

3-TIER ARCHITECTURE (PROJECT)

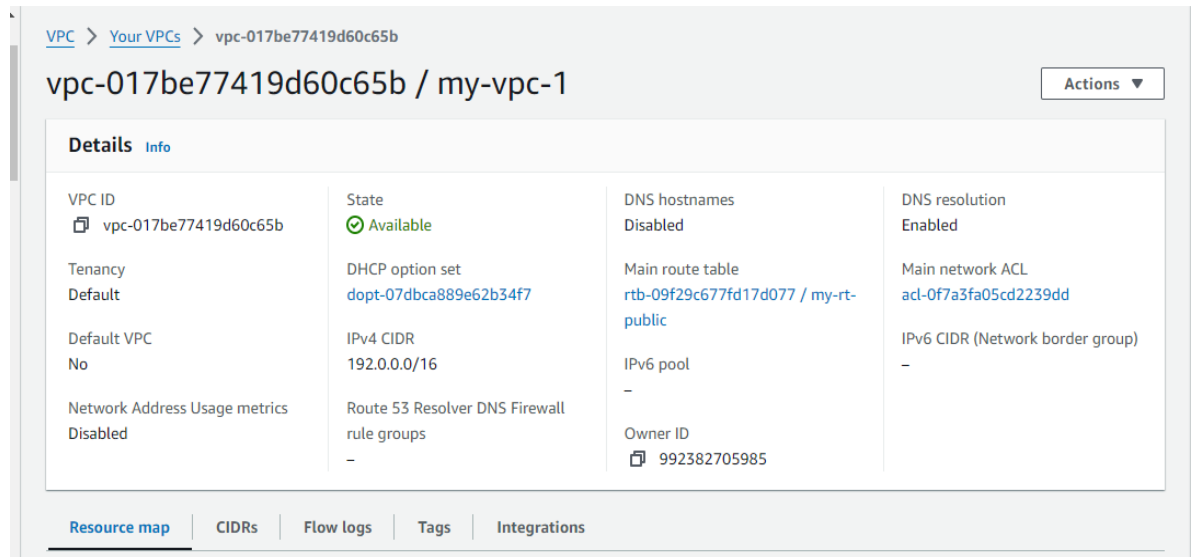
Name: Mylarapu Ravi Kumar

Batch: 119(7am)

Phone no: 6302540471

E-Mail id: ravimylarapu39@gmail.com

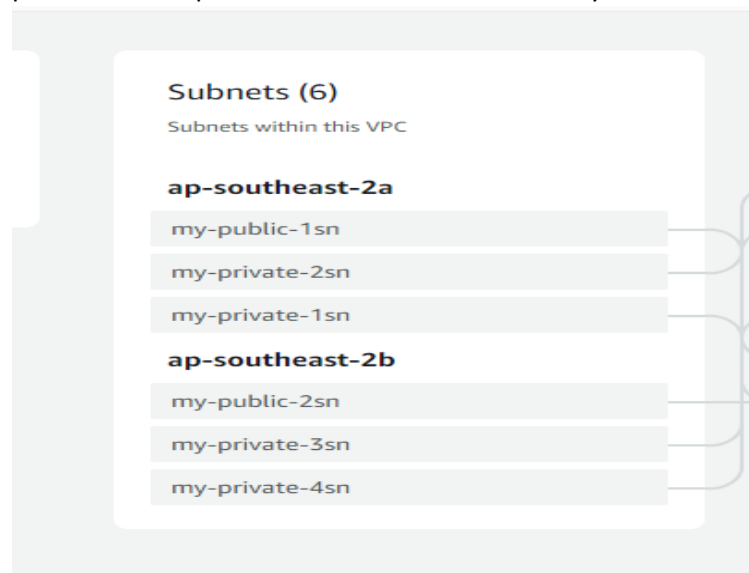
1. Create one new VPC



The screenshot shows the AWS VPC console for a VPC named 'vpc-017be77419d60c65b / my-vpc-1'. The 'Details' tab is selected, displaying a table of VPC configuration. The VPC is in an 'Available' state. It has a main route table 'rtb-09f29c677fd17d077 / my-rt-public' and a main network ACL 'acl-0f7a3fa05cd2239dd'. The VPC ID is 'vpc-017be77419d60c65b' and the Owner ID is '992382705985'.

