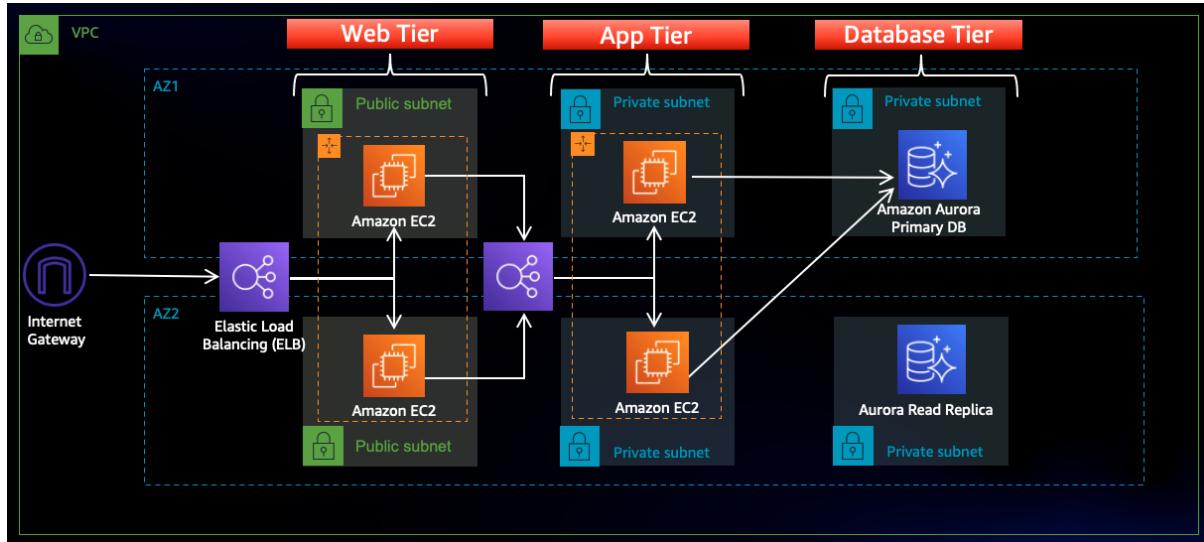


Building Scalable, Secure, and High-Performance Web Applications with AWS 3-Tier Architecture

Architecture Overview



In today's digital age, building a scalable and secure web application is crucial for business success. The [AWS 3-Tier Web Application Architecture](#) stands out as a best practice for deploying cloud-based applications with efficiency and reliability. By leveraging [Amazon Web Services \(AWS\)](#) to create a three-tier architecture, businesses can achieve enhanced scalability, security, and performance. This architecture divides the application into three layers: the Web Tier, Application Tier, and Database Tier — each designed to handle specific tasks, ensuring seamless operation and optimal user experience. In this blog post, we'll delve into how the AWS 3-Tier architecture can be leveraged to build resilient and high-performing web applications. You can be a Automation Geek, checkout my [2 tier architecture](#) in AWS using Terraform.



Practical Implementation

Step 1: Start by creating an S3 bucket which will hold the code for you tier.

Create bucket Info

Buckets are containers for data stored in S3. [Learn more](#)

General configuration

Bucket name

Bucket name must be unique within the global namespace and follow the bucket naming rules. [See rules for bucket naming](#)

AWS Region

▼

Copy settings from existing bucket - *optional*
Only the bucket settings in the following configuration are copied.

[Choose bucket](#)

STEP 2: Create an instance role for ec2 service, allow access to S3 but only restricted read access.

AWS service

Allow AWS services like EC2, Lambda, or others to perform actions in this account.

AWS account

Allow entities in other AWS accounts belonging to you or a 3rd party to perform actions in this account.

Web identity

Allows users federated by the specified external web identity provider to assume this role to perform actions in this account.

SAML 2.0 federation

Allow users federated with SAML 2.0 from a corporate directory to perform actions in this account.

Custom trust policy

Create a custom trust policy to enable others to perform actions in this account.

Use case

Allow an AWS service like EC2, Lambda, or others to perform actions in this account.

