AWS Employee Profile Management System

Application: Employee profile of XYZ company – New employees input their information and upload photos. Existing employees can get their information.

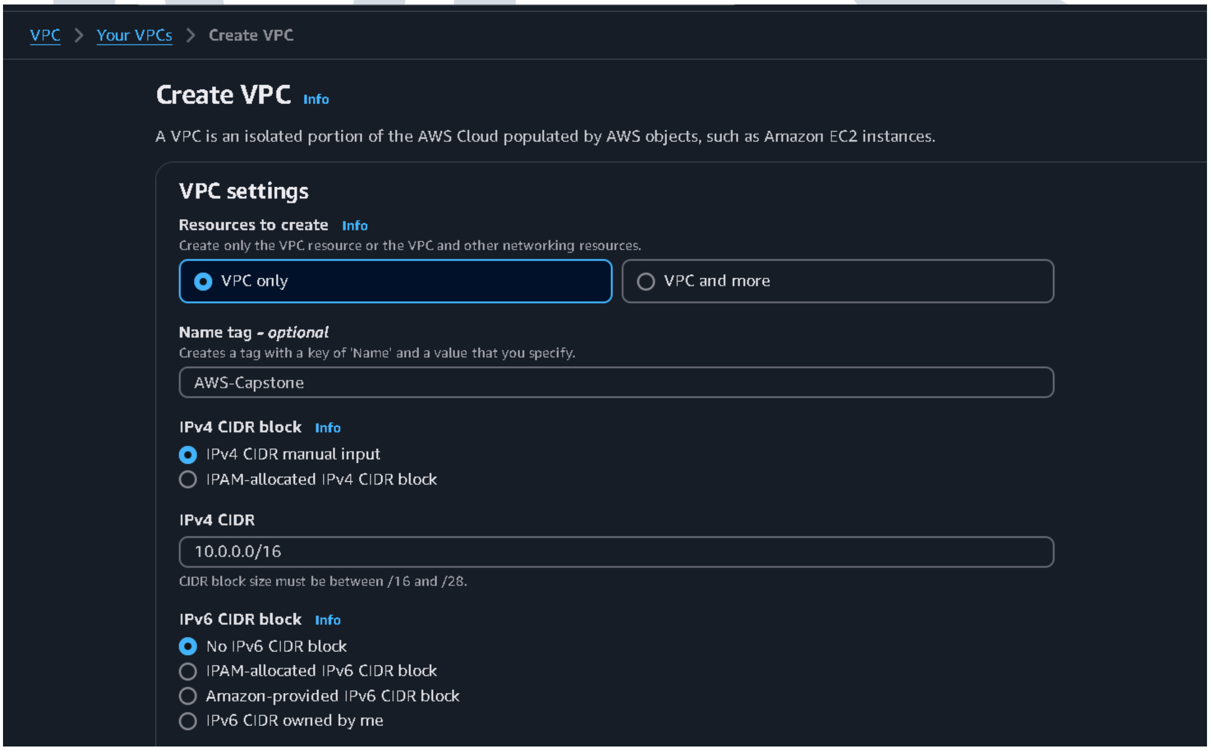
Solution:

Step 1:

Setup VPC for Load Balancer, Application EC2 instance and RDS Database - one public and two private subnets.

***Create a VPC**

- Go to the **VPC Dashboard** in the AWS Management Console.
- Click **Create VPC**.
- Choose the default IPv4 CIDR block (e.g., 10.0.0.0/16).
- Leave the other settings as default (you can customize them later if needed).



VPC > Your VPCs > Create VPC

Create VPC Info

A VPC is an isolated portion of the AWS Cloud populated by AWS objects, such as Amazon EC2 instances.

VPC settings

Resources to create Info
Create only the VPC resource or the VPC and other networking resources.

● VPC only ○ VPC and more

Name tag - *optional*
Creates a tag with a key of 'Name' and a value that you specify.

AWS-Capstone

IPv4 CIDR block Info

● IPv4 CIDR manual input
○ IPAM-allocated IPv4 CIDR block

IPv4 CIDR

10.0.0.0/16

CIDR block size must be between /16 and /28.

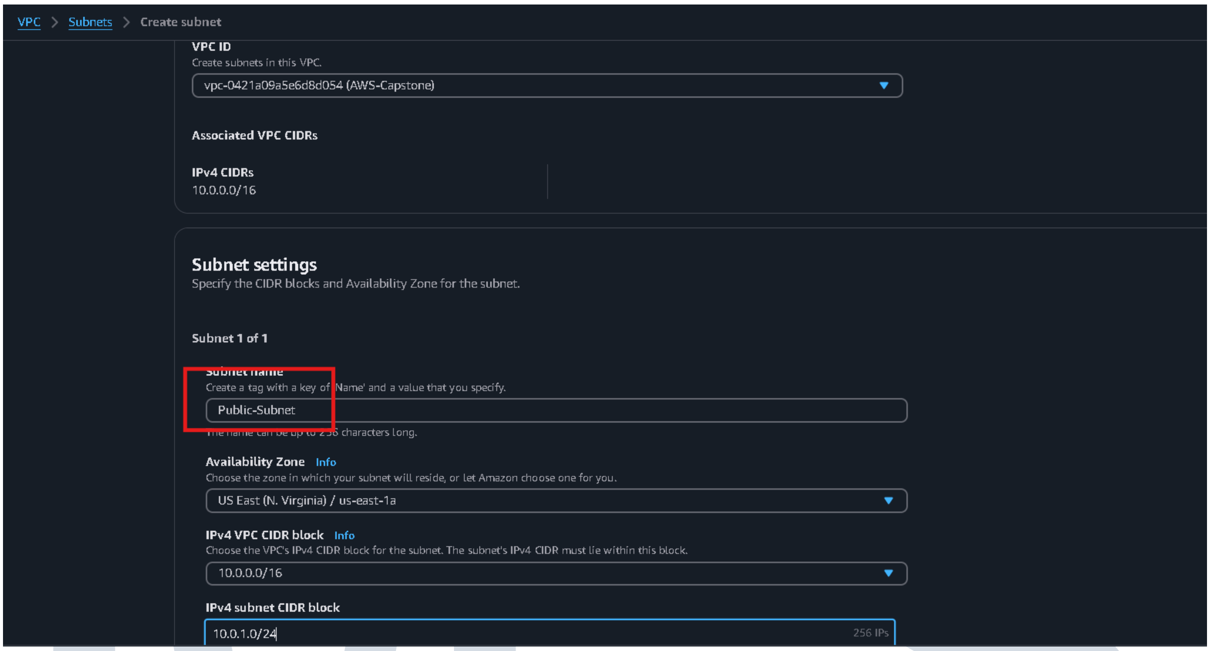
IPv6 CIDR block Info

● No IPv6 CIDR block
○ IPAM-allocated IPv6 CIDR block
○ Amazon-provided IPv6 CIDR block
○ IPv6 CIDR owned by me

***Create Subnets**

- Go to the **Subnets** section in the VPC Dashboard.
- Click **Create subnet**.

- **Public Subnet**:
  - Name: Public-Subnet
  - Availability Zone: Choose any AZ.
  - CIDR block: 10.0.1.0/24 (or any suitable range).



VPC > Subnets > Create subnet

VPC ID
Create subnets in this VPC.
vpc-0421a09a5e6d8d054 (AWS-Capstone)

Associated VPC CIDRs

IPv4 CIDRs
10.0.0.0/16

Subnet settings
Specify the CIDR blocks and Availability Zone for the subnet.

Subnet 1 of 1

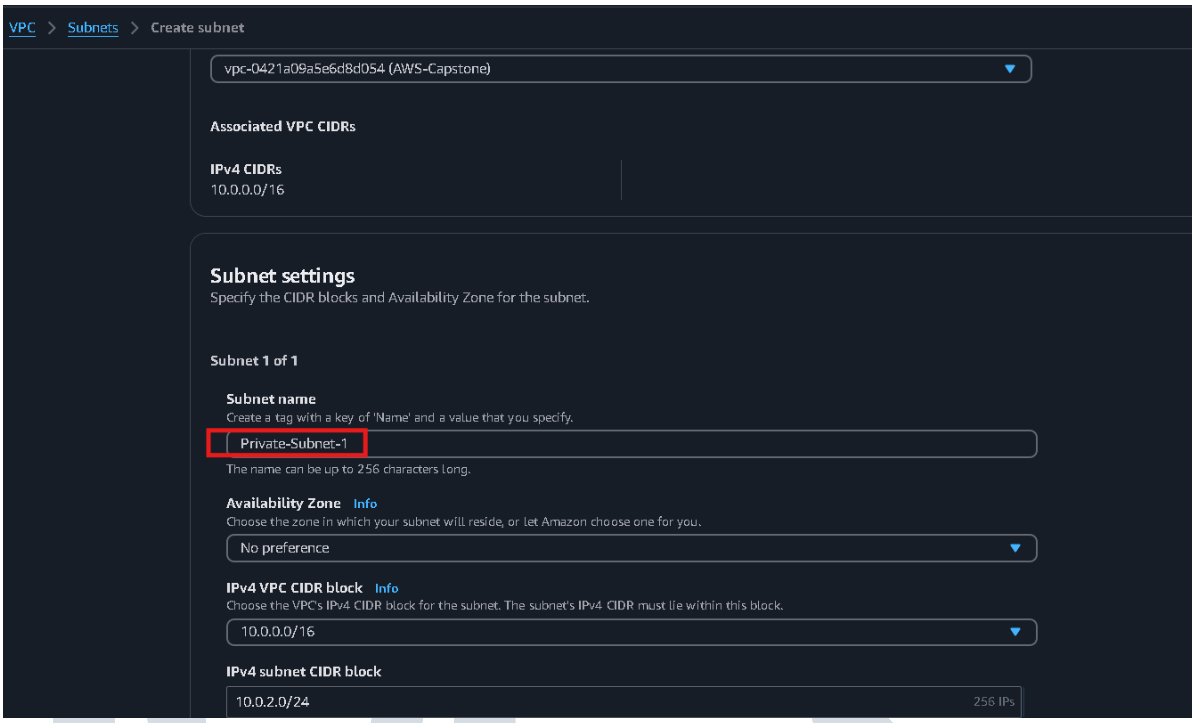
Subnet name
Create a tag with a key of 'Name' and a value that you specify.
Public-Subnet
The name can be up to 256 characters long.

Availability Zone Info
Choose the zone in which your subnet will reside, or let Amazon choose one for you.
US East (N. Virginia) / us-east-1a

IPv4 VPC CIDR block Info
Choose the VPC's IPv4 CIDR block for the subnet. The subnet's IPv4 CIDR must lie within this block.
10.0.0.0/16

IPv4 subnet CIDR block
10.0.1.0/24 256 IPs

- **Private Subnet 1**:
  - Name: Private-Subnet-EC2
  - Availability Zone: Choose a different AZ (if desired).
  - CIDR block: 10.0.2.0/24.



vpc-0421a09a5e6d8d054 (AWS-Capstone)

Associated VPC CIDRs

IPv4 CIDRs
10.0.0.0/16

Subnet settings
Specify the CIDR blocks and Availability Zone for the subnet.

Subnet 1 of 1

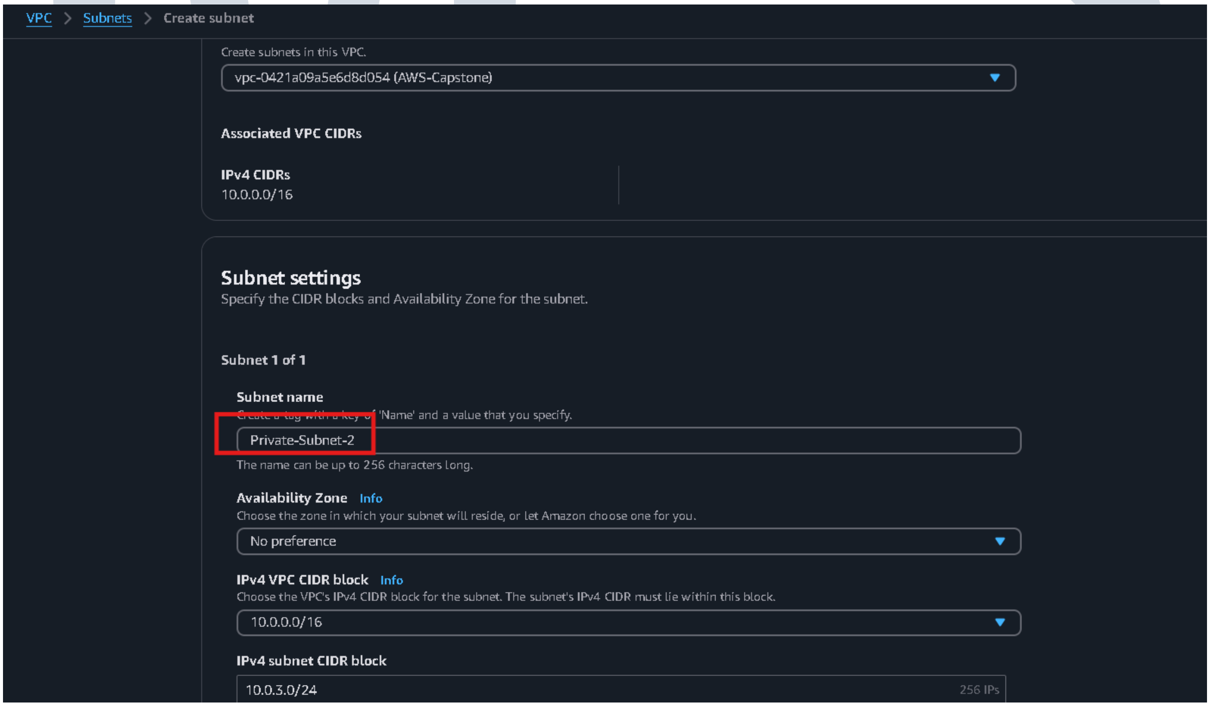
Subnet name
Create a tag with a key of 'Name' and a value that you specify.
Private-Subnet-1
The name can be up to 256 characters long.

Availability Zone Info
Choose the zone in which your subnet will reside, or let Amazon choose one for you.
No preference

IPv4 VPC CIDR block Info
Choose the VPC's IPv4 CIDR block for the subnet. The subnet's IPv4 CIDR must lie within this block.
10.0.0.0/16

IPv4 subnet CIDR block
10.0.2.0/24 256 IPs

- **Private Subnet 2**:
    - Name: Private-Subnet-RDS
    - Availability Zone: Same or different AZ.
    - CIDR block: 10.0.3.0/24.



Create subnets in this VPC.
vpc-0421a09a5e6d8d054 (AWS-Capstone)

Associated VPC CIDRs

IPv4 CIDRs
10.0.0.0/16

Subnet settings
Specify the CIDR blocks and Availability Zone for the subnet.

Subnet 1 of 1

Subnet name
Create a tag with a key of 'Name' and a value that you specify.
Private-Subnet-2
The name can be up to 256 characters long.

Availability Zone Info
Choose the zone in which your subnet will reside, or let Amazon choose one for you.
No preference

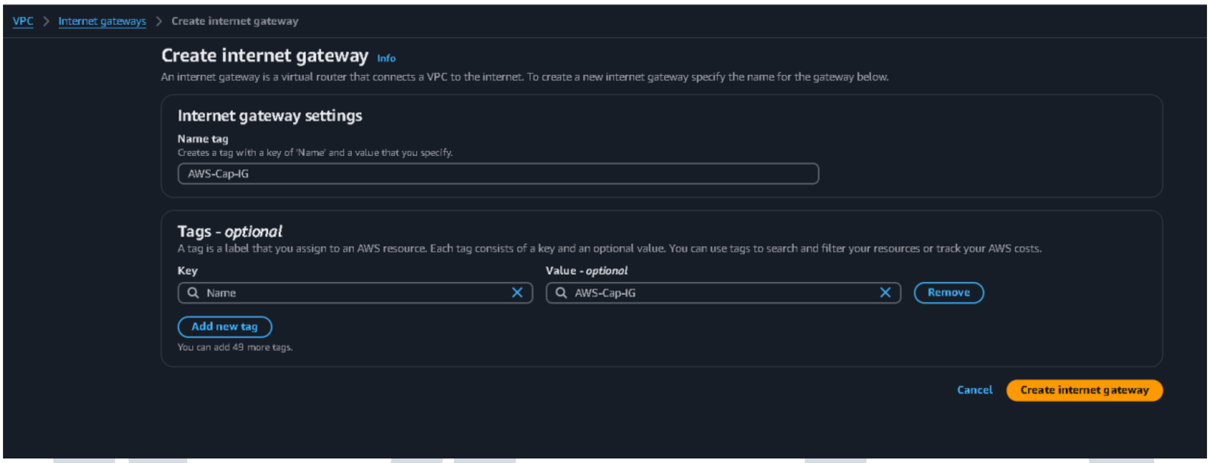
IPv4 VPC CIDR block Info
Choose the VPC's IPv4 CIDR block for the subnet. The subnet's IPv4 CIDR must lie within this block.
10.0.0.0/16

IPv4 subnet CIDR block
10.0.3.0/24 256 IPs

***Set Up Internet Gateway**

- In the VPC Dashboard, go to **Internet Gateways**.

- Click **Create internet gateway**.
- Name it Internet-Gateway.
- Attach it to the VPC you created earlier.



Create internet gateway Info

An internet gateway is a virtual router that connects a VPC to the internet. To create a new internet gateway specify the name for the gateway below.

Internet gateway settings

Name tag

Creates a tag with a key of 'Name' and a value that you specify.

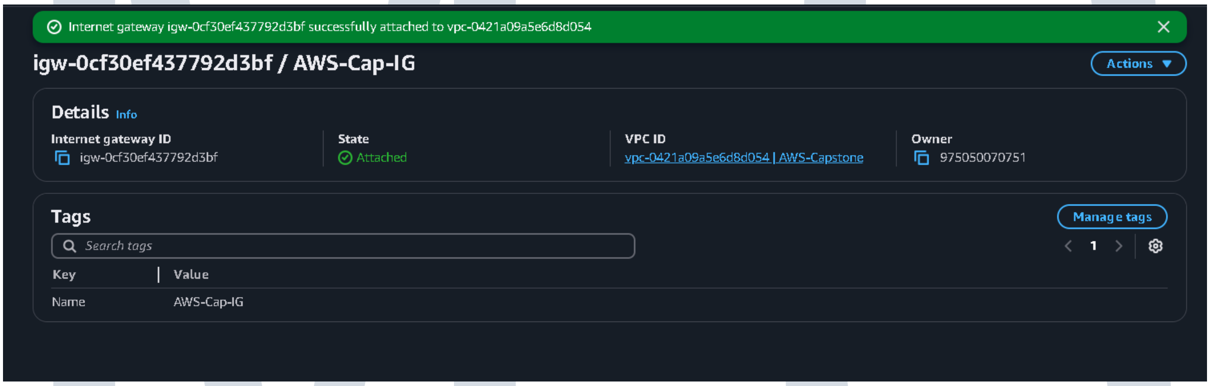
AWS-Cap-IG

Tags - *optional*

A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

Key	Value - *optional*
Name	AWS-Cap-IG

You can add 49 more tags.