Details			
VPC ID	State	DNS hostnames	DNS resolution
vpc-017be77419d60c65b	Available	Disabled	Enabled
Tenancy	DHCP option set	Main route table	Main network ACL
Default	dopt-07dbca889e62b34f7	rtb-09f29c677fd17d077 / my-rt-public	acl-0f7a3fa05cd2239dd
Default VPC	IPv4 CIDR	IPv6 pool	IPv6 CIDR (Network border group)
No	192.0.0.0/16	-	-
Network Address Usage metrics	Route 53 Resolver DNS Firewall rule groups	Owner ID	
Disabled	-	992382705985	

2. Create one public and two private subnets in one availability zone and also create one public and two private subnet in other availability zone



- Next create one internet gateway and attached to VPC

[VPC](#) > [Internet gateways](#) > igw-0527104185c5daabb

igw-0527104185c5daabb / my-igw1

Actions

Details Info

Internet gateway ID igw-0527104185c5daabb	State Attached	VPC ID vpc-017be77419d60c65b / my-vpc-1	Owner 992382705985
--	-------------------	--	-----------------------

Tags Manage tags

Search tags

Key	Value
Name	my-igw1

- Next create nat gateway and attach with private subnets

[VPC](#) > [Route tables](#) > [rtb-0e1faa92c34e7c2ea](#) > Edit routes

Edit routes

Destination	Target	Status	Propagated
192.0.0.0/16	local	Active	No
0.0.0.0/0	NAT Gateway	Active	No

Add route

Cancel Preview Save changes

- next go to edit subnet association and attach public subnets in public routes

[VPC](#) > [Route tables](#) > [rtb-09f29c677fd17d077](#) > Edit subnet associations

Edit subnet associations

Change which subnets are associated with this route table.

Available subnets (2/6)

Filter subnet associations

	Name	Subnet ID	IPv4 CIDR	IPv6 CIDR	Route table ID
<input checked="" type="checkbox"/>	my-public-1sn	subnet-0b99ffa9fda90213a	192.0.0.0/24	-	rtb-09f29c677fd17d077 / my-rt-
<input type="checkbox"/>	my-private-1sn	subnet-03e6faf68e68e05ba	192.0.1.0/24	-	rtb-0797365d55bcb2d77 / my-rt-
<input type="checkbox"/>	my-private-2sn	subnet-0cc0fb52977212258	192.0.4.0/24	-	rtb-0e1faa92c34e7c2ea / my-rt-
<input checked="" type="checkbox"/>	my-public-2sn	subnet-0ae392f23c40346be	192.0.5.0/24	-	rtb-09f29c677fd17d077 / my-rt-
<input type="checkbox"/>	my-private-3sn	subnet-060254ce91332ec81	192.0.6.0/24	-	rtb-06abd75cfffb0c02 / my-rt-
<input type="checkbox"/>	my-private-4sn	subnet-0873fef48cda59f8a	192.0.7.0/24	-	rtb-06a4d1f86f4e311cf / my-rt-

6.next go to public route table click the action button and set main route table and type set

Route tables (1/7) Info

Find resources by attribute or tag

	Name	Route table ID	Explicit subnet associ...
<input type="checkbox"/>	-	rtb-0761c185b1aeca872	-
<input type="checkbox"/>	-	rtb-0b121f0cef5caa20b	-
<input checked="" type="checkbox"/>	my-rt-public	rtb-09f29c677fd17d077	2 subnets
<input type="checkbox"/>	my-rt-private-1	rtb-0797365d55bcb2d77	subnet-03e6faf68e68e05...
<input type="checkbox"/>	my-rt-private-2	rtb-0e1faa92c34e7c2ea	subnet-0cc0fb52977712...

Actions

Create route table

View details

Set main route table

Edit subnet associations

Edit edge associations

Edit route propagation

Edit routes

Manage tags

Delete route table

7.next go to public subnet and click and edit subnet settings and click the enable auto assign public ipv4 address

VPC > Subnets > [subnet-0b99ffa9fda90213a](#) > Edit subnet settings

Edit subnet settings Info

Subnet

Subnet ID

subnet-0b99ffa9fda90213a

Name

my-public-1sn

Auto-assign IP settings Info

Enable AWS to automatically assign a public IPv4 or IPv6 address to a new primary network interface for an instance in this subnet.

☒ Enable auto-assign public IPv4 address Info

☐ Enable auto-assign customer-owned IPv4 address Info

Option disabled because no customer owned pools found.

Resource-based name (RBN) settings Info

Specify the hostname type for EC2 instances in this subnet and optional RBN DNS query settings.

