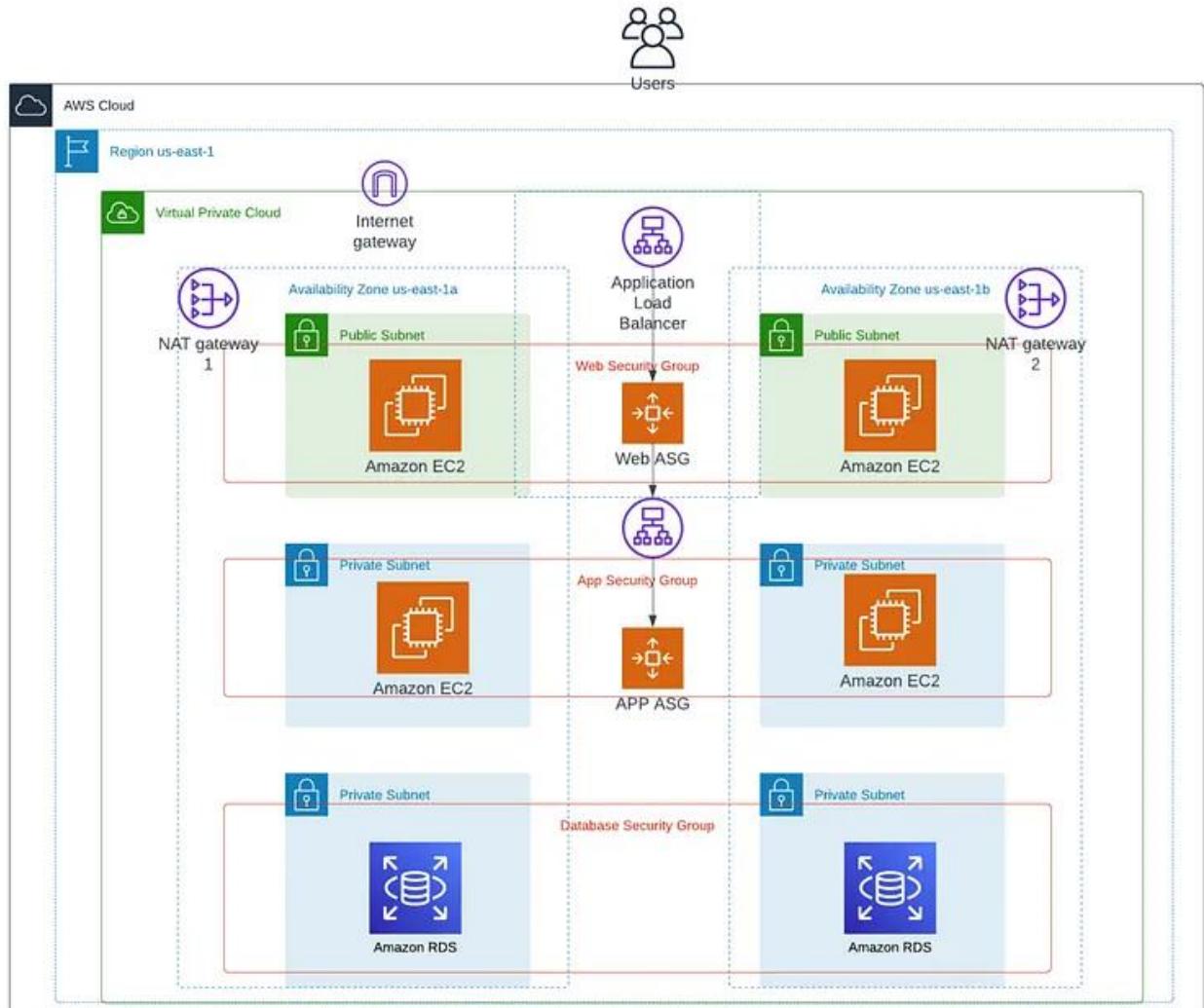


Creating a 3-Tier Architecture for Web Applications in AWS



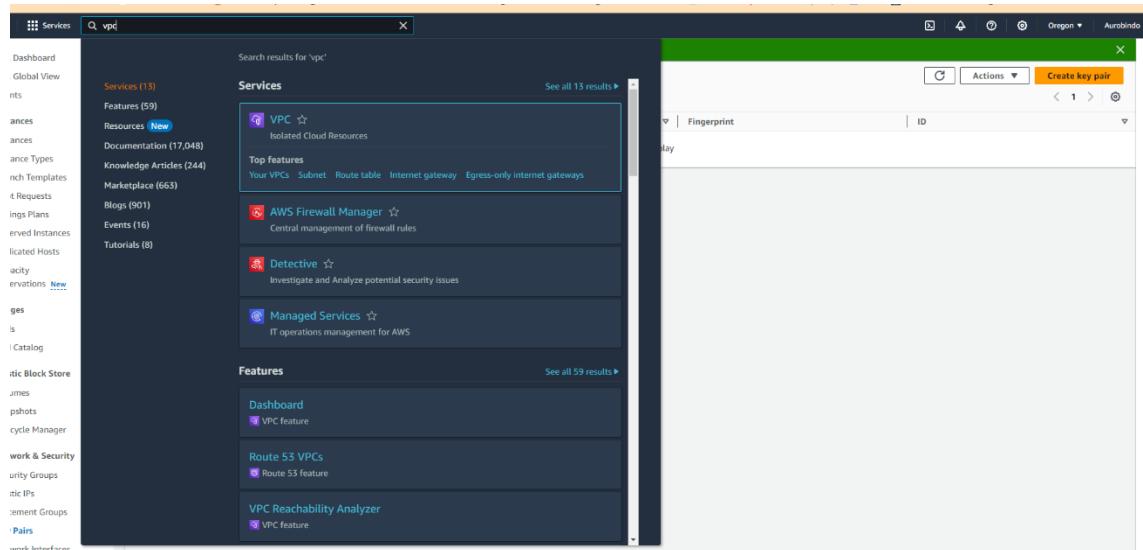
What is a 3-Tier Architecture?

A three-tier architecture comprises three layers, namely the presentation tier, the application tier, and the data tier. The presentation tier serves as the front-end, hosting the user interface, such as the website that users or clients interact with. The application tier, commonly referred to as the back-end, processes the data. Finally, the data tier is responsible for data storage and management.

Part 1: Creating a VPC and Subnets

Using the architecture diagram as a reference, we will need to start by creating a new VPC with 2 public subnets and 4 private subnets.

Log into the AWS management console and click the Create VPC button.



We will create a VPC with multiple public and private subnets, availability zones, and more, so let's choose "VPC and more."

Name your VPC. I am using the auto-assigned IPV4-CIDR block of "10.0.0.0/16." Choose these settings:

- no IPV6
- default Tenancy
- 2 Availability Zones
- 2 public subnets
- 4 private subnets
- Nat gateway chosen in 1 az
- None for vpc endpoints

Create VPC Info

A VPC is an isolated portion of the AWS Cloud populated by AWS objects, such as Amazon EC2 instances. Mouse over a resource to highlight the related resources.

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 auto-generation Info
Enter a value for the Name tag. This value will be used to auto-generate Name tags for all resources in the VPC.

Auto-generate

IPv4 CIDR block Info
Determine the starting IP and the size of your VPC using CIDR notation.

65,536 IPs
CIDR block size must be between /16 and /28.

IPv6 CIDR block Info
 No IPv6 CIDR block Amazon-provided IPv6 CIDR block

Tenancy Info

Number of Availability Zones (AZs) Info
Choose the number of AZs in which to provision subnets. We recommend at least two AZs for high availability.

1 2 3

Preview

VPC Show details
Your AWS virtual network

Subnets (4)
Subnets within this VPC

us-west-2a
project-A-subnet-public1-us-west-2a
project-A-subnet-private1-us-west-2a

us-west-2b
project-A-subnet-public2-us-west-2b
project-A-subnet-private2-us-west-2b

Route tables (3)
Route network traffic to resources

project-A-rtb-public
project-A-rtb-private1-us-west-2a
project-A-rtb-private2-us-west-2b

Network connection
Connections to other network

project-A-igw
project-A-vpc-e-s3

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences 27°C Light rain ENG IN 10:25 29-08-2024

Create VPC Info

No IPv6 CIDR block Amazon-provided IPv6 CIDR block

Tenancy Info

Number of Availability Zones (AZs) Info
Choose the number of AZs in which to provision subnets. We recommend at least two AZs for high availability.

1 2 3

Customize AZs

First availability zone

Second availability zone

Number of public subnets Info
The number of public subnets to add to your VPC. Use public subnets for web applications that need to be publicly accessible over the internet.

0 2 4

Number of private subnets Info
The number of private subnets to add to your VPC. Use private subnets to secure backend resources that don't need public access.