Internet gateway igw-0cf30ef437792d3bf successfully attached to vpc-0421a09a5e6d8d054

igw-0cf30ef437792d3bf / AWS-Cap-IG

Details Info

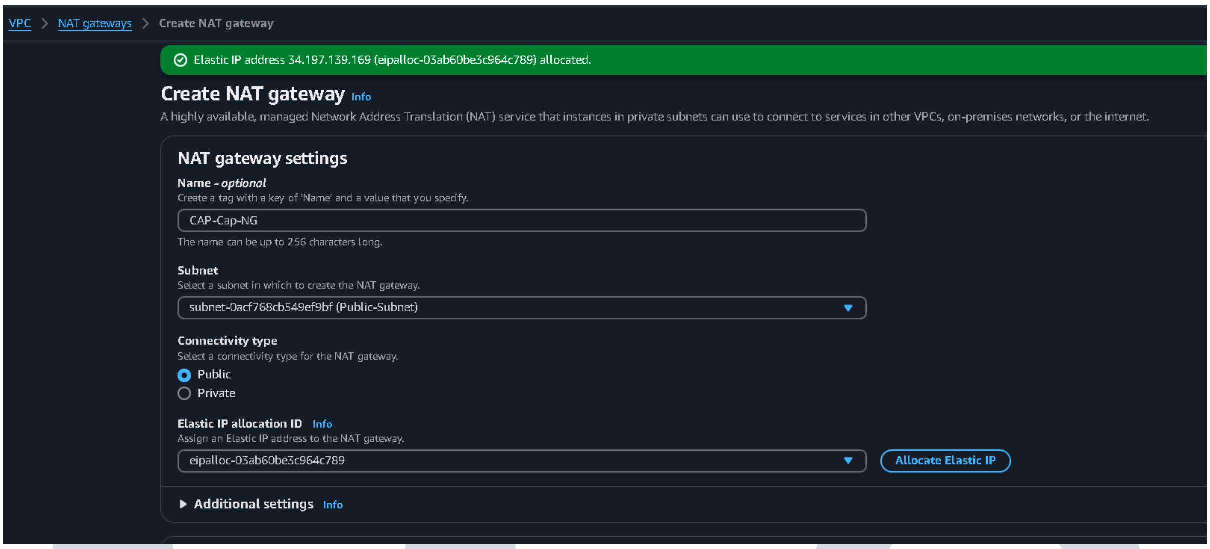
Internet gateway ID	State	VPC ID	Owner
igw-0cf30ef437792d3bf	Attached	vpc-0421a09a5e6d8d054 | AWS-Capstone	975050070751

Tags

Key	Value
Name	AWS-Cap-IG

*Set Up NAT Gateway (Optional)**

- Go to the **NAT Gateways** section in the VPC Dashboard.
- Create a NAT Gateway in the **Public Subnet**.
- Allocate an Elastic IP for the NAT Gateway.
- Update the private subnet route tables to route internet-bound traffic through the NAT Gateway.



VPC > NAT gateways > Create NAT gateway

Elastic IP address 34.197.139.169 (eipalloc-03ab60be3c964c789) allocated.

Create NAT gateway Info

A highly available, managed Network Address Translation (NAT) service that instances in private subnets can use to connect to services in other VPCs, on-premises networks, or the internet.

NAT gateway settings

Name - *optional*
Create a tag with a key of 'Name' and a value that you specify.
CAP-Cap-NG
The name can be up to 256 characters long.

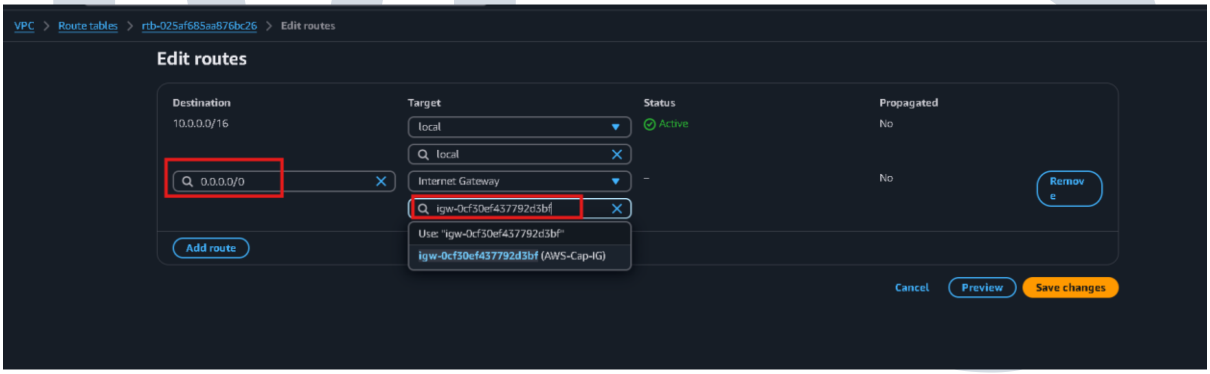
Subnet
Select a subnet in which to create the NAT gateway.
subnet-0acf768cb549ef9bf (Public-Subnet)

Connectivity type
Select a connectivity type for the NAT gateway.
● Public
○ Private

Elastic IP allocation ID Info
Assign an Elastic IP address to the NAT gateway.
eipalloc-03ab60be3c964c789

***Configure Route Tables**

- Go to the **Route Tables** section in the VPC Dashboard.
- By default, a main route table is created.need to modify it:
    - For the **Public Subnet**, modify the route table to have a route:
        - Destination: 0.0.0.0/0
        - Target: **Internet Gateway**



VPC > Route tables > rtb-025af685aa876bc26 > Edit routes

Edit routes

Destination	Target	Status	Propagated
10.0.0.0/16	local	Active	No
0.0.0.0/0	Internet Gateway igw-0cf30ef437792d3bf	-	No

Use: "igw-0cf30ef437792d3bf"
igw-0cf30ef437792d3bf (AWS-Cap-IG)

- For the **Private Subnets**:
    - Create a new route table for private subnets.
    - Add routes for outbound traffic through a **NAT Gateway**

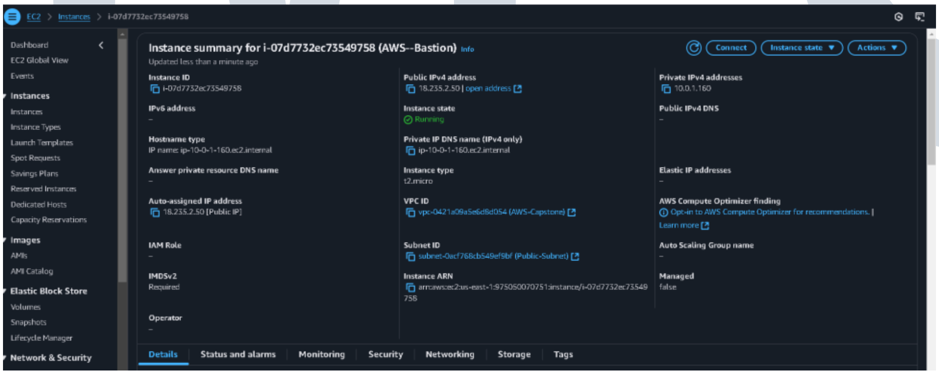
Step 2:

Create Bastion Host Launch an EC2 instance in the Public Subnet

Launch the Bastion Host in the Public Subnet

1. **Go to EC2 Dashboard**: In the AWS Management Console, navigate to the **EC2 Dashboard**.
2. **Launch a New EC2 Instance**:
    - Click **Launch Instance** to create a new EC2 instance.
    - Choose an AMI, such as **Ubuntu**.
    - Choose an instance type (e.g., t2.micro for testing purposes).

3. **Network Settings**:
    - In the **Network** section, select the **VPC** already created.
    - Select the **Public Subnet** for this instance.
    - Assign a **Public IP** to the Bastion Host (this is important because the Bastion Host needs to be reachable from the internet for SSH access).
4. **Configure Security Group**:
    - Create a **new Security Group** for the Bastion Host, or use an existing one. Add the following inbound rules:
        - **Type**: SSH
        - **Protocol**: TCP
        - **Port**: 22
        - **Source**: your_ip/32 (for security, restrict access to your own IP address).
        - This will allow SSH access to the Bastion Host only from your IP address.
    - Add the following outbound rules (default is usually fine):
        - **Type**: All traffic
        - **Protocol**: All traffic
        - **Port range**: All
        - **Destination**: 0.0.0.0/0 (allowing traffic to any destination).
5. **Review and Launch**:
    - Review your settings and click **Launch**.
    - Make sure to use a **new key pair** or an existing key pair to access the instance (you will need this for SSH access).



Instance summary for i-07d7732ec73549758 (AWS--Bastion)

Updated less than a minute ago

Instance ID
i-07d7732ec73549758

IPv6 address
–

Hostname type
IP name: ip-10-0-1-160.ec2.internal

Answer private resource DNS name
–

Auto-assigned IP address
18.235.2.50 [Public IP]

IAM Role
–

IMDSv2
Required

Operator
–

Public IPv4 address
18.235.2.50 | open address

Instance state
Running

Private IP DNS name (IPv4 only)
ip-10-0-1-160.ec2.internal

Instance type
t2.micro

VPC ID
vpc-0421a09a5e6d8d054 (AWS-Capstone)

Subnet ID
subnet-0acf768cb549ef9bf (Public-Subnet)

Instance ARN
arn:aws:ec2:us-east-1:975050070751:instance/i-07d7732ec73549758

Private IPv4 addresses
10.0.1.160

Public IPv4 DNS
–

Elastic IP addresses
–

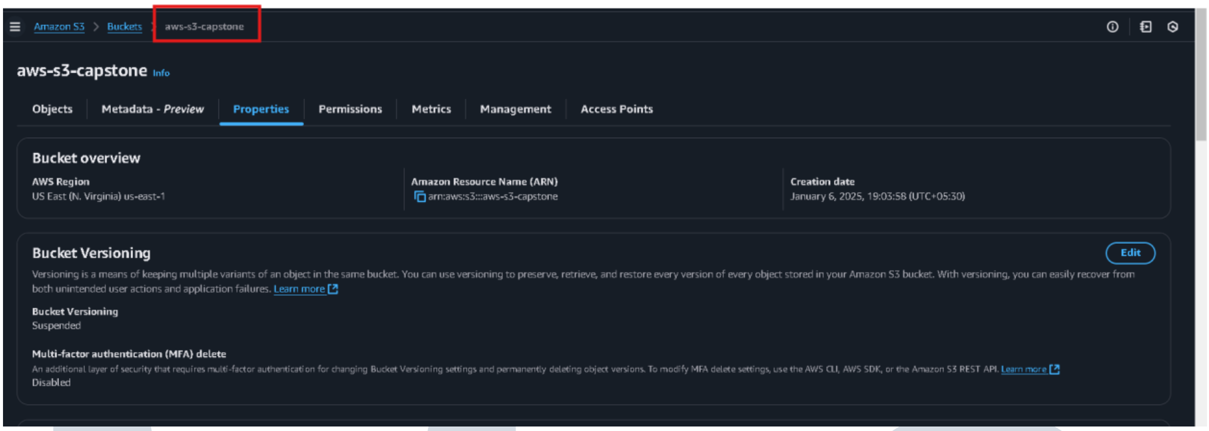
AWS Compute Optimizer finding
Opt-in to AWS Compute Optimizer for recommendations. | Learn more

Auto Scaling Group name
–

Managed
false

Step 4:

Create S3 Bucket To store images For Static Website Hosting



aws-s3-capstone Info

Bucket overview

AWS Region
US East (N. Virginia) us-east-1

Amazon Resource Name (ARN)
arn:aws:s3:::aws-s3-capstone

Creation date
January 6, 2025, 19:03:58 (UTC+05:30)

Bucket Versioning

Versioning is a means of keeping multiple variants of an object in the same bucket. You can use versioning to preserve, retrieve, and restore every version of every object stored in your Amazon S3 bucket. With versioning, you can easily recover from both unintended user actions and application failures. Learn more

Bucket Versioning
Suspended

Multi-factor authentication (MFA) delete
An additional layer of security that requires multi-factor authentication for changing Bucket Versioning settings and permanently deleting object versions. To modify MFA delete settings, use the AWS CLI, AWS SDK, or the Amazon S3 REST API. Learn more
Disabled

Step 5:

Create RDS Instance Connect to it and create “employees” table

Create an RDS Instance:

Create an RDS MySQL Database Instance

1. Navigate to RDS Dashboard:

o Go to RDS from the AWS Management Console.

2. Create a Database:

o Click Create Database.

o Choose MySQL and Standard Create.

o Configuration:

*B Instance Identifier: Employe-db.

*Master Username: admin.

* Password.

* Instance Size: Choose an appropriate instance (e.g., db.t2.micro for testing).

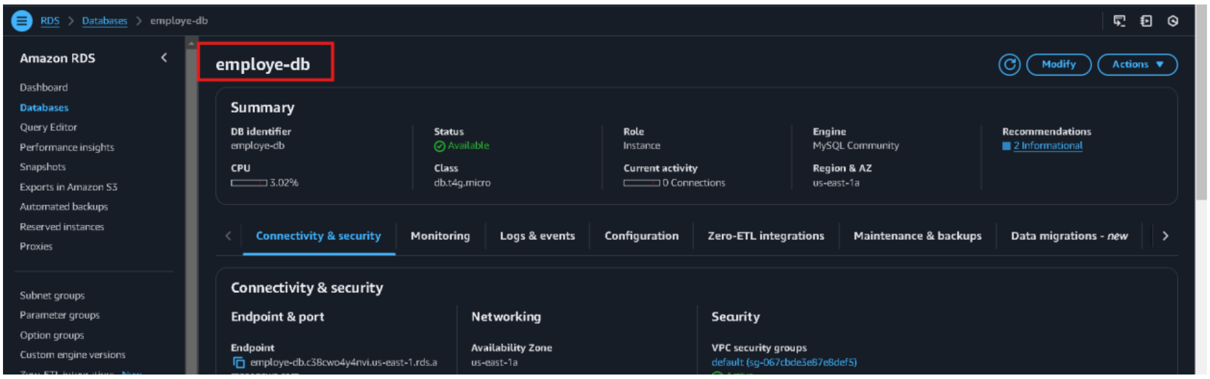
* Connectivity:

* Choose the VPC that includes your EC2 instances.

* Set Publicly Accessible to No.

* Security Group: Allow inbound MySQL traffic (port 3306).

* Review and create the RDS instance



RDS > Databases > employe-db

Amazon RDS

- Dashboard
- Databases
- Query Editor
- Performance insights
- Snapshots
- Exports in Amazon S3
- Automated backups
- Reserved instances
- Proxies
- Subnet groups
- Parameter groups
- Option groups
- Custom engine versions

employe-db

Summary

DB identifier: employe-db
Status: Available
Role: Instance
Engine: MySQL Community
Recommendations: 2 Informational
CPU: 3.02%
Class: db.t4g.micro
Current activity: 0 Connections
Region & AZ: us-east-1a

Connectivity & security

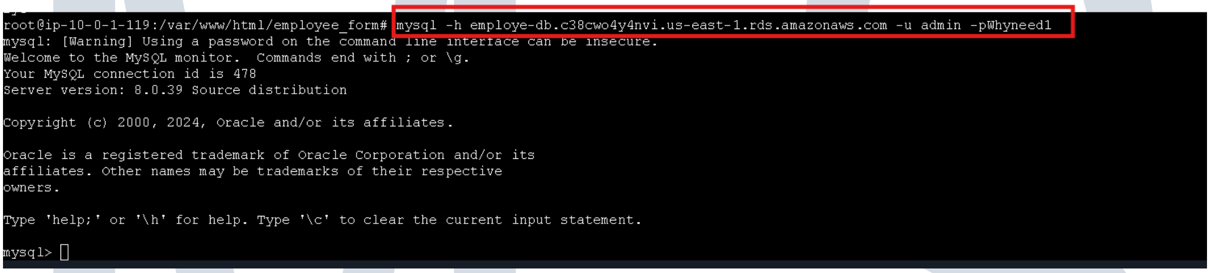
Endpoint & port
Endpoint: employe-db.c38cwo4y4nvi.us-east-1.rds.a

Networking
Availability Zone: us-east-1a

Security
VPC security groups: default (sg-067cbde3e87e8def5)

Now install MySQL in ubuntu instance.

Once RDS instance comes in available state and connect database instance to the EC2 instance by copying RDS Endpoint following credentials.



```
root@ip-10-0-1-119:/var/www/html/employee_form# mysql -h employe-db.c38cwo4y4nvi.us-east-1.rds.amazonaws.com -u admin -pWhyneed1
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 478
Server version: 8.0.39 Source distribution

Copyright (c) 2000, 2024, 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> 
```

Create Database &amp; Table in RDS instance:

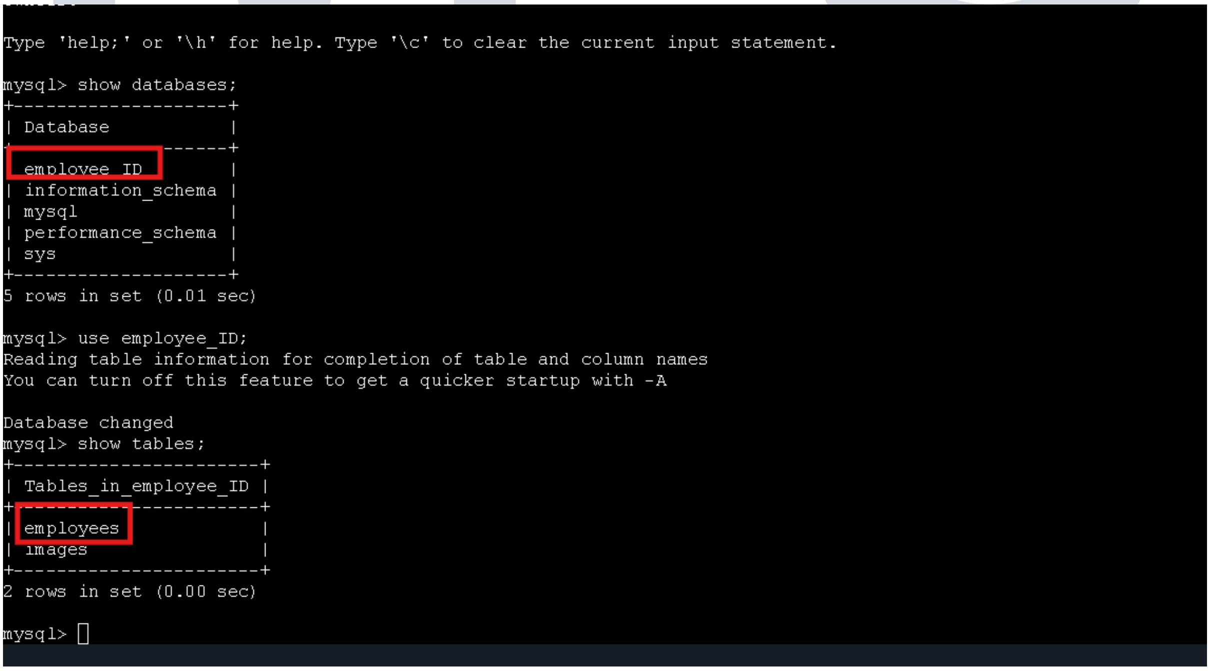
a. Database name: employee_ID

b. Table name: employees

c. Database password: Whyneed1

Verify the “intel” database is present or not.

Create a “data” table.



