLOW
*	ALL	ALL	0.0.0.0/0	DENY

### Outbound rules

Rule #	Type	Port Range	Destination	Allow/Deny
100	Ephemeral	1024–65535	0.0.0.0/0	ALLOW
*	ALL	ALL	0.0.0.0/0	DENY



VPC | us-east-2

SecurityGroups | EC2 | us-east-2

+

us-east-2.console.aws.amazon.com/vpcconsole/home?region=us-east-2#Net...

laptops and netboo... Ontario Immigrant... IELTS Immigration NCPL C-16 NCPL 830-930 Scribe | Workspace Buddy All Bookmarks

aws Search [Alt+S]

United States (Ohio) dipen-project @ dipenk

VPC > Network ACLs > acl-0ccbea2889fd11d1d / Private-NACL

Details info

Network ACL ID

acl-0ccbea2889fd11d1d

Associated with

4 Subnets

Default

No

VPC ID

vpc-0401b7102745ad69a / ThreeTier-VPC

Owner

167611893883

Inbound rules

Outbound rules

Subnet associations

Tags

Subnet associations (4)

Filter subnet associations

Edit subnet associations

Name	Subnet ID	Associated with	Availability Zone	IPv4 CIDR	IPv6 CIDR
App-Priv-2	subnet-06efb4847f3088...	acl-0ccbea2889fd11d1d / Private-NACL	us-east-2b	10.0.4.0/24	-
DB-Priv-2	subnet-0e1b8ad794e32...	acl-0ccbea2889fd11d1d / Private-NACL	us-east-2b	10.0.6.0/24	-
App-Priv-1	subnet-07eb8462b470d...	acl-0ccbea2889fd11d1d / Private-NACL	us-east-2a	10.0.3.0/24	-
DB-Priv-1	subnet-04096aba77919...	acl-0ccbea2889fd11d1d / Private-NACL	us-east-2a	10.0.5.0/24	-

CloudShell Feedback

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

VPC | us-east-2

SecurityGroups | EC2 | us-east-2

+

us-east-2.console.aws.amazon.com/vpcconsole/home?region=us-east-2#Net...

laptops and netboo... Ontario Immigrant... IELTS Immigration NCPL C-16 NCPL 830-930 Scribe | Workspace Buddy All Bookmarks

aws Search [Alt+S]

United States (Ohio) dipen-project @ dipenk

VPC > Network ACLs > acl-0ccbea2889fd11d1d / Private-NACL

Details info

Network ACL ID

acl-0ccbea2889fd11d1d

Associated with

4 Subnets

Default

No

VPC ID

vpc-0401b7102745ad69a / ThreeTier-VPC

Inbound rules

Outbound rules

Subnet associations

Tags

Inbound rules (4)

Filter inbound rules

Edit inbound rules

Rule number	Type	Protocol	Port range	Source	Allow/Deny
100	Custom TCP	TCP (6)	3000	10.0.1.0/24	Allow
110	MySQL/Aurora (3306)	TCP (6)	3306	10.0.3.0/24	Allow
120	Custom TCP	TCP (6)	1024 - 65535	10.0.1.0/24	Allow
*	All traffic	All	All	0.0.0.0/0	Deny

CloudShell Feedback

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



Actions ▼

VPC ID  
vpc-0401b7102745ad69a / ThreeTier-VPC

**Inbound rules**      **Outbound rules**      Subnet associations      Tags

[Edit outbound rules](#)

 *Filter outbound rules*

Rule number	Type	Protocol	Port range	Destination	Allow/Deny
100	Custom TCP	TCP (6)	1024 - 65535	0.0.0.0/0	 Allow
*	All traffic	All	All	0.0.0.0/0	 Deny

## Create a Key Pair (for SSH into EC2)

1. Go to **EC2 Dashboard** → **Key Pairs** → **Create Key Pair**
2. **Name:** ThreeTier-KeyPair
3. **Key pair type:** RSA (recommended for SSH)
4. **Private key format:** .pem (for Linux/Mac)
5. Click **Create key pair**
6. File will download as: ThreeTier-KeyPair.pem

The screenshot shows the AWS Management Console for the 'us-east-2' region, specifically the 'Key pairs' page. A green success banner at the top indicates 'Successfully created key pair'. Below this, the 'Key pairs (2)' section shows a table with two entries. The first entry is 'c161' with type 'rsa' and creation time '2025/07/11 10:54 GMT-4'. The second entry is 'ThreeTier-KeyPair' with type 'rsa' and creation time '2025/07/19 13:22 GMT-4'. The table also shows fingerprints and IDs for each key pair. At the bottom of the console, there are links for 'CloudShell', 'Feedback', and copyright information for Amazon Web Services, Inc.