Service or use case

EC2

```
12
13
14
15
16 }
```

Step 2: Add permissions

Edit

Permissions policy summary

Policy name

Type

Attached as

[AmazonS3ReadOnlyAccess](#)

AWS managed

Permissions policy

[AmazonSSMManagedInstanceCore](#)

AWS managed

Permissions policy

Step 3: Add tags

STEP 3: Create an Separate VPC for this architecture, Separate VPC provides network Isolation.

Create VPC Info

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

VPC settings

Resources to create Info

Create only the VPC resource or the VPC and other networking resources.

VPC only

VPC and more

Name tag - *optional*

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

my-3tier-priyanshu

IPv4 CIDR block Info

- IPv4 CIDR manual input
 IPAM-allocated IPv4 CIDR block

IPv4 CIDR

10.0.0.0/16

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

IPv6 CIDR block Info

- No IPv6 CIDR block
 IPAM-allocated IPv6 CIDR block

STEP 4: Create Subnets in different AZ, to create a multi AZ setup. We Will be having two Public and Two private subnets.

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 [Info](#)

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



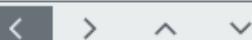
IPv4 VPC CIDR block [Info](#)

Choose the IPv4 VPC CIDR block to create a subnet in.



IPv4 subnet CIDR block

256 IPs



▼ Tags - optional

Key

Value - optional

Subnet 2 of 2

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 [Info](#)

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



IPv4 VPC CIDR block [Info](#)

Choose the IPv4 VPC CIDR block to create a subnet in.



IPv4 subnet CIDR block

256 IPs



▼ Tags - optional

Subnet 3 of 3

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 [Info](#)

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



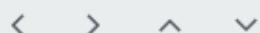
IPv4 VPC CIDR block [Info](#)

Choose the IPv4 VPC CIDR block to create a subnet in.



IPv4 subnet CIDR block

256 IPs



▼ Tags - optional

Key

Value - optional



Subnet 4 of 4

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 [Info](#)

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



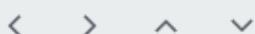
IPv4 VPC CIDR block [Info](#)

Choose the IPv4 VPC CIDR block to create a subnet in.



IPv4 subnet CIDR block

256 IPs



▼ Tags - optional

Key

Value - optional



STEP 5: Create a 5th and 6th subnet for the private Database, again in multi AZ.

Subnet 5 of 5

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 [Info](#)

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



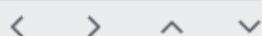
IPv4 VPC CIDR block [Info](#)

Choose the IPv4 VPC CIDR block to create a subnet in.



IPv4 subnet CIDR block

256 IPs



▼ Tags - *optional*

Key

Value - *optional*



Subnet 6 of 6

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 [Info](#)

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



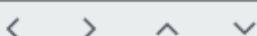
IPv4 VPC CIDR block [Info](#)

Choose the IPv4 VPC CIDR block to create a subnet in.



IPv4 subnet CIDR block

256 IPs



▼ Tags - optional

Key

Value - optional



After all the configuration we'll be having 6 subnets, As shown below:

<input type="checkbox"/>	Name	Subnet ID	State
<input type="checkbox"/>	DB-private-Subnet-AZ2	subnet-0d91f1533863a65c5	Available
<input type="checkbox"/>	DB-private-subnet-AZ1	subnet-016c28dc844de2a02	Available
<input type="checkbox"/>	public-web-subnet-az2	subnet-001a75db22f70898d	Available
<input type="checkbox"/>	public-web-subnet-AZ1	subnet-076df2f16efa400e7	Available
<input type="checkbox"/>	private-subnet-AZ2	subnet-071856a3d56840d6e	Available
<input type="checkbox"/>	private-subnet-AZ1	subnet-0e5a9b907a0f3d15b	Available

STEP 6: Create an internet Gateway, IGW is used to route traffic from internet to instance and vice versa, if all the network restrictions are allowed.

Create internet gateway Info

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

Internet gateway settings

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

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

Key	Value - optional
<input type="text" value="Name"/>	<input type="text" value="my-3tier-priyanshu-igw"/> X

Add new tag
You can add 49 more tags.