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

mysql> show databases;
+--------------------+
| Database           |
+--------------------+
| employee_ID        |
| information_schema |
| mysql              |
| performance_schema |
| sys                |
+--------------------+
5 rows in set (0.01 sec)

mysql> use employee_ID;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql> show tables;
+-----------------------+
| Tables_in_employee_ID |
+-----------------------+
| employees             |
| images                |
+-----------------------+
2 rows in set (0.00 sec)

mysql> 
```

Now installs PHP 5.6 along with the php5.6-mysqli extension. PHP is a scripting language commonly used for web development, and the mysqli extension allows PHP to interact with MySQL databases, enabling it to connect to and execute SQL queries on a MySQL database.

Step 6:

Domain Name Setup Get a Domain Name Setup Route 53

Create a Hosted Zone in Route 53

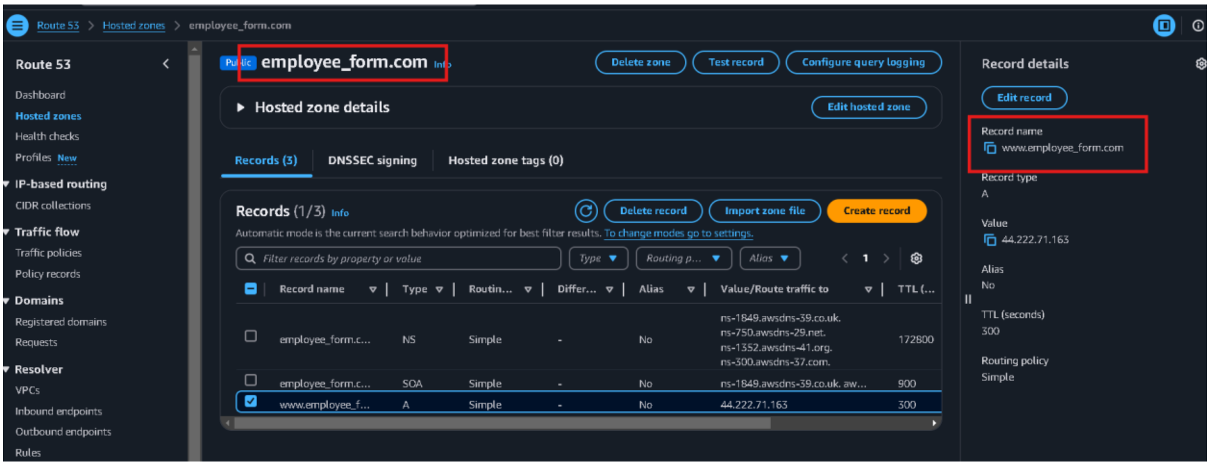
A **Hosted Zone** is required to manage DNS records for your domain.

1. **Go to the Route 53 Console**
   Navigate to **Hosted Zones** in the Route 53 Dashboard.
2. **Create a Hosted Zone**
   - Click **Create Hosted Zone**.
   - Enter your domain name (e.g., example.com).
   - Set the **Type** to **Public Hosted Zone** (for public websites).
   - Add a description (optional).
   - Click **Create Hosted Zone**

3.Add DNS Records in the Hosted Zone

Add an A Record for an EC2 Instance

1. Click **Create Record** in the hosted zone.
2. Enter the following details:
   - **Record Name**: Leave blank for the root domain (e.g., example.com), or enter a subdomain (e.g., www).
   - **Record Type**: Select A - IPv4 address.
   - **Value**: Enter the public IPv4 address of your EC2 instance.
3. Click **Create Records**.



employee_form.com

Hosted zone details

Records (1/3) Info

Automatic mode is the current search behavior optimized for best filter results. To change modes go to settings.

	Record name	Type	Routin...	Differ...	Alias	Value/Route traffic to	TTL (...
□	employee_form.c...	NS	Simple	-	No	ns-1849.awsdns-39.co.uk. ns-750.awsdns-29.net. ns-1352.awsdns-41.org. ns-300.awsdns-37.com.	172800
□	employee_form.c...	SOA	Simple	-	No	ns-1849.awsdns-39.co.uk. aw...	900
☑	www.employee_f...	A	Simple	-	No	44.222.71.163	300

Record details

Record name
www.employee_form.com

Record type
A

Value
44.222.71.163

Alias
No

TTL (seconds)
300

Routing policy
Simple

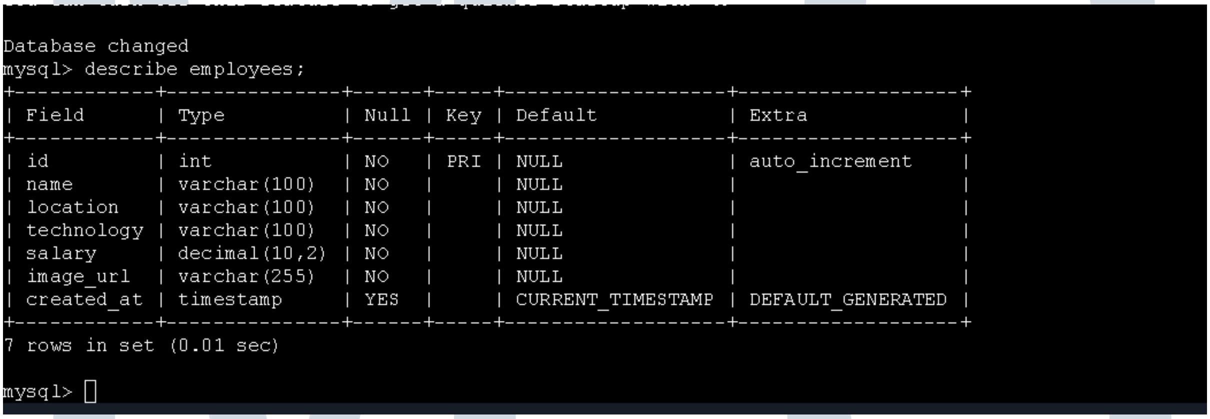
Step 7:

Now we will build application

Steps to Build

Step 1: Setup RDS Database st, you need to set up an **RDS database**.

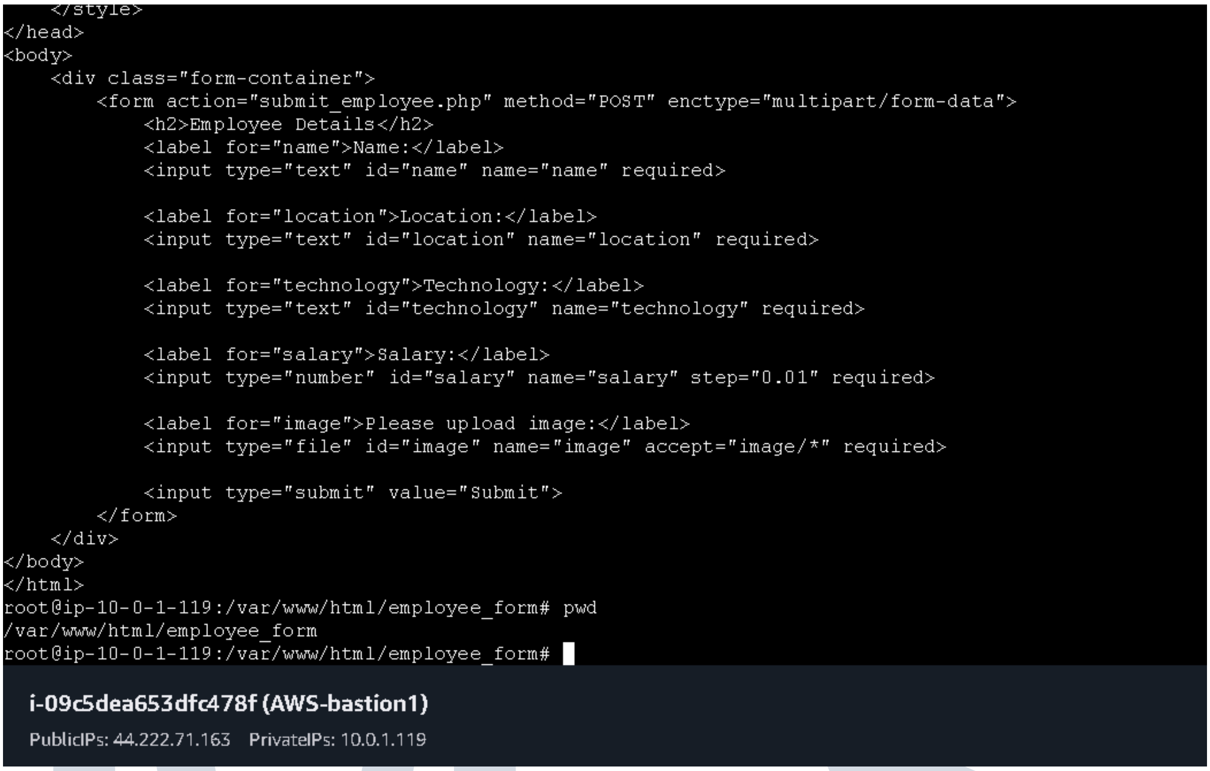
- **Create a database** in RDS
- **Create a table** to store the user data with the following columns:
  - id (auto-increment primary key)
  - name
  - location
  - technology
  - salary
  - image_url



```
Database changed
mysql> describe employees;
+------------+---------------+------+-----+-------------------+-------------------+
| Field      | Type          | Null | Key | Default           | Extra             |
+------------+---------------+------+-----+-------------------+-------------------+
| id         | int           | NO   | PRI | NULL              | auto_increment    |
| name       | varchar(100)  | NO   |     | NULL              |                   |
| location   | varchar(100)  | NO   |     | NULL              |                   |
| technology | varchar(100)  | NO   |     | NULL              |                   |
| salary     | decimal(10,2) | NO   |     | NULL              |                   |
| image_url  | varchar(255)  | NO   |     | NULL              |                   |
| created_at | timestamp     | YES  |     | CURRENT_TIMESTAMP | DEFAULT_GENERATED |
+------------+---------------+------+-----+-------------------+-------------------+
7 rows in set (0.01 sec)

mysql> 
```

Step 2: HTML Form

This is the HTML form that collect user inputs: Name, Location, Technology, Salary, and an image.



```
    </style>
</head>
<body>
    <div class="form-container">
        <form action="submit_employee.php" method="POST" enctype="multipart/form-data">
            <h2>Employee Details</h2>
            <label for="name">Name:</label>
            <input type="text" id="name" name="name" required>

            <label for="location">Location:</label>
            <input type="text" id="location" name="location" required>

            <label for="technology">Technology:</label>
            <input type="text" id="technology" name="technology" required>

            <label for="salary">Salary:</label>
            <input type="number" id="salary" name="salary" step="0.01" required>

            <label for="image">Please upload image:</label>
            <input type="file" id="image" name="image" accept="image/*" required>

            <input type="submit" value="Submit">
        </form>
    </div>
</body>
</html>
root@ip-10-0-1-119:/var/www/html/employee_form# pwd
/var/www/html/employee_form
root@ip-10-0-1-119:/var/www/html/employee_form# 
```

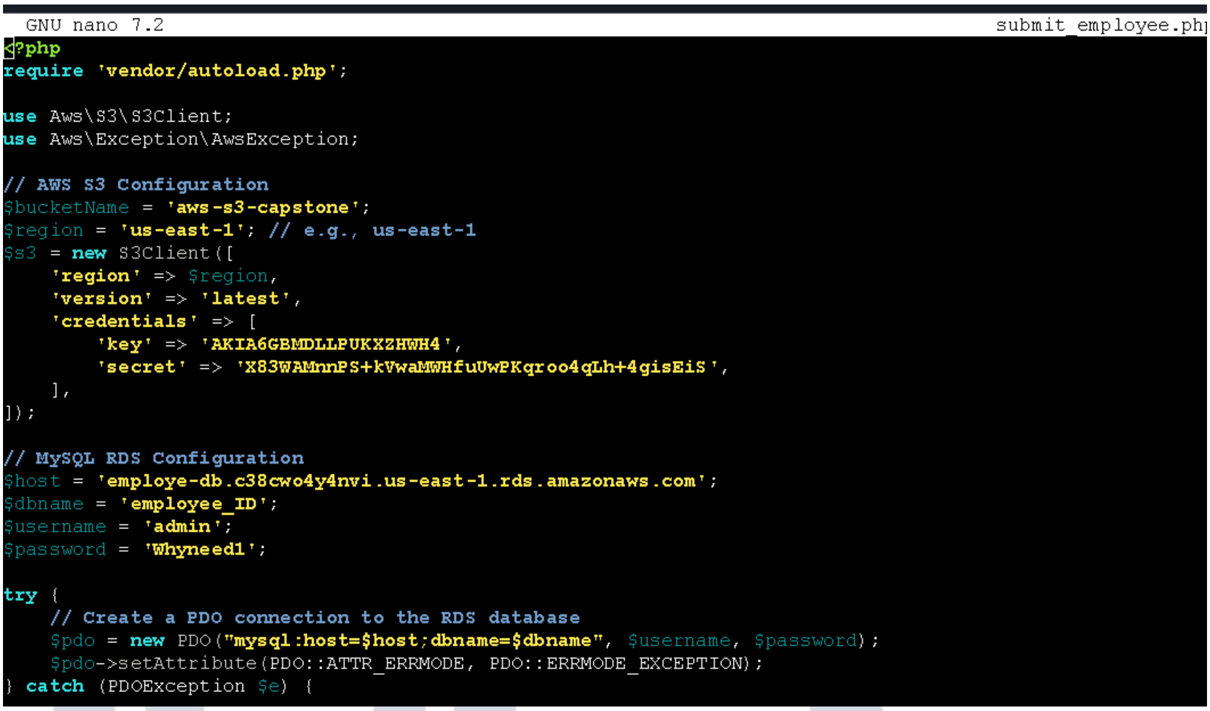
i-09c5dea653dfc478f (AWS-bastion1)

PublicIPs: 44.222.71.163 PrivateIPs: 10.0.1.119

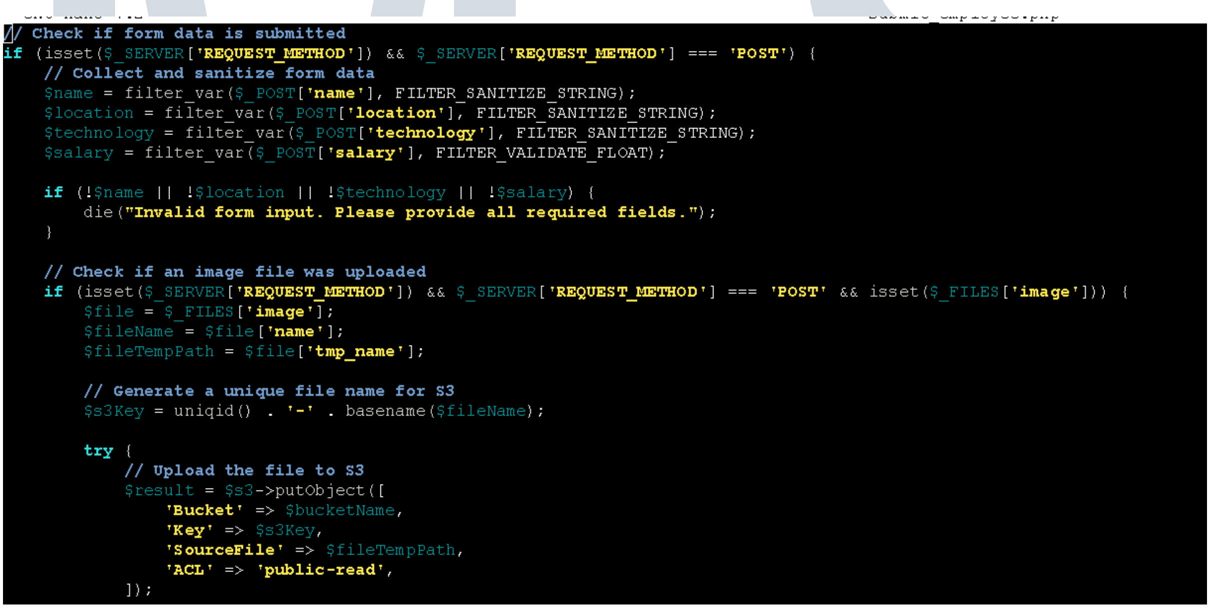
Step 3: PHP Script to Handle Form Submission

Now let’s create the PHP script to handle form submission and save the data to the RDS database.

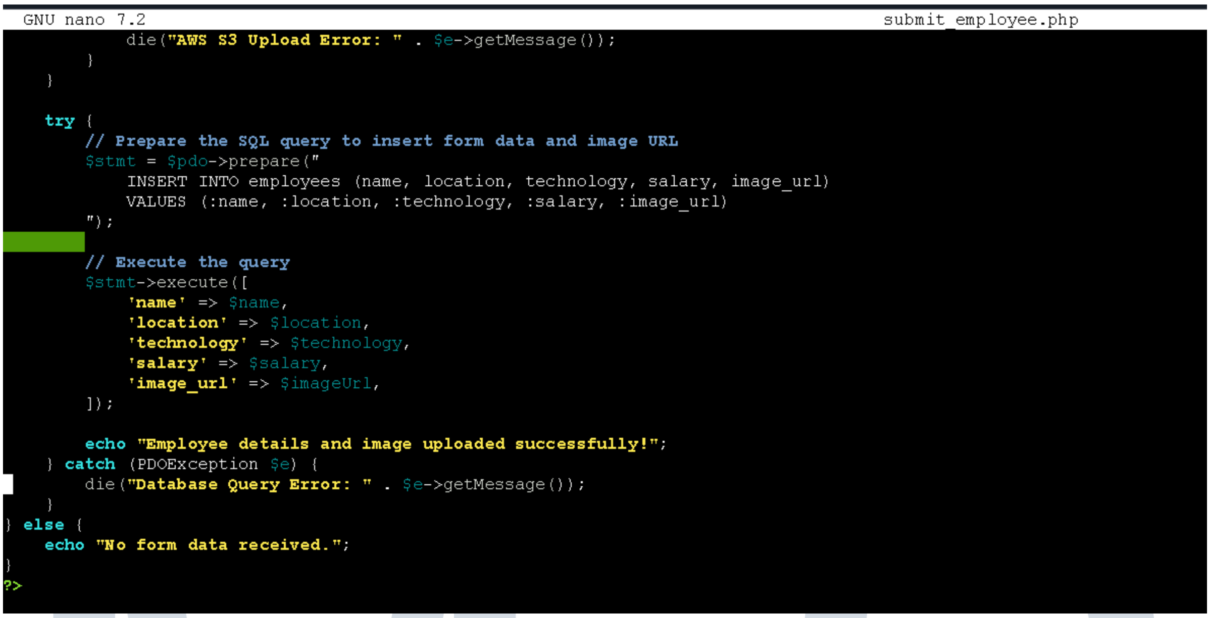
1. **Database Connection:** We need to connect to the MySQL database on RDS. Update the connection parameters (host, dbname, username, password).
2. **File Upload Handling:** The script will process the uploaded image and store it on the server.
3. **Save Data in Database:** We will insert the form data (including the image URL) into the RDS database.