<input type="checkbox"/>	Name	Type	Created	Fingerprint	ID
<input type="checkbox"/>	c161	rsa	2025/07/11 10:54 GMT-4	03:e7:cc:f3:28:81:5e:65:46:19:7b:3b:...	key-038c6e7c6b77f8d07
<input type="checkbox"/>	ThreeTier-KeyPair	rsa	2025/07/19 13:22 GMT-4	88:bf:2e:ced4:55:a8:3f:42:c7:9d:bc:...	key-039870d12d9e8b803

Phase 3

# Launching Ec2 Instances

## Public Instance:

The screenshot displays the AWS Management Console for a public EC2 instance. The instance is named 'i-09ec1a7dc485d468b' and is in the 'us-east-2' region. The instance summary shows it is a 'bastion-host-p2' type, running on an 'm5.xlarge' instance type. The instance is in the 'Running' state. The public IPv4 address is '3.15.196.240'. The private IP DNS name is 'ip-10-0-2-91.us-east-2.compute.internal'. The instance is associated with the 'vpc-0159638b511253570' VPC and the 'subnet-904f8f08b5d5a8d2f' subnet. The instance is associated with the 'arn:aws:ec2:us-east-2:167611893883:instance/i-09ec1a7dc485d468b' ARN. The instance is associated with the 'ami-0eb9d6f9fab44c24' AMI and the 'al2023-ami-2023.8.20250707.0-kernel-6.1-x86\_64' AMI name. The instance is associated with the 'stop-protection-disabled' stop protection. The instance is associated with the 'monitoring-disabled' monitoring. The instance is associated with the 'platform-linux/unix' platform details. The instance is associated with the 'termination-protection-disabled' termination protection. The instance is associated with the 'amazon/al2023-ami-2023.8.20250707.0-kernel-6.1-x86\_64' AMI location. The instance is associated with the 'managed-false' managed state.

## App-tier Instance:

The screenshot displays the AWS Management Console for an app-tier EC2 instance. The instance is named 'i-01ee36f84fb8a4854' and is in the 'us-east-2' region. The instance summary shows it is an 'app-tier-p2' type, running on an 'm5.xlarge' instance type. The instance is in the 'Running' state. The public IPv4 address is '10.0.4.178'. The private IP DNS name is 'ip-10-0-4-178.us-east-2.compute.internal'. The instance is associated with the 'vpc-0159638b511253570' VPC and the 'subnet-904f8f08b5d5a8d2f' subnet. The instance is associated with the 'arn:aws:ec2:us-east-2:167611893883:instance/i-01ee36f84fb8a4854' ARN. The instance is associated with the 'ami-0eb9d6f9fab44c24' AMI and the 'al2023-ami-2023.8.20250707.0-kernel-6.1-x86\_64' AMI name. The instance is associated with the 'stop-protection-disabled' stop protection. The instance is associated with the 'monitoring-disabled' monitoring. The instance is associated with the 'platform-linux/unix' platform details. The instance is associated with the 'termination-protection-disabled' termination protection. The instance is associated with the 'amazon/al2023-ami-2023.8.20250707.0-kernel-6.1-x86\_64' AMI location. The instance is associated with the 'managed-false' managed state.

## DB-tier-Instance:

The screenshot displays the AWS Management Console interface for an EC2 instance. The browser address bar shows the URL: `us-east-2.console.aws.amazon.com/ec2/home?region=us-east-2#InstanceDetails:instanceId=i-042560431f7fb3be3`. The console header includes the AWS logo, a search bar, and the user's profile information (United States (Ohio), dipen-project @ dipenk).

