

3 – TIER ARCHITECTURE

Three-tier architecture is a software development model that organizes applications into three logical and physical computing tiers:

- 1.Presentation tier
- 2.Application tier
- 3.Data tier

1.Presentation tier:

Also known as the user interface, web tier or frontend this is where the end –user interacts with the system

2.Application tier:

Also known as the middle tier or logic tier this is the core of the application where information is processed using business logic

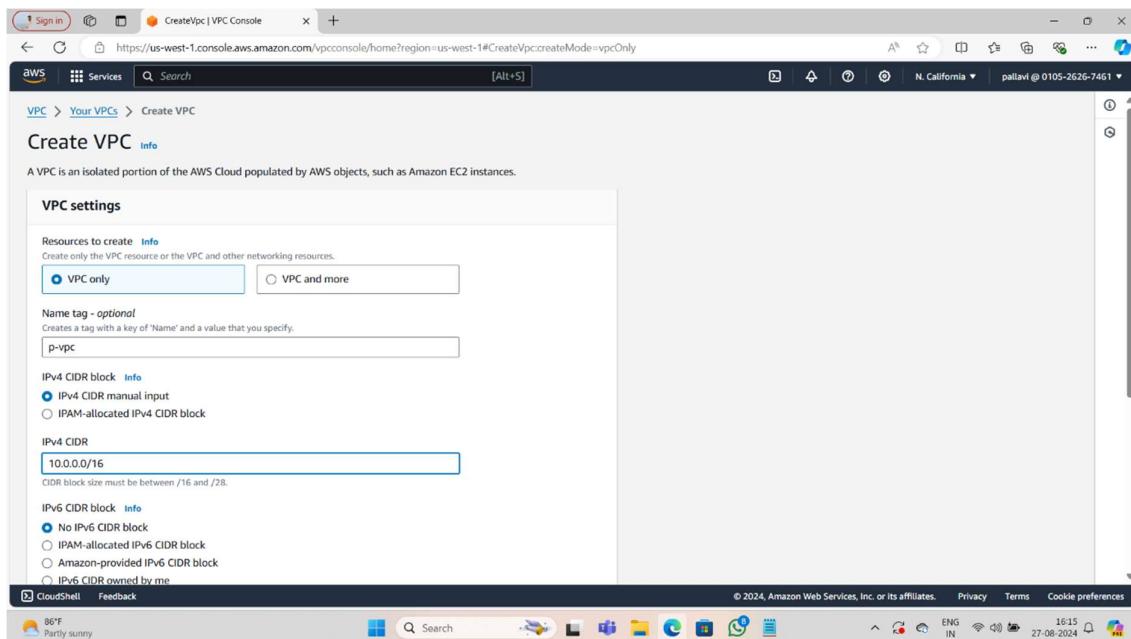
3.Data tier:

Also known as the databases tier,back-end, or data access tier, this is where the application's data is stored ,managed ,retrieved and manipulated



➤ Create VPC in N.CALIFORNIA :

- 1.login to the AWS account & select VIRGINIA region & do search for VPC in the search box.
2. click on create VPC
3. select VPC only & give name & give IPv4 CIDR and then
4. click on create VP



➤ Create SUBNETS (2- Public & 4 - Private):

- 1.Click on SUBNETS & Click on create subnet & choose VPC ID (Own VPC not Default).
2. give subnet name & select availability zone (1b or 1c) & give IPv4 subnet CIDR BLOCK
3. click on create subnet.
4. like that create 6 subnets – 2 public subnets in 1b & 1c zone and 4 private subnets – take 2 private subnets in 1b & remaining 2 private subnets in 1c zone.
5. some snapshots of subnets are added below

Create subnet

VPC

VPC ID
Create subnets in this VPC.

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.

The name can be up to 256 characters long.

Availability Zone
Choose the zone in which your subnet will reside, or let Amazon choose one for you.

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

IPv4 subnet CIDR block
 256 IPs

Tags - optional
Key Value - optional

You can add 49 more tags.

CloudShell Feedback

86°F Partly sunny

Search

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

16:16 ENG IN 27-08-2024

Create subnet

VPC

VPC ID
Create subnets in this VPC.

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.

The name can be up to 256 characters long.

Availability Zone
Choose the zone in which your subnet will reside, or let Amazon choose one for you.

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

IPv4 subnet CIDR block
 256 IPs

Tags - optional
Key Value - optional

You can add 49 more tags.

CloudShell Feedback

86°F Partly sunny

Search

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

16:16 ENG IN 27-08-2024

Create subnet

VPC

VPC ID
Create subnets in this VPC.

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.

The name can be up to 256 characters long.

Availability Zone
Choose the zone in which your subnet will reside, or let Amazon choose one for you.

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

IPv4 subnet CIDR block
 256 IPs

Tags - optional
Key Value - optional

You can add 49 more tags.

CloudShell Feedback

86°F Partly sunny

Search

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

16:17 ENG IN 27-08-2024

Screenshot of the AWS VPC console showing the 'Create subnet' wizard step 1 of 3.

Subnet details:

- Name:** sub-public2
- Availability Zone:** US West (N. California) / us-west-1c
- IPv4 CIDR block:** 10.0.0.0/16
- IPv4 subnet CIDR block:** 10.0.0.0/24 (256 IPs)

Tags - optional:

Key	Value - optional
Q Name	Q sub-public2

Buttons: Cancel, Create subnet

Screenshot of the AWS VPC console showing the 'Create subnet' wizard step 2 of 3.

VPC ID: vpc-0fd9b04fef095cae (p-vpc)

Associated VPC CIDRs: 10.0.0.0/16

Subnet settings:

Subnet 1 of 1:

Subnet name: sub-private1

Availability Zone: US West (N. California) / us-west-1c

Buttons: CloudShell, Feedback, Cancel, Create subnet

Screenshot of the AWS VPC console showing the 'Create subnet' wizard step 3 of 3.

Subnet details:

- Name:** sub-private1
- Availability Zone:** US West (N. California) / us-west-1b
- IPv4 CIDR block:** 10.0.0.0/16
- IPv4 subnet CIDR block:** 10.0.2.0/24 (256 IPs)

Tags - optional:

Key	Value - optional
Q Name	Q sub-private1

Buttons: Cancel, Create subnet

Create subnet

VPC

VPC ID
Create subnets in this VPC.

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.

The name can be up to 256 characters long.

Availability Zone
Choose the zone in which your subnet will reside, or let Amazon choose one for you.

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

IPv4 subnet CIDR block
 256 IPs

Tags - optional
Key Value - optional

You can add 49 more tags.

CloudShell **Feedback**

86°F Partly sunny

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

16:19 27-08-2024

SUBNET NAME
Create a tag with a key of 'Name' and a value that you specify.

The name can be up to 256 characters long.

Availability Zone
Choose the zone in which your subnet will reside, or let Amazon choose one for you.

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

IPv4 subnet CIDR block
 256 IPs

Tags - optional
Key Value - optional

You can add 49 more tags.