Cancel Create internet gateway

STEP 7: Attach this to the VPC which we just created above.

igw-0321cd02f0d95722b / my-3tier-priyanshu-igw

Actions ▾			
Attach to VPC			
Detach from VPC			
Manage tags			
Delete			

Details Info

Internet gateway ID <input type="text" value="igw-0321cd02f0d95722b"/>	State Detached	VPC ID -	Owner <input type="text" value="840260673873"/>
---	--------------------------------	-------------	--

Attach to VPC (igw-0321cd02f0d95722b) Info

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.

<input type="text" value="vpc-0d10d036dac169e48"/> X
Use: "vpc-0d10d036dac169e48"
vpc-0d10d036dac169e48 - my-3tier-priyanshu

Cancel Attach internet gateway

STEP 8: Create NAT Gateway, NAT in simpler allows instance to access network outside the instance, but doesn't allow external Traffic to come inside the instance. As its A subnet level resource we attach the first NAT to the public subnet 1 in AZ1.

NAT gateway settings

Name - optional

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

The name can be up to 256 characters long.

Subnet

Select a subnet in which to create the NAT gateway.



Connectivity type

Select a connectivity type for the NAT gateway.

- Public
- Private

Elastic IP allocation ID Info

Assign an Elastic IP address to the NAT gateway.



► Additional settings Info

STEP 9: Create a second NAT Gateway and attach it to the second public AZ subnet.

NAT gateway settings

Name - optional

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

The name can be up to 256 characters long.

Subnet

Select a subnet in which to create the NAT gateway.



Connectivity type

Select a connectivity type for the NAT gateway.

- Public
- Private

Elastic IP allocation ID Info

Assign an Elastic IP address to the NAT gateway.



Name	NAT gateway ID	Connectivit...	State
AZ2-NATGW-3tier	nat-0d4f8bf1e9b67c5da	Public	<input checked="" type="checkbox"/> Available
AZ1-NATGW-3tier	nat-07c1144ab752bef3e	Public	<input checked="" type="checkbox"/> Available

STEP 10: Create the route Tables with the specified subnet and routes.

Route table settings

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

VPC
The VPC to use for this route table.

Tags

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

Key	Value - optional
<input type="text" value="Name"/>	<input type="text" value="my-Public-3tier-rt"/> X

Add new tag

You can add 49 more tags.

Cancel Create route table

STEP 11: Edit route table for public destination internet and source IGW.

Edit routes

Destination	Target	Status	Propagated
10.0.0.0/16	<input type="text" value="local"/> X	Active	No
<input type="text" value="0.0.0.0"/> X	<input type="text" value="Internet Gateway"/> X	-	No Remove
	<input type="text" value="igw-0321cd02f0d95722b"/> X		
	Use: *igw-0321cd02f0d95722b*		
	igw-0321cd02f0d95722b (my-3tier-priyanshu-igw)		

Add route

Cancel Preview Save changes

STEP 12: Subnet Association with the public subnet.

my-Public-3tier-rt	rtb-047402d9d0ff28ec8	-	-	No								
<ul style="list-style-type: none"> Details Routes Subnet associations Edge associations Route propagation Tags 												
Explicit subnet associations (0) <p>Edit subnet associations</p> <table border="1"> <thead> <tr> <th colspan="2">Find subnet association</th> <th>< 1 ></th> <th>Reset</th> </tr> </thead> <tbody> <tr> <td>Name</td> <td>Subnet ID</td> <td>IPv4 CIDR</td> <td>IPv6 CIDR</td> </tr> </tbody> </table>					Find subnet association		< 1 >	Reset	Name	Subnet ID	IPv4 CIDR	IPv6 CIDR
Find subnet association		< 1 >	Reset									
Name	Subnet ID	IPv4 CIDR	IPv6 CIDR									

Available subnets (2/6)