The left-hand navigation pane is expanded to show the 'Instances' section under 'EC2'. The main content area displays the 'Instance summary for i-042560431f7fb3be3 (db-tier-p2)'. The summary is organized into three columns:

- Instance ID:** i-042560431f7fb3be3
- Instance state:** Running
- Private IPv4 addresses:** 10.0.6.125
- IPV6 address:** -
- Private IP DNS name (IPv4 only):** ip-10-0-6-125.us-east-2.compute.internal
- Public DNS:** -
- Hostname type:** IP name: ip-10-0-6-125.us-east-2.compute.internal
- Answer private resource DNS name:** -
- Elastic IP addresses:** -
- Auto-assigned IP address:** -
- Instance type:** t2.micro
- IAM Role:** -
- VPC ID:** vpc-0159683b511253570 (VPC-P2)
- Subnet ID:** subnet-0eb7084ae5e4adbee
- IMDSv2:** Required
- Instance ARN:** arn:aws:ec2:us-east-2:167611893883:instance/i-042560431f7fb3be3
- Operator:** -
- Managed:** false

Below the summary, there are tabs for 'Details', 'Status and alarms', 'Monitoring', 'Security', 'Networking', 'Storage', and 'Tags'. The 'Details' tab is selected, showing the following information:

- AMI ID:** ami-0eb9d6f63fab44d24
- AMI name:** al2023-ami-2023.8.20250707.0-kernel-6.1-x86\_64
- Stop protection:** Disabled
- Monitoring:** disabled
- Allowed image:** -
- Launch time:** Sun Jul 20 2025 22:47:50 GMT-0400 (Eastern Daylight Time) (37 minutes)
- Platform details:** Linux/UNIX
- Termination protection:** Disabled
- AMI location:** amazon/al2023-ami-2023.8.20250707.0-kernel-6.1-x86\_64

The footer of the console shows the copyright notice: © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences.

Phase 4

# Creating Target Group & Application Load Balancer

## Target Group:

The screenshot displays the AWS Management Console interface for a Target Group named 'app-tier-alb-p2'. The console is in the 'us-east-2' region. The left sidebar shows the navigation menu with categories like Savings Plans, Images, Elastic Block Store, Network & Security, Load Balancing, and Auto Scaling. The main content area shows the 'Details' tab for the Target Group. The details include the Target type (Instance), Protocol (HTTP: 80), Protocol version (HTTP1), and VPC (vpc-0159638b511253570). Below the details, there is a 'Targets' section showing a single target with ID 'i-01ee36f84fb9a4854' and name 'app-tier-p2'. The target is in a 'Healthy' state. The 'Registered targets' section shows a table with columns for Instance ID, Name, Port, Zone, Health status, Health status details, Admin..., Override..., Launch..., and Anomaly detection... The table contains one entry for the target 'app-tier-p2' with a 'Healthy' status.

## Application Load Balancer:

The screenshot displays the AWS Management Console interface for an Application Load Balancer named 'private-internal-alb-p2'. The console is in the 'us-east-2' region. The left sidebar shows the navigation menu with categories like Savings Plans, Images, Elastic Block Store, Network & Security, Load Balancing, and Auto Scaling. The main content area shows the 'Details' tab for the Load Balancer. The details include the Load balancer type (Application), Status (Active), Scheme (Internal), Hosted zone (Z3AADJGX6KTTL2), VPC (vpc-0159638b511253570), Availability Zones (subnet-00864e94a739ae347 and subnet-09d0836730f8ba846), and Date created (July 21, 2025, 00:16 (UTC-04:00)). Below the details, there is a 'Listeners and rules' section showing a table with columns for Protocol:Port, Default action, Rules, ARN, Security policy, Default SSL/TLS certificate, mTLS, and Trust store. The table contains one entry for the listener 'HTTP:80' with a default action of 'Forward to target group' and a single rule.



**Phase 5**

# Database Tier (RDS Free Tier)

## Step 1: Create a DB Subnet Group

1. Go to **RDS > Subnet groups > Create DB subnet group**
2. Name: db-subnet-group-p2
3. VPC: Select your VPC (vpc-p2)
4. Add **2 subnets** (must be in different AZs):
  - db1-p2 (e.g., 10.0.5.0/24)
  - db2-p2 (e.g., 10.0.6.0/24)

The screenshot displays the AWS Management Console interface for a DB Subnet Group. The left sidebar shows the navigation menu with 'Aurora and RDS' selected. The main content area shows the details for the subnet group 'db-subnet-group-p2'.

**Subnet group details**

- VPC ID: [vpc-0159688b611255570](#)
- ARN: [arn:aws:rds:us-east-2:167611893883:subgrp:db-subnet-group-p2](#)
- Supported network types: IPv4
- Description: db-subnet-group-p2

**Subnets (2)**

Availability zone	Subnet name	Subnet ID	CIDR block
us-east-2b	db2-p2	<a href="#">subnet-0eb7094ae5e4adbee</a>	10.0.6.0/24
us-east-2a	db1-p1	<a href="#">subnet-0b43872986766c470</a>	10.0.5.0/24

**Tags (0)**

Filter by Tag key

Tags

You don't have any tags associated with this resource.