CloudShell **Feedback**

86°F Partly sunny

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

16:20 27-08-2024

Name	Subnet ID	State	VPC	IPv4 CIDR	IPv6 CIDR
sub-private2	subnet-0b536fdb8791a5e657	Available	vpc-0fd9b04fef095cae p-vpc	10.0.3.0/24	-
-	subnet-0c28b64133f1e50a1	Available	vpc-0fd9b04fef095cae p-vpc	172.31.16.0/20	-
sub-private4	subnet-0e47774e73103498	Available	vpc-0fd9b04fef095cae p-vpc	10.0.5.0/24	-
sub-public2	subnet-0eeef48d5fe8902	Available	vpc-0fd9b04fef095cae p-vpc	10.0.1.0/24	-
sub-private1	subnet-0316e54ab86f24200	Available	vpc-0fd9b04fef095cae p-vpc	10.0.2.0/24	-
-	subnet-03e151e840124cd0f	Available	vpc-0fd9b04fef095cae p-vpc	172.31.0.0/20	-
sub-public1	subnet-050c4335f1d867b6e	Available	vpc-0fd9b04fef095cae p-vpc	10.0.0.0/24	-
sub-private3	subnet-03036f6bc41757c26	Available	vpc-0fd9b04fef095cae p-vpc	10.0.4.0/24	-

➤ Create INTERNET GATEWAY:

- 1.click on internet gateway &create internet gateway.
2. after the creation of internet gateway, click on internet gateway , click on actions & attach it to VPC.
- 3.Snapshots of internet gateway are attached below

VPC
Attach an Internet gateway to a VPC to enable the VPC to communicate with the internet. Specify the VPC to attach below.

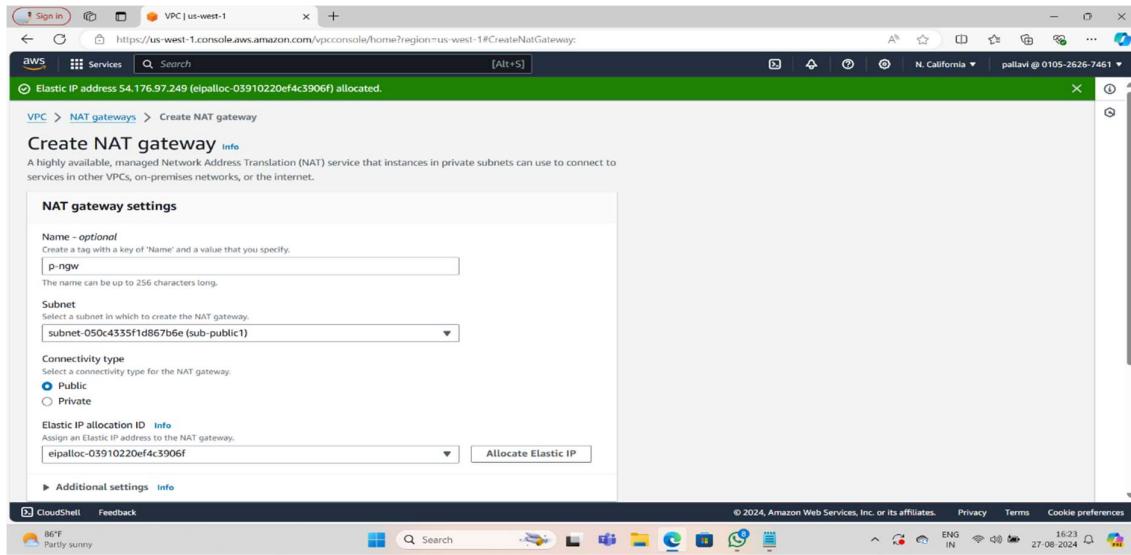
Available VPCs
Attach the internet gateway to this VPC:
vpc-0fd9b04fef095cae

AWS Command Line Interface command

Cancel Attach internet gateway

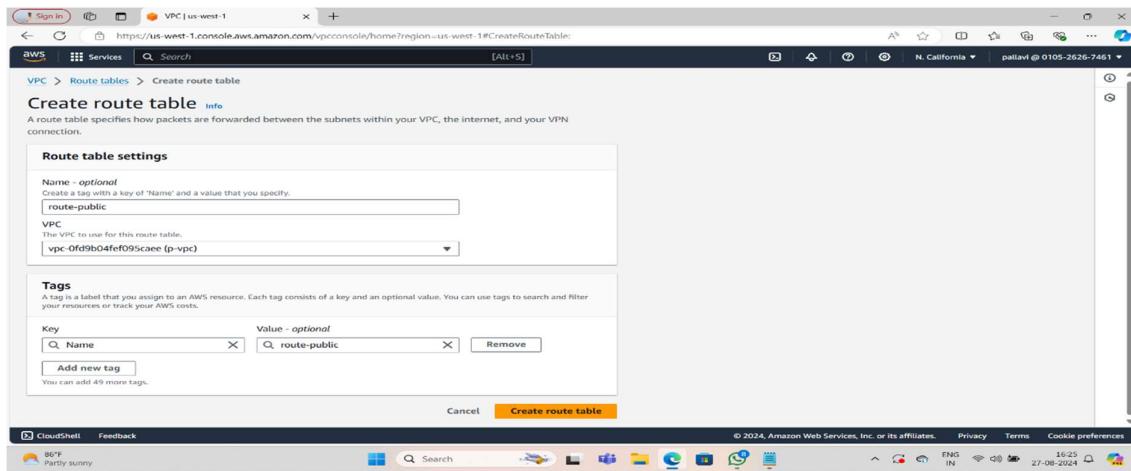
➤ Create NATGATEWAY:

- 1.click on Nat gateway & click on create.
- 2.select PRIVATE SUBNET & choose connectivity type as PUBLIIC & Allocate ELASTIC IP
- 3.Click on create NAT GATEWAY



➤ Create ROUTE TABLES:

1. We have to create 2 route tables – one is PUBLIC & another one is PRIVATE.
2. goto route table – click on create route- select VPC & create route table.
3. click on route- actions-edit subnet associations-select PUBLIC SUBNETS – save associations.
4. create another route table as PRIVATE.
5. Select VPC – do edit subnet associations – select 4 private subnets – save associations
6. snapshots are attached below.



- 7.click on public route table – edit routes – add rules- attach internet gateway – save changes.
8. for private route table – attach internet gateway & Nat gateway – save changes.
9. Now go to the subnets - click on public subnet -01, click on actions – edit subnet settings – ENABLE Auto assign public IPv4 address.
- 10.Do the same for remaining subnets also.

➤ Create SECURITY GROUPS:

1.we have to create TWO security groups.

2. Go to security groups - click on create security groups- select VPC – add INBOUND (SSH & HTTP) & OUTBOUND RULES (All traffic) – click on create security group.

3.snapshots are attached below

The image contains two side-by-side screenshots of the AWS VPC console. Both screenshots show the 'Create security group' page.