```
  GNU nano 7.2                                                                  submit_employee.ph
<?php
require 'vendor/autoload.php';

use Aws\S3\S3Client;
use Aws\Exception\AwsException;

// AWS S3 Configuration
$bucketName = 'aws-s3-capstone';
$region = 'us-east-1'; // e.g., us-east-1
$s3 = new S3Client([
    'region' => $region,
    'version' => 'latest',
    'credentials' => [
        'key' => 'AKIA6GBMDLLPUKXZHWH4',
        'secret' => 'X83WAMnnPS+kVwaMWHfuUwPKqroo4qLh+4gisEiS',
    ],
]);

// MySQL RDS Configuration
$host = 'employe-db.c38cwo4y4nvi.us-east-1.rds.amazonaws.com';
$dbname = 'employee_ID';
$username = 'admin';
$password = 'Whyneed1';

try {
    // Create a PDO connection to the RDS database
    $pdo = new PDO("mysql:host=$host;dbname=$dbname", $username, $password);
    $pdo->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
} catch (PDOException $e) {
```



```
// Check if form data is submitted
if (isset($_SERVER['REQUEST_METHOD']) && $_SERVER['REQUEST_METHOD'] === 'POST') {
    // Collect and sanitize form data
    $name = filter_var($_POST['name'], FILTER_SANITIZE_STRING);
    $location = filter_var($_POST['location'], FILTER_SANITIZE_STRING);
    $technology = filter_var($_POST['technology'], FILTER_SANITIZE_STRING);
    $salary = filter_var($_POST['salary'], FILTER_VALIDATE_FLOAT);

    if (!$name || !$location || !$technology || !$salary) {
        die("Invalid form input. Please provide all required fields.");
    }

    // Check if an image file was uploaded
    if (isset($_SERVER['REQUEST_METHOD']) && $_SERVER['REQUEST_METHOD'] === 'POST' && isset($_FILES['image'])) {
        $file = $_FILES['image'];
        $fileName = $file['name'];
        $fileTempPath = $file['tmp_name'];

        // Generate a unique file name for S3
        $s3Key = uniqid() . '-' . basename($fileName);

        try {
            // Upload the file to S3
            $result = $s3->putObject([
                'Bucket' => $bucketName,
                'Key' => $s3Key,
                'SourceFile' => $fileTempPath,
                'ACL' => 'public-read',
            ]);
```



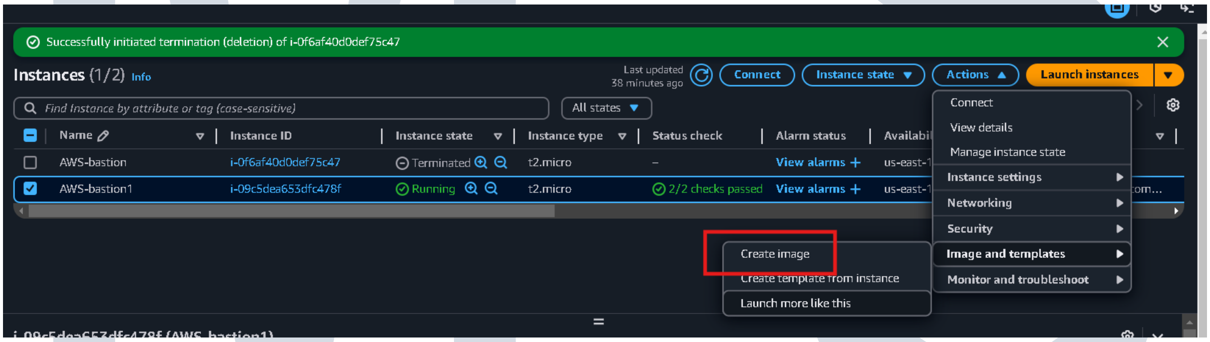
```
  GNU nano 7.2                                                            submit_employee.php
            die("AWS S3 Upload Error: " . $e->getMessage());
        }
    }

    try {
        // Prepare the SQL query to insert form data and image URL
        $stmt = $pdo->prepare("
            INSERT INTO employees (name, location, technology, salary, image_url)
            VALUES (:name, :location, :technology, :salary, :image_url)
        ");

        // Execute the query
        $stmt->execute([
            'name' => $name,
            'location' => $location,
            'technology' => $technology,
            'salary' => $salary,
            'image_url' => $imageUrl,
        ]);