Web tier:

1. Create new instance template

EC2 > [Launch templates](#) > my-template1

my-template1 (lt-087157bc8a38aec82)

Actions ▾ Delete template

Launch template details

Launch template ID lt-087157bc8a38aec82	Launch template name my-template1	Default version 1	Owner arn:aws:iam::992382705985:root
--	--------------------------------------	----------------------	---

Details Versions Template tags

Launch template version details

Actions ▾ Delete template version

Version 1 (Default) ▾	Description -	Date created 2024-03-07T04:52:59.000Z	Created by arn:aws:iam::992382705985:root
--------------------------	------------------	--	--

2. Next create auto scaling groups and follow the steps

EC2 > [Auto Scaling groups](#) > my-scalling

my-scalling

Details Activity Automatic scaling Instance management Monitoring Instance refresh

Group details Edit

Auto Scaling group name my-scalling	Desired capacity 2	Desired capacity type Units (number of instances)	Amazon Resource Name (ARN) arn:aws:autoscaling:ap-southeast-2:992382705985:autoScalingGroup:31138feb-7502-4f27-85ea-88663bd70ef3:autoScalingGroupName/my-scalling
Date created Thu Mar 07 2024 10:24:42 GMT+0530 (India Standard Time)	Minimum capacity 2	Status -	
	Maximum capacity 5		

3. Next create instance security groups

EC2 > [Security Groups](#) > sg-0246afec850f47888 - my-sg-1 > Edit inbound rules

Edit inbound rules [Info](#)

Inbound rules control the incoming traffic that's allowed to reach the instance.

Inbound rules [Info](#)

Security group rule ID	Type Info	Protocol Info	Port range Info	Source Info	Description - optional Info	
sg-r01529332653c61903	SSH ▾	TCP	22	Cus... ▾ 0.0.0.0/0 ✕		Delete
sg-r0a2d7feb23dc64fd9	HTTP ▾	TCP	80	Cus... ▾ 0.0.0.0/0 ✕		Delete

Add rule

4. After creating auto scaling now create instance automatically

Instances (2) Info

Refresh

Connect

Instance state

Actions

Launch instances

Find Instance by attribute or tag (case-sensitive)

Instance state = running

Clear filters

Any state

	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability
		i-001f3ba4ca727cd03	Running	t2.micro	2/2 checks passed	View alarms	ap-southea
		i-02740648e7fceec18	Running	t2.micro	2/2 checks passed	View alarms	ap-southea

5.next connect to one instance

ap-southeast-2.console.aws.amazon.com/ec2-instance-connect/ssh?region=ap-southeast-2&connType=standard&instanceId=i-001f3ba4ca727cd03&osUser=ec2-user...

Services

VPC

EC2

RDS

S3

IAM

Lambda

Amazon Linux 2023

https://aws.amazon.com/linux/amazon-linux-2023

last login: Thu Mar 7 04:56:07 2024 from 13.239.158.5

ec2-user@ip-192-0-5-176 ~]\$ sudo -i

root@ip-192-0-5-176 ~]#

i-001f3ba4ca727cd03

PublicIPs: 54.253.17.159 PrivateIPs: 192.0.5.176

Application tier

1.first create instance template 2

EC2 > Launch templates > my-template2

my-template2 (lt-0b0d32b01cd54cc0b)

Actions

Delete template

Launch template details

Launch template ID	Launch template name	Default version	Owner
lt-0b0d32b01cd54cc0b	my-template2	1	arn:aws:iam::992382705985:root

Details

Versions

Template tags

Launch template version details

Actions

Delete template version

Version	Description	Date created	Created by
1 (Default)	-	2024-03-07T05:21:08.000Z	arn:aws:iam::992382705985:root

2.next create auto scaling by creating load balancer

EC2 > Auto Scaling groups > my-scal-2

my-scal-2

Details | Activity | Automatic scaling | Instance management | Monitoring | Instance refresh

Group details Edit

Auto Scaling group name my-scal-2	Desired capacity 2	Desired capacity type Units (number of instances)	Amazon Resource Name (ARN) arn:aws:autoscaling:ap-southeast-2:992382705985:autoScalingGroup:2b94ea75-abf9-4062-a0e3-fcb5632f524b:autoScalingGroupName/my-scal-2
Date created Thu Mar 07 2024 10:53:50 GMT+0530 (India Standard Time)	Minimum capacity 2	Status Updating capacity	
	Maximum capacity 5		

3. we can see that now we can see two new instance (pri1, pri2)running automatically with attached private subnets

Instances (5) Info Refresh Connect Instance state Actions Launch instances

Find Instance by attribute or tag (case-sensitive)