Screenshot 1 (Top):

- Basic details:**
 - Security group name: private sg
 - Description: allow
 - VPC: vpc-0fd9b04fef095cae (p-vpc)
- Inbound rules:**

Type	Protocol	Port range	Source	Description - optional
All traffic	All	All	Anyw...	0.0.0.0/0
SSH	TCP	22	Anyw...	0.0.0.0/0
HTTP	TCP	80	Anyw...	0.0.0.0/0

Screenshot 2 (Bottom):

- Inbound rules:**

Type	Protocol	Port range	Source	Description - optional
All traffic	All	All	Anyw...	0.0.0.0/0
SSH	TCP	22	Anyw...	0.0.0.0/0
HTTP	TCP	80	Anyw...	0.0.0.0/0

Warning: 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.
- Outbound rules:**

Type	Protocol	Port range	Destination	Description - optional
All traffic	All	All	Anyw...	0.0.0.0/0

➤ NOW LAUNCH TWO TEMPLATES (Public &Private):

PUBLIC TEMPLATE:

1. Search EC2 – Click on LAUNCH TEMPLATES – Click on CREATE LAUNCH TEMPLATES.
2. Select AMI – UBUNTU & instance type - t2. Micro (1GB- Free Tier).
3. Select KEY PAIR – a new or existing.
4. In Network Settings I am not going to specify subnets, but security group (public -sg) that I am created is selected. Make sure the proper VPC is selected.
5. Snapshots of Public Template are attached below.

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 versions.

Launch template name and description

Launch template name - required
temp-public
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.

CloudShell Feedback
High UV Now

Summary

Software Image (AMI)
Amazon Linux

Virtual server type (instance type)

Firewall (security group)

Storage (volumes)

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 GB of EBS storage, 2 million IOs, 1 GB of snapshots, and 100 GB of bandwidth available on this instance.

Create launch template

Create launch template

Search our full catalog including 1000s of application and OS images

Recent **Quick Start**

Amazon Machine Image (AMI)

Ubuntu Server 24.04 LTS (HVM), SSD Volume Type Free tier eligible
ami-0d53d72369335a9d6 / ami-05c99ba7750f139a2262 (64-bit (Arm))
Virtualization: hvm ENA enabled: true Root device type: ebs

Description
Ubuntu Server 24.04 LTS (HVM),EBS General Purpose (SSD) Volume Type. Support available from Canonical (<http://www.ubuntu.com/cloud/services>).

Architecture 64-bit (x86) **AMI ID** ami-0d53d72369335a9d6 **Verified provider**

Instance type Info | Get advice Advanced

CloudShell Feedback
High UV Now

Summary

Software Image (AMI) Canonical, Ubuntu, 24.04, amd6...read more
ami-0d53d72369335a9d6

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 GB of EBS storage, 2 million IOs, 1 GB of snapshots, and 100 GB of bandwidth available on this instance.

Create launch template

Create launch template

VPC | us-west-1

Subnet Info

Network settings Info

Subnet Info
Don't include in launch template Create new subnet

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
public-sg sg-010091d0f6429d61c
VPC: vpc-0fd9104fe0f95caee

Compare security group rules

Advanced network configuration

Storage (volumes) Info

EBS Volumes

Summary

Software Image (AMI) Canonical, Ubuntu, 24.04, amd6...read more
ami-0d53d72369335a9d6

Virtual server type (instance type)

Firewall (security group) public-sg

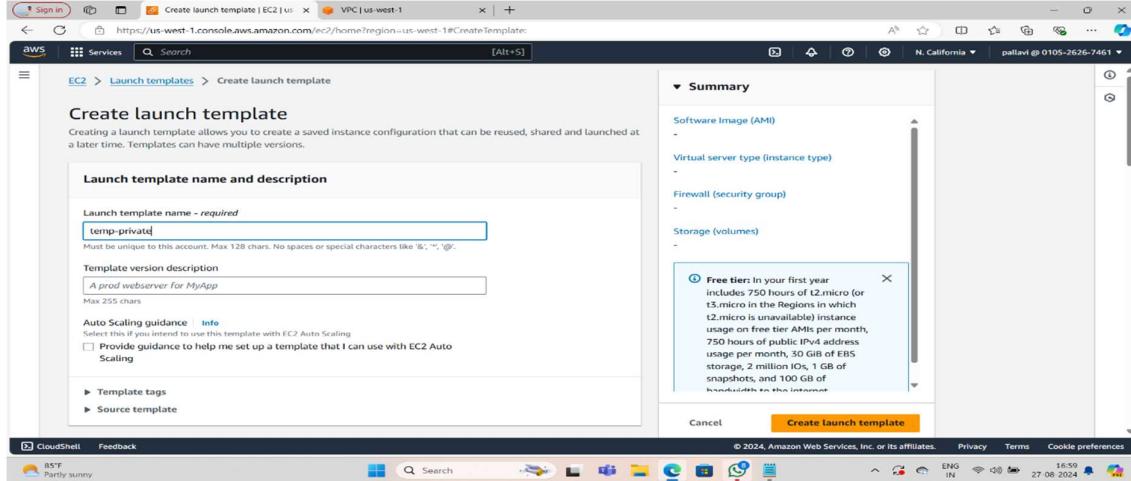
Storage (volumes)
1 volume(s) - 8 GiB

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 GB of EBS storage, 2 million IOs, 1 GB of snapshots, and 100 GB of bandwidth available on this instance.

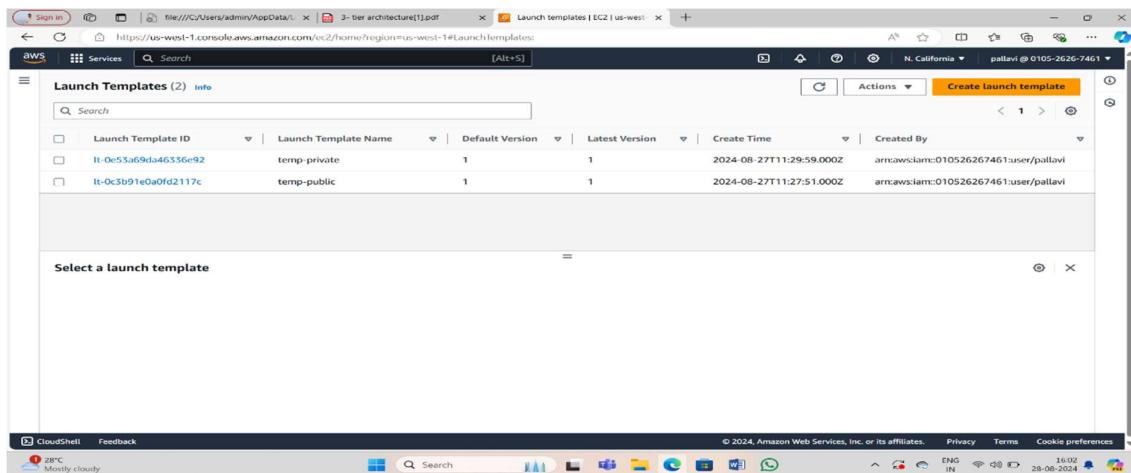
Create launch template

PRIVATE TEMPLATE :

- 1.Create same as previous template, but at SECURITY GROUP select security group-2(private -sg)
- 2.Snapshots of PRIVATE TEMPLATE as attached below.