Available subnets (2/6)						
<input type="text"/> Filter subnet associations						
Name	Subnet ID	IPv4 CIDR	IPv6 CIDR	Route table ID		
DB-private-Subnet-AZ2	subnet-0d91f1533863a65c5	10.0.5.0/24	-	Main (rtb-0b1c49f967034c92e)		
DB-private-subnet-AZ1	subnet-016c28dc844de2a02	10.0.4.0/24	-	Main (rtb-0b1c49f967034c92e)		
<input checked="" type="checkbox"/> public-web-subnet-az2	subnet-001a75db22f70898d	10.0.1.0/24	-	Main (rtb-0b1c49f967034c92e)		
<input checked="" type="checkbox"/> public-web-subnet-AZ1	subnet-076df2f16efa400e7	10.0.0.0/24	-	Main (rtb-0b1c49f967034c92e)		
private-subnet-AZ2	subnet-071856a3d56840d6e	10.0.3.0/24	-	Main (rtb-0b1c49f967034c92e)		
private-subnet-AZ1	subnet-0e5a9b907a0f3d15b	10.0.2.0/24	-	Main (rtb-0b1c49f967034c92e)		

Selected subnets

subnet-001a75db22f70898d / public-web-subnet-az2 <input type="button" value="X"/>	subnet-076df2f16efa400e7 / public-web-subnet-AZ1 <input type="button" value="X"/>
---	---

Create route table for the private subnet, but this time in routes add Nat Gateway, Private Subnet.

Route table settings

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

VPC
The VPC to use for this route table.

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

Key	Value - optional
<input type="text" value="Name"/>	<input type="text" value="my-Private-3tier-rt"/> <input type="button" value="X"/> <input type="button" value="Remove"/>
<input type="button" value="Add new tag"/>	

You can add 49 more tags.

Available subnets (1/6)

Available subnets (1/6)						
<input type="text"/> Filter subnet associations						
Name	Subnet ID	IPv4 CIDR	IPv6 CIDR	Route table ID		
DB-private-Subnet-AZ2	subnet-0d91f1533863a65c5	10.0.5.0/24	-	Main (rtb-0b1c49f967034c92e)		
DB-private-subnet-AZ1	subnet-016c28dc844de2a02	10.0.4.0/24	-	Main (rtb-0b1c49f967034c92e)		
<input type="checkbox"/> public-web-subnet-az2	subnet-001a75db22f70898d	10.0.1.0/24	-	rtb-047402d9d0ff28ec8 / my-Publi		
<input type="checkbox"/> public-web-subnet-AZ1	subnet-076df2f16efa400e7	10.0.0.0/24	-	rtb-047402d9d0ff28ec8 / my-Publi		
private-subnet-AZ2	subnet-071856a3d56840d6e	10.0.3.0/24	-	rtb-0c7a1099f25617b62 / my-Priva		
<input checked="" type="checkbox"/> private-subnet-AZ1	subnet-0e5a9b907a0f3d15b	10.0.2.0/24	-	rtb-0c7a1099f25617b62 / my-Priva		

Selected subnets

subnet-0e5a9b907a0f3d15b / private-subnet-AZ1 <input type="button" value="X"/>
--

Edit subnet associations

Change which subnets are associated with this route table.

Available subnets (2/6)

Name	Subnet ID	IPv4 CIDR	IPv6 CIDR	Route table ID
DB-private-Subnet-AZ2	subnet-0d91f1533863a65c5	10.0.5.0/24	-	Main (rtb-0b1c49f967034c92e)
DB-private-subnet-AZ1	subnet-016c28dc844de2a02	10.0.4.0/24	-	Main (rtb-0b1c49f967034c92e)
public-web-subnet-az2	subnet-001a75db22f70898d	10.0.1.0/24	-	rtb-047402d9d0ff28ec8 / my-Pub
public-web-subnet-AZ1	subnet-076df2f16ef400e7	10.0.0.0/24	-	rtb-047402d9d0ff28ec8 / my-Pub
<input checked="" type="checkbox"/> private-subnet-AZ2	subnet-071856a3d56840d6e	10.0.3.0/24	-	Main (rtb-0b1c49f967034c92e)
<input checked="" type="checkbox"/> private-subnet-AZ1	subnet-0e5a9b907a0f3d15b	10.0.2.0/24	-	Main (rtb-0b1c49f967034c92e)

Selected subnets

subnet-071856a3d56840d6e / private-subnet-AZ2 X
subnet-0e5a9b907a0f3d15b / private-subnet-AZ1 X

Cancel Save associations

Edit routes

Destination	Target	Status	Propagated
10.0.0.0/16	local Q local	<input checked="" type="radio"/> Active	No
Q 0.0.0.0/0	NAT Gateway Q nat-07c1144ab752bef3e	-	No Remove

Add route

Cancel Preview Save changes

CREATING SECURITY GROUPS

In this section we will be creating security groups for all requirements. Security groups play an integral part of the AWS security best practices, Regulating them can be beneficial for your overall security.