Manage tags

## Step 2: Launch RDS Instance (MySQL)

1. Go to **RDS > Databases > Create database**
2. Choose **Standard Create**
3. Engine: **MySQL**
4. Version: Latest **Free Tier eligible** (e.g., 8.x)
5. Template: **Free Tier**
6. DB Instance Identifier: **dipen-db**
7. Master username: **admin**
8. Password: Choose strong password (store it safely)
9. DB instance size: **db.t3.micro**
10. Storage: 20 GB (General Purpose SSD)

The screenshot displays the AWS Management Console interface for an Amazon RDS instance. The browser address bar shows the URL: `us-east-2.console.aws.amazon.com/rds/home?region=us-east-2#database-id=dipen-db&cluster=false`. The console page title is "Aurora and RDS" with a breadcrumb trail "Databases > dipen-db".

**Summary**

DB identifier	Status	Role	Engine	Recommendations
dipen-db	Available	Instance	MySQL Community	

**CPU** 3.45%

**Class** db.t3.micro

**Current activity** 0 Connections

**Region & AZ** us-east-2b

**Connectivity & security** | Monitoring | Logs & events | Configuration | Zero-ETL integrations | Maintenance & backups | Data migrations - new | Tags | Recommendations

**Connectivity & security**

Endpoint & port	Networking	Security
<b>Endpoint</b> dipen-db.cj9e2a8e5.us-east-2.rds.amazonaws.com	<b>Availability Zone</b> us-east-2b	<b>VPC security groups</b> db-sg-p2 (sg-00e23433e9f1c76d6) Active
<b>Port</b> 3306	<b>VPC</b> VPC-P2 (vpc-0159638b511255570)	<b>Publicly accessible</b> No
	<b>Subnet group</b> db-subnet-group-p2	<b>Certificate authority</b> <a href="#">Info</a> rds-ca-rsa2048-g1
	<b>Subnets</b> subnet-0a43572986766c470 subnet-0eb70944ae5e4adbee	<b>Certificate authority date</b> May 21, 2061, 20:04 (UTC-04:00)
	<b>Network type</b> IPv4	<b>DB instance certificate expiration date</b> July 21, 2026, 00:49 (UTC-04:00)