0 2 4

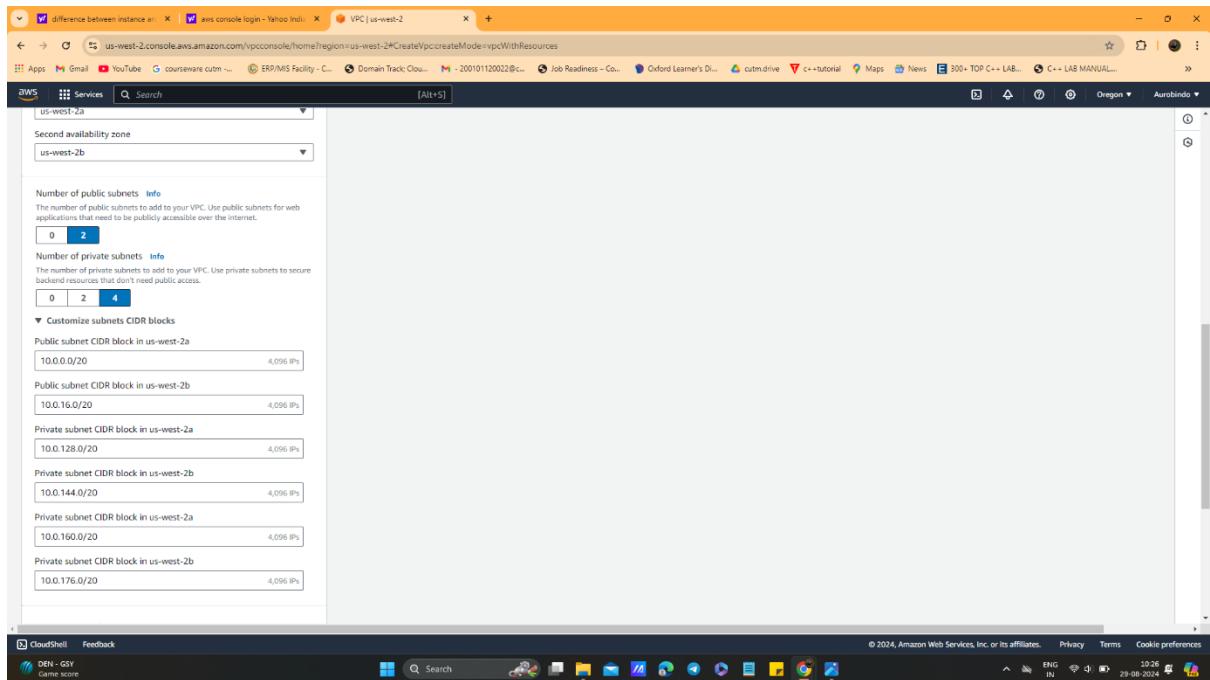
Customize subnets CIDR blocks

NAT gateways (5) Info
Choose the number of Availability Zones (AZs) in which to create NAT gateways.

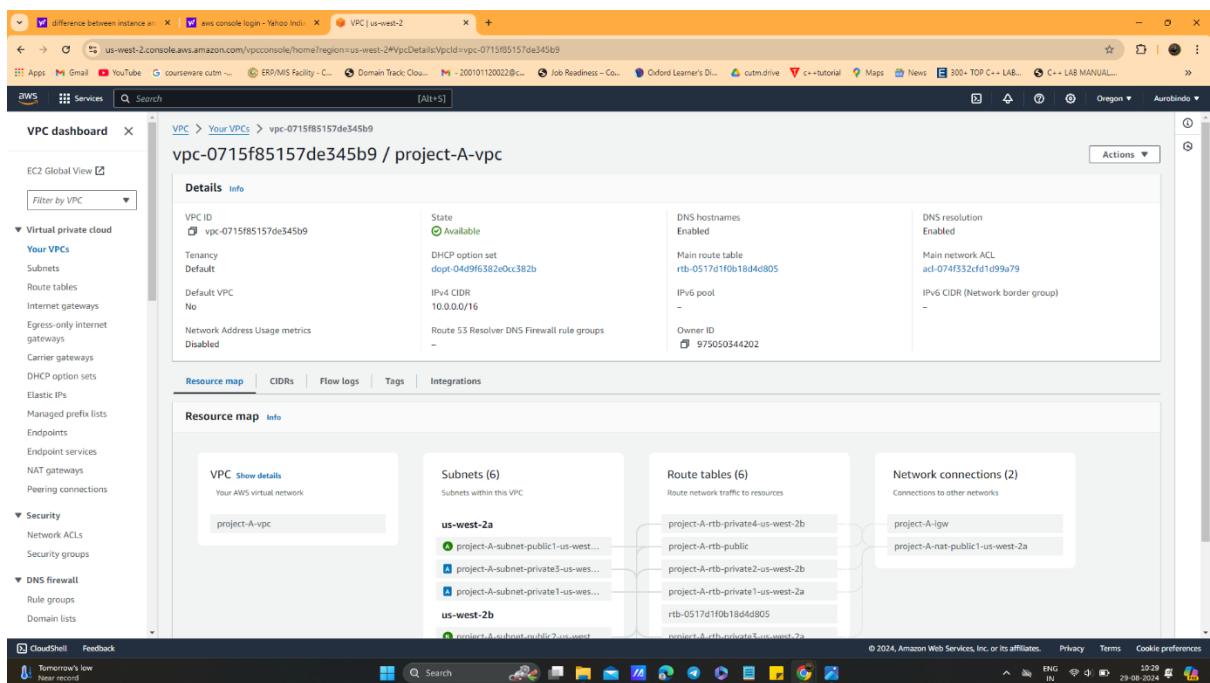
Note that there is a charge for each NAT gateway.

None In 1 AZ 1 per AZ

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences DEN - GSY Game score ENG IN 10:26 29-08-2024



View your new VPC after everything is created.



Next click on the Subnets tab in the VPC console. Select one of the new subnets that was created, then under the "Actions" tab, expand the down arrow and select "Edit subnet settings."

Screenshot of the AWS VPC Subnets page showing a list of subnets across multiple projects and regions.

Name	Subnet ID	VPC	IPv4 CIDR	IPv6 CIDR
-	subnet-0ef1ef51c0b67ca9a	vpc-0692e4587c641b352	172.31.0.0/20	-
project-A-subnet-public2-us-west-2b	subnet-087e3be7afddde5dd2	vpc-0715f85157de345b9 proj...	10.0.16.0/20	-
project-A-subnet-private3-us-west-2a	subnet-0b3559b05990b54b	vpc-0715f85157de345b9 proj...	10.0.160.0/20	-
project-A-subnet-public1-us-west-2a	subnet-0bd8384f319f6960f	vpc-0715f85157de345b9 proj...	10.0.0.0/20	-
-	subnet-0a48971711298ea52	vpc-0692e4587c641b352	172.31.48.0/20	-
-	subnet-075d10ae4c9e7b841	vpc-0692e4587c641b352	172.31.32.0/20	-
project-A-subnet-private4-us-west-2b	subnet-073a015fbab319d67	vpc-0715f85157de345b9 proj...	10.0.176.0/20	-
project-A-subnet-private2-us-west-2b	subnet-071ad0a3ea1157b9b	vpc-0715f85157de345b9 proj...	10.0.144.0/20	-
project-A-subnet-private1-us-west-2a	subnet-076e0d0a1cfcd9e2	vpc-0715f85157de345b9 proj...	10.0.128.0/20	-
-	subnet-097a21983a369d6de	vpc-0692e4587c641b352	172.31.16.0/20	-

Details for subnet-087e3be7afddde5dd2 / project-A-subnet-public2-us-west-2b:

- Subnet ID: subnet-087e3be7afddde5dd2
- Subnet ARN: arn:aws:ec2:us-west-2:2975650344202:subnet/subnet-087e3be7afddde5dd2
- Available IPv4 addresses: 4091
- IPv6 CIDR: -
- Availability Zone ID: usw2-az1
- Network border group: us-west-2
- VPC: vpc-0715f85157de345b9 | project-A-vpc
- Auto-assign public IPv4 address: Enabled

Screenshot of the AWS Subnet Settings page for subnet-087e3be7afddde5dd2.

Auto-assign IP settings

- Enable auto-assign public IPv4 address
- Enable auto-assign customer-owned IPv4 address

Resource-based name (RBN) settings

- Enable resource name DNS A record on launch
- Enable resource name DNS AAAA record on launch

DNS64 settings

- Enable DNS64

Part 2: Creating a Web Server Tier

Next we will create our first tier that represents our front end user interface (web interface).we will go to ec2 in that there is launch template click on that and create template.

The screenshot shows the AWS EC2 Launch Templates landing page. The left sidebar contains navigation links for EC2 Dashboard, Global View, Events, Instances, Launch Templates (which is selected), Images, Elastic Block Store, Network & Security, and more. The main content area features a section titled "EC2 launch templates" with the sub-section "Streamline, simplify and standardize instance launches". Below this, there's a brief description of using launch templates to automate instance launches and a "New launch template" button. A "Benefits and features" section lists "Streamline provisioning", "Simplify permissions", and "Governance". The bottom right corner shows the AWS footer with copyright information and links.

The screenshot shows the "Create launch template" wizard, Step 1: Summary. The left panel contains fields for "Launch template name and description" (name: "web-tier-template", version: "allow"), "Template version description" (description: "allow"), and "Auto Scaling guidance" (checkbox checked: "Provide guidance to help me set up a template that I can use with EC2 Auto Scaling"). The right panel displays a summary of the configuration, including the software image (AMI), virtual server type (instance type), firewall (security group), and storage (volumes). A tooltip for the "Free tier" provides details about instance usage. The bottom right corner shows the AWS footer.