1. ALB security group:

This allows internet traffic to reach your ALB.

Security group name Info

Name cannot be edited after creation.

Description Info

VPC Info

Inbound rules Info

Type Info	Protocol Info	Port range Info	Source Info	Description - optional Info
HTTP	TCP	80	Any... <input type="button" value="X"/>	<input type="text" value="0.0.0.0/0"/> <input type="button" value="Delete"/>
HTTP	TCP	80	Any... <input type="button" value="X"/>	<input type="text" value="::/0"/> <input type="button" value="Delete"/>

2. Web Tier Security Group:

This allows traffic from ALB to web Tier.

Security group name Info

Name cannot be edited after creation.

Description Info

VPC Info

Inbound rules Info

Type Info	Protocol Info	Port range Info	Source Info	Description - optional Info
HTTP	TCP	80	Cus... <input type="button" value="X"/>	<input type="text" value="sg-0e4d7dfb095a98e4a"/> <input type="button" value="Delete"/>

3. Internal LB SG:

There's an Internal ALB security group, which internally used for networking.

Security group name info
 Name cannot be edited after creation.

Description info

VPC Info

Inbound rules info

Type info	Protocol info	Port range info	Source info	Description - optional info
HTTP	TCP	80	<input type="text" value="Cus..."/> <input type="button" value="X"/>	<input type="text" value="Q_ vpc-0d10d036dac169e48"/> <input type="button" value="X"/>
<input type="text" value="sg-01f7efe2cc326ac2"/> <input type="button" value="X"/>				
<input type="button" value="Delete"/>				

4. SG for private instance:

These allows only Certain IP(MY IP) to access the private instance.

Security group name info
 Name cannot be edited after creation.

Description info

VPC Info

Inbound rules info

Type info	Protocol info	Port range info	Source info	Description - optional info
Custom TCP	TCP	4000	<input type="text" value="Cus..."/> <input type="button" value="X"/>	<input type="text" value="Q_ ::/64"/> <input type="button" value="X"/>
Custom TCP	TCP	4000	<input type="text" value="Cus..."/> <input type="button" value="X"/>	<input type="text" value="This is my IP"/> <input type="button" value="X"/>
<input type="button" value="Delete"/>				

5. DB private SG:

This allows the DB to be accessible from the instance.

The screenshot shows the AWS Subnet Group configuration page. At the top, there is a 'Name' field containing 'DB-sg-3tier' with a note: 'Name cannot be edited after creation.' Below it is a 'Description' field with the value 'DB-sg-3tier'. Under 'VPC Info', a search bar shows 'vpc-0d10d036dac169e48'. The main section is titled 'Inbound rules' and contains a table with columns: Type, Protocol, Port range, Source info, and Description - optional. A rule is listed: Type is 'MySQL/Aurora', Protocol is 'TCP', Port range is '3306', Source info is 'Cus...', and Description is 'sg-09bbca10323a5d 565'. An 'Add rule' button is at the bottom left.

Creating Database

In this Setup We are using RDS as a Database, In This secrion we'll be doing configurations related to the RDS intance.

STEP 1: Creating Subnet Group for RDS

Exports in Amazon S3

Automated backups

Reserved instances

Proxies

Subnet groups

Parameter groups

Option groups

Custom engine versions

Events

my-3tier-sb-sg

VPC

Choose a VPC identifier that corresponds to the subnets you want to use for your DB subnet group. You can choose a different VPC identifier after your subnet group has been created.

my-3tier-priyanshu (vpc-0d10d036dac169e48)



Add subnets

Availability Zones

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

Choose an availability zone



ap-south-1a X

ap-south-1b X

Subnets

Choose the subnets that you want to add. The list includes the subnets in the selected Availability Zone.