## 2. private-sg-p2 → Enables outbound MySQL traffic to RDS

## RDS Connection

```

❯ ec2-user@ip-10.0.6.125:~$
Address: 10.0.6.29

[ec2-user@ip-10-0-6-125 ~]$ mysql -h 10.0.6.29 -u admin -p
Enter password:
ERROR 2003 (HY000): Can't connect to MySQL server on '10.0.6.29' (110)

[ec2-user@ip-10-0-6-125 ~]$ telnet dipen-db.cj062a8elx5.us-east-2.rds.amazonaws.com
Trying 10.0.6.29...
telnet: connect to address 10.0.6.29: Connection timed out
[ec2-user@ip-10-0-6-125 ~]$ exit
logout
Connection to 10.0.6.125 closed.
[root@ip-10-0-4-178 ~]# ssh -t "p2.pem" ec2-user@10.0.6.125
ec2-user@10.0.6.125 ~$
Amazon Linux 2023
https://aws.amazon.com/linux/amazon-linux-2023

Last login: Mon Jul 21 05:12:07 2025 from 10.0.4.178
[ec2-user@ip-10-0-6-125 ~]$ exit
logout
Connection to 10.0.6.125 closed.
[root@ip-10-0-4-178 ~]# ssh -t "p2.pem" ec2-user@10.0.6.125
ec2-user@10.0.6.125 ~$
Amazon Linux 2023
https://aws.amazon.com/linux/amazon-linux-2023

Last login: Mon Jul 21 05:04:11 2025 from 10.0.4.178
[ec2-user@ip-10-0-6-125 ~]$ mysql -h 10.0.6.29 -u admin -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 40
Server version: 8.0.41 Source distribution

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> exit
bye
[ec2-user@ip-10-0-6-125 ~]$ mysql -h dipen-db.cj062a8elx5.us-east-2.rds.amazonaws.com -u admin -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 41
Server version: 8.0.41 Source distribution

Copyright (c) 2000, 2023, Oracle and/or its affiliates.

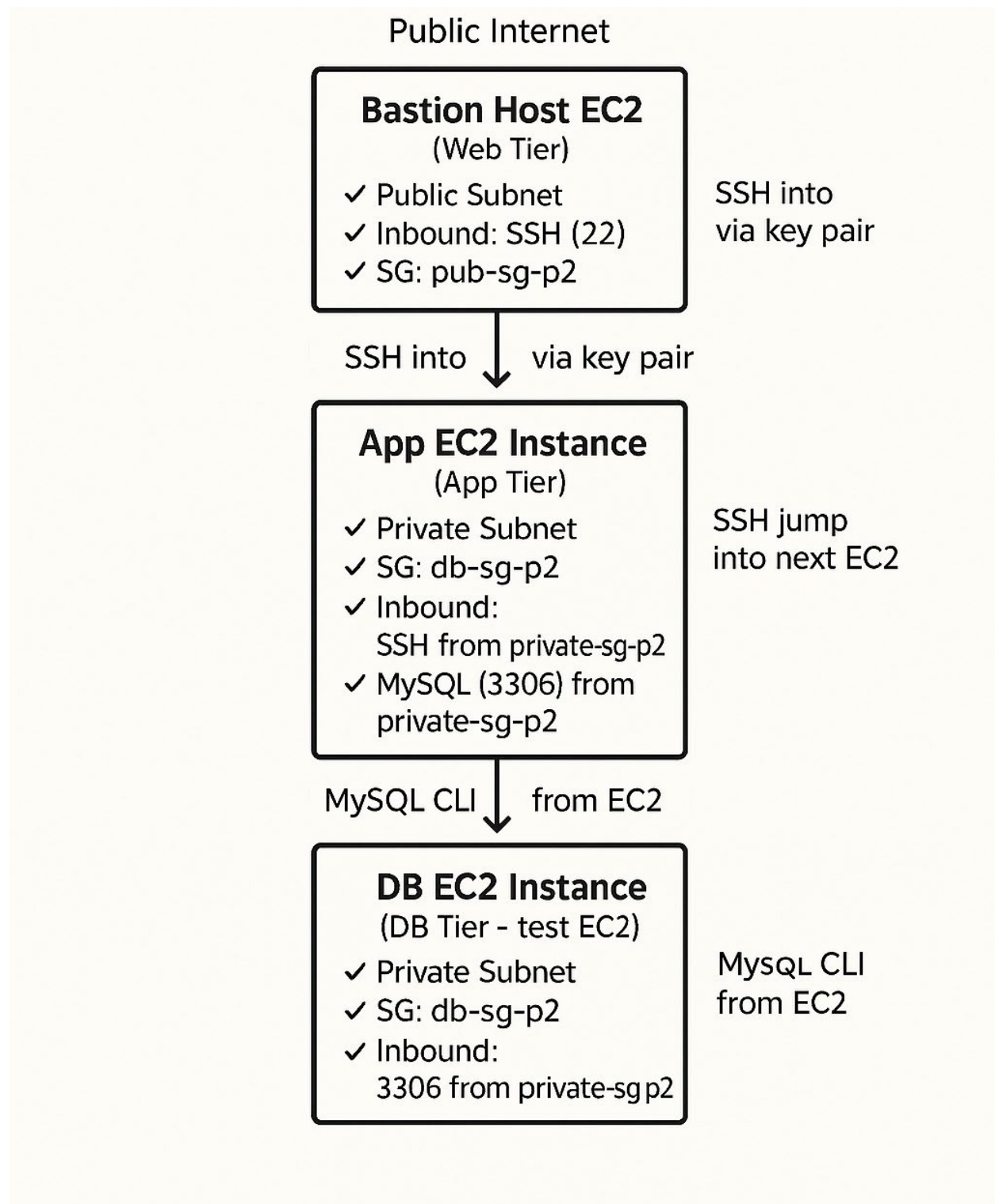
Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> |

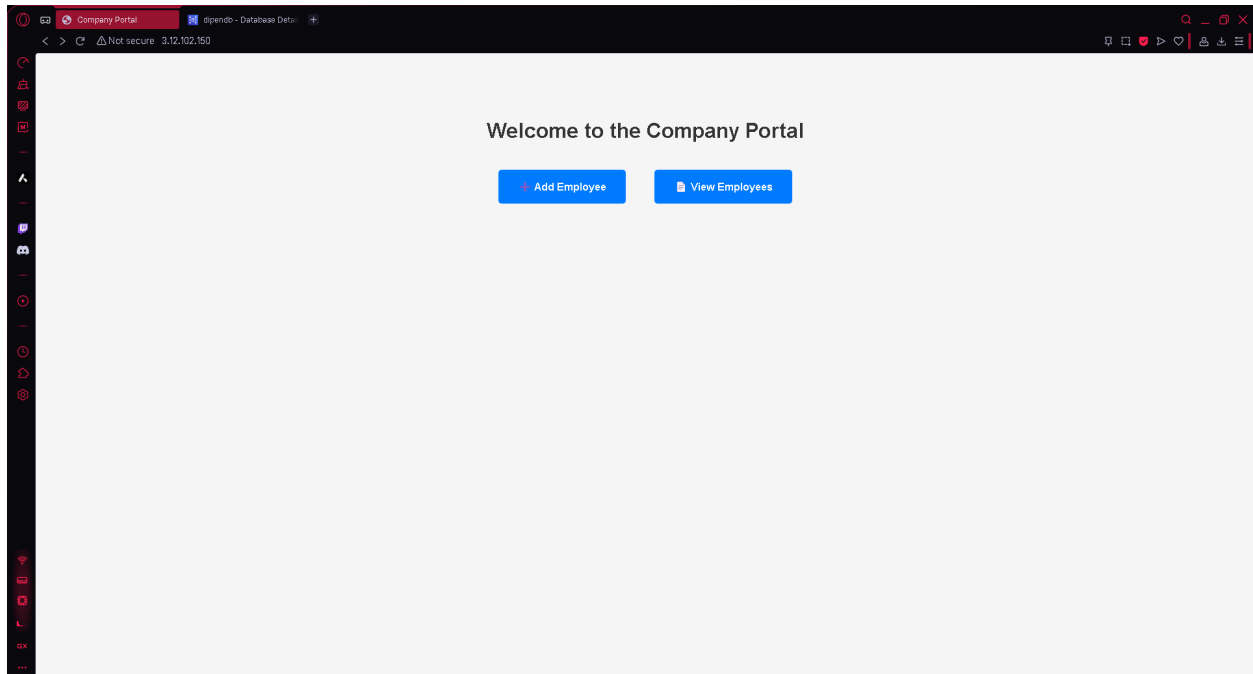
```