The screenshot shows the AWS Cloud Console with the URL <https://us-west-2.console.aws.amazon.com/ec2/home?region=us-west-2#CreateLaunchTemplate>. The 'Create launch template | EC2' tab is active. In the main content area, the 'Application and OS Images (Amazon Machine Image) - required' section is selected. A tooltip for the 'Free tier' information is displayed, stating: 'Free tier: In your first year includes 750 hours of t2.micro (or t2.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 IOPS, 1 GB of snapshots, and 100 GB of bandwidth to the internet.' Below the tooltip, there are tabs for 'Recent' and 'Quick Start'. Under 'Recent', there are links for Amazon Linux, macOS, Ubuntu, Windows, Red Hat, and SUSE. Under 'Quick Start', there are links for Amazon Machine Image (AMI), Ubuntu Server 24.04 LTS (HVM), SSD Volume Type, and a note that it's 'Free tier eligible'. The 'Architecture' dropdown is set to '64-bit (x86)'. The 'AMI ID' is listed as 'ami-05134c8ef96964280'. The 'Create launch template' button is highlighted in orange at the bottom right.

Here I am creating security group for web tier

The screenshot shows the AWS Cloud Console with the URL <https://us-west-2.console.aws.amazon.com/vpcconsole/home?region=us-west-2#CreateSecurityGroup>. The 'VPC | us-west-2' tab is active. In the main content area, the 'Create security group' page is displayed. The 'Basic details' section is filled with the following values: 'Security group name' is 'websecuritygroup', 'Description' is 'allow', and 'VPC info' is 'vpc-0715fb5157de345b9 (project-A-vpc)'. Below this, the 'Inbound rules' section shows a note that 'This security group has no inbound rules.' and a 'Add rule' button. The 'Outbound rules' section shows a table with columns: Type, Info, Protocol, Info, Port range, Info, Destination, Info, and Description - optional. The table currently has one row: 'Type' is 'CloudShell', 'Protocol' is 'Feedback', 'Port range' is 'Info', 'Destination' is 'Info', and 'Description - optional' is 'Info'. At the bottom of the page, there is a note: '© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences ENG IN 10:58 29-08-2024 Oregon AutoBindo'.

Inbound rules: add rules for ssh, HTTP, custom ip: source -anywhere

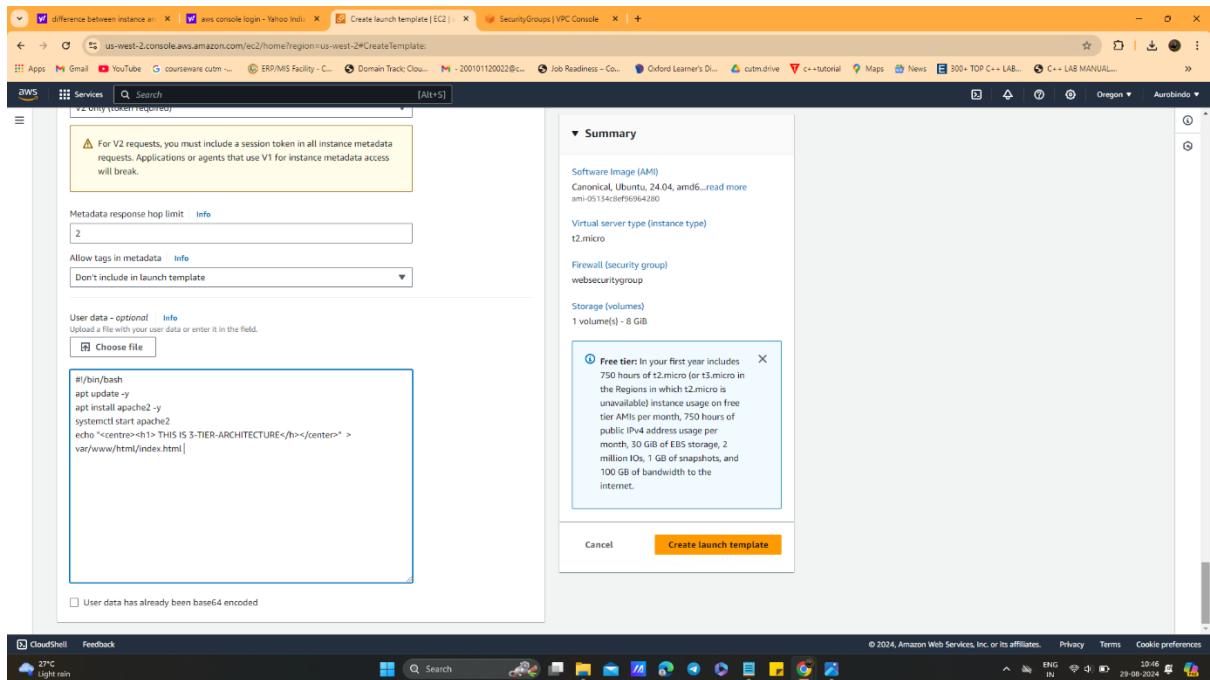
The screenshot shows the AWS VPC Security Groups console. A new security group named 'websecuritygroup' is being created. In the 'Inbound rules' section, there are three rules: one for Custom ICMP - IPv4 (allowing all ports from anywhere), one for HTTP (TCP port 80 from anywhere), and one for SSH (TCP port 22 from anywhere). In the 'Outbound rules' section, there is a single rule allowing all traffic. A note at the bottom states: '⚠ Rules with source of 0.0.0.0/0 or ::/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.'

The screenshot shows the AWS Create launch template | EC2 console. A new launch template is being created. It includes the following configurations:

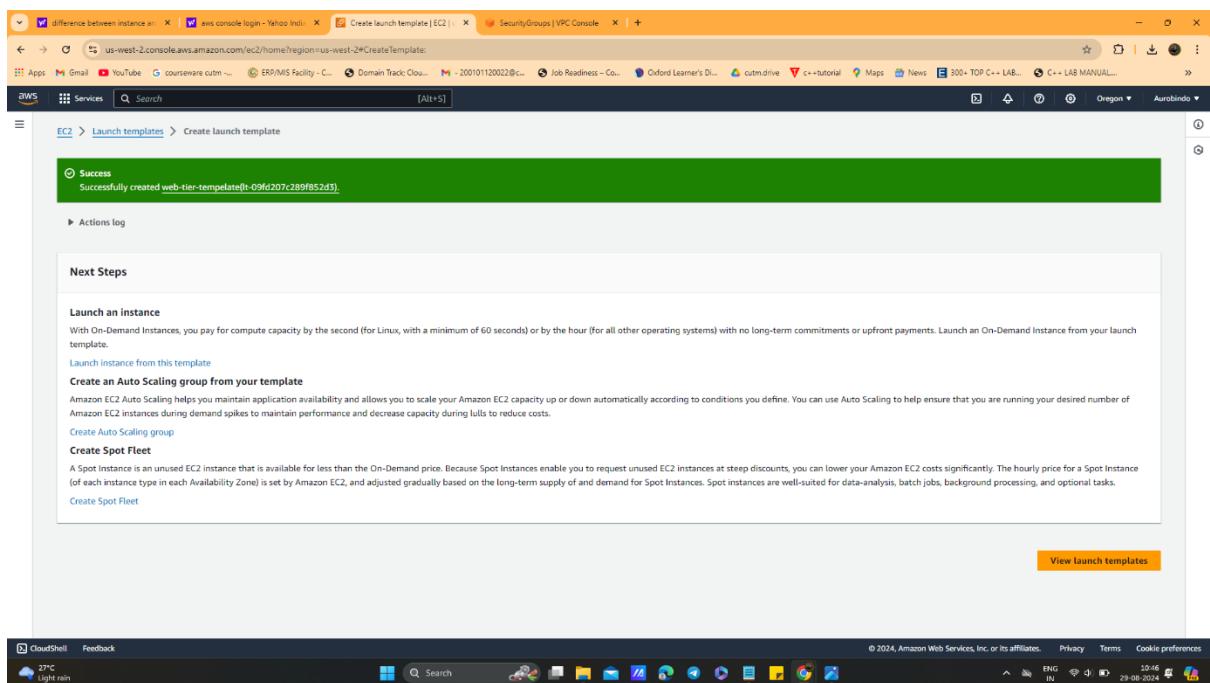
- Key pair name:** projectA
- Network settings:** Subnet: Don't include in launch template; Firewall (security groups): Select existing security group (websecuritygroup sg-0f218dbcc2eb42667).
- Storage (volumes):** EBS Volumes: Hide details.

A callout box highlights the 'Free tier' information: '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 IOPS, 1 GB of snapshots, and 100 GB of bandwidth to the internet.'

And at last in additional information give userdata



And click on create launch tempelate



For this project to work we need to create an auto scaling group that we attach to our EC2 instance. This will increase our reliability and availability.

avigate to the autoscaling tab at the bottom of the EC2 dashboard. Click “Create auto scaling group.” The launch template that we just finished creating is the template that our auto scaling group will use to launch new EC2 instances when scaling up.

Name your auto scaling group (ASG), choose the launch template that you created, then click the Next button

us-west-2.console.aws.amazon.com/ec2/home?region=us-west-2#AutoScalingGroups

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.

How it works

An Auto Scaling group is a collection of Amazon EC2 instances that are treated as a logical unit. You configure settings for a group and its instances as well as define the group's minimum, maximum, and desired capacity.

Pricing

Amazon EC2 Auto Scaling features have no additional fees beyond the service fees for Amazon EC2, CloudWatch (for scaling policies), and the other AWS resources that you use. Visit the pricing page of each service to learn more.