- 3.two templates launched successfully



➤ CREATE TWO AUTOSCALING GROUPS (Public&Private):

PUBLIC ASG:

- 1.In EC2, go to autoscaling group – click on create autoscaling group.
- 2.give name to ASG – Select PUBLIC TEMPLATE (which is already created)
- 3.In network settings - choose VPC – choose 2 public subnets.
4. After that click on NEXT.
5. We have to attach LOAD BALANCER to ASG.

The screenshots show the AWS Create Auto Scaling group wizard in three stages:

- Step 1: Choose launch template** (Screenshot 1): Shows the 'Name' field set to 'public-asg'. A note states: "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." The 'Launch template' dropdown shows 'temp-public'.
- Step 2: Choose instance launch options** (Screenshot 2): Shows the 'Network' section. Under 'VPC', 'vpc-0fd9b04fe0f095cae (p-vpc)' is selected. Under 'Availability Zones and subnets', two subnets are listed: 'us-west-1b | subnet-050c4335f1d867b6e (sub-public1)' and 'us-west-1c | subnet-0eeefef4bd5fea8902 (sub-public2)'. A 'Create a subnet' button is also present.
- Step 3 - optional: Configure advanced options** (Screenshot 3): Shows the 'Load balancing' section. It offers three options: '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), and 'Attach to a new load balancer' (Quickly create a basic load balancer and attach to your Auto Scaling group). The 'Attach to a new load balancer' option is selected. Under 'Attach to a new load balancer', it says 'Load balancer type' and lists 'Application Load Balancer (HTTP, HTTPS)' and 'Network Load Balancer (TCP, UDP, TLS)', with 'Application Load Balancer' selected.

6. Attach load balancer- choose application load balancer- LB name should be same as ASG , if you want to edit it you can edit the name.

7.select subnets – give PORT NO: 80 for HTTP – Select TARGET GROUP (new or existing).

8.Give HEALTH CHECK GRACE PERIOD as your wish.

Sign in https://us-west-1.console.aws.amazon.com/ec2/home?region=us-west-1#CreateAutoScalingGroup: VPC | us-west-1

aws Services Search [Alt+S] N. California pallavi @ 0105-2626-7461

Add tags Step 7 Review

Attach to a new load balancer

Define a new load balancer to create for attachment to this Auto Scaling group.

Load balancer type

Choose from the load balancer types offered below. Type selection cannot be changed after the load balancer is created. If you need a different type of load balancer than those offered here, visit the Load Balancing console.

Application Load Balancer HTTP, HTTPS

Network Load Balancer TCP, UDP, TLS

Load balancer name

Name cannot be changed after the load balancer is created.

public-asg-1

Load balancer scheme

Scheme cannot be changed after the load balancer is created.

Internal

Internet-facing

Network mapping

Your new load balancer will be created using the same VPC and Availability Zone selections as your Auto Scaling group. You can select different subnets and add subnets from additional Availability Zones.

VPC

vpc-0fd9b04fef095cae [] p-vpc

Availability Zones and subnets

You must select a single subnet for each Availability Zone enabled. Only public subnets are available for selection to support DNS resolution.

us-west-1c subnet-0eceeef48d5fea8902

us-west-1b subnet-050c4335f1d867b6e

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences 85°F Partly sunny 17:02 27-08-2024

Sign in https://us-west-1.console.aws.amazon.com/ec2/home?region=us-west-1#CreateAutoScalingGroup: VPC | us-west-1

aws Services Search [Alt+S] N. California pallavi @ 0105-2626-7461

Availability zones and subnets

You must select a single subnet for each Availability Zone enabled. Only public subnets are available for selection to support DNS resolution.

us-west-1c subnet-0eceeef48d5fea8902

us-west-1b subnet-050c4335f1d867b6e

Listeners and routing

If you require secure listeners, or multiple listeners, you can configure them from the Load Balancing console after your load balancer is created.

Protocol	Port	Default routing (forward to)
HTTP	80	Create a target group New target group name An instance target group with default settings will be created. public-asg

Tags - optional

Consider adding tags to your load balancer. Tags enable you to categorize your AWS resources so you can more easily manage them.

Add tag 50 remaining

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

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences 85°F Partly sunny 17:02 27-08-2024

Sign in https://us-west-1.console.aws.amazon.com/ec2/home?region=us-west-1#CreateAutoScalingGroup: 3-tier architecture(1).pdf

aws Services Search [Alt+S] N. California pallavi @ 0105-2626-7461

Additional health check types - optional

Turn on Elastic Load Balancing health checks

Elastic Load Balancing monitors whether instances are available to handle requests. When it reports an unhealthy instance, EC2 Auto Scaling can replace it on its next periodic check.

Turn on VPC Lattice health checks

VPC Lattice can monitor whether instances are available to handle requests. If it considers a target as failed a health check, EC2 Auto Scaling replaces it after its next periodic check.

Turn on Amazon EBS health checks

EBS monitors whether an instance's root volume or attached volume stalls. When it reports an unhealthy volume, EC2 Auto Scaling can replace the instance on its next periodic health check.

Health check grace period

This time period delays the first health check until your instances finish initializing. It doesn't prevent an instance from terminating when placed into a non-running state.

30 seconds

Additional settings

Monitoring

Enable group metrics collection within CloudWatch

Default instance warmup

The amount of time that CloudWatch metrics for new instances do not contribute to the group's aggregated instance metrics, as their usage data is not reliable yet.