        echo "Employee details and image uploaded successfully!";
    } catch (PDOException $e) {
        die("Database Query Error: " . $e->getMessage());
    }
} else {
    echo "No form data received.";
}
?>
```

We will save above file and install all pre-required plugins.

Step 8:

Create AMI with application Login to a EC2 instance and setup application

Create an AMI from this EC2 instance to launch the template.



Successfully initiated termination (deletion) of i-0f6af40d0def75c47

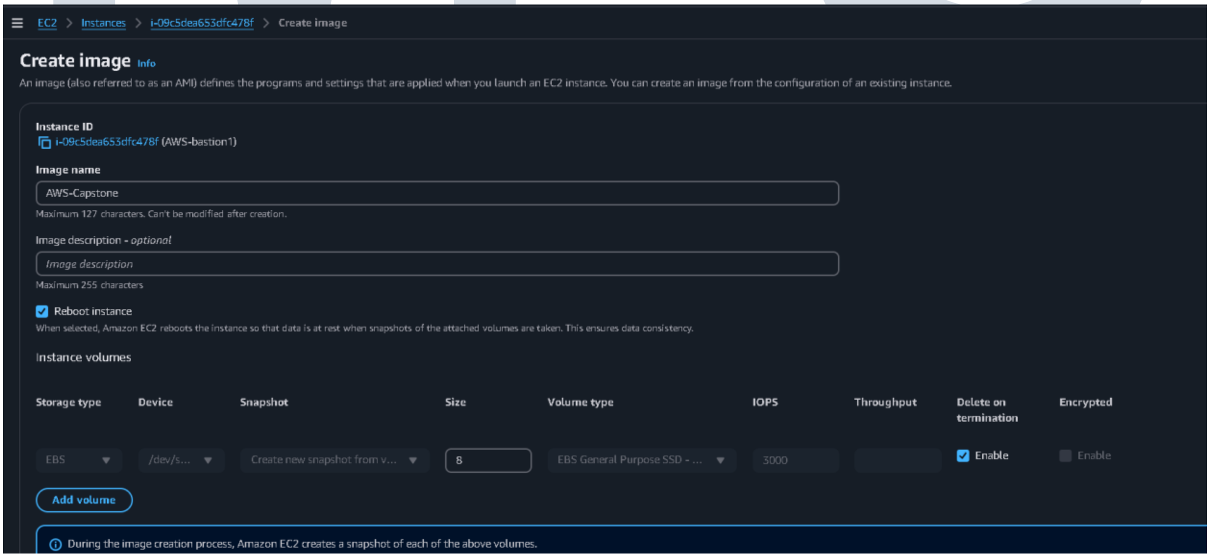
Instances (1/2) Info

Last updated 38 minutes ago

Find Instance by attribute or tag (case-sensitive)

All states

	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availabi
□	AWS-bastion	i-0f6af40d0def75c47	Terminated	t2.micro	–	View alarms +	us-east-1
☑	AWS-bastion1	i-09c5dea653dfc478f	Running	t2.micro	2/2 checks passed	View alarms +	us-east-1



EC2 > Instances > i-09c5dea653dfc478f > Create image

Create image Info

An image (also referred to as an AMI) defines the programs and settings that are applied when you launch an EC2 instance. You can create an image from the configuration of an existing instance.

Instance ID
i-09c5dea653dfc478f (AWS-bastion1)

Image name
AWS-Capstone
Maximum 127 characters. Can't be modified after creation.

Image description - *optional*
Image description
Maximum 255 characters

☑ Reboot instance
When selected, Amazon EC2 reboots the instance so that data is at rest when snapshots of the attached volumes are taken. This ensures data consistency.

Instance volumes

Storage type	Device	Snapshot	Size	Volume type	IOPS	Throughput	Delete on termination	Encrypted
EBS	/dev/s...	Create new snapshot from v...	8	EBS General Purpose SSD - ...	3000		☑ Enable	□ Enable

During the image creation process, Amazon EC2 creates a snapshot of each of the above volumes.

Click on create.

Enable Auto Scaling on EC2 Instances.

Create a Launch Template:

* Go to EC2 Dashboard &gt; Launch Templates.

* Click Create Launch Template.

* Set up the template details:

o Template Name: Enter a name.

o AMI ID: Use the same AMI ID as your initial EC2 instance.

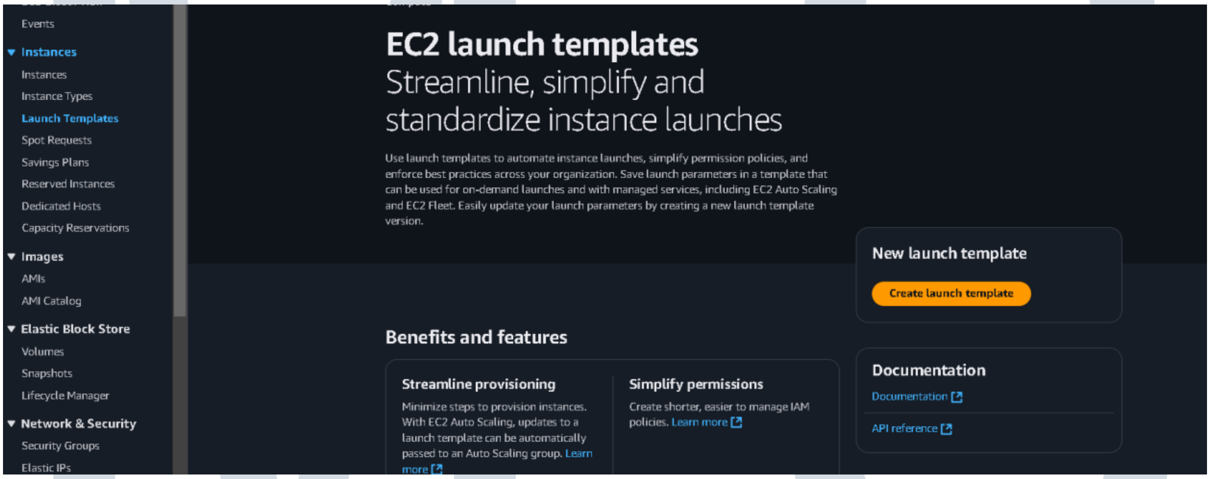
o Instance Type: Choose the same type as the first instance.

o Key Pair: Select the key pair used for the first instance.

* Network Settings:

o Security Groups: Select the security group used for the initial instance.

* Click Create Launch Template.



EC2 launch templates

Streamline, simplify and standardize instance launches

Use launch templates to automate instance launches, simplify permission policies, and enforce best practices across your organization. Save launch parameters in a template that can be used for on-demand launches and with managed services, including EC2 Auto Scaling and EC2 Fleet. Easily update your launch parameters by creating a new launch template version.

New launch template

Create launch template

Benefits and features

Streamline provisioning

Minimize steps to provision instances. With EC2 Auto Scaling, updates to a launch template can be automatically passed to an Auto Scaling group. Learn more

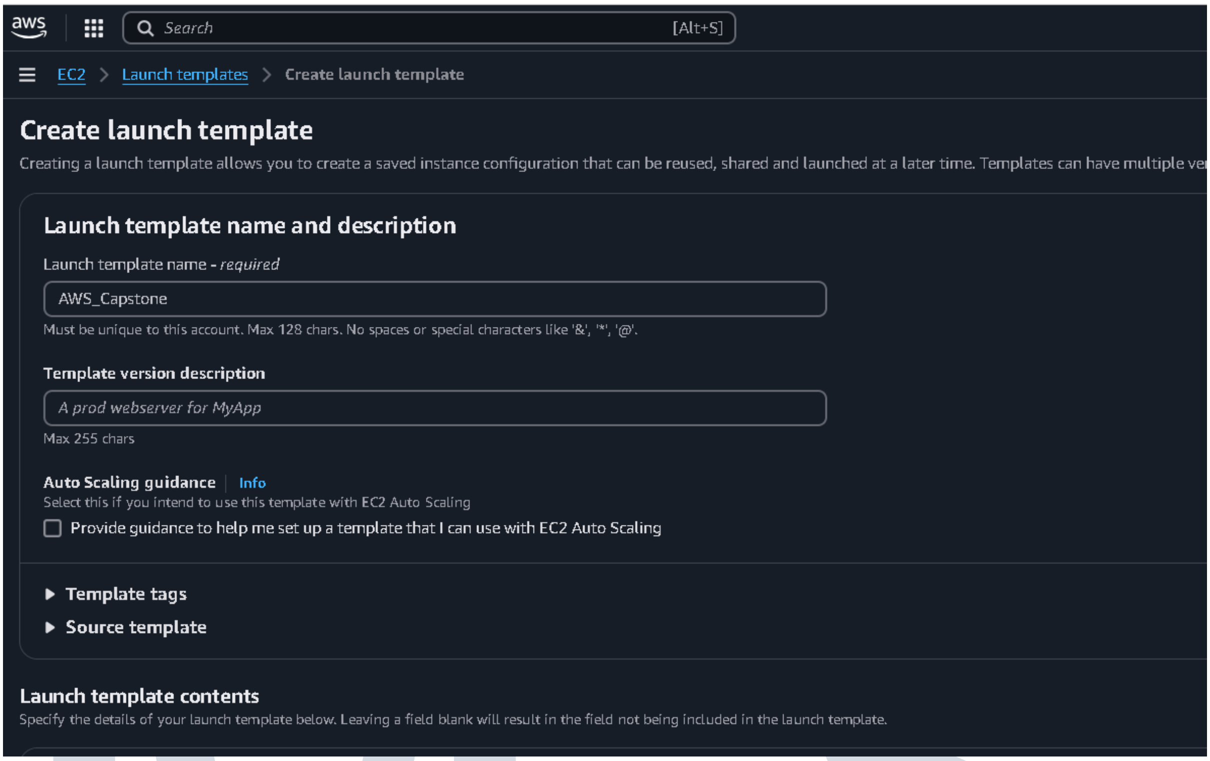
Simplify permissions

Create shorter, easier to manage IAM policies. Learn more

Documentation

Documentation

API reference



Create launch template

Creating a launch template allows you to create a saved instance configuration that can be reused, shared and launched at a later time. Templates can have multiple ve

Launch template name and description

Launch template name - *required*

AWS_Capstone

Must be unique to this account. Max 128 chars. No spaces or special characters like '&', '*', '@'.

Template version description

A prod webserver for MyApp

Max 255 chars

Auto Scaling guidance | Info

Select this if you intend to use this template with EC2 Auto Scaling

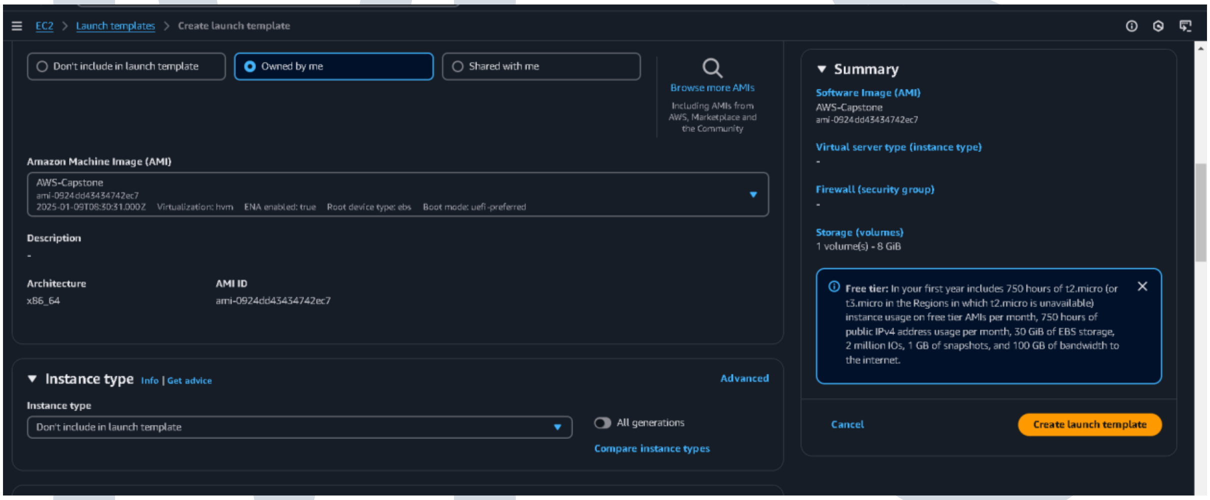
□ Provide guidance to help me set up a template that I can use with EC2 Auto Scaling

▶ Template tags

▶ Source template

Launch template contents

Specify the details of your launch template below. Leaving a field blank will result in the field not being included in the launch template.



EC2 > Launch templates > Create launch template
Browse more AMIs
Including AMIs from AWS, Marketplace and the Community
Advanced
Compare instance types
Cancel
Create launch template

○ Don't include in launch template ● Owned by me ○ Shared with me

Amazon Machine Image (AMI)

AWS-Capstone
ami-0924dd43434742ec7
2025-01-09T08:30:31.000Z Virtualization: hvm ENA enabled: true Root device type: ebs Boot mode: uefi-preferred

Description
-

Architecture: x86_64

AMI ID: ami-0924dd43434742ec7

▼ Instance type Info | Get advice

Instance type: Don't include in launch template

○ All generations

▼ Summary

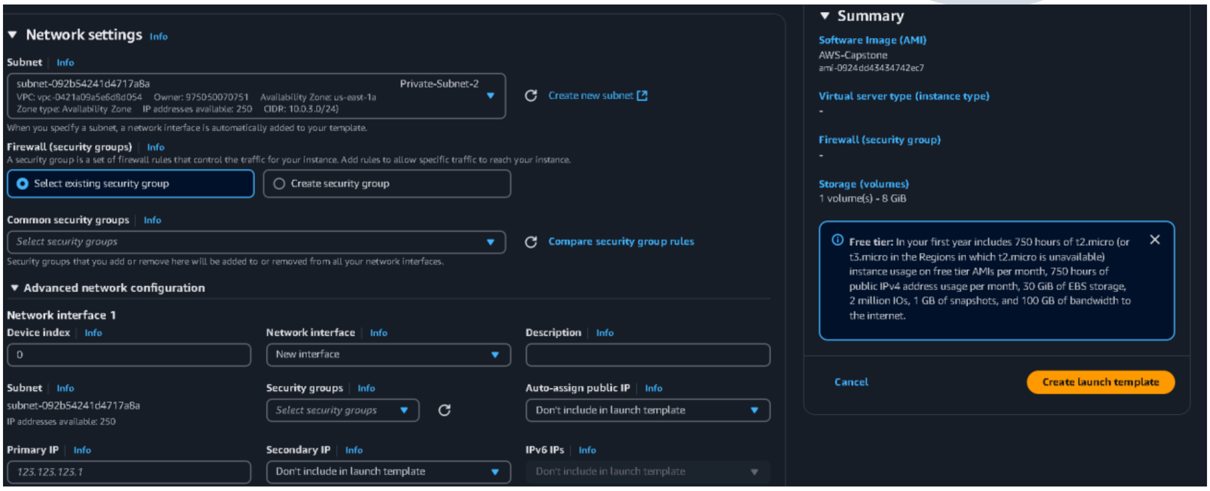
Software Image (AMI)
AWS-Capstone
ami-0924dd43434742ec7

Virtual server type (instance type)
-

Firewall (security group)
-

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

Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 750 hours of public IPv4 address usage per month, 30 GiB of EBS storage, 2 million IOs, 1 GB of snapshots, and 100 GB of bandwidth to the internet.



▼ Network settings Info

Subnet | Info

subnet-092b54241d4717a8a Private-Subnet-2
VPC: vpc-0421a09a5e6d8d054 Owner: 975050070751 Availability Zone: us-east-1a
Zone type: Availability Zone IP addresses available: 250 CIDR: 10.0.3.0/24)

When you specify a subnet, a network interface is automatically added to your template.

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.

● Select existing security group ○ Create security group

Common security groups | Info

Select security groups

Security groups that you add or remove here will be added to or removed from all your network interfaces.

▼ Advanced network configuration

Network interface 1

Device index | Info 0	Network interface | Info New interface	Description | Info
Subnet | Info subnet-092b54241d4717a8a IP addresses available: 250	Security groups | Info *Select security groups*	Auto-assign public IP | Info Don't include in launch template
Primary IP | Info *123.123.123.1*	Secondary IP | Info Don't include in launch template	IPv6 IPs | Info Don't include in launch template

▼ Summary

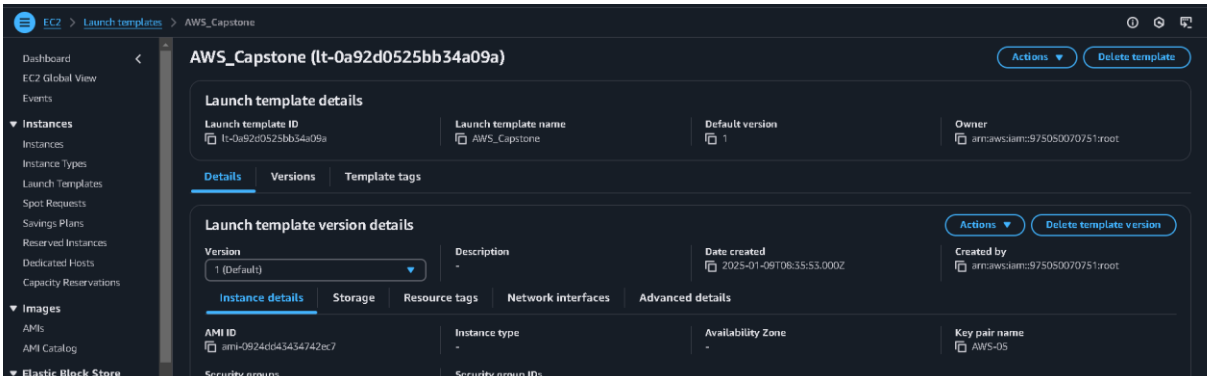
Software Image (AMI)
AWS-Capstone
ami-0924dd43434742ec7

Virtual server type (instance type)
-

Firewall (security group)
-

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

Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 750 hours of public IPv4 address usage per month, 30 GiB of EBS storage, 2 million IOs, 1 GB of snapshots, and 100 GB of bandwidth to the internet.



AWS_Capstone (lt-0a92d0525bb34a09a)

Launch template details

Launch template ID: lt-0a92d0525bb34a09a
Launch template name: AWS_Capstone
Default version: 1
Owner: arn:aws:iam::975050070751:root

Launch template version details

Version: 1 (Default)
Description: -
Date created: 2025-01-09T08:35:53.000Z
Created by: arn:aws:iam::975050070751:root

AMI ID: ami-0924dd43434742ec7
Instance type: -
Availability Zone: -
Key pair name: AWS-05
Security groups
Security group IDs

Enable Auto Scaling on these instances (minimum 2):

Create an Auto Scaling Group.

Go to Auto Scaling Groups in the EC2 dashboard.

Click Create Auto Scaling Group.

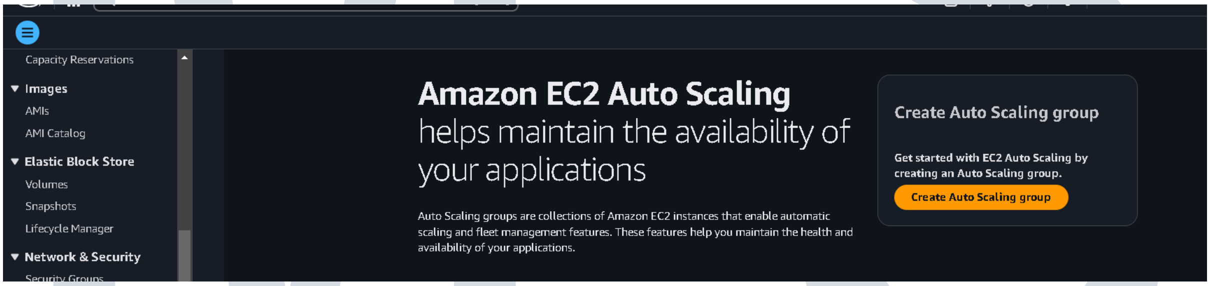
Choose the launch template created above.

Configure Group Details:

 Set Minimum Desired Capacity to 2 and Maximum Capacity to a higher value i.e. 3.

Configure Scaling Policies to manage scaling actions based on load.

Review and create the Auto Scaling group.

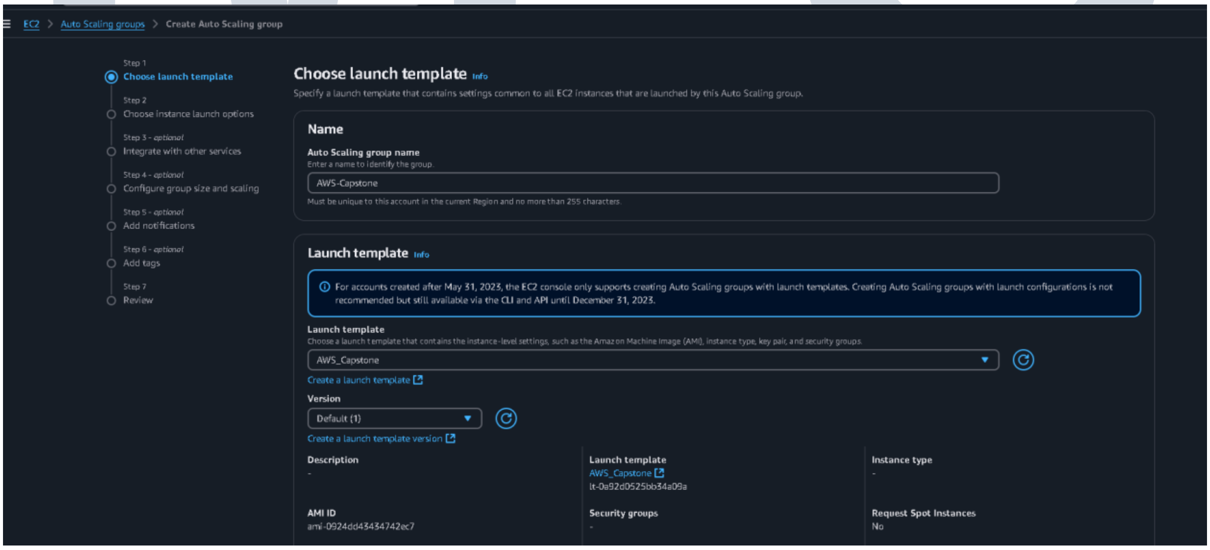


Amazon EC2 Auto Scaling helps maintain the availability of your applications

Auto Scaling groups are collections of Amazon EC2 instances that enable automatic scaling and fleet management features. These features help you maintain the health and availability of your applications.

Create Auto Scaling group

Get started with EC2 Auto Scaling by creating an Auto Scaling group.



EC2 > Auto Scaling groups > Create Auto Scaling group
Step 1
Choose launch template
Step 2
Choose instance launch options
Step 3 - optional
Integrate with other services
Step 4 - optional
Configure group size and scaling
Step 5 - optional
Add notifications
Step 6 - optional
Add tags
Step 7
Review
Info
Create a launch template
Create a launch template version

Choose launch template

Specify a launch template that contains settings common to all EC2 instances that are launched by this Auto Scaling group.

Name

Auto Scaling group name: AWS-Capstone
Enter a name to identify the group.
Must be unique to this account in the current Region and no more than 255 characters.

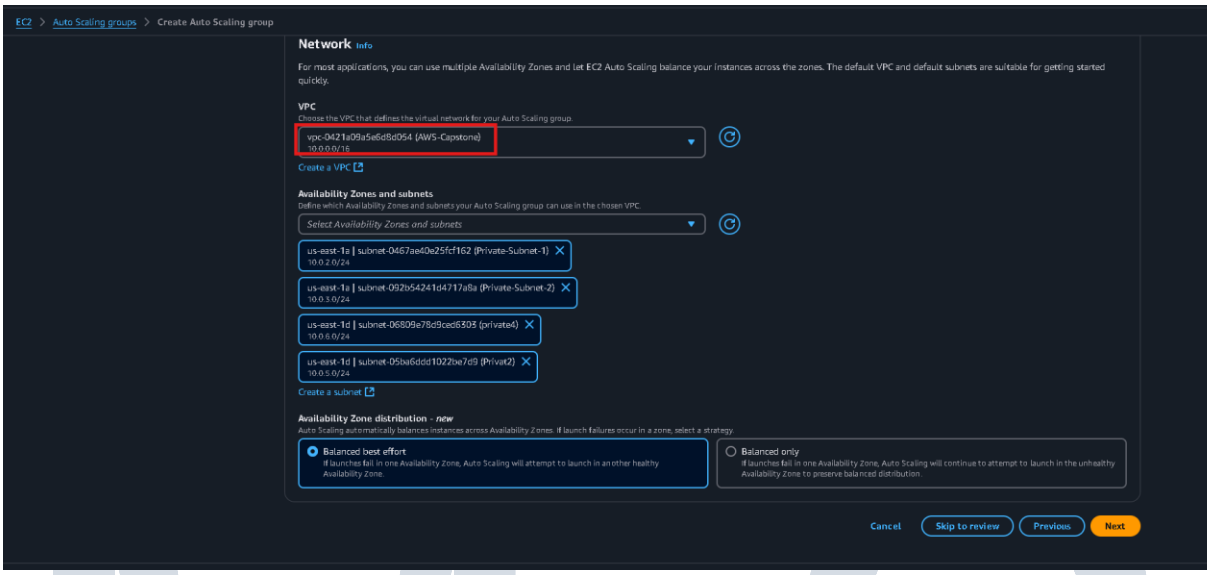
Launch template

For accounts created after May 31, 2023, the EC2 console only supports creating Auto Scaling groups with launch templates. Creating Auto Scaling groups with launch configurations is not recommended but still available via the CLI and API until December 31, 2023.

Launch template: AWS_Capstone
Choose a launch template that contains the instance-level settings, such as the Amazon Machine Image (AMI), instance type, key pair, and security groups.

Version: Default (1)

Description: -
Launch template: AWS_Capstone lt-0a92d0525bb34a09a
Instance type: -
AMI ID: ami-0924dd43434742ec7
Security groups: -
Request Spot Instances: No



Network Info

For most applications, you can use multiple Availability Zones and let EC2 Auto Scaling balance your instances across the zones. The default VPC and default subnets are suitable for getting started quickly.

VPC
Choose the VPC that defines the virtual network for your Auto Scaling group.
vpc-0421a09a5e6d8d054 (AWS-Capstone)
10.0.0.0/16
Create a VPC

Availability Zones and subnets
Define which Availability Zones and subnets your Auto Scaling group can use in the chosen VPC.
Select Availability Zones and subnets

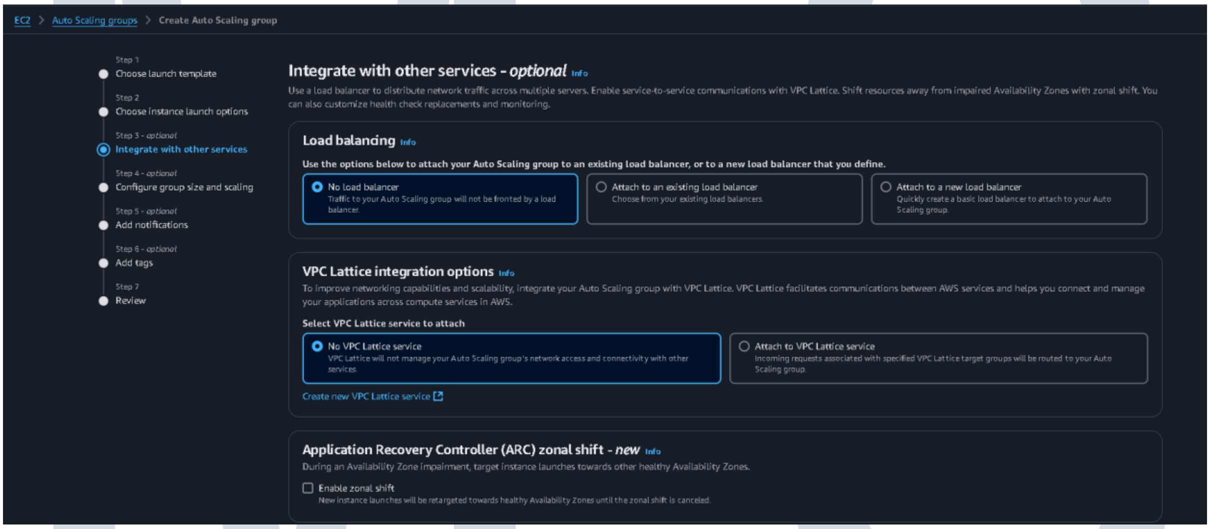
- us-east-1a | subnet-0467ae40e25fcf162 (Private-Subnet-1) 10.0.2.0/24
- us-east-1a | subnet-092b54241d4717a8a (Private-Subnet-2) 10.0.3.0/24
- us-east-1d | subnet-06809e78d9ced6303 (private4) 10.0.6.0/24
- us-east-1d | subnet-05ba6ddd1022be7d9 (Privat2) 10.0.5.0/24

Create a subnet

Availability Zone distribution - *new*
Auto Scaling automatically balances instances across Availability Zones. If launch failures occur in a zone, select a strategy.

● Balanced best effort
If launches fail in one Availability Zone, Auto Scaling will attempt to launch in another healthy Availability Zone.

○ Balanced only
If launches fail in one Availability Zone, Auto Scaling will continue to attempt to launch in the unhealthy Availability Zone to preserve balanced distribution.


EC2 > Auto Scaling groups > Create Auto Scaling group
Step 1
Choose launch template
Step 2
Choose instance launch options
Step 3 - optional
Integrate with other services
Step 4 - optional
Configure group size and scaling
Step 5 - optional
Add notifications
Step 6 - optional
Add tags
Step 7
Review

Integrate with other services - *optional* Info

Use a load balancer to distribute network traffic across multiple servers. Enable service-to-service communications with VPC Lattice. Shift resources away from impaired Availability Zones with zonal shift. You can also customize health check replacements and monitoring.

Load balancing Info

Use the options below to attach your Auto Scaling group to an existing load balancer, or to a new load balancer that you define.

● No load balancer
Traffic to your Auto Scaling group will not be fronted by a load balancer.

○ Attach to an existing load balancer
Choose from your existing load balancers.

○ Attach to a new load balancer
Quickly create a basic load balancer to attach to your Auto Scaling group.

VPC Lattice integration options Info

To improve networking capabilities and scalability, integrate your Auto Scaling group with VPC Lattice. VPC Lattice facilitates communications between AWS services and helps you connect and manage your applications across compute services in AWS.

Select VPC Lattice service to attach

● No VPC Lattice service
VPC Lattice will not manage your Auto Scaling group's network access and connectivity with other services.

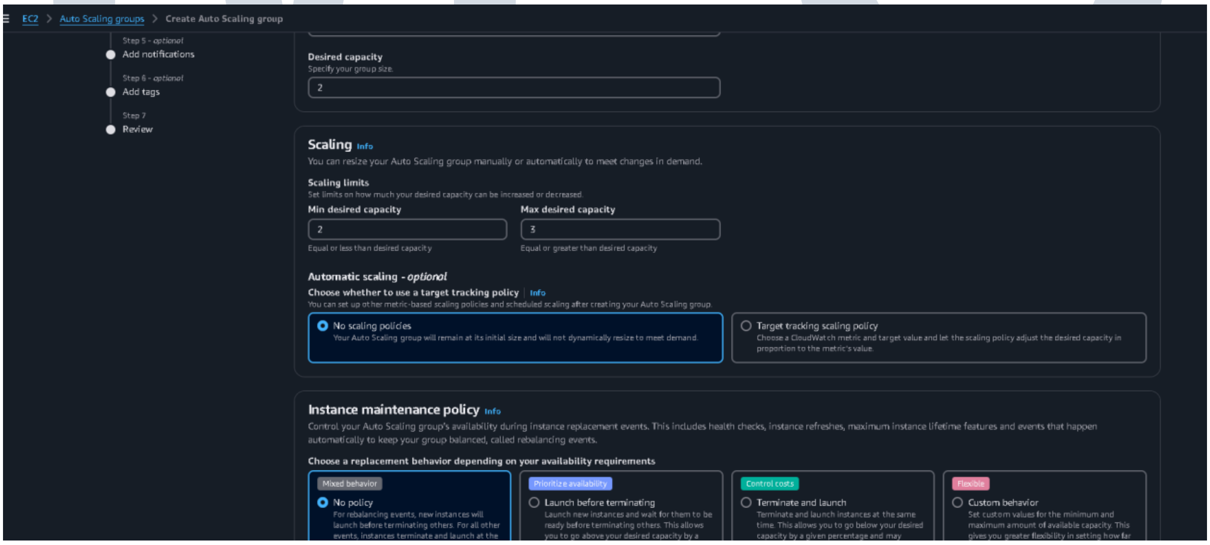
○ Attach to VPC Lattice service
Incoming requests associated with specified VPC Lattice target groups will be routed to your Auto Scaling group.

Create new VPC Lattice service

Application Recovery Controller (ARC) zonal shift - *new* Info

During an Availability Zone impairment, target instance launches towards other healthy Availability Zones.

□ Enable zonal shift
New instance launches will be retargeted towards healthy Availability Zones until the zonal shift is canceled.



Desired capacity
Specify your group size.
2

Scaling Info

You can resize your Auto Scaling group manually or automatically to meet changes in demand.

Scaling limits
Set limits on how much your desired capacity can be increased or decreased.

Min desired capacity
2
Equal or less than desired capacity

Max desired capacity
3
Equal or greater than desired capacity

Automatic scaling - *optional*
Choose whether to use a target tracking policy Info
You can set up other metric-based scaling policies and scheduled scaling after creating your Auto Scaling group.

● No scaling policies
Your Auto Scaling group will remain at its initial size and will not dynamically resize to meet demand.

○ Target tracking scaling policy
Choose a CloudWatch metric and target value and let the scaling policy adjust the desired capacity in proportion to the metric's value.

Instance maintenance policy Info

Control your Auto Scaling group's availability during instance replacement events. This includes health checks, instance refreshes, maximum instance lifetime features and events that happen automatically to keep your group balanced, called rebalancing events.

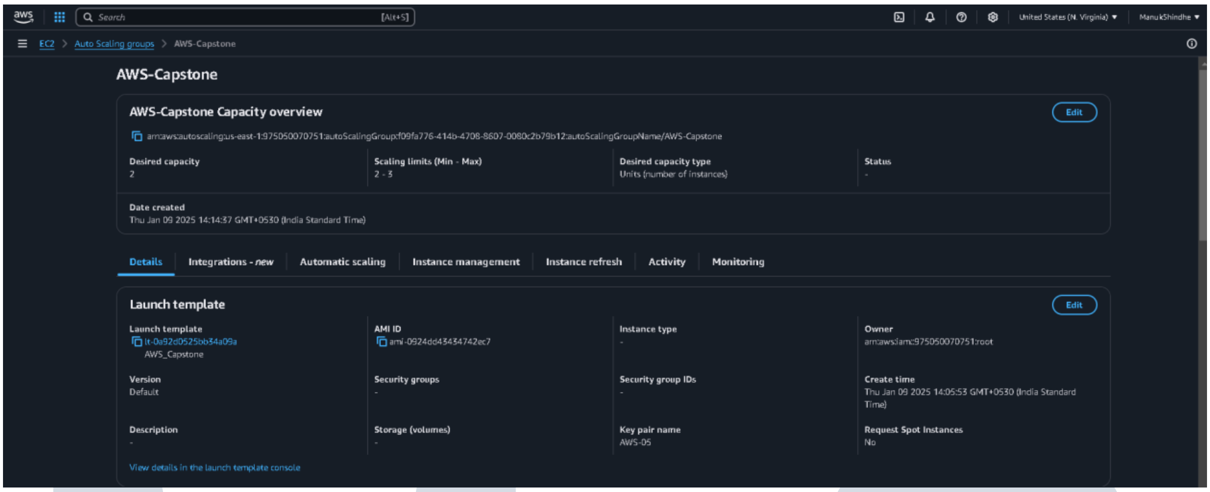
Choose a replacement behavior depending on your availability requirements

Mixed behavior
● No policy
For rebalancing events, new instances will launch before terminating others. For all other events, instances terminate and launch at the

Prioritize availability
○ Launch before terminating
Launch new instances and wait for them to be ready before terminating others. This allows you to go above your desired capacity by a

Control costs
○ Terminate and launch
Terminate and launch instances at the same time. This allows you to go below your desired capacity by a given percentage and may

Flexible
○ Custom behavior
Set custom values for the minimum and maximum amount of available capacity. This gives you greater flexibility in setting how far



EC2 > Auto Scaling groups > AWS-Capstone

AWS-Capstone

AWS-Capstone Capacity overview

arn:aws:autoscaling:us-east-1:975050070751:autoScalingGroup:f09fa776-414b-4708-8607-0080c2b79b12:autoScalingGroupName/AWS-Capstone

Desired capacity	Scaling limits (Min - Max)	Desired capacity type	Status
2	2 - 3	Units (number of instances)	-

Date created
Thu Jan 09 2025 14:14:37 GMT+0530 (India Standard Time)

Launch template

Launch template	AMI ID	Instance type	Owner
lt-0a92d0525bb34a09a AWS_Capstone	ami-0924dd43434742ec7	-	arn:aws:iam::975050070751:root

Version	Security groups	Security group IDs	Create time
Default	-	-	Thu Jan 09 2025 14:05:53 GMT+0530 (India Standard Time)

Description	Storage (volumes)	Key pair name	Request Spot Instances
-	-	AWS-05	No

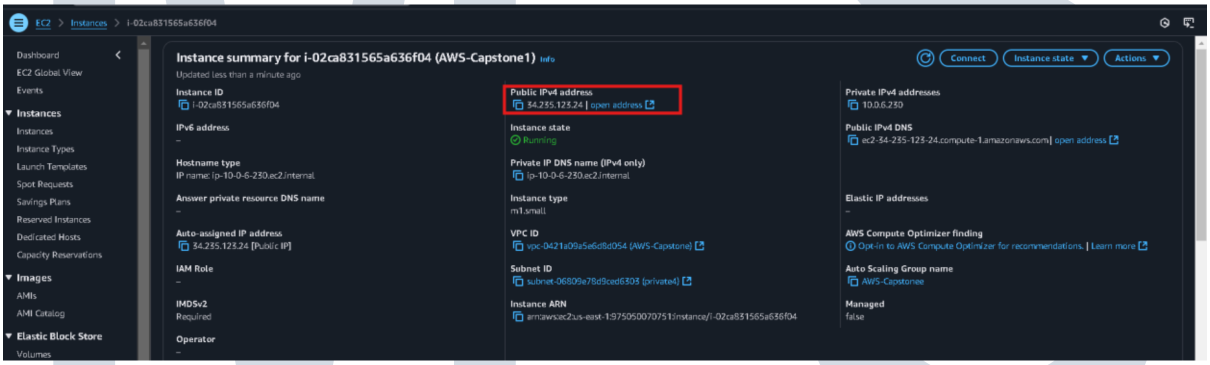
View details in the launch template console

Confirm Application Functionality.

Access the website to confirm that PHP scripts on all instances can connect to the RDS database and function correctly.

So copy the IP address of the newly created instances of Auto-Scaling group's instance then very the and try to add the data and verify in database.

1)

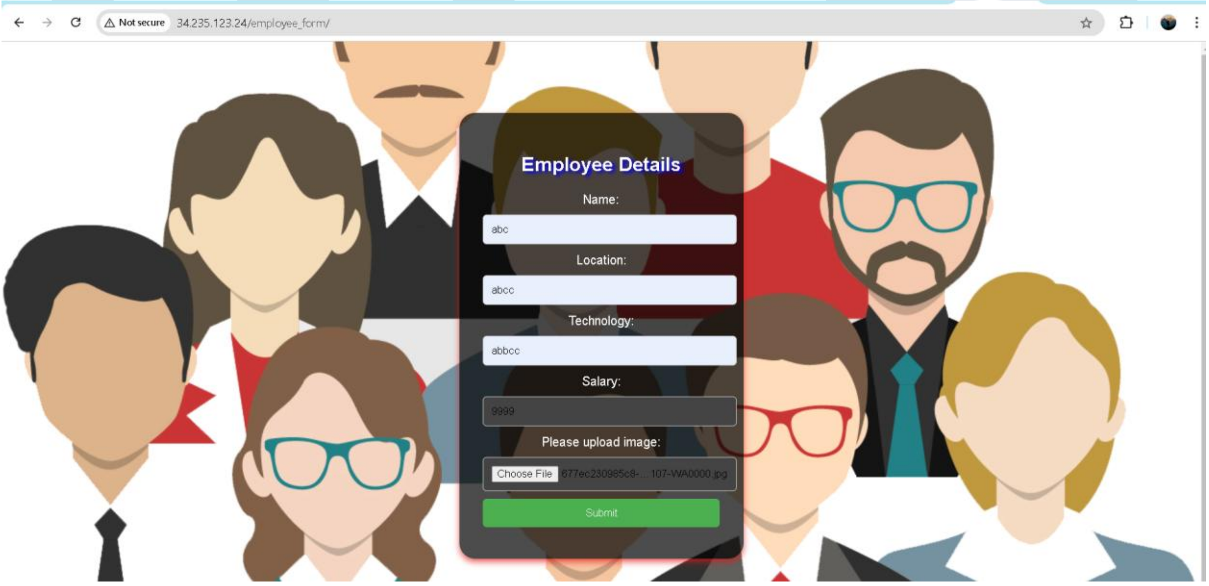


EC2 > Instances > i-02ca831565a636f04

Instance summary for i-02ca831565a636f04 (AWS-Capstone1)

Updated less than a minute ago

Instance ID i-02ca831565a636f04	Public IPv4 address 34.235.123.24 | open address	Private IPv4 addresses 10.0.6.230
IPv6 address –	Instance state Running	Public IPv4 DNS ec2-34-235-123-24.compute-1.amazonaws.com| open address
Hostname type IP name: ip-10-0-6-230.ec2.internal	Private IP DNS name (IPv4 only) ip-10-0-6-230.ec2.internal	
Answer private resource DNS name –	Instance type m1.small	Elastic IP addresses –
Auto-assigned IP address 34.235.123.24 [Public IP]	VPC ID vpc-0421a09a5e6d8d054 (AWS-Capstone)	AWS Compute Optimizer finding Opt-in to AWS Compute Optimizer for recommendations. | Learn more
IAM Role –	Subnet ID subnet-06809e78d9ced6303 (private4)	Auto Scaling Group name AWS-Capstonee
IMDSv2 Required	Instance ARN arn:aws:ec2:us-east-1:975050070751:instance/i-02ca831565a636f04	Managed false
Operator –		



Not secure 34.235.123.24/employee_form/

Employee Details

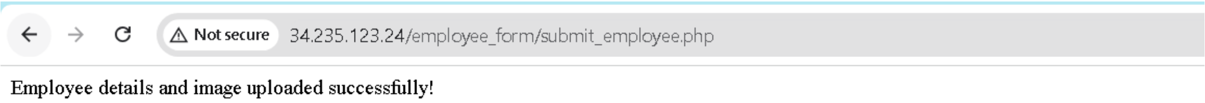
Name: abc

Location: abcc

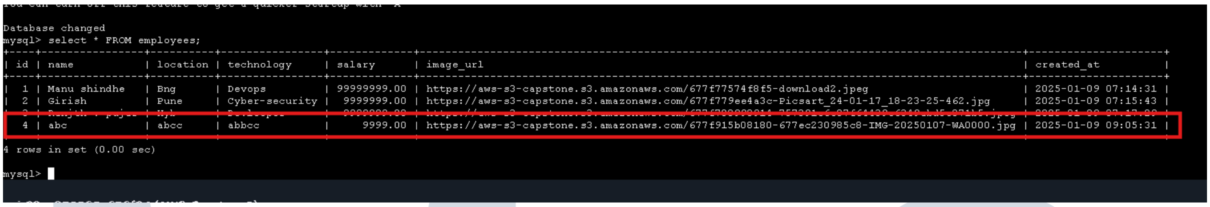
Technology: abbcc

Salary: 9999

Please upload image: Choose File 677ec230985c8-....107-WA0000.jpg

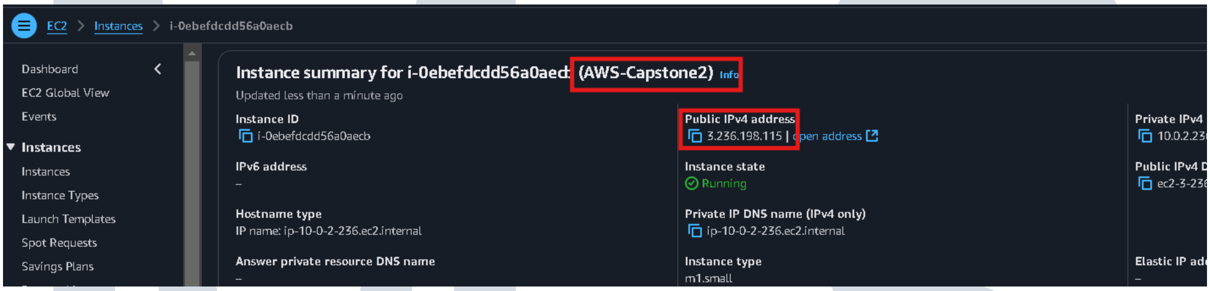


Employee details and image uploaded successfully!