Getting started

What is Amazon EC2 Auto Scaling?
Getting started with Amazon EC2 Auto Scaling
Set up a scaled and load-balanced application
FAQ

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences ENG IN 10:47 29-08-2024

us-west-2.console.aws.amazon.com/ec2/home?region=us-west-2#CreateAutoScalingGroup

Step 1 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
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
Choose a launch template that contains the instance-level settings, such as the Amazon Machine Image (AMI), instance type, key pair, and security groups.

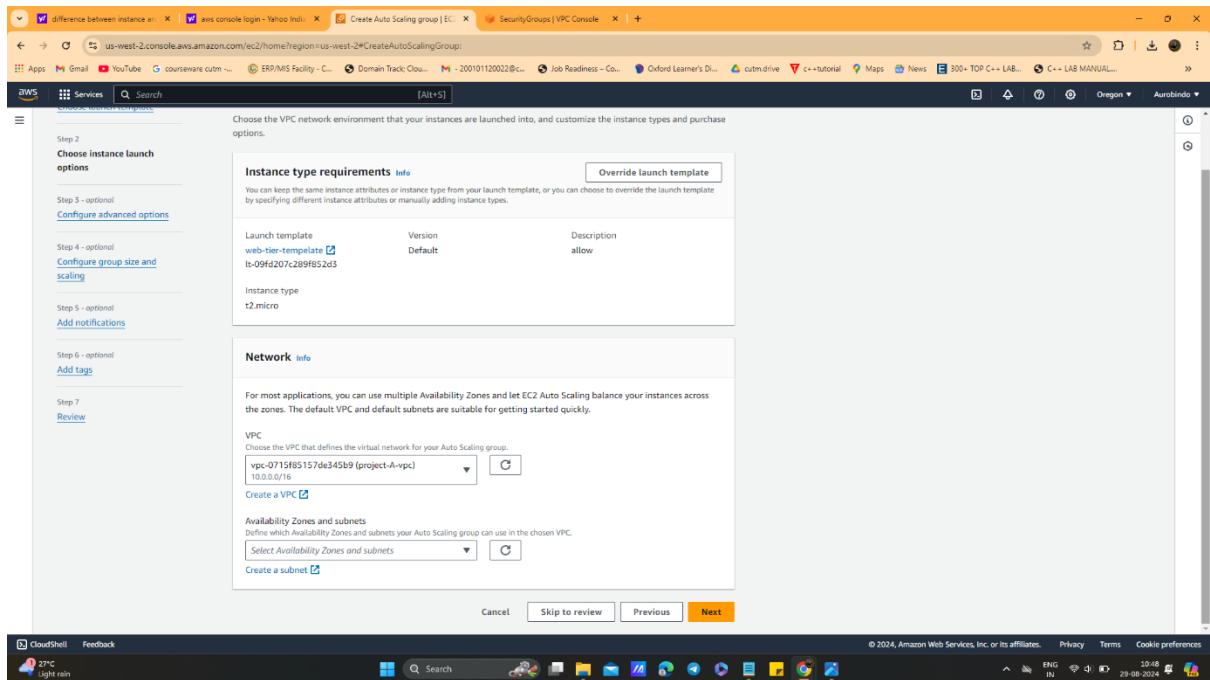
Version

Description
allow

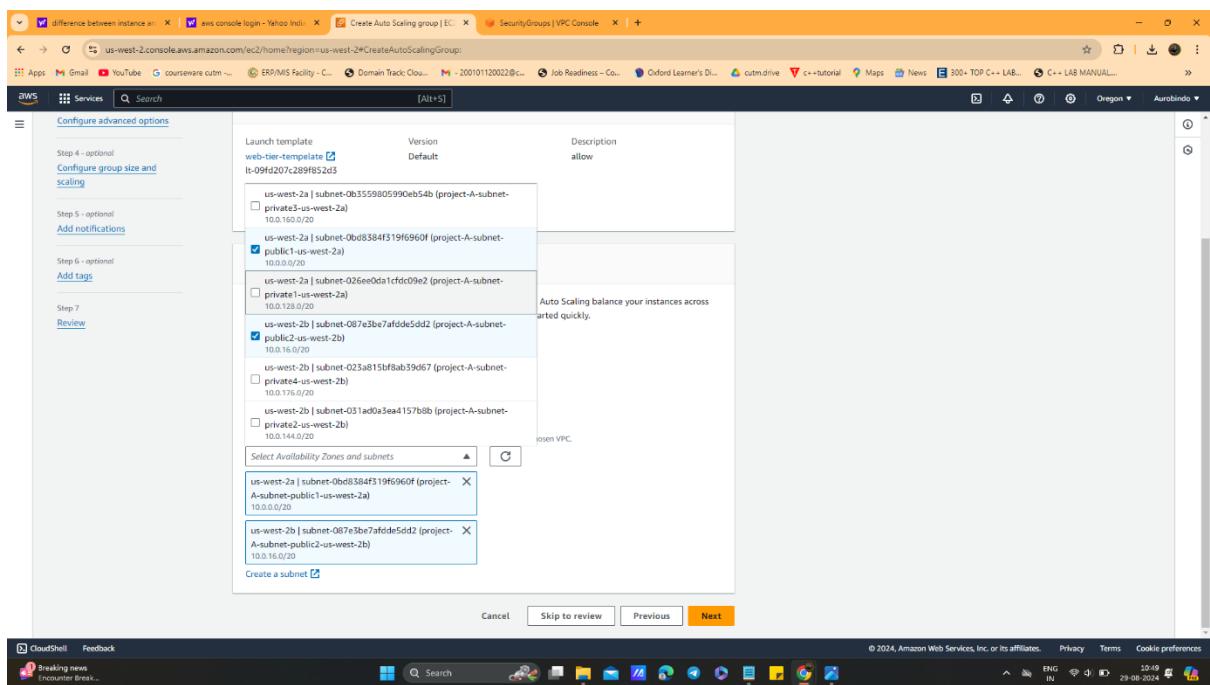
Launch template
web-tier-template

Instance type
t2.micro

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences ENG IN 10:48 29-08-2024

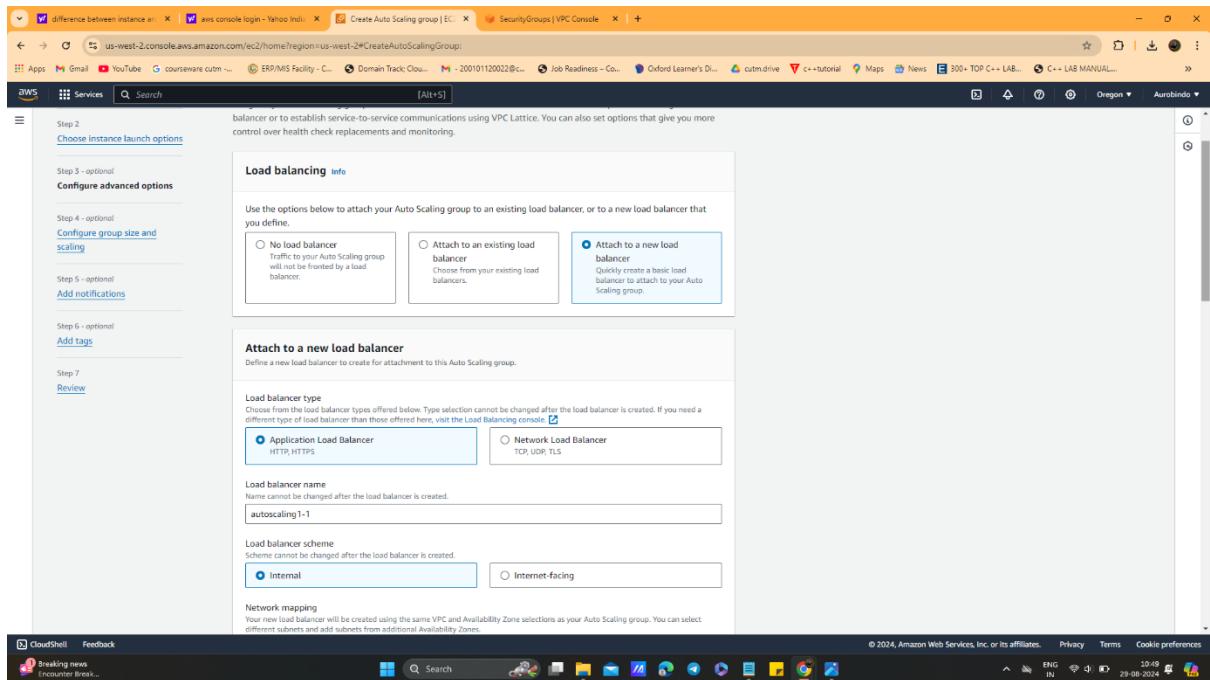


Select the public subnet which we created in first while creating vpc in place of select az and subnets