Enable default instance warmup

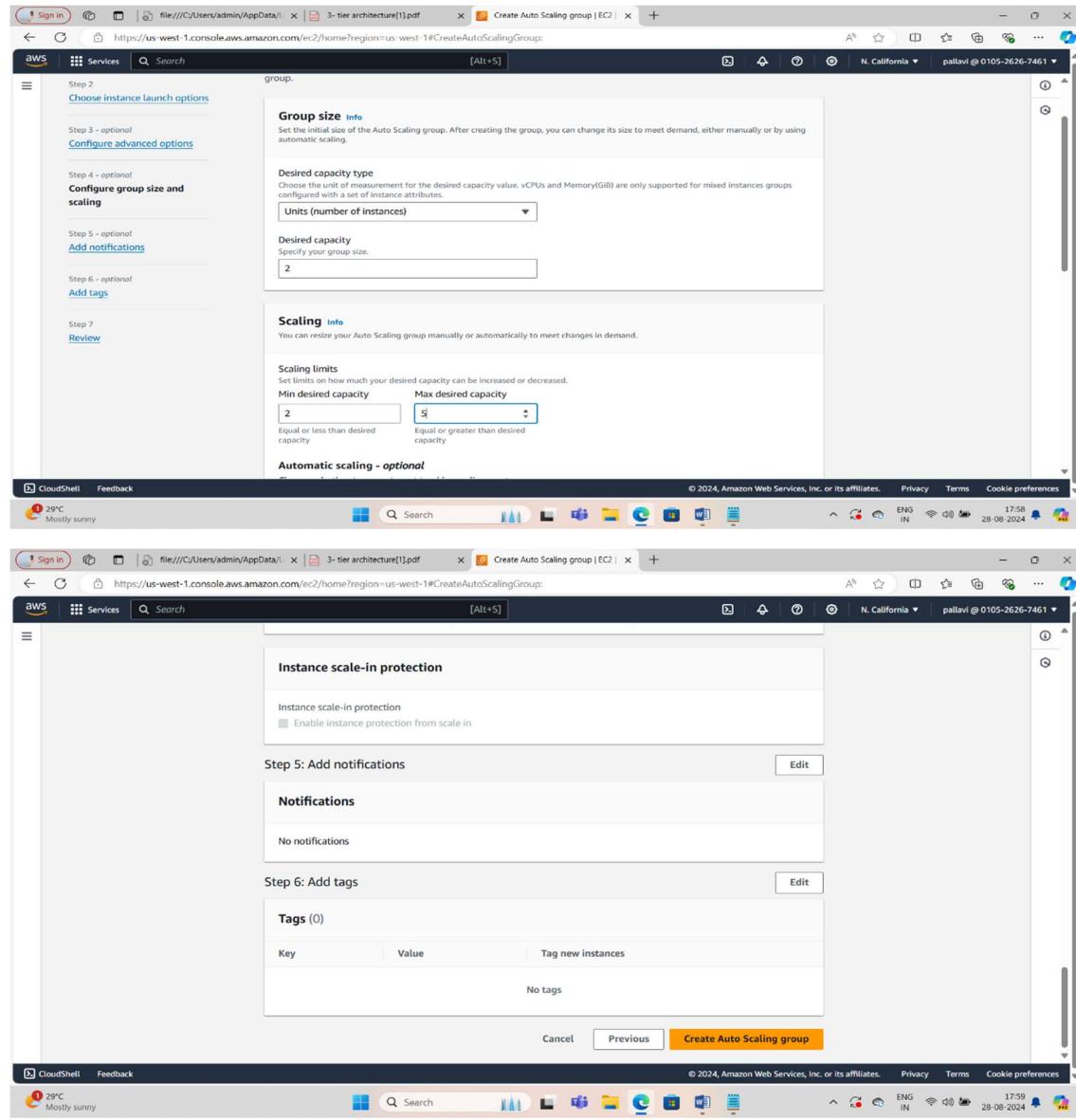
Cancel Skip to review Previous Next

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences 85°F Heavy rain Tomorrow 17:58 28-08-2024

9. Select Group size We want to set a minimum and maximum number of instances the ASG can provision:

- Desired capacity: 2
- Minimum capacity: 2
- Maximum capacity: 5

10. After that, click on next – next – create auto scaling group.



CREATE ANOTHER ASG AS PRIVATE:

1. follow all the steps as above.
2. But at network settings, choose 4 private subnets

Screenshot of the AWS EC2 Create Auto Scaling group wizard Step 1: Choose launch template.

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.
private-asg

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.

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

Create a launch template

CloudShell Feedback 85°F Partly sunny Search © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences N. California pallavi @ 0105-2626-7461 17:04 27-08-2024

Screenshot of the AWS EC2 Create Auto Scaling group wizard Step 2: Choose instance launch options.

VPC

Choose the VPC that defines the virtual network for your Auto Scaling group.
vpc-0f0b04fe095cae (p-vpc)

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-west-1b | subnet-03036f6bc41757c26 (sub-private3)
10.0.4.0/24

us-west-1c | subnet-0b536fdb8791a5e657 (sub-private4)
10.0.3.0/24

us-west-1c | subnet-0c647774e73103498 (sub-private4)
10.0.5.0/24

us-west-1b | subnet-0316e54ab86f24200 (sub-private1)
10.0.2.0/24

Create a subnet

Cancel Skip to review Previous Next

CloudShell Feedback 85°F Partly sunny Search © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences N. California pallavi @ 0105-2626-7461 17:05 27-08-2024

Screenshot of the AWS EC2 Create Auto Scaling group wizard Step 3: Configure advanced options - optional.

Configure advanced options - optional

Integrate your Auto Scaling group with other services to distribute network traffic across multiple servers using a load balancer or to establish service-to-service communications using VPC Lattice. You can also set options that give you more control over health check replacements and monitoring.

Load balancing

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.

Attach to a new load balancer

Define a new load balancer to create for attachment to this Auto Scaling group.

Load balancer type

Choose from the load balancer types offered below. Type selection cannot be changed after the load balancer is created. If you need a different type of load balancer than those offered here, visit the Load Balancing console.

Application Load Balancer
HTTP, HTTPS

Network Load Balancer
TCP, UDP, SSL

CloudShell Feedback 85°F Partly sunny Search © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences N. California pallavi @ 0105-2626-7461 17:05 27-08-2024

Create Auto Scaling group | EC2

Step 6 - optional
Add tags

Step 7
Review

Attach to a new load balancer
Define a new load balancer to create for attachment to this Auto Scaling group.

Load balancer type
Choose from the load balancer types offered below. Type selection cannot be changed after the load balancer is created. If you need a different type of load balancer than those offered here, visit the Load Balancing console.

Application Load Balancer
HTTP, HTTPS

Network Load Balancer
TCP, UDP, TLS

Load balancer name
Name cannot be changed after the load balancer is created.
private-asg

Load balancer scheme
Scheme cannot be changed after the load balancer is created.

Internal

Internet-facing

Network mapping
Your new load balancer will be created using the same VPC and Availability Zone selections as your Auto Scaling group. You can select different subnets and add subnets from additional Availability Zones.

VPC
vpc-0fd9b04fef095cae

Availability Zones and subnets
You must select a single subnet for each Availability Zone enabled. Only public subnets are available for selection to support DNS resolution.

us-west-1c Select a subnet

CloudShell Feedback
85°F Partly sunny

Create Auto Scaling group | EC2

subnets | VPC Console

Internal **Internet-facing**

Network mapping
Your new load balancer will be created using the same VPC and Availability Zone selections as your Auto Scaling group. You can select different subnets and add subnets from additional Availability Zones.

VPC
vpc-0fd9b04fef095cae

Availability Zones and subnets
You must select a single subnet for each Availability Zone enabled. Only public subnets are available for selection to support DNS resolution.

us-west-1c subnet-0b536fd8791a5e657

us-west-1b subnet-0316e54ab86f24200

Listeners and routing
If you require secure listeners, or multiple listeners, you can configure them from the Load Balancing console after your load balancer is created.

Protocol	Port	Default routing (forward to)
HTTP	80	Create a target group New target group name An instance target group with default settings will be created. private-asg

Tags - optional
Consider adding tags to your load balancer. Tags enable you to categorize your AWS resources so you can more easily manage them.

Add tag

CloudShell Feedback
85°F Partly sunny

4.Successfully created TWO AUTO SCALING GROUPS.

AWS THREE-TIER ARCHITECTURE **Auto Scaling groups | EC2**

EC2 > Auto Scaling groups

Auto Scaling groups (2) info

Search your Auto Scaling groups

Name	Launch template/configuration	Instances	Status	Desired capacity	Min	Max	Available
private-asg	temp-private Version Default	2	-	2	2	5	us-west-1...
public-asg	temp-public Version Default	2	-	2	2	5	us-west-1...