```
Database changed
mysql> select * FROM employees;
+----+----------------+----------+----------------+-------------+-----------------------------------------------------------------------------------------------+---------------------+
| id | name           | location | technology     | salary      | image_url                                                                                     | created_at          |
+----+----------------+----------+----------------+-------------+-----------------------------------------------------------------------------------------------+---------------------+
|  1 | Manu shindhe   | Bng      | Devops         | 99999999.00 | https://aws-s3-capstone.s3.amazonaws.com/677f77574f8f5-download2.jpeg                         | 2025-01-09 07:14:31 |
|  2 | Girish         | Pune     | Cyber-security |  9999999.00 | https://aws-s3-capstone.s3.amazonaws.com/677f779ee4a3c-Picsart_24-01-17_18-23-25-462.jpg      | 2025-01-09 07:15:43 |
|  3 | [illegible]
|  4 | abc            | abcc     | abbcc          |     9999.00 | https://aws-s3-capstone.s3.amazonaws.com/677f915b08180-677ec230985c8-IMG-20250107-WA0000.jpg  | 2025-01-09 09:05:31 |
4 rows in set (0.00 sec)

mysql>
```

2)



Instance summary for i-0ebefdcdd56a0aecb (AWS-Capstone2)

Updated less than a minute ago

Instance ID: i-0ebefdcdd56a0aecb

Public IPv4 address: 3.236.198.115 | open address

Private IPv4: 10.0.2.23

IPv6 address: –

Instance state: Running

Public IPv4 D: ec2-3-23

Hostname type: IP name: ip-10-0-2-236.ec2.internal

Private IP DNS name (IPv4 only): ip-10-0-2-236.ec2.internal

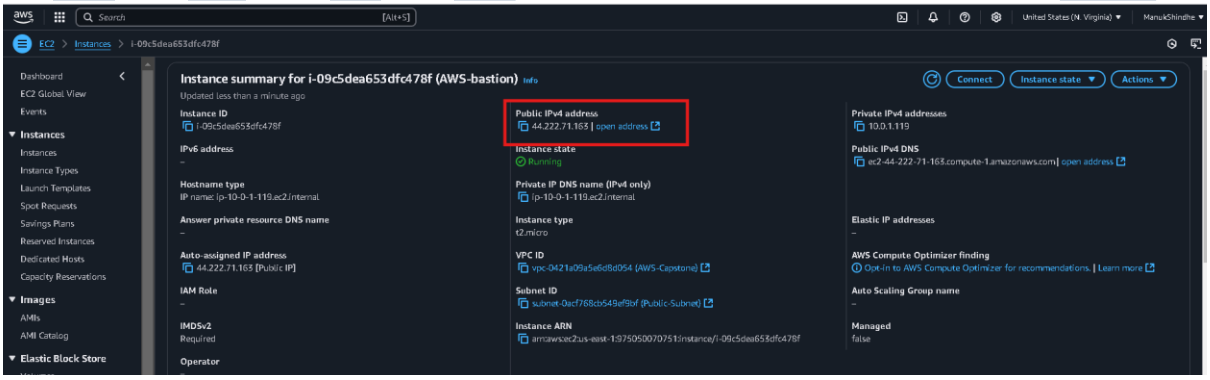
Answer private resource DNS name: –

Instance type: m1.small

Elastic IP ad: –

Similarly, we will get same output for AWS-Capstone2 also. Once we information is provided over browser/internet by user this information will store backend in RDS table.

Now we will access AWS-bastion host



Instance summary for i-09c5dea653dfc478f (AWS-bastion)

Updated less than a minute ago

Instance ID: i-09c5dea653dfc478f

Public IPv4 address: 44.222.71.163 | open address

Private IPv4 addresses: 10.0.1.119

IPv6 address: –

Instance state: Running

Public IPv4 DNS: ec2-44-222-71-163.compute-1.amazonaws.com| open address

Hostname type: IP name: ip-10-0-1-119.ec2.internal

Private IP DNS name (IPv4 only): ip-10-0-1-119.ec2.internal

Answer private resource DNS name: –

Instance type: t2.micro

Elastic IP addresses: –

Auto-assigned IP address: 44.222.71.163 [Public IP]

VPC ID: vpc-0421a09a5e6d8d054 (AWS-Capstone)

AWS Compute Optimizer finding: Opt-in to AWS Compute Optimizer for recommendations. | Learn more

IAM Role: –

Subnet ID: subnet-0acf768cb549ef9bf (Public-Subnet)

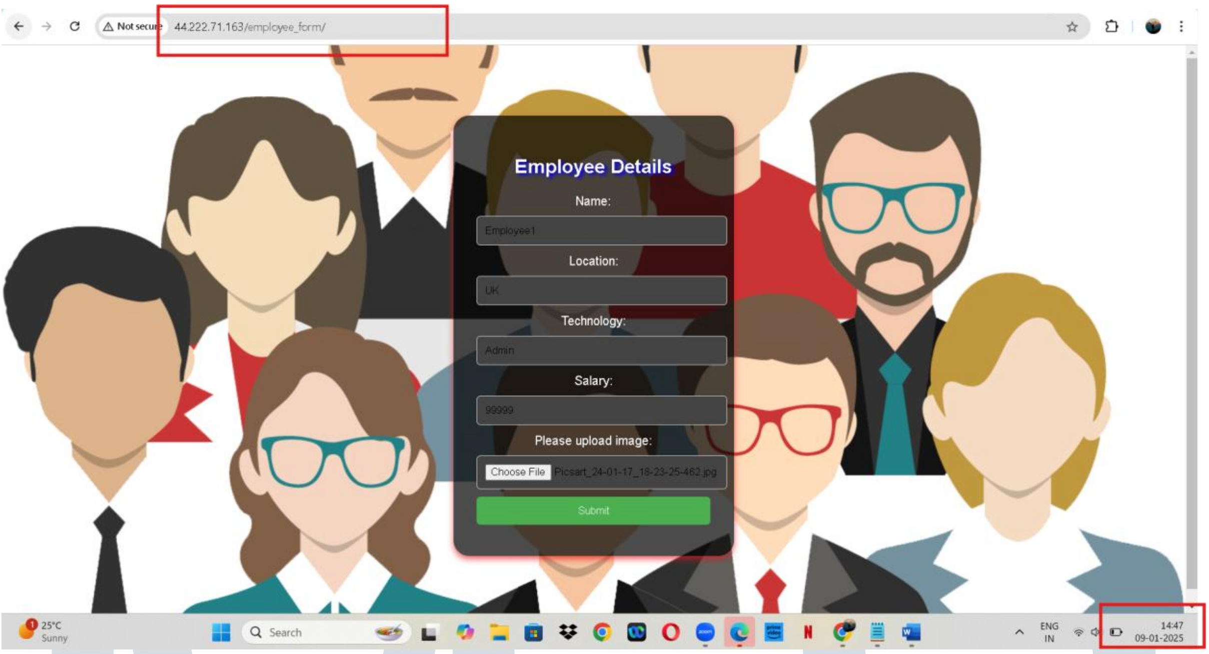
Auto Scaling Group name: –

IMDSv2: Required

Instance ARN: arn:aws:ec2:us-east-1:975050070751:instance/i-09c5dea653dfc478f

Managed: false

Operator



44.222.71.163/employee_form/

Employee Details

Name: Employee1

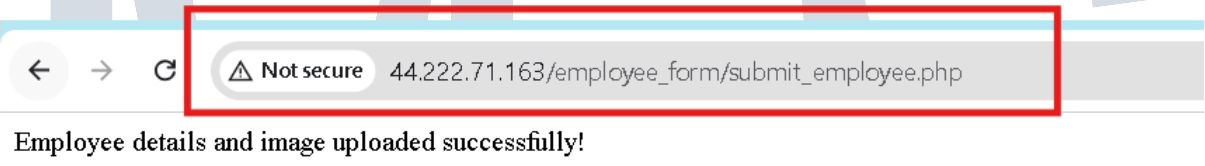
Location: UK

Technology: Admin

Salary: 99999

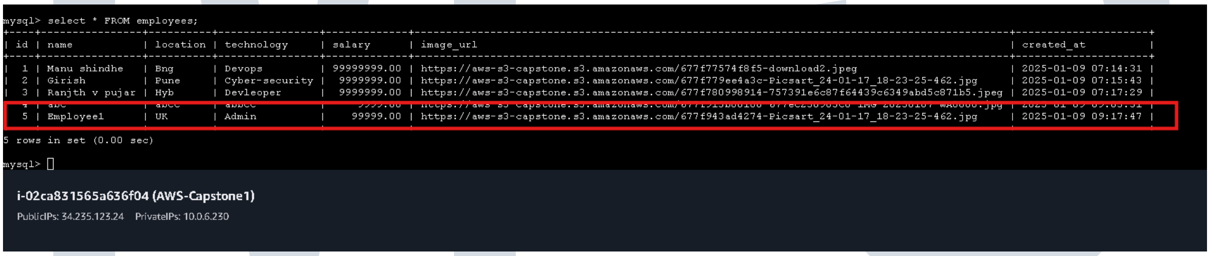
Please upload image: Choose File Picsart_24-01-17_18-23-25-462.jpg

Submit



Not secure 44.222.71.163/employee_form/submit_employee.php

Employee details and image uploaded successfully!



```
mysql> select * FROM employees;
+----+-----------------+----------+-----------------+-------------+--------------------------------------------------------------------------------------------------+---------------------+
| id | name            | location | technology      | salary      | image_url                                                                                        | created_at          |
+----+-----------------+----------+-----------------+-------------+--------------------------------------------------------------------------------------------------+---------------------+
|  1 | Manu shindhe    | Bng      | Devops          | 99999999.00 | https://aws-s3-capstone.s3.amazonaws.com/677f77574f8f5-download2.jpeg                            | 2025-01-09 07:14:31 |
|  2 | Girish          | Pune     | Cyber-security  |  9999999.00 | https://aws-s3-capstone.s3.amazonaws.com/677f779ee4a3c-Picsart_24-01-17_18-23-25-462.jpg          | 2025-01-09 07:15:43 |
|  3 | Ranjth v pujar  | Hyb      | Devleoper       |  9999999.00 | https://aws-s3-capstone.s3.amazonaws.com/677f780998914-757391e6c87f64439c6349abd5c871b5.jpeg     | 2025-01-09 07:17:29 |
|  4 | abc             | abcc     | abbcc           |     9999.00 | https://aws-s3-capstone.s3.amazonaws.com/[illegible]-IMG-[illegible]-WA0000.jpg                   | 2025-01-09 [illegible] |
|  5 | Employee1       | UK       | Admin           |    99999.00 | https://aws-s3-capstone.s3.amazonaws.com/677f943ad4274-Picsart_24-01-17_18-23-25-462.jpg          | 2025-01-09 09:17:47 |
+----+-----------------+----------+-----------------+-------------+--------------------------------------------------------------------------------------------------+---------------------+
5 rows in set (0.00 sec)