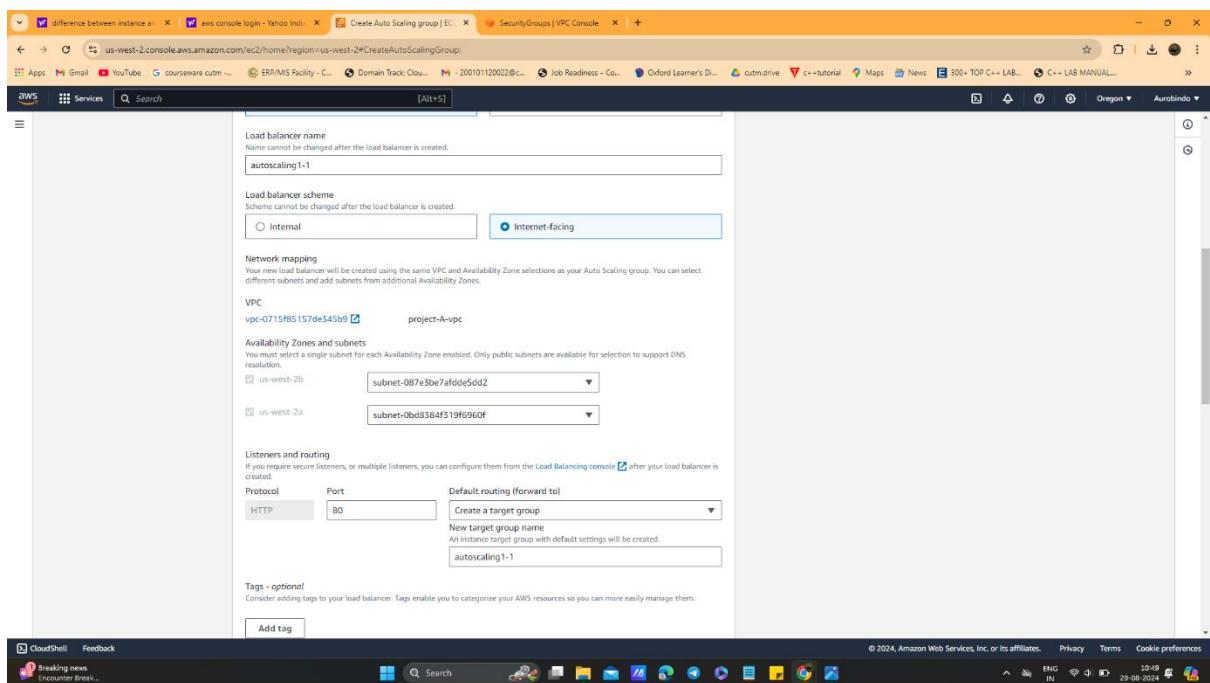


Now we are given to option to allocate a load balancer for our ASG. A load balancer will distribute the load from incoming traffic across multiple servers. This helps with availability and performance.

Select “Attach to a new load balancer” and “Application load balancer,” name your load balancer, then select “Internet facing” as this is for our web tier.



Click on internet-facing scheme:



The screenshot shows the 'Create Auto Scaling group' wizard on the AWS Management Console. The current step is 'Configure Auto Scaling group details'. A section titled 'VPC Lattice integration options' is expanded, showing two options: 'No VPC Lattice service' (selected) and 'Attach to VPC Lattice service'. Below this, the 'Health checks' section is shown, containing options for EC2 health checks (selected), ELB health checks (recommended), and VPC Lattice health checks. The 'Additional settings' section includes monitoring (CloudWatch Metrics collection enabled) and default instance warmup (disabled). Navigation buttons at the bottom include 'Cancel', 'Skip to review', 'Previous', and 'Next'.

Here, we have to configure group size and scaling.you can select your own desired capacity according to your project requirements

The screenshot shows the AWS Auto Scaling group creation wizard at Step 4: Configure group size and scaling - optional. The page title is "Configure group size and scaling - optional". It includes sections for "Desired capacity" (set to 3), "Scaling limits" (min desired capacity 1, max desired capacity 3), and "Automatic scaling - optional" (target tracking scaling policy selected). Other options like "No scaling policies" and "Scale in only" are also shown.

Configure group size and scaling - optional

Desired capacity

Scaling limits

Automatic scaling - optional

Target tracking scaling policy

No scaling policies

Scale in only

And click to create:

The screenshot shows the AWS EC2 Auto Scaling Groups console. At the top, there are tabs for 'Auto Scaling groups' and 'SecurityGroups | VPC Console'. Below the tabs, a search bar and a toolbar with buttons for 'Launch configurations', 'Launch templates', 'Actions', and 'Create Auto Scaling group' are visible. A table lists one Auto Scaling group: 'autoscaling1'. The table columns include Name, Launch template/configuration, Instances, Status, Desired capacity, Min, Max, and Availability Zones. The 'Availability Zones' column shows 'us-west-2a, us-west-2b'. The status of the instance is '3'. The 'Launch template/configuration' column shows 'web-tier-template | Version Default'. The 'Instances' column shows '3'. The 'Status' column shows '-'. The 'Desired capacity' column shows '3'. The 'Min' column shows '1'. The 'Max' column shows '3'. The 'Availability Zones' column shows 'us-west-2a, us-west-2b'. The table has a header row with filters for Name, Launch template/configuration, Instances, Status, Desired capacity, Min, Max, and Availability Zones.

After creating autoscaling go to ec2 instance and click on instance, copy the public ip, and check whether server working or not .if working proceed

The screenshot shows a web browser window with the URL '54.187.29.91'. The page content is not fully visible but appears to be a simple response from the server. The browser's address bar shows 'Not secure' and the IP address '54.187.29.91'. The browser interface includes a search bar, a toolbar with various icons, and a taskbar at the bottom with icons for File, Edit, View, Insert, Format, Tools, Help, and a system tray with battery, signal, and date/time information.

Part 3: Creating an Application Tier

Next up we are going to create the back-end of our 3-tier architecture

I will start by heading to the Launch Templates tab under the EC2 dashboard.

Here I am creating another security group 2 for app-tier

In inbound rules I have taken custom tcp, ssh, all icmp-ipv4

For Security Group 2, use Custom TCP; the source here is the security group from our web tier (tier-1). For the third group, select All ICMP-IPv4, and set the Source type as anywhere. This will allow us to ping our application tier from the public internet to test if traffic is routed properly.

Screenshot of the AWS VPC console showing the configuration of a security group (sg-0f218dbcc2eb42667) for a specific VPC.

Inbound rules

Type	Protocol	Port range	Source	Description - optional
Custom TCP	TCP	0	Custom	
SSH	TCP	22	Anywhere (0.0.0.0/0)	
All ICMP - IPv4	ICMP	All	Anywhere (0.0.0.0/0)	

Outbound rules

Type	Protocol	Port range	Destination	Description - optional
All traffic	All	All	Custom	

Security group sg-073c01367874e254e - appsecuritygroup was created successfully

Details

Security group name	Security group ID	Description	VPC ID
appsecuritygroup	sg-073c01367874e254e	allow	vpc-0715f85157de345b9
Owner	975050344202	Inbound rules count 3 Permission entries	Outbound rules count 1 Permission entry

Inbound rules (3)

Name	Security group rule ID	IP version	Type	Protocol	Port range	Source	Description
-	sgr-003b29c018d867...	-	Custom TCP	TCP	0	sg-0f218dbcc2eb42667	-
-	sgr-07febfd4df97ce61	IPv4	SSH	TCP	22	0.0.0.0/0	-
-	sgr-0cc39ec2850ad74c4	IPv4	All ICMP - IPv4	ICMP	All	0.0.0.0/0	-

Screenshot of the AWS Cloud Console showing the creation of a launch template for an EC2 instance.

Subnet: Info
 Don't include in launch template
 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
Security groups info
 Select security groups
 appsecuritygroup sg-075c01367874e254e X
 VPC vpc-0715fb5157de54080
 Advanced network configuration

Storage (volumes): Info
EBS Volumes
 Hide details
 Volume 1 (AMI Root) (8 GiB, EBS, General purpose SSD (gp3))
 AMI Volumes are not included in the template unless modified
 Free tier eligible customers can get up to 30 GB of EBS General Purpose (SSD) or Magnetic storage
 Add new volume
 The selected AMI contains more instance store volumes than the instance allows. Only the first 0 instance store

Summary
Software Image (AMI)
 Canonical, Ubuntu, 24.04, amd64... read more
 ami-05134c8ef96964280
Virtual server type (instance type)
 t2.micro
Firewall security group
 appsecuritygroup
Storage (volumes)
 1 volume(s) - 8 GiB

Free tier: In your first year includes
 750 hours of t2.micro (or t1.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 IOPS, 1 GB of snapshots, and 100 GB of bandwidth to the internet.

Create launch template

Success
 Successfully created apptiertemplate(t-0088febf3225a185).

Actions log

Next Steps

Launch an instance
 With On-Demand Instances, you pay for compute capacity by the second (for Linux, with a minimum of 60 seconds) or by the hour (for all other operating systems) with no long-term commitments or upfront payments. Launch an On-Demand Instance from your launch template.
[Launch instance from this template](#)

Create an Auto Scaling group from your template
 Amazon EC2 Auto Scaling helps you maintain application availability and allows you to scale your Amazon EC2 capacity up or down automatically according to conditions you define. You can use Auto Scaling to help ensure that you are running your desired number of Amazon EC2 instances during demand spikes to maintain performance and decrease capacity during lulls to reduce costs.
[Create Auto Scaling group](#)

Create Spot Fleet
 A Spot Instance is an unused EC2 instance that is available for less than the On-Demand price. Because Spot Instances enable you to request unused EC2 instances at steep discounts, you can lower your Amazon EC2 costs significantly. The hourly price for a Spot Instance (of each instance type in each Availability Zone) is set by Amazon EC2, and adjusted gradually based on the long-term supply of and demand for Spot Instances. Spot instances are well-suited for data-analysis, batch jobs, background processing, and optional tasks.
[Create Spot Fleet](#)

View Launch Templates

Go to autoscaling:

Screenshot of the AWS Cloud Console showing the "Create Auto Scaling group" wizard - Step 1: Choose launch template.