Any state

<input type="checkbox"/>	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability
<input type="checkbox"/>	pri1	i-087b175a46953f52d	Running	t2.micro	2/2 checks passed	View alarms	ap-southeast-2
<input type="checkbox"/>		i-001f3ba4ca727cd03	Running	t2.micro	2/2 checks passed	View alarms	ap-southeast-2
<input type="checkbox"/>	pri2	i-0610ac787a53eb5b0	Running	t2.micro	2/2 checks passed	View alarms	ap-southeast-2
<input type="checkbox"/>		i-03a93afbfb6e130c4	Terminated	t2.micro	-	View alarms	ap-southeast-2
<input type="checkbox"/>		i-02740648e7fcec18	Running	t2.micro	2/2 checks passed	View alarms	ap-southeast-2

4.to connect pri 1 instance we need to connect public instance and do the following task

```
aws
Services
Search [Alt+S]
VPC EC2 RDS S3 IAM Lambda
[root@ip-192-0-5-176 ~]# vi key.pem
[root@ip-192-0-5-176 ~]# ls
key.pem
[root@ip-192-0-5-176 ~]# chmod 400 "key.pem"
[root@ip-192-0-5-176 ~]# ssh -i "key.pem" ec2-user@192.0.1.149
Amazon Linux 2023
https://aws.amazon.com/linux/amazon-linux-2023
[ec2-user@ip-192-0-1-149 ~]$
```

i-001f3ba4ca727cd03

PublicIPs: 54.253.17.159 PrivateIPs: 192.0.5.176

Database tier:

1. Create security groups

[EC2](#) > [Security Groups](#) > sg-01884de618d0fe742 - my-sg-3

sg-01884de618d0fe742 - my-sg-3 Actions ▼

Details

Security group name my-sg-3	Security group ID sg-01884de618d0fe742	Description allow ssh	VPC ID vpc-017be77419d60c65b
Owner 992382705985	Inbound rules count 3 Permission entries	Outbound rules count 1 Permission entry	

[Inbound rules](#) | [Outbound rules](#) | [Tags](#)

2. Next create Relational Database subnet groups

[RDS](#) > [Subnet groups](#) > my-sn -groups-1

my-sn -groups-1

Subnet group details

VPC ID vpc-017be77419d60c65b
ARN arn:aws:rds:ap-southeast-2:992382705985:subgrp:my-sn -groups-1
Supported network types IPv4
Description fkfdkh

Subnets (2)

3.next create a Database following steps

RDS > Databases > database-1

database-1

🔄

Modify

Actions ▾

Summary

DB identifier database-1	Status 🟢 Available	Role Instance	Engine MySQL Community	Recommendations
CPU 📊 5.59%	Class db.t2.micro	Current activity 📊 0	Region & AZ ap-southeast-2a	
Connections				

<

Connectivity & security

Monitoring

Logs & events

Configuration

Maintenance & backups

Tags

R

>

Connectivity & security

4.connect to an instance doing following steps

aws Services 🔍 Search [Alt+S] Sydney mylarapu ravi ▾

VPC EC2 RDS S3 IAM Lambda

```
Amazon Linux 2023
https://aws.amazon.com/linux/amazon-linux-2023

Last login: Thu Mar  7 04:56:47 2024 from 13.239.158.3
[ec2-user@ip-192-0-5-176 ~]$ sudo dnf update -y
Last metadata expiration check: 1:40:51 ago on Thu Mar  7 04:55:43 2024.
Dependencies resolved.
Nothing to do.
Complete!
```

5.install MariaDB 105

```
[ec2-user@ip-192-0-5-176 ~]$ sudo dnf install mariadb105
Last metadata expiration check: 1:41:14 ago on Thu Mar  7 04:55:43 2024.
Dependencies resolved.
```

Package	Architecture	Version	Repository	Size
Installing:				
mariadb105	x86_64	3:10.5.23-1.amzn2023.0.1	amazonlinux	1.6 M
Installing dependencies:				
mariadb-connector-c	x86_64	3.1.13-1.amzn2023.0.3	amazonlinux	196 k
mariadb-connector-c-config	noarch	3.1.13-1.amzn2023.0.3	amazonlinux	9.2 k
mariadb105-common	x86_64	3:10.5.23-1.amzn2023.0.1	amazonlinux	30 k
perl-Sys-Hostname	x86_64	1.23-477.amzn2023.0.6	amazonlinux	18 k

Transaction Summary

i-001f3ba4ca727cd03

PublicIPs: 54.253.17.159 PrivateIPs: 192.0.5.176

Final overview