mysql> 
```

i-02ca831565a636f04 (AWS-Capstone1)

PublicIPs: 34.235.123.24 PrivateIPs: 10.0.6.230

Step 10:

Create S3 Event To send notification to SNS

Steps to Set up S3 Event Notification with SNS

Step 1: Create an SNS Topic

First, you need to create an SNS topic that will receive the notifications from your S3 bucket.

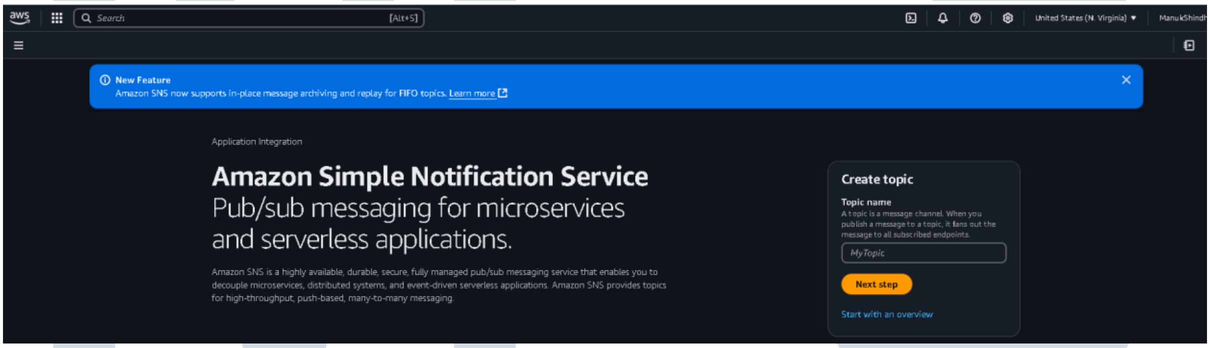
1. Log in to the AWS Management Console.
2. Navigate to SNS:
   - In the AWS Management Console, search for and select SNS (Simple Notification Service).
3. Create a Topic:
   - Click on Create topic.

- Choose Standard as the type.
- Give your topic a name
- You can also add a display name if you like
- Click Create topic to create the SNS topic.

4. Get the Topic ARN:
5. Step 2: Subscribe to the SNS Topic

Next, you'll want to subscribe to this SNS topic so that you can receive notifications.

1. In the SNS dashboard, select the topic you just created.
2. Click on Create subscription.
3. Select the protocol (e.g., Email, SMS, Lambda, or HTTP/HTTPS) to receive the notifications.
    - For example, choose Email if you want to receive notifications via email.
4. Enter the endpoint (e.g., your email address for an email subscription).
5. Click Create subscription.
    - You will need to confirm the subscription if you chose Email or another endpoint that requires confirmation.
    - Check your email and click the confirmation link to confirm your subscription.



Amazon SNS now supports in-place message archiving and replay for FIFO topics. Learn more

Application Integration

Amazon Simple Notification Service
Pub/sub messaging for microservices and serverless applications.

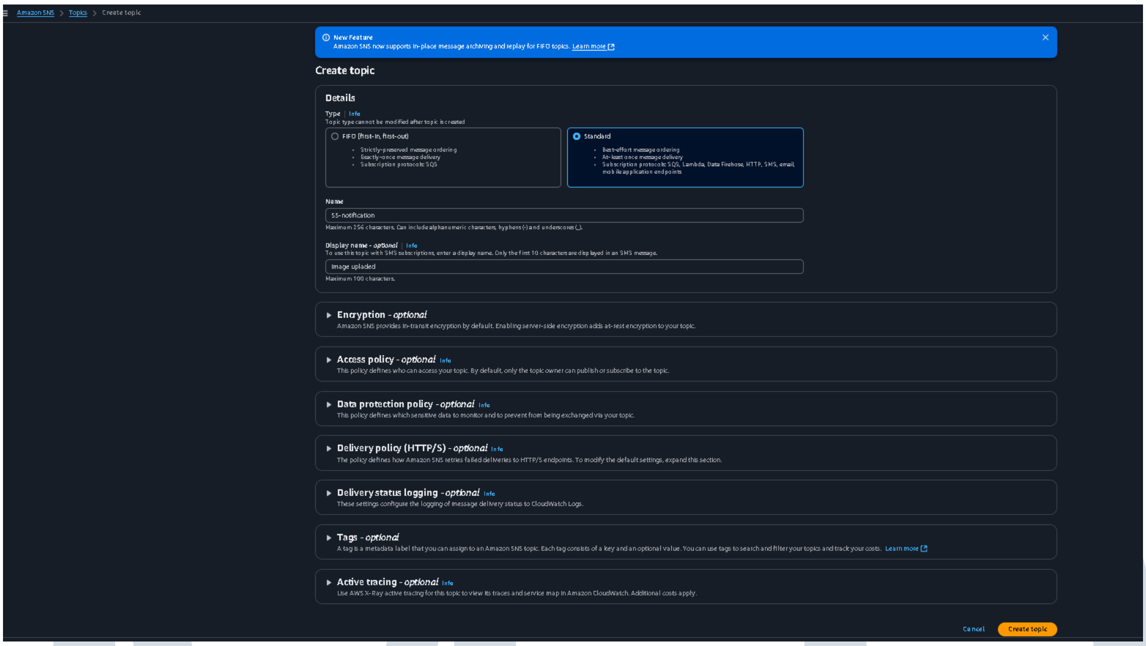
Amazon SNS is a highly available, durable, secure, fully managed pub/sub messaging service that enables you to decouple microservices, distributed systems, and event-driven serverless applications. Amazon SNS provides topics for high-throughput, push-based, many-to-many messaging.

Create topic

Topic name

A topic is a message channel. When you publish a message to a topic, it fans out the message to all subscribed endpoints.

MyTopic


Amazon SNS > Topics > Create topic

New Feature
Amazon SNS now supports in-place message archiving and replay for FIFO topics. Learn more

Create topic

Details

Type | Info
Topic type cannot be modified after topic is created

○ FIFO (first-in, first-out)
- Strictly-preserved message ordering
- Exactly-once message delivery
- Subscription protocols: SQS

● Standard
- Best-effort message ordering
- At-least once message delivery
- Subscription protocols: SQS, Lambda, Data Firehose, HTTP, SMS, email, mobile application endpoints

Name
S3-notification
Maximum 256 characters. Can include alphanumeric characters, hyphens (-) and underscores (_).

Display name - *optional* | Info
To use this topic with SMS subscriptions, enter a display name. Only the first 10 characters are displayed in an SMS message.
Image upladed
Maximum 100 characters.

▸ Encryption - *optional*
Amazon SNS provides in-transit encryption by default. Enabling server-side encryption adds at-rest encryption to your topic.

▸ Access policy - *optional* Info
This policy defines who can access your topic. By default, only the topic owner can publish or subscribe to the topic.

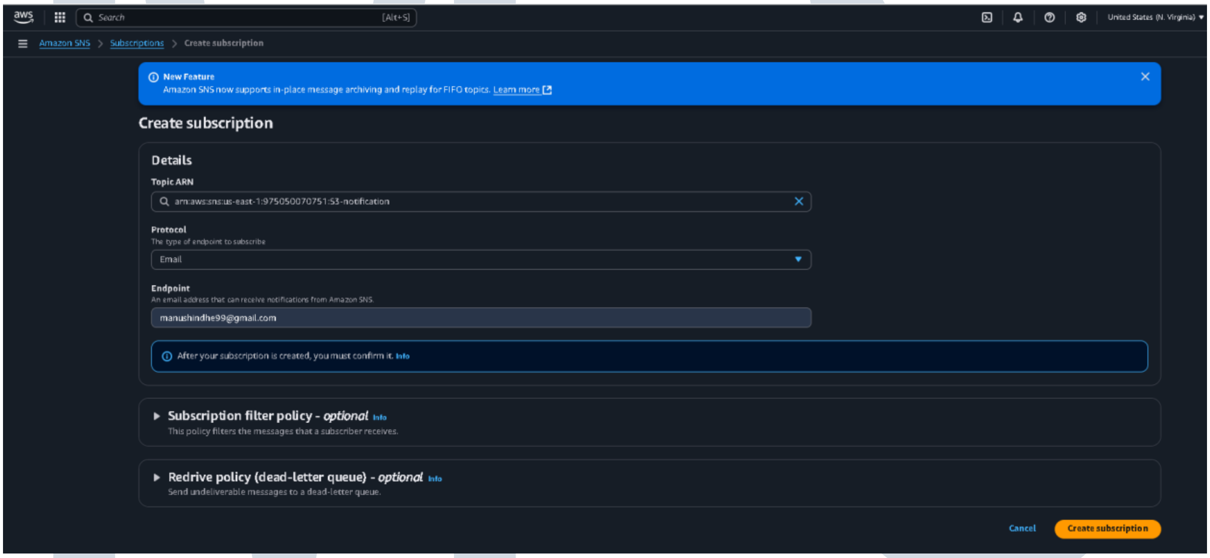
▸ Data protection policy - *optional* Info
This policy defines which sensitive data to monitor and to prevent from being exchanged via your topic.

▸ Delivery policy (HTTP/S) - *optional* Info
The policy defines how Amazon SNS retries failed deliveries to HTTP/S endpoints. To modify the default settings, expand this section.

▸ Delivery status logging - *optional* Info
These settings configure the logging of message delivery status to CloudWatch Logs.

▸ Tags - *optional*
A tag is a metadata label that you can assign to an Amazon SNS topic. Each tag consists of a key and an optional value. You can use tags to search and filter your topics and track your costs. Learn more

▸ Active tracing - *optional* Info
Use AWS X-Ray active tracing for this topic to view its traces and service map in Amazon CloudWatch. Additional costs apply.



Amazon SNS > Subscriptions > Create subscription

New Feature
Amazon SNS now supports in-place message archiving and replay for FIFO topics. Learn more

Create subscription

Details

Topic ARN
arn:aws:sns:us-east-1:975050070751:S3-notification

Protocol
The type of endpoint to subscribe
Email

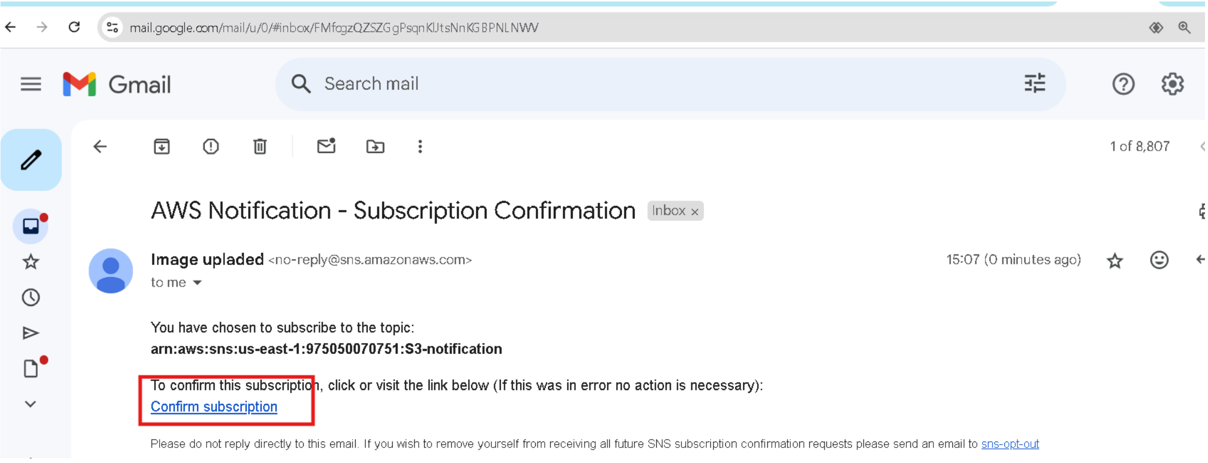
Endpoint
An email address that can receive notifications from Amazon SNS.
manushindhe99@gmail.com

After your subscription is created, you must confirm it. Info

▸ Subscription filter policy - *optional* Info
This policy filters the messages that a subscriber receives.

▸ Redrive policy (dead-letter queue) - *optional* Info
Send undeliverable messages to a dead-letter queue.

Will receive mail for subscription


mail.google.com/mail/u/0/#inbox/FMfcgzQZSZGgPsqnKJtsNnKGBPNLNWV
Gmail
Search mail
1 of 8,807

AWS Notification - Subscription Confirmation Inbox ×

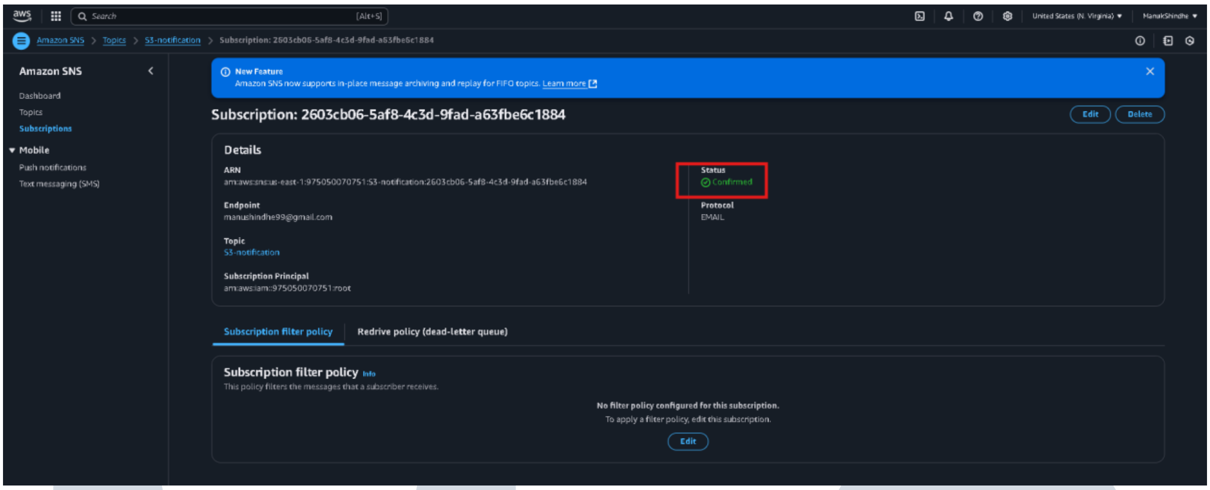
Image upladed <no-reply@sns.amazonaws.com>
to me
15:07 (0 minutes ago)

You have chosen to subscribe to the topic:
arn:aws:sns:us-east-1:975050070751:S3-notification

To confirm this subscription, click or visit the link below (If this was in error no action is necessary):
Confirm subscription

Please do not reply directly to this email. If you wish to remove yourself from receiving all future SNS subscription confirmation requests please send an email to sns-opt-out

Click on confirm


aws
Search [Alt+S]
United States (N. Virginia)
ManakShindhe
Amazon SNS > Topics > S3-notification > Subscription: 2603cb06-5af8-4c3d-9fad-a63fbe6c1884
Amazon SNS
Dashboard
Topics
Subscriptions
Mobile
Push notifications
Text messaging (SMS)
Edit
Delete
Subscription filter policy
Redrive policy (dead-letter queue)
Info
Edit

New Feature
Amazon SNS now supports in-place message archiving and replay for FIFO topics. Learn more

Subscription: 2603cb06-5af8-4c3d-9fad-a63fbe6c1884

Details

ARN
arn:aws:sns:us-east-1:975050070751:S3-notification:2603cb06-5af8-4c3d-9fad-a63fbe6c1884

Status
Confirmed

Endpoint
manushindhe99@gmail.com

Protocol
EMAIL

Topic
S3-notification

Subscription Principal
arn:aws:iam::975050070751:root

Subscription filter policy
This policy filters the messages that a subscriber receives.

No filter policy configured for this subscription.
To apply a filter policy, edit this subscription.

Step 3: Configure S3 Event Notification to Trigger SNS

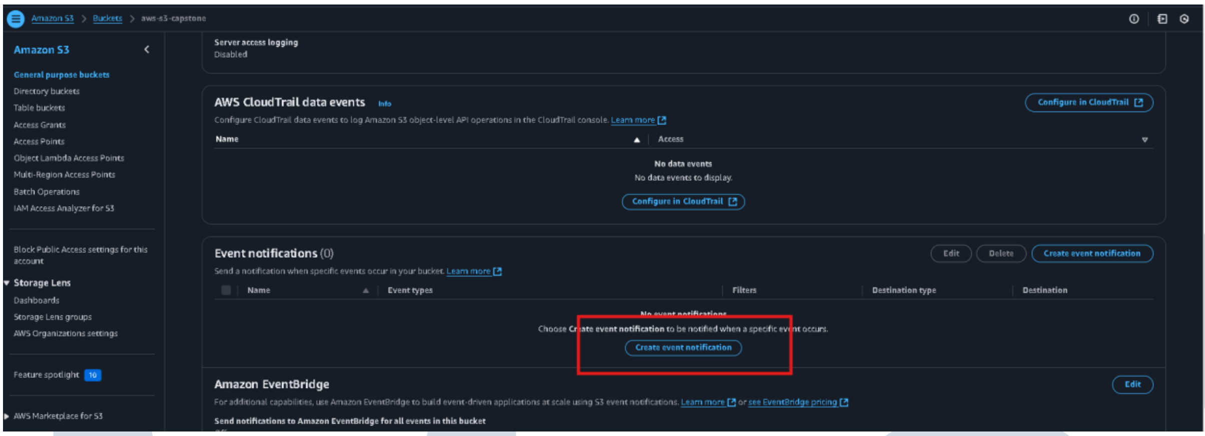
Now, you will configure the S3 bucket to send notifications to the SNS topic when specific events happen in the bucket (e.g., when an object is uploaded).

1. Navigate to S3:
   - In the AWS Management Console, search for and select S3.
2. Select Your S3 Bucket:
   - In the S3 console, select the bucket for which you want to configure the event notification.
3. Go to Properties:
   - In the bucket dashboard, go to the Properties tab.
4. Scroll to "Event Notifications":
   - Under the Event Notifications section, click Create event notification.
5. Configure the Event:
   - Event name: Give your event a name
   - Event types: Select the event type(s) you want to trigger the notification (e.g., ObjectCreated).
   - Prefix/Suffix (optional): You can specify a prefix (to filter by folder or object path) or a suffix (to filter by file extension). Leave blank for all objects.
   - Send to: Select SNS Topic.
   - SNS Topic ARN: Paste the SNS Topic ARN that you copied earlier.
6. Save the Notification:
   - After filling out the necessary fields, click Save changes.

Step 4: Test the Setup

To test whether the S3 event notification to SNS is working:

1. Upload an object to your S3 bucket (either via the console or programmatically).
2. Check the SNS subscription (e.g., your email) to see if you receive a notification.



Server access logging
Disabled

AWS CloudTrail data events

Configure CloudTrail data events to log Amazon S3 object-level API operations in the CloudTrail console. Learn more

Name	Access

No data events
No data events to display.

Event notifications (0)

Send a notification when specific events occur in your bucket. Learn more

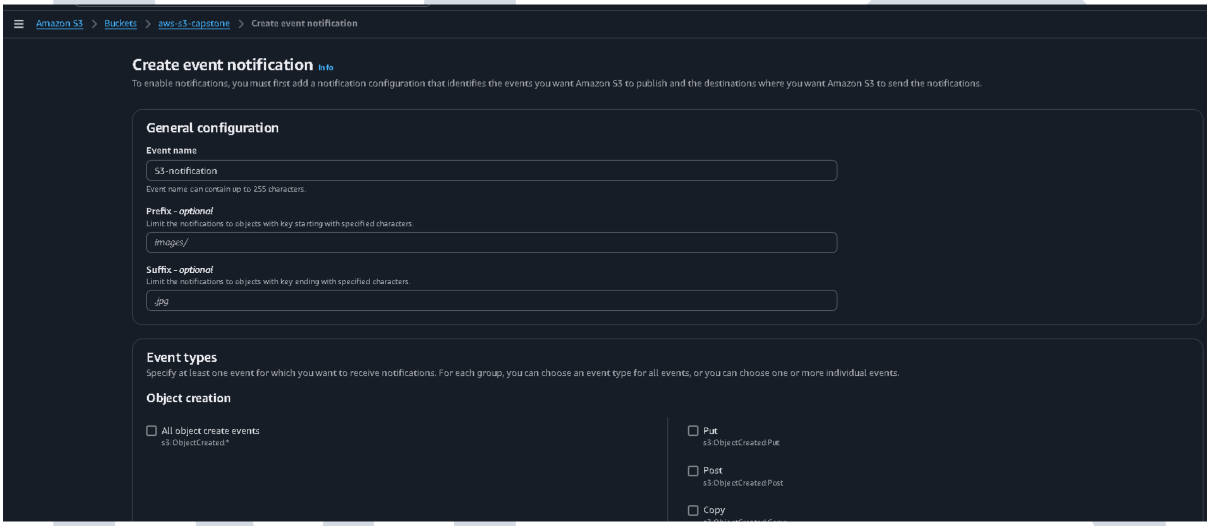
Name	Event types	Filters	Destination type	Destination

No event notifications
Choose Create event notification to be notified when a specific event occurs.

Amazon EventBridge

For additional capabilities, use Amazon EventBridge to build event-driven applications at scale using S3 event notifications. Learn more or see EventBridge pricing

Send notifications to Amazon EventBridge for all events in this bucket



Amazon S3 > Buckets > aws-s3-capstone > Create event notification
Info

Create event notification

To enable notifications, you must first add a notification configuration that identifies the events you want Amazon S3 to publish and the destinations where you want Amazon S3 to send the notifications.

General configuration

Event name: S3-notification
Event name can contain up to 255 characters.

Prefix - *optional*
Limit the notifications to objects with key starting with specified characters.
images/

Suffix - *optional*
Limit the notifications to objects with key ending with specified characters.
.jpg

Event types

Specify at least one event for which you want to receive notifications. For each group, you can choose an event type for all events, or you can choose one or more individual events.

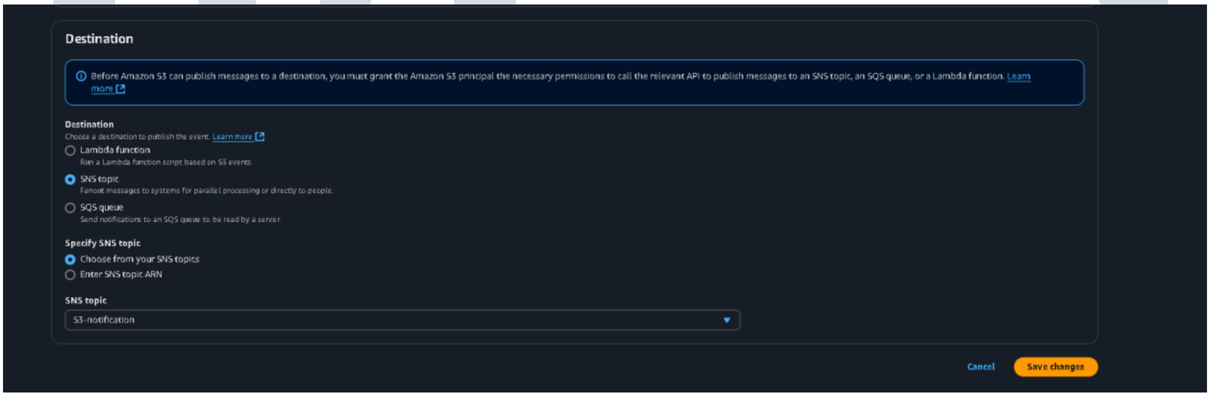
Object creation

□ All object create events
s3:ObjectCreated:*

□ Put
s3:ObjectCreated:Put

□ Post
s3:ObjectCreated:Post

□ Copy



Destination

Before Amazon S3 can publish messages to a destination, you must grant the Amazon S3 principal the necessary permissions to call the relevant API to publish messages to an SNS topic, an SQS queue, or a Lambda function. Learn more

Destination
Choose a destination to publish the event. Learn more

○ Lambda function
Run a Lambda function script based on S3 events.

● SNS topic
Fanout messages to systems for parallel processing or directly to people.

○ SQS queue
Send notifications to an SQS queue to be read by a server.

Specify SNS topic

● Choose from your SNS topics
○ Enter SNS topic ARN

SNS topic: S3-notification

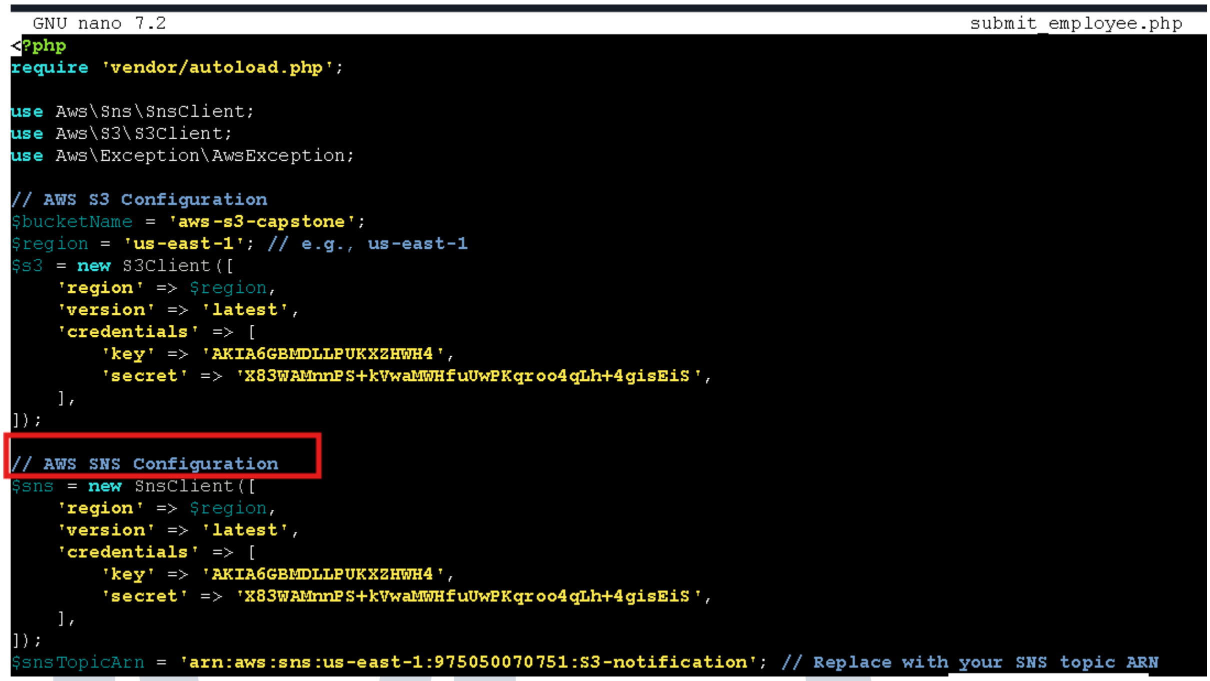
Step 5: Update PHP Code to Publish to SNS

Now, modify your existing PHP code to send a notification to the SNS topic upon successful image upload to S3.

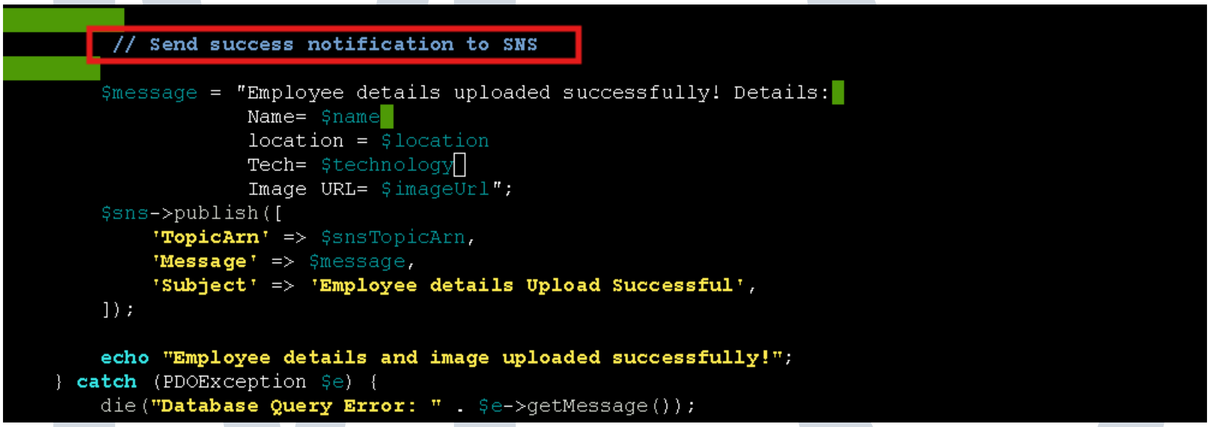
Modify the PHP Code:

In submit_employee.php file, after successfully uploading the image to S3 and storing the data in your MySQL database, you will send a notification to the SNS topic.

Update code with the following to send a success message to the SNS topic:



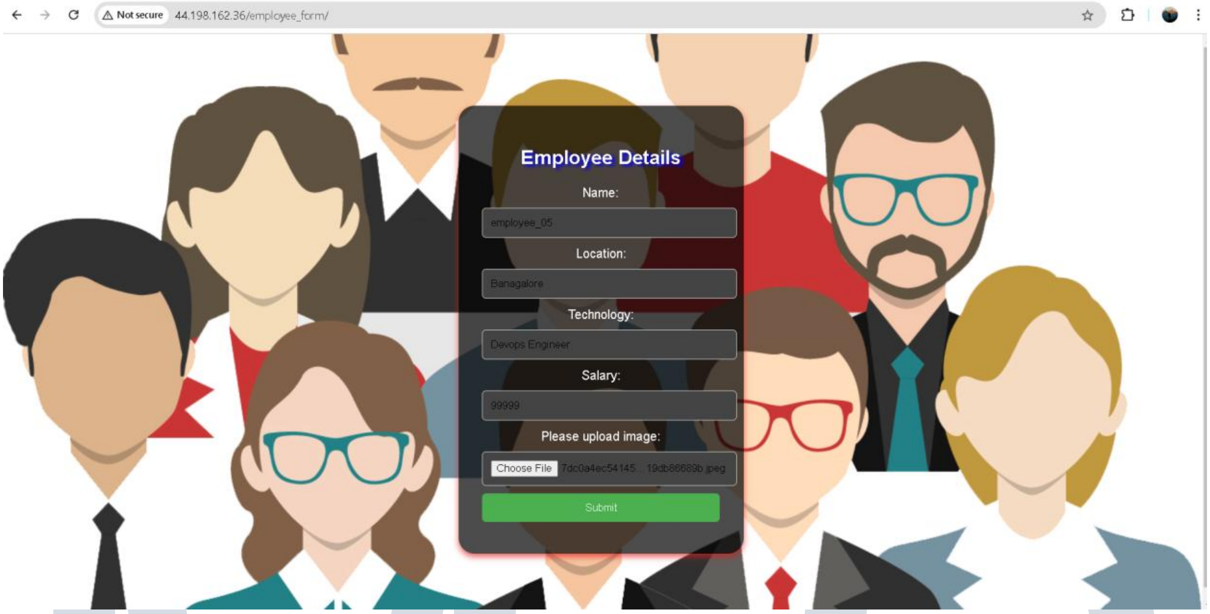
```
  GNU nano 7.2                                                                          submit_employee.php
<?php
require 'vendor/autoload.php';