The "Name" field contains "autoscaling".

The "Launch template" dropdown shows "apptiertemplate" selected.

The "Version" dropdown shows "Default (1)".

The "Description" field contains "allow".

The "Launch template" field shows "apptiertemplate lt-0088febf5225a185".

The "Instance type" field shows "t2.micro".

Screenshot of the AWS Cloud Console showing the "Create Auto Scaling group" wizard - Step 4: Configure group size and scaling.

The "Launch template" dropdown shows "apptiertemplate lt-0088febf5225a185".

The "Version" dropdown shows "Default (1)".

The "Description" field contains "allow".

The "Instance type" field shows "t2.micro".

Screenshot of the AWS Cloud Console showing the "Create Auto Scaling group" wizard - Step 7: Review.

The "Network" section shows the VPC "vpc-0715f85157de345b9 (project-A-vpc)" selected.

The "Availability Zones and subnets" section shows two subnets selected: "us-west-2a | subnet-0b3559805990eb54b (project-A-subnet-private5-us-west-2a) 10.3.160.0/20" and "us-west-2b | subnet-031ad0a3ea4157b8b (project-A-subnet-private2-us-west-2b) 10.0.144.0/20".

Buttons at the bottom include "Cancel", "Skip to review", "Previous", and "Next".

difference between instance a... ✓ [aws console login - Yahoo India](#) ✓ Create Auto Scaling group Info ● VPC us-west-2 ● 34.187.29.91 +

us-west-2.console.aws.amazon.com/ec2/home?region=us-west-2#CreateAutoScalingGroup:

Apps Gmail YouTube G coursework cutm ~ ERP MIS Facility - C... Domain Track Clo... 2001-12-20022@C... Job Readiness - Co... Oxford Learner's Di... cutn.drive C+++Tutorial Maps News 300+ TOP C++ LAB... C++ LAB MANUAL... Oregon AuroraBind... ...

EC2 > Auto Scaling groups > Create Auto Scaling group

Step 1
Choose launch template

Step 2
Choose instance launch options

Step 3 - optional
Configure advanced options

Step 4 - optional
Configure group size and scaling

Step 5 - optional
Add notifications

Step 6 - optional
Add tags

Step 7
Review

Configure group size and scaling - optional Info

Define your group's desired capacity and scaling limits. You can optionally add automatic scaling to adjust the size of your group.

Group size Info

Set the initial size of the Auto Scaling group. After creating the group, you can change its size to meet demand, either manually or by using automatic scaling.

Desired capacity type
Choose the unit of measurement for the desired capacity value. vCPUs and Memory(GiB) are only supported for mixed instances groups configured with a set of instance attributes.

Units (number of instances) ▼

Desired capacity
Specify your group size:
3 ▼

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 Max desired capacity
1 5
Equal or less than desired capacity 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.

The screenshot displays the AWS CloudShell interface with two browser tabs open. The top tab is titled 'Create Auto Scaling group | EC2 | us-west-2' and shows the configuration for a new Auto Scaling group. It includes sections for 'Automatic scaling - optional', 'Scaling policy name' (set to 'Target Tracking Policy'), 'Metric type' (set to 'Average CPU utilization'), 'Target value' (set to 50), 'Instance warmup' (set to 300 seconds), and 'Instance maintenance policy' (with 'No policy' selected). The bottom tab is titled 'Auto Scaling groups | EC2 | us-west-2' and lists two existing Auto Scaling groups: 'autoscaling2' and 'autoscaling1', both of which are currently active ('In Service').

Part 4: Created a Database Tier

Almost there! We have created 2 out of the 3 tiers and tested both successfully.

We are now going to build our database; AWS offers several types of databases but for this exercise we are going to use a MySQL RDS database.

Create a DB Subnet Group

We will begin by creating a subnet groups. Navigate to the RDS console and on the left side menu, click "Subnet Groups"

Screenshot of the AWS RDS Dashboard showing the introduction of Aurora I/O-Optimized. The dashboard includes sections for Resources, Recommended services, and Recommended for you.

Resources

- You are using the following Amazon RDS resources in the US West (Oregon) region (used/quota)
- DB Instances (0/40): Allocated storage (0 TB/100 TB), Instances and storage include Neptune and DocumentDB. Increase DB instances limit.
- DB Clusters (0/40): Reserved instances (0/40), Snapshots (0): Manual, DB Cluster (0/50), DB Instance (0/50), Automated, DB Cluster (0), DB Instance (0).
- Recent events (0): Event subscriptions (0/20)

Recommended services

- OpsWorks: Configuration Management with Chef and Puppet
- IAM: Manage access to AWS resources
- Systems Manager: AWS Systems Manager is a Central Place to View and Manage AWS Resources
- Amazon Prometheus: A fully managed Prometheus-compatible monitoring service.
- CloudFormation: Create and Manage Resources with Templates

Recommended for you

- Implementing Cross-Region DR: Learn how to set up Cross-Region disaster recovery (DR) for Aurora PostgreSQL, using an Aurora global database spanning multiple Regions. [Learn more](#)
- Migrate SSRS to RDS for SQL Server: Learn how you can migrate existing SSRS content to an Amazon RDS for SQL Server instance using a PowerShell module. [Learn more](#)

Create database

Amazon Relational Database Service (RDS) makes it easy to set up, operate, and scale a relational database in the cloud.

[Restore from S3](#) [Create database](#)

Note: your DB instances will launch in the US West (Oregon) region

Screenshot of the AWS RDS Subnet Groups creation page.

Create DB subnet group

To create a new subnet group, give it a name and a description, and choose an existing VPC. You will then be able to add subnets related to that VPC.

Subnet group details

Name: You won't be able to modify the name after your subnet group has been created.
Description:
VPC: Choose a VPC identifier that corresponds to the subnets you want to use for your DB subnet group. You won't be able to choose a different VPC identifier after your subnet group has been created.

Add subnets

Availability Zones: Choose the Availability Zones that include the subnets you want to add.

Subnets: Choose the subnets that you want to add. The list includes the subnets in the selected Availability Zones.

The screenshot shows the AWS RDS console interface. In the top navigation bar, the URL is `us-west-2.console.aws.amazon.com/rds/home?region=us-west-2#db-subnet-group`. The main content area is titled "Create DB subnet group". A dropdown menu under "VPC" shows "project-A-vpc (vpc-0715f85157de345b9)". The "Add subnets" section lists two availability zones: "us-west-2a" and "us-west-2b". Under "Subnets", two subnets are selected: "subnet-023a815bf8ab39d67 (10.0.176.0/20)" and "subnet-026eed0da1cfcd09e2 (10.0.128.0/20)". A note at the bottom states: "For Multi-AZ DB clusters, you must select 3 subnets in 3 different Availability Zones." The "Subnets selected (2)" table shows the details of the selected subnets. The table has columns: Availability zone, Subnet ID, and CIDR block. The data is as follows:

Availability zone	Subnet ID	CIDR block
us-west-2b	subnet-023a815bf8ab39d67	10.0.176.0/20
us-west-2a	subnet-026eed0da1cfcd09e2	10.0.128.0/20

At the bottom right of the page, there is a green success message: "Successfully created subnet-datiier. View subnet group".

Back in the RDS console click Create database.

Select the MySQL DB

Screenshot of the AWS RDS console showing the "Create database" wizard. The "Choose a database creation method" step is selected, with "Standard create" chosen. The "Engine options" section shows various engine types: Aurora (MySQL Compatible), Aurora (PostgreSQL Compatible), MySQL (selected), MariaDB, PostgreSQL, and Oracle. A sidebar on the right provides information about MySQL.

Choose a database creation method

Standard create You set all of the configuration options, including ones for availability, security, backups, and maintenance.

Easy create Use recommended best-practice configurations. Some configuration options can be changed after the database is created.

Engine options

Engine type: MySQL

Aurora (MySQL Compatible) Aurora (PostgreSQL Compatible)

MySQL MariaDB

PostgreSQL Oracle

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences 29°C Light rain 11:33 IN 29-08-2024 Oregon Aurebindo

Screenshot of the AWS RDS console showing the "Create database" wizard. The "Engine options" step is selected, showing MySQL Community as the edition and MySQL 8.0.35 as the engine version. The "Show versions that support the Multi-AZ DB cluster" filter is selected. The "Enable RDS Extended Support" checkbox is unchecked. The "Templates" section shows Production, Dev/Test, and Free tier options. A sidebar on the right provides information about MySQL.

Engine options

Edition: MySQL Community

Engine version: MySQL 8.0.35

Show versions that support the Multi-AZ DB cluster

Show versions that support the Amazon RDS Optimized Writes

Enable RDS Extended Support

Templates

Production Use defaults for high availability

Dev/Test This instance is intended for development and testing environments.

Free tier Use RDS Free Tier to develop new applications, test existing ones, and prototype.

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences 11:33 IN 29-08-2024 Oregon Aurebindo

Screenshot of the AWS RDS console showing the 'Launch DB instance' wizard for MySQL 8.0.35.

Step 1: Engine Version

Selected: MySQL 8.0.35

Enable RDS Extended Support Info

Amazon RDS Extended Support is a paid offering. By selecting this option, you consent to being charged for this offering if you are running your database major version past the EBS end of standard support date for that version. Check the end of standard support date for your major version in the RDS for MySQL documentation.

Step 2: Templates

Choose a sample template to meet your use case.