0 Auto Scaling groups selected

CloudShell Feedback
24°C Haze

5. Now go to EC2 dashboard- click on instances.

6. Here we can see the 4 FOUR NEW RUNNING INSTANCES.

7. Give the names to those instances as – public-1, public-2, private-1 & private-2.

The screenshot shows the AWS EC2 Instances page. The left sidebar has 'Instances' selected under 'Instances'. The main area displays a table of instances with the following details:

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPs
private1	i-0346c61f80966f5cb	Running	m1.small	2/2 checks passed	View alarms	us-west-1b	-
public1	i-0e1007eb9ecbfae4a	Running	m1.small	2/2 checks passed	View alarms	us-west-1b	-
private2	i-0553a20545cf8bec	Running	m1.small	2/2 checks passed	View alarms	us-west-1c	-
public2	i-01127d4c6bb6bcd4e	Running	m1.small	2/2 checks passed	View alarms	us-west-1c	-

The screenshot shows a terminal session on the public-1 instance (i-0e1007eb9ecbfae4a). The user runs several commands to update packages and install Apache2:

```
root@ip-10-0-97-# apt update -y
Hit:1 http://us-west-1.ec2.archive.ubuntu.com/ubuntu noble InRelease
Get:2 http://us-west-1.ec2.archive.ubuntu.com/ubuntu noble-updates InRelease [126 kB]
Hit:3 http://us-west-1.ec2.archive.ubuntu.com/ubuntu noble-backports InRelease
Hit:4 http://security.ubuntu.com/ubuntu noble-security InRelease
Get:5 http://us-west-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 Packages [446 kB]
Get:6 http://us-west-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 c-n-f Metadata [7716 B]
Get:7 http://us-west-1.ec2.archive.ubuntu.com/ubuntu noble-updates/universe amd64 Packages [337 kB]
Get:8 http://us-west-1.ec2.archive.ubuntu.com/ubuntu noble-updates/universe amd64 c-n-f Metadata [13.7 kB]
Fetched 953 kB in 1s (1141 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
102 packages can be upgraded. Run 'apt list --upgradable' to see them.
root@ip-10-0-97-# apt install apache2 -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
apache2 is already the newest version (2.4.58-1ubuntu8.4).
0 upgraded, 0 newly installed, 0 to remove and 102 not upgraded.
root@ip-10-0-97-# cd /var/www/html
root@ip-10-0-97-# /var/www/html ls
index.html
root@ip-10-0-97-# /var/www/html rm index.html
root@ip-10-0-97-# /var/www/html vi index.html
root@ip-10-0-97-# /var/www/html cd
root@ip-10-0-97-# systemctl status apache2
```

Below the terminal window, the status bar shows:

i-0e1007eb9ecbfae4a (public1)
PublicIPs: 54.215.185.246 PrivateIPs: 10.0.0.97

8. Click on public -1 instance – click on connect- connect to the EC2 server.

9. After connecting to the UBUNTU, give the following commands:

- sudo -I -to become a root user
- apt update -y – to update packages
- apt install apache2 – to install apache2

- cd /var/www/html – path
 - ls – list
 - rm index.html – to remove index.html
 - vi index.html – enter – press I –insert date --:x (to save) – enter
 - systemctl status apache2 – to check the server status.
 - Ping google.com – to check whether the server is ping or not.

10.screenshots attached below.

11. Copy the PUBLIC IP address & paste it in google chrome.

12. It will shows the data which we inserted.

```
Sign In RDS | us-west-1 Amazon Web Services Sign-In How to Install MySQL on Ubuntu Instance details | EC2 | us-west-1 EC2 Instance Connect | us-west-1 + N. California pallavi @ 0105-2626-7461

aws Services Search [Alt+S] https://us-west-1.console.aws.amazon.com/ec2-instance-connect/shell?connType=standard&instanceId=i-0e1007eb9ecfae4a8osUser=ubuntu&region=us-west-1...
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
python3-pip is already the newest version (2.4.58-1ubuntu0.4).
0 upgraded, 0 newly installed, 0 to remove and 102 not upgraded.
root@ip-10-0-0-97:~# cd /var/www/html
root@ip-10-0-0-97:/var/www/html# ls
index.html
root@ip-10-0-0-97:/var/www/html# rm index.html
root@ip-10-0-0-97:/var/www/html# vi index.html
root@ip-10-0-0-97:/var/www/html# cd
root@ip-10-0-0-97:~# systemctl status apache2
● apache2.service - Apache2 - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/apache2.service; enabled; preset: enabled)
   Active: active (running) since Tue 2024-08-27 11:43:43 UTC; 5h 7min ago
     Docs: https://httpd.apache.org/docs/2.4/
Main PID: 1135 (apache2)
   Tasks: 53 (limit: 1973)
      Memory: 8.6M (peak: 8.9M)
        CPU: 1.164s
      CGroup: /system.slice/apache2.service
              └─1135 /usr/sbin/apache2 -k start
                 ├─2401 /usr/sbin/apache2 -k start
                 ├─2402 /usr/sbin/apache2 -k start
                 └─2403 /usr/sbin/apache2 -k start

Aug 27 11:43:43 ip-10-0-0-97 systemd[1]: Starting apache2.service - The Apache HTTP Server...
Aug 27 11:43:43 ip-10-0-0-97 apachectl[2403]: AH00558: apache2: Could not reliably determine the server's fully qualified domain name, using 10.0.0.97. Set the F
Aug 27 11:43:43 ip-10-0-0-97 systemd[1]: Started apache2.service - The Apache HTTP Server.
lines 1-16/16 (END)
```

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

NOW CONNECT TO THE PRIVATE INSTANCE THROUGH THE PUBLIC INSTANCE:

- 1.Follow the steps as mentioned in the snapshots.
- 2.Successfully connected to the PRIVATE INSTANCE

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

System information as of Tue Aug 27 17:01:51 UTC 2024
System load: 0.0 Processes: 109
Usage of /: 33.5% of 6.71GB Users logged in: 0
Memory usage: 40% IPv4 address for enX0: 10.0.2.142
Swap usage: 0%

* Ubuntu Pro delivers the most comprehensive open source security and
  compliance features.
  https://ubuntu.com/aws/pro

Expanded Security Maintenance for Applications is not enabled.
100 updates can be applied immediately.
30 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable

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

Last login: Tue Aug 27 17:15:16 2024 from 10.0.0.97
ubuntu@ip-10-0-2-142:~$ i-0e1007eb9ecbfae4a (public1)
PublicIPs: 54.215.185.246 PrivateIPs: 10.0.0.97
```

➤ Create SUBNET GROUP

- 1.select create subnet group on rds dashboard
- 2.give a name to new subnet group-select vpc
- 3.add availability zones (us-west-1b & us-west-1c) and 6 subnets

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.

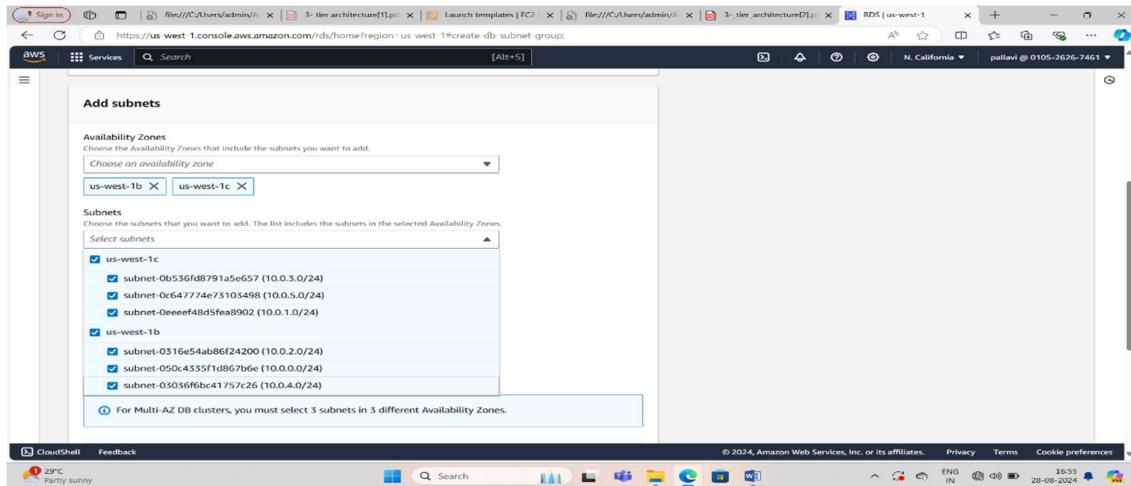
Must contain from 1 to 255 characters. Alphanumeric characters, spaces, hyphens, underscores, and periods are allowed.

Description

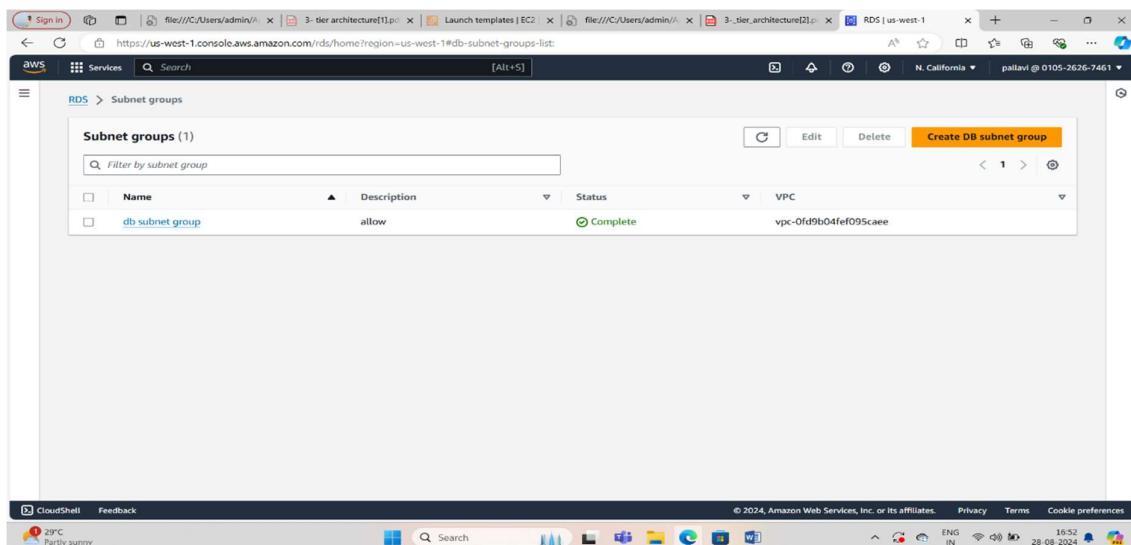
VPC
Choose a VPC identifier that corresponds to the subnets you want to use for your DB subnet group. You won't be able to choose a different VPC identifier after your subnet group has been created.

Add subnets

Availability Zones



4.successfully created a subnet group



➤ Create DATABASE

- 1.select database on rds dashboard and click create database
- 2.choose STANDARD in creation method and MYSQL in engine methods
- 3.select FREE TIER in template
4. choose SELF MANAGED in settings and give PASSWORD
- 5.go to connectivity and choose CONNECT TO EC2 INSTANCE and select public instance
- 6.we don't have to give vpc and db subnet group it will take by default
- 7.choose existing security groups and select both public (public-sg) and private (private-sg) security groups

Screenshot of the AWS RDS console showing the 'Create database' wizard. The first step, 'Choose a database creation method', offers 'Standard create' (selected) and 'Easy create'. The second step, 'Engine options', lists engine types: Aurora (MySQL Compatible), Aurora (PostgreSQL Compatible), MySQL (selected), and MariaDB. A detailed description of MySQL is provided on the right, highlighting its popularity and various features.

Screenshot of the AWS RDS console showing the 'Templates' step of the creation wizard. It offers three template options: 'Production' (selected), 'Dev/Test', and 'Free tier'. The 'Free tier' option is described as using the Free Tier to develop new applications, test existing ones, or gain hands-on experience with Amazon RDS. The 'Settings' section allows specifying the DB instance identifier ('database-1') and master username ('admin'). A detailed description of MySQL is provided on the right.

Screenshot of the AWS RDS console showing the 'Credentials Settings' step of the creation wizard. It includes fields for 'Master username' ('admin'), 'Auto generate password' (unchecked), 'Master password' ('pallavareddy10'), and 'Confirm master password' ('pallavareddy10'). The 'Self managed' option is selected for password management. The 'Instance configuration' section at the bottom is noted as being limited to supported options. A detailed description of MySQL is provided on the right.

Connectivity Info

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.

EC2 instance Info
Choose the EC2 instance to add as the compute resource for this database. A VPC security group is added to this EC2 instance. A VPC security group is also added to the database with an inbound rule that allows the EC2 instance to access the database.

i-0e1007db5ecfae4a
public1

Some VPC settings can't be changed when a compute resource is added
Adding an EC2 compute resource automatically selects the VPC, DB subnet group, and public access settings for this database. To allow the EC2 instance to access this database, a VPC security group rds-ec2-X is added to the database and another called ec2-rds-X to the EC2 instance. You can remove the new security group for the database only by removing the compute resource.

Network type Info
To use dual-stack mode, make sure that you associate an IPv6 CIDR block with a subnet in the VPC you specify.

IPv4
Your resources can communicate only over the IPv4 addressing protocol.

Dual-stack mode
Your resources can communicate over IPv4, IPv6, or both.

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.

CloudShell Feedback

CloudShell provides a terminal window for running AWS Lambda functions and AWS Lambda layers. It includes a history of previous commands and a log viewer for monitoring function execution.

CloudShell is currently running in the CAD/INR environment. The current memory usage is +0.25%

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.

Virtual private cloud (VPC) Info
Choose the VPC. The VPC defines the virtual networking environment for this DB instance.

p-vpc (vpc-0fd9504fe1055ceef)
0 Subnets, 0 Availability Zones

Only VPCs with a corresponding DB subnet group are listed.

After a database is created, you can't change its VPC.

DB subnet group Info
Choose the DB subnet group. The DB subnet group defines which subnets and IP ranges the DB instance can use in the VPC that you selected.

Choose existing
Choose existing DB subnet group

Automatic setup
RDS creates a new subnet group for you or reuses an existing subnet group

Existing DB subnet groups
db subnet group
6 Subnets, 2 Availability Zones

Public access Info
Yes
RDS assigns a public IP address to the database. Amazon EC2 instances and other resources outside of the VPC can connect to your database. Resources inside the VPC can also connect to the database. Choose one or more VPC security groups that specify which

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.

VPC security group (firewall) Info
Choose one or more VPC security groups to allow access to your database. Make sure that the security group rules allow the appropriate incoming traffic.

Choose existing
Choose existing VPC security groups

Create new
Create new VPC security group

Additional VPC security group
Choose one or more options
private-sg X public-sg X

Amazon RDS will add a new VPC security group rds-ec2-2 to allow connectivity with your compute resource.

Availability Zone Info
us-west-1b

Certificate authority - optional Info
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-MsA2048-g1 (default)
Expires: May 20, 2061

If you don't select a certificate authority, RDS chooses one for you.

Additional configuration

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.

CloudShell Feedback

CloudShell provides a terminal window for running AWS Lambda functions and AWS Lambda layers. It includes a history of previous commands and a log viewer for monitoring function execution.

CloudShell is currently running in the CAD/INR environment. The current memory usage is +0.25%

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.

8.successfully database is created

The screenshot shows the AWS RDS Databases page. On the left, there's a sidebar with options like Dashboard, Databases, Query Editor, Performance insights, Snapshots, Exports in Amazon S3, Automated backups, Reserved instances, Proxies, Subnet groups, Parameter groups, Option groups, Zero-ETL Integrations, Events, and Event subscriptions. The main area is titled 'Databases (1)' and shows a table with one row. The row details are: DB identifier: database-1, Status: Available, Instance: MySQL Community, Region & ...: us-west-1b, Size: db.t3.micro. There's also a 'Create database' button.

➤ Take DB SNAPSHOT

1.Go to rds dashboard - select snapshot –click on take snapshot

2.select db instance and give a new name to snapshot

3.click on take snapshot

The screenshot shows the 'Take DB Snapshot' dialog box. It has a 'Preferences' section with instructions: 'To take a DB Snapshot, choose a database and name your DB Snapshot.' Below it, 'Snapshot type' is set to 'DB instance'. Under 'DB instance', 'DB instance identifier' is set to 'database-1'. In the 'Snapshot name' field, the value 'P-SNAPSHOT' is entered. At the bottom right is a 'Take snapshot' button.

4.successfully snapshot is created

The screenshot shows the AWS RDS Snapshots page. The 'Manual' tab is selected. A table titled 'Manual snapshots (1)' shows one entry: Snapshot name: p-snapshot, DB instance or cluster: database-1, Snapshot creation time: August 27, 2024, 20:48 (UTC+05:30), and DB instance created time: August 27, 2024, 20:41 (UTC+05:30). There's a 'Take snapshot' button at the top right of the table.

➤ MYSQL installation

1.Go to public instance web and give the following commands

- Sudo apt update -y
- Sudo apt install mysql –sever -y
- mysql -h database-1.cv2muissww5k.us-west-1.rds.amazonaws.com -u admin -p
(database-1.cv2muissww5k.us-west-1.rds.amazonaws.com – end point of database)

2.once you complete giving the commands it displays as following in below snapshots