## Flowchart of RDS Connection

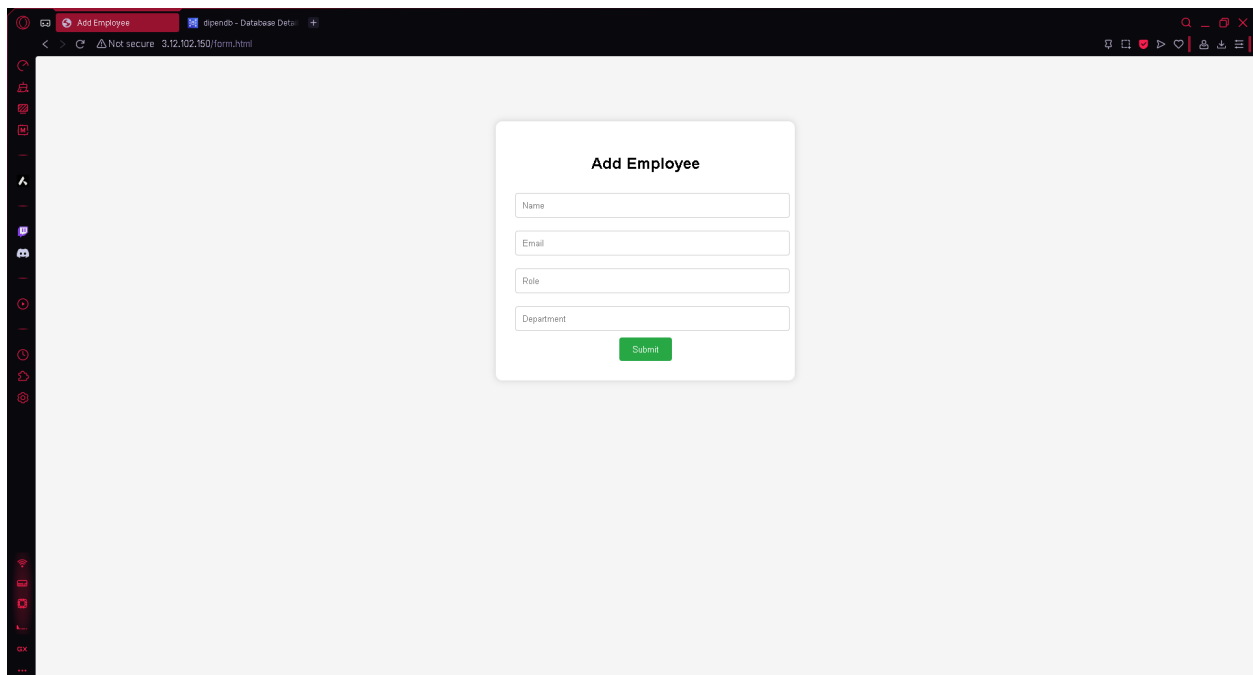


<b>Tier</b>	<b>Security Group</b>	<b>Inbound Allows From</b>	<b>Port(s)</b>
Web Tier	pub-sg-p2	My IP (SSH/HTTP)	22, 80
App Tier	private-sg-p2	pub-sg-p2	22, 80
DB Tier	db-sg-p2	private-sg-p2	22, 3306
RDS	db-sg-p2	db-sg-p2 (self-ref)	3306

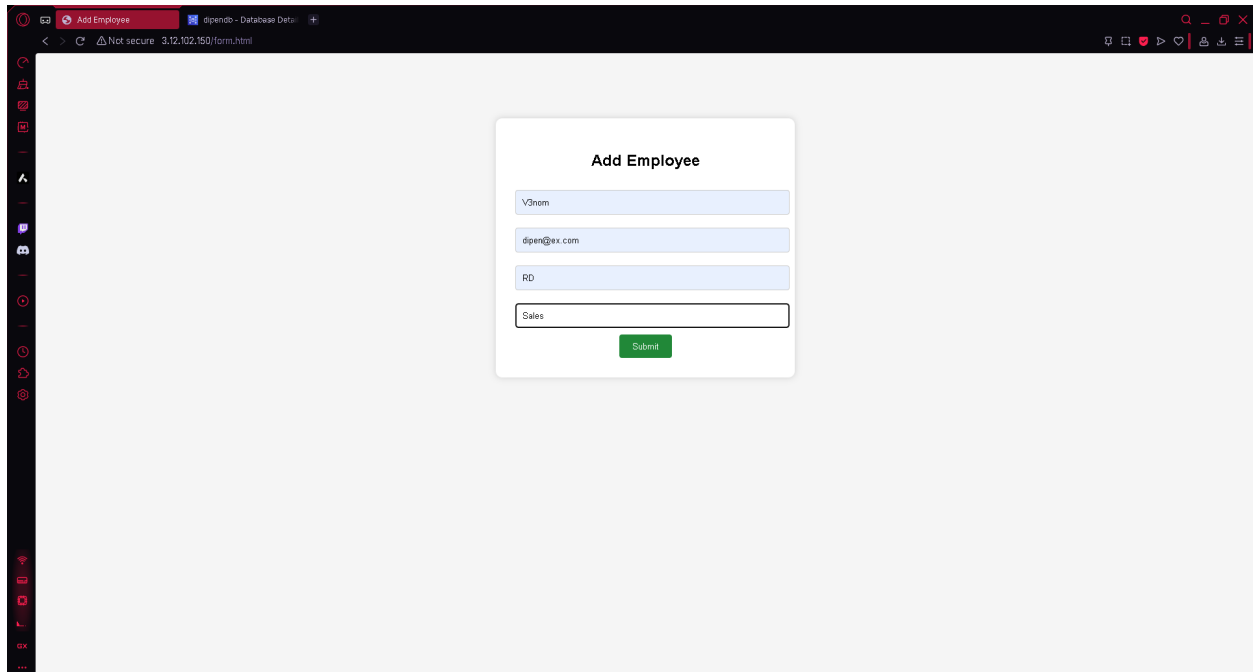
## Home Page (Web-tier)



## Add Employee



## Adding data



A screenshot of a web browser window. The address bar shows the URL `3.12.102.150/form.html`. The page title is "Add Employee". The form contains four input fields: "V3nom", "dpen@ex.com", "RD", and "Sales". A green "Submit" button is located below the "Sales" field.