- Production: Use defaults for high availability and fast, consistent performance.
- Dev/Test: This template is intended for development use outside of a production environment.
- Free tier: Use RDS Free Tier to develop new applications, test existing applications, or gain hands-on experience with Amazon RDS.

Step 3: Availability and durability

Deployment options Info

The deployment options below are limited to those supported by the engine you selected above.

- Multi-AZ DB Cluster: Creates a DB cluster with a primary DB instance and two readable standby DB instances, with each DB instance in a different Availability Zone (AZ). Provides high availability, data redundancy, and increases capacity to serve read workloads.
- Multi-AZ DB instance (not supported for Multi-AZ DB cluster snapshot): Creates a primary DB instance and a standby DB instance in a different AZ. Provides high availability and data redundancy, but the standby DB instance doesn't support connections for read workloads.
- Single DB instance (not supported for Multi-AZ DB cluster snapshot): Creates a single DB instance with no standby DB instances.

CloudShell Feedback 29°C Light rain Search Privacy Terms Cookie preferences © 2024, Amazon Web Services, Inc. or its affiliates. ENG IN 11:33 29-08-2024 Oregon Aurebindo

Step 4: Settings

DB instance identifier Info

Type a name for your DB instance. The name must be unique across all DB instances owned by your AWS account in the current AWS Region.

database-1

The DB instance identifier is case-insensitive, but is stored as all lowercase (as in "mydbinstance"). Constraints: 1 to 60 alphanumeric characters or hyphens. First character must be a letter. Can't contain two consecutive hyphens. Can't end with a hyphen.

Credentials Settings

Master username Info

Type a login ID for the master user of your DB instance.

admin

1 to 16 alphanumeric characters. The first character must be a letter.

Credentials management

You can use AWS Secrets Manager or manage your master user credentials.

- Managed in AWS Secrets Manager - most secure: RDS generates a password for you and manages it throughout its lifecycle using AWS Secrets Manager.
- Self managed: Create your own password or have RDS create a password that you manage.

Auto generate password

Amazon RDS can generate a password for you, or you can specify your own password.

Master password Info

Password strength Strong

Minimum constraints: At least 8 printable ASCII characters. Can't contain any of the following symbols: / * @

Confirm master password Info

Instance configuration

The DB instance configuration options below are limited to those supported by the engine that you selected above.

CloudShell Feedback 29°C Light rain Search Privacy Terms Cookie preferences © 2024, Amazon Web Services, Inc. or its affiliates. ENG IN 11:33 29-08-2024 Oregon Aurebindo

Screenshot of the AWS RDS console showing the configuration of a MySQL database instance.

Storage

- Storage type:** General Purpose SSD (gp2)
- Allocated storage:** 20 GiB
- Note:** After you modify the storage for a DB instance, the status of the DB instance will be in storage-optimization. Your instance will remain available as the storage-optimization operation completes. [Learn more](#)

Storage autoscaling

Storage autoscaling info: Provides dynamic scaling support for your database's storage based on your application's needs.

Enable storage autoscaling: Enabling this feature will allow the storage to increase after the specified threshold is exceeded.

Maximum storage threshold: 1000 GiB

Connectivity

Compute resource: Choose whether to set up a connection to a compute resource for this database. Setting up a connection will automatically change connectivity settings so that the compute resource can connect to this database.

- Don't connect to an EC2 compute resource:** Don't set up a connection to a compute resource for this database. You can manually set up a connection to a compute resource later.
- Connect to an EC2 compute resource:** Set up a connection to an EC2 compute resource for this database.

Certificate authority - optional

Using a server certificate provides an extra layer of security by validating that the connection is being made to an Amazon database. It does so by checking the server certificate that is automatically installed on all databases that you provision.

rds-ca-na2048-g1 (default)
Expiry: May 25, 2061

Additional configuration

Tags - optional: A tag consists of a case-sensitive key-value pair. No tags associated with the resource. [Add new tag](#)

Database authentication

Database authentication options:

- Password authentication:** Authenticates using database passwords.
- Password and IAM database authentication:** Authenticates using the database password and user credentials through AWS IAM users and roles.
- Password and Kerberos authentication:** Choose a directory in which you want to allow authorized users to authenticate with this DB instance using Kerberos Authentication.

Monitoring

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences ENG IN 11:33 29-08-2024

MySQL

MySQL is the most popular open source database in the world. MySQL on RDS offers the rich features of the MySQL community edition with the flexibility to easily scale compute resources or storage capacity for your database.

- Supports database size up to 64 TiB.
- Supports General Purpose, Memory Optimized, and Burstable Performance instance classes.
- Supports automated backup and point-in-time recovery.
- Supports up to 15 Read Replicas per instance, within a single Region or 5 read replicas cross-region.

The screenshot shows the AWS RDS console interface. On the left, there's a sidebar with options like Dashboard, Databases (which is selected), Query Editor, Performance insights, Snapshots, Exports in Amazon S3, Automated backups, Reserved instances, Proxies, Subnet groups, Parameter groups, Option groups, Custom engine versions, Zero-ETL integrations, Events, Event subscriptions, Recommendations, and Certificate update. The main area has a title 'Creating database database-1' with a note: 'Your database might take a few minutes to launch. You can use settings from database-1 to simplify configuration of suggested database add-ons while we finish creating your DB for you.' Below this is a 'Introducing Aurora I/O-Optimized' section. The 'Databases' section shows a table with one row:

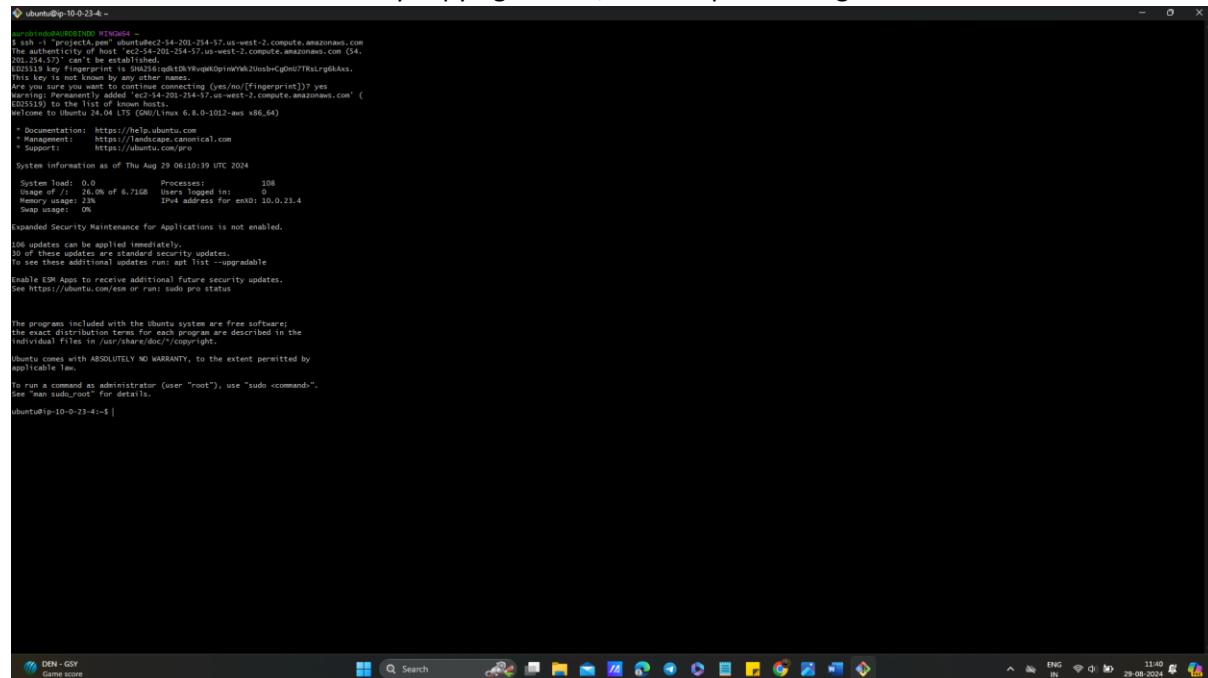
DB identifier	Status	Role	Engine	Region & ...	Size	Recommendations	C...	Current activity	Maintena...	VPC
database-1	Creating	Instance	MySQL Community	us-west-2a	db.t3.micro	-	-	-	none	vpc-071

At the bottom, there are buttons for Group resources, Modify, Actions, Restore from S3, and Create database. The status bar at the bottom right shows '© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences ENG IN 11:55 29-08-2024'.

This screenshot is nearly identical to the previous one, showing the AWS RDS console. The main difference is in the status of the database 'database-1'. It is now shown as 'Backing-up' instead of 'Creating'. The rest of the interface, including the sidebar, notifications, and status bar, remains the same.