```
Get:42 http://security.ubuntu.com/ubuntu noble-security/universe amd64 . Metadata (9756 B)
Get:43 http://security.ubuntu.com/ubuntu noble-security/universe amd64 Packages [2200 kB]
Get:44 http://security.ubuntu.com/ubuntu noble-security/restricted Translation-en [54.8 kB]
Get:45 http://security.ubuntu.com/ubuntu noble-security/multiverse amd64 Packages [10.6 kB]
Get:46 http://security.ubuntu.com/ubuntu noble-security/multiverse Translation-en [2808 B]
Get:47 http://security.ubuntu.com/ubuntu noble-security/multiverse amd64 Components [208 B]
Get:48 http://security.ubuntu.com/ubuntu noble-security/multiverse amd64 c-n-f Metadata [344 B]
Fetched 28.7 MB in 6s (4967 kB/s)
rc
ubuntu@10-0-2-142:~$ sudo apt install mysql-server
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
libbcgi-fast-perl libbcgi-pm-perl libclone-perl libcode-locale-perl libevent-pthreads-2.1-7t64 libfcgi-bin libfcgi-perl libfcgi0t64 libhtml-parser-perl
libhttplib-tagent-perl libhttp-template-perl libhttp-date-perl libhttp-message-perl libio-html-perl liblwp-mediatypes-perl libmecab2 libprotobuf-lite32t64
libmymysqlclient-perl liburi-perl mecab-ipadic mecab-ipadic-utf8 mecab-utils mysql-client-8.0 mysql-client-core-8.0 mysql-common mysql-server-8.0
mysql-client-core-8.0
Suggested packages:
libdata-dump-perl libipc-sharedcache-perl libipcre-compress-brotli-perl libbusiness-isbn perl libregexp-ipv6-perl libwww-perl mailx tinyca
The following NEW packages will be installed:
libbcgi-fast-perl libbcgi-pm-perl libclone-perl libevent-pthreads-2.1-7t64 libfcgi-bin libfcgi-perl libfcgi0t64 libhtml-parser-perl
libhttplib-tagent-perl libhttp-template-perl libhttp-date-perl libhttp-message-perl libio-html-perl liblwp-mediatypes-perl libmecab2 libprotobuf-lite32t64
libmymysqlclient-perl liburi-perl mecab-ipadic mecab-ipadic-utf8 mecab-utils mysql-client-8.0 mysql-client-core-8.0 mysql-common mysql-server mysql-server-8.0
mysql-server-core-8.0
0 upgraded, 29 newly installed, 0 to remove and 102 not upgraded.
Need to get 29.1 MB of archives.
After this operation, 242 MB of additional disk space will be used.
Do you want to continue? [Y/n] i

i-0e1007eb9ecbfae4a (public1)
PublicIPs: 54.215.185.246 PrivateIPs: 10.0.0.97
```



```
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.
ubuntu@10-0-2-142:~$ mysql -h database-1.cv2muissww5k.us-west-1.rds.amazonaws.com -u admin -p
Command 'mysql -h' not found, did you mean:
command mysql-h
See 'man info <commandname>' for additional versions.
ubuntu@10-0-2-142:~$ mysql -h database-1.cv2muissww5k.us-west-1.rds.amazonaws.com -u admin -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 28
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> i

i-0e1007eb9ecbfae4a (public1)
PublicIPs: 54.215.185.246 PrivateIPs: 10.0.0.97
```



THE END