**Add Employee**

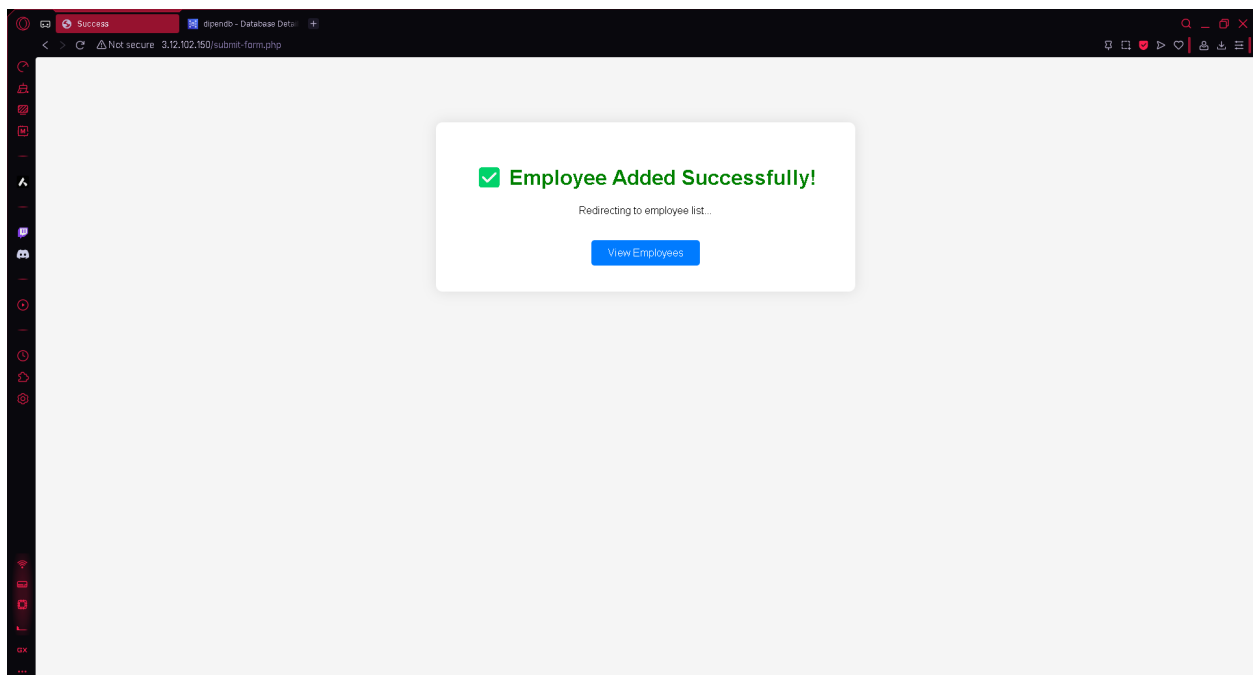
V3nom

dpen@ex.com

RD

Sales

Submit





## View Employees

Employee Records

dipend - Database Data

Not secure 3.12.102.150/view-employees.html

Employee Records

Id	Name	Email	Role	Department	Created At
8	V3nom	dipen@ex.com	RD	Sales	2025-07-23 05:36:25
7	abc	dipen@ex.com	Dev	IT	2025-07-23 05:34:26
6	V3nom	v3nom@v3nom.com	RD	IT	2025-07-23 05:28:53
5	V3nom	v3nom@v3nom.com	RD	IT	2025-07-23 05:28:50
4	John Doe	john@example.com	DevOps	Engineering	2025-07-23 05:28:37
1	Dipen Patel	dipen@example.com	Cloud Engineer	DevOps	2025-07-23 04:34:32
2	Aarav Shah	aaravshah@example.com	Backend Developer	Engineering	2025-07-23 04:34:32
3	Mira Joshi	mira.j@example.com	Project Manager	Operations	2025-07-23 04:34:32

## Files and Purpose

File Name	Purpose
form.html	Employee data input form
submit-form.php	Validates & inserts data to RDS
view-employees.html	Displays data in styled table
get-employees.php	Sends employee data as JSON (API)
index.html	Optional landing/home page

## Flow Structure

### 1. Form Submission (Frontend)

- **File:** form.html
  - **Function:** Presents a form to collect:
    - Name, Email, Role, Department
  - **Action:** On submit, sends a POST request to submit-form.php on the Web Tier EC2.
- 

### 2. Submit Data (Backend PHP)

- **File:** submit-form.php
  - **Runs on:** Web Tier EC2
  - **Tasks:**
    - Connects to RDS MySQL using mysqli
    - Validates input fields
    - Executes INSERT INTO employees (...) VALUES (...)
    - Displays a success message with redirect
- 

### 3. View Data (Frontend + Backend)

- **Frontend File:** view-employees.html
  - Uses JavaScript to fetch employee data from get-employees.php
  - Dynamically updates the table on load
- **Backend File:** get-employees.php
  - Fetches all records from the employees table
  - Returns data as a JSON response