use Aws\Sns\SnsClient;
use Aws\S3\S3Client;
use Aws\Exception\AwsException;

// AWS S3 Configuration
$bucketName = 'aws-s3-capstone';
$region = 'us-east-1'; // e.g., us-east-1
$s3 = new S3Client([
    'region' => $region,
    'version' => 'latest',
    'credentials' => [
        'key' => 'AKIA6GBMDLLPUKXZHWH4',
        'secret' => 'X83WAMnnPS+kVwaMWHfuUwPKqroo4qLh+4gisEiS',
    ],
]);

// AWS SNS Configuration
$sns = new SnsClient([
    'region' => $region,
    'version' => 'latest',
    'credentials' => [
        'key' => 'AKIA6GBMDLLPUKXZHWH4',
        'secret' => 'X83WAMnnPS+kVwaMWHfuUwPKqroo4qLh+4gisEiS',
    ],
]);
$snsTopicArn = 'arn:aws:sns:us-east-1:975050070751:S3-notification'; // Replace with your SNS topic ARN
```



```
        // Send success notification to SNS

        $message = "Employee details uploaded successfully! Details:
                    Name= $name
                    location = $location
                    Tech= $technology
                    Image URL= $imageUrl";
        $sns->publish([
            'TopicArn' => $snsTopicArn,
            'Message' => $message,
            'Subject' => 'Employee details Upload Successful',
        ]);

        echo "Employee details and image uploaded successfully!";
    } catch (PDOException $e) {
        die("Database Query Error: " . $e->getMessage());
```

Step 6: Test the Integration

1. **Upload an image** and employee details using form.
2. Once the upload is successful, check email inbox for the notification from SNS.
3. If the email is successfully sent, you should see a message containing the Employee details, image URL and success notification.



Employee Details

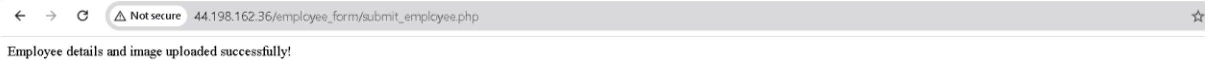
Name: employee_05

Location: Banagalore

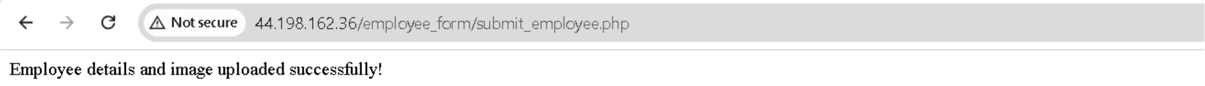
Technology: Devops Engineer

Salary: 99999

Please upload image: Choose File 7dc0a4ec54145...19db86689b.jpeg

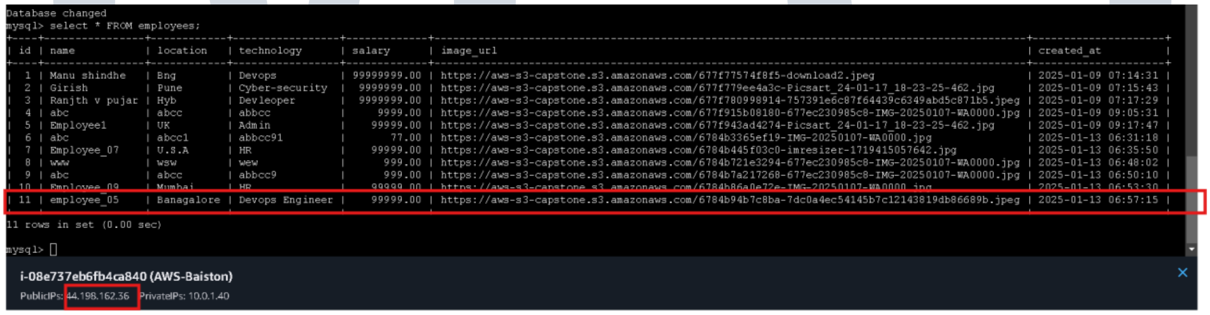


Employee details and image uploaded successfully!



Employee details and image uploaded successfully!

Now we will check in database whether details are stored.

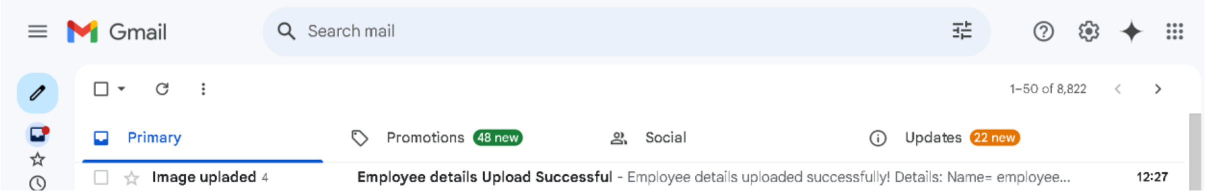


```
Database changed
mysql> select * FROM employees;
+----+----------------+-----------+-----------------+-------------+---------------------------------------------------------------------------------------------------------------+---------------------+
| id | name           | location  | technology      | salary      | image_url                                                                                                     | created_at          |
+----+----------------+-----------+-----------------+-------------+---------------------------------------------------------------------------------------------------------------+---------------------+
|  1 | Manu shindhe   | Bng       | Devops          | 99999999.00 | https://aws-s3-capstone.s3.amazonaws.com/677f77574f8f5-download2.jpeg                                         | 2025-01-09 07:14:31 |
|  2 | Girish         | Pune      | Cyber-security  |  9999999.00 | https://aws-s3-capstone.s3.amazonaws.com/677f779ee4a3c-Picsart_24-01-17_18-23-25-462.jpg                      | 2025-01-09 07:15:43 |
|  3 | Ranjth v pujar | Hyb       | Devleoper       |  9999999.00 | https://aws-s3-capstone.s3.amazonaws.com/677f780998914-757391e6c87f64439c6349abd5c871b5.jpeg                  | 2025-01-09 07:17:29 |
|  4 | abc            | abcc      | abbcc           |     9999.00 | https://aws-s3-capstone.s3.amazonaws.com/677f915b08180-677ec230985c8-IMG-20250107-WA0000.jpg                  | 2025-01-09 09:05:31 |
|  5 | Employee1      | UK        | Admin           |    99999.00 | https://aws-s3-capstone.s3.amazonaws.com/677f943ad4274-Picsart_24-01-17_18-23-25-462.jpg                      | 2025-01-09 09:17:47 |
|  6 | abc            | abcc1     | abbcc91         |       77.00 | https://aws-s3-capstone.s3.amazonaws.com/6784b3365ef19-IMG-20250107-WA0000.jpg                                | 2025-01-13 06:31:18 |
|  7 | Employee_07    | U.S.A     | HR              |    99999.00 | https://aws-s3-capstone.s3.amazonaws.com/6784b445f03c0-imresizer-1719415057642.jpg                            | 2025-01-13 06:35:50 |
|  8 | www            | wsw       | wew             |      999.00 | https://aws-s3-capstone.s3.amazonaws.com/6784b721e3294-677ec230985c8-IMG-20250107-WA0000.jpg                  | 2025-01-13 06:48:02 |
|  9 | abc            | abcc      | abbcc9          |      999.00 | https://aws-s3-capstone.s3.amazonaws.com/6784b7a217268-677ec230985c8-IMG-20250107-WA0000.jpg                  | 2025-01-13 06:50:10 |
| 10 | Employee_09    | Mumbai    | HR              |    99999.00 | https://aws-s3-capstone.s3.amazonaws.com/6784b86a0e72e-IMG-20250107-WA0000.jpg                                | 2025-01-13 06:53:30 |
| 11 | employee_05    | Banagalore | Devops Engineer |    99999.00 | https://aws-s3-capstone.s3.amazonaws.com/6784b94b7c8ba-7dc0a4ec54145b7c12143819db86689b.jpeg                  | 2025-01-13 06:57:15 |

11 rows in set (0.00 sec)

mysql> 
```

i-08e737eb6fb4ca840 (AWS-Baiston)

PublicIPs: 44.198.162.36 PrivateIPs: 10.0.1.40

Now we can verify email inbox for the notification from SNS.



☐ ☆ Image upladed 4 — **Employee details Upload Successful** - Employee details uploaded successfully! Details: Name= employee... 12:27

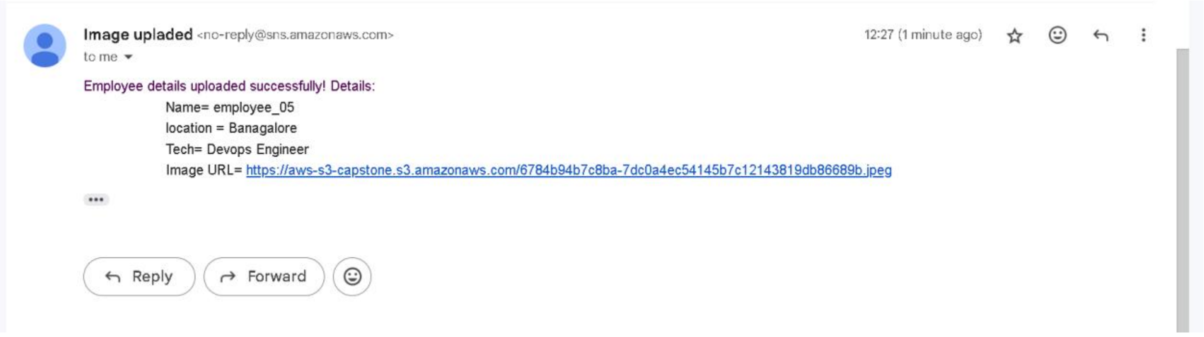


Image upladed <no-reply@sns.amazonaws.com>

to me

12:27 (1 minute ago)

Employee details uploaded successfully! Details:

Name= employee_05
location = Banagalore
Tech= Devops Engineer
Image URL= https://aws-s3-capstone.s3.amazonaws.com/6784b94b7c8ba-7dc0a4ec54145b7c12143819db86689b.jpeg

This is how we can verify our application - – *New employees input their information and upload photos. Existing employees can get their information*.