Part 5: Testing

We connect to web-tier server by copying the ssh, after we paste into gitbash to connect



```
Ubuntu@ip-10-0-23-4 ~
$ ls -l
total 0
$ cd /tmp
$ curl -L https://git.io/v1UoI
$ ./git-ssh
$ ssh -i "projectA.pem" ubuntu@ec2-54-201-254-57.us-west-2.compute.amazonaws.com
The authenticity of host 'ec2-54-201-254-57.us-west-2.compute.amazonaws.com (54.201.254.57)' can't be established.
ED25519 key fingerprint: SHA256:nkt7kVvqKOpinWYw2Uosb+Cg0nU7TrsLrg0kAxs.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'ec2-54-201-254-57.us-west-2.compute.amazonaws.com' (ED25519) to the list of known hosts.
Welcome to Ubuntu 24.04 LTS (GNU/Linux 6.8.0-1012-aws x86_64)

 * Documentation: https://help.ubuntu.com
 * Management: https://landscape.canonical.com
 * Support: https://ubuntu.com/pro

System information as of Thu Aug 29 06:10:39 UTC 2024

System load: 0.0      Processes:          108
Usage of /: 26.0% of 6.71GB  Users logged in:   0
Memory usage: 23%           IPv4 address for enx0: 10.0.23.4
Swap usage: 0%          

expanded Security Maintenance for Applications is not enabled.

106 updates can be applied immediately.
90 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

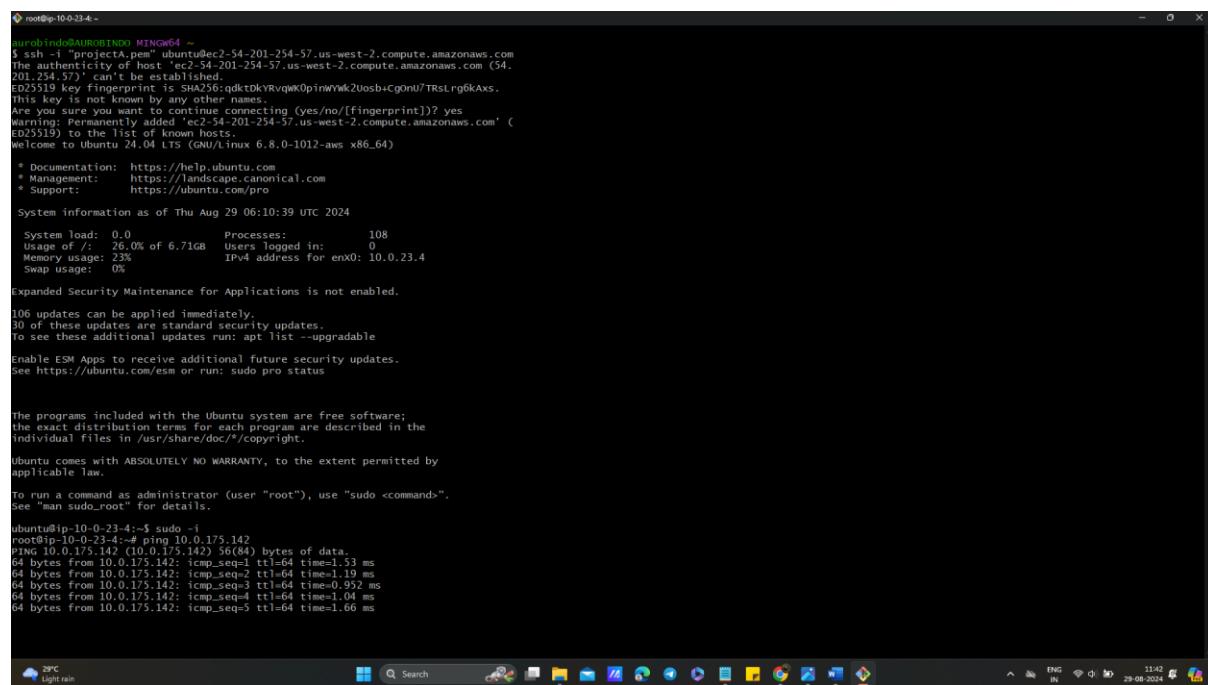
The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".
see "man sudo_root" for details.

ubuntu@ip-10-0-23-4:~$
```

In web-tier server we have to connect apptier by ping command give ping private ip of app-tier



```
root@ip-10-0-23-4 ~
$ ls -l
total 0
$ cd /tmp
$ curl -L https://git.io/v1UoI
$ ./git-ssh
$ ssh -i "projectA.pem" root@ip-10-0-23-4
The authenticity of host 'ip-10-0-23-4 (10.0.23.4)' can't be established.
ED25519 key fingerprint: SHA256:nkt7kVvqKOpinWYw2Uosb+Cg0nU7TrsLrg0kAxs.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'ip-10-0-23-4 (10.0.23.4)' (ED25519) to the list of known hosts.
Welcome to Ubuntu 24.04 LTS (GNU/Linux 6.8.0-1012-aws x86_64)

 * Documentation: https://help.ubuntu.com
 * Management: https://landscape.canonical.com
 * Support: https://ubuntu.com/pro

System information as of Thu Aug 29 06:10:39 UTC 2024

System load: 0.0      Processes:          108
Usage of /: 26.0% of 6.71GB  Users logged in:   0
Memory usage: 23%           IPv4 address for enx0: 10.0.23.4
Swap usage: 0%          

Expanded Security Maintenance for Applications is not enabled.

106 updates can be applied immediately.
90 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".
see "man sudo_root" for details.

ubuntu@ip-10-0-23-4:~$ sudo -i
root@ip-10-0-23-4:~# ping 10.0.1/5.142
PING 10.0.1/5.142(10.0.1.5) 56(80) bytes of data.
64 bytes from 10.0.1/5.142: icmp_seq=1 ttl=64 time=1.53 ms
64 bytes from 10.0.1/5.142: icmp_seq=2 ttl=64 time=1.19 ms
64 bytes from 10.0.1/5.142: icmp_seq=3 ttl=64 time=0.952 ms
64 bytes from 10.0.1/5.142: icmp_seq=4 ttl=64 time=1.04 ms
64 bytes from 10.0.1/5.142: icmp_seq=5 ttl=64 time=1.66 ms
```

```

2.7.0-20070801+main-3 [4384 B]
Get:28 http://us-west-2.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 mysql-server a
11 8.0.39-0ubuntu0.24.04.2 [9520 B]
Fetched 29.6 MB in 1s (56.8 MB/s)
Preconfiguring packages ...
Selecting previously unselected package mysql-common.
(Reading database ... 68464 files and directories currently installed.)
Preparing to unpack .../0-mysql-common_5.8+1.1.0build1_all.deb ...
Unpacking mysql-common (5.8+1.1.0build1) ...
Selecting previously unselected package mysql-client-core-8.0.
Preparing to unpack .../1-mysql-client-core-8.0_8.0.39-0ubuntu0.24.04.2_amd64.deb ...
Unpacking mysql-client-core-8.0 (8.0.39-0ubuntu0.24.04.2) ...
Selecting previously unselected package mysql-client-8.0.
Preparing to unpack .../2-mysql-client-8.0_8.0.39-0ubuntu0.24.04.2_amd64.deb ...
Unpacking mysql-client-8.0 (8.0.39-0ubuntu0.24.04.2) ...
Selecting previously unselected package libevent-pthreads-2.1-7t64:amd64.
Preparing to unpack .../3-libevent-pthreads-2.1-7t64_2.1.12-stable-9ubuntu2_amd64.deb ...
Unpacking libevent-pthreads-2.1-7t64:amd64 (2.1.12-stable-9ubuntu2) ...
Selecting previously unselected package libmecab2:amd64.
Preparing to unpack .../4-libmecab2_0.996-14ubuntu4_amd64.deb ...
Unpacking libmecab2:amd64 (0.996-14ubuntu4) ...
Selecting previously unselected package libprotobuf-lite32t64:amd64.
Preparing to unpack .../5-libprotobuf-lite32t64_3.21.12-8.2build1_amd64.deb ...
Unpacking libprotobuf-lite32t64:amd64 (3.21.12-8.2build1) ...
Selecting previously unselected package mysql-server-core-8.0.
Preparing to unpack .../6-mysql-server-core-8.0_8.0.39-0ubuntu0.24.04.2_amd64.deb ...
Unpacking mysql-server-core-8.0 (8.0.39-0ubuntu0.24.04.2) ...
Setting up mysql-common (5.8+1.1.0build1) ...
update-alternatives: using /etc/mysql/my.cnf.fallback to provide /etc/mysql/my.cnf (my.cnf) in auto mode

Progress: [ 14%] [#####
.....]
```

Now let's test connectivity to the Database Tier. Run

```
#sudo apt update
```

```
#sudo apt install mysql-server
```

```
#sudo systemctl start mysql.service
```

After this type my sql -h "end point of database we created" -u admin -p

```

Scanning linux images...
Running kernel seems to be up-to-date.
No services need to be restarted.
No containers need to be restarted.
No user sessions are running outdated binaries.

No VM guests are running outdated hypervisor (qemu) binaries on this host.
root@ip-10-0-23-4:~# sudo systemctl start mysql.service
root@ip-10-0-23-4:~# mysql -h database-1.c18guqsm2ovz.us-west-2.rds.amazonaws.com -u admin -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 29
Server version: 8.0.35 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>
```

If it is showing mysql> it means we connected to SQL server

```
owners.

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

mysql> show databases;
+-----+
| Database      |
+-----+
| information_schema |
| mysql          |
| performance_schema |
| sys            |
+-----+
4 rows in set (0.00 sec)

mysql>
```

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

mysql> show databases;
+-----+
| Database      |
+-----+
| information_schema |
| mysql          |
| performance_schema |
| sys            |
+-----+
4 rows in set (0.00 sec)

mysql> create aurobindo;
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near 'create database aurobindo' at line 1
mysql> create database aurobindo;
Query OK, 1 row affected (0.01 sec)

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

mysql> |
```

We have now successfully connected to our MySQL database from the application tier. We have connectivity with all of our tiers!