Select subnets



subnet-0d91f1533863a65c5 (10.0.5.0/24) X

subnet-016c28dc844de2a02 (10.0.4.0/24) X

STEP 2: After Subnet Group create Database. We'll be creating a minimal setup of the RDS database as this is for testing purpose, you should not use this for a production setup.

Dashboard

Databases

Query Editor

Performance insights

Snapshots

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 [Info](#)

- Aurora (MySQL Compatible)



- Aurora (PostgreSQL Compatible)



- MySQL



- MariaDB



- PostgreSQL



- Oracle



2. Using the Dev/Test version for lesser costing.

- Production

Use defaults for high availability and fast, consistent performance.

- Dev/Test

This instance is intended for development use outside of a production environment.

Settings

DB cluster identifier [Info](#)

Enter a name for your DB cluster. The name must be unique across all DB clusters owned by your AWS account in the current AWS Region.

database-1

The DB cluster identifier is case-insensitive, but is stored as all lowercase (as in "mydbcluster"). 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

STEP 3: Create a solid Master password for your database access.

Master password [Info](#)

.....

Constraints: At least 8 printable ASCII characters. Can't contain any of the following: / (slash), '(single quote), "(double quote) and @ (at sign).

Confirm master password [Info](#)

.....

Cluster storage configuration - new [Info](#)

Choose the storage configuration for the Aurora DB cluster that best fits your application's price predictability and price performance needs.

Configuration options

Database instance, storage, and I/O charges vary depending on the configuration. [Learn more](#)

Aurora Standard

- Cost-effective pricing for many applications with moderate I/O usage (I/O costs <25% of total database costs).
- Pay-per-request I/O charges apply. DB instance and storage prices don't include I/O usage.

Aurora I/O-Optimized

- Predictable pricing for all applications. Improved price performance for I/O-intensive applications (I/O costs >25% of total database costs).
- No additional charges for read/write I/O operations. DB instance and storage prices include I/O usage.

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

DB instance class [Info](#)

Serverless v2

Memory optimized classes (includes r classes)

Burstable classes (includes t classes)

db.r5.2xlarge ▼

8 vCPUs 64 GiB RAM Network: 4,750 Mbps

Include previous generation classes

STEP 4: Disable the ec2 connect options as we'll be managing that independently using security groups.

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.

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.

Virtual private cloud (VPC) [Info](#)

Choose the VPC. The VPC defines the virtual networking environment for this DB cluster.

my-3tier-priyanshu (vpc-0d10d036dac169e48)

6 Subnets, 2 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 cluster can use in the VPC that you selected.

my-3tier-sb-sg

2 Subnets, 2 Availability Zones



STEP 5: Attach the VPC and security group for the database that we just created in the security group section.

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

Existing VPC security groups

Choose one or more options



DB-sg-3tier

As we have used most of the things for demo, This setup can cost you for database. You can fine tune more settings for reducing you infra cost.

DB instance 963.60 USD
Total 963.60 USD

This billing estimate is based on on-demand usage as described in [Amazon Aurora Pricing](#). Estimate does not consider reserved instance benefits and costs for instance storage, IOs, or data transfer.

Estimate your monthly costs for the DB Instance using the [AWS Simple Monthly Calculator](#).

ⓘ You are responsible for ensuring that you have all of the necessary rights for any third-party products or services that you use with AWS services.

Cancel **Create database**

Wait for some time and you can see your database instance:

Databases (3)		<input checked="" type="checkbox"/> Group resources		
		<input type="text"/> Filter by databases		
	DB identifier	Status	Role	Engine
<input type="radio"/>	database-1	Available	Regional cluster	Aurora
<input type="radio"/>	database-1-instance-1	Available	Writer instance	Aurora
<input type="radio"/>	database-1-instance-1-ap-south-1b	Available	Reader instance	Aurora

Instance Management

STEP 1: Create first instance in the private subnet with no public IP, attach the created security group to it.

▼ Network settings [Info](#)

VPC - required [Info](#)

vpc-0d10d036dac169e48 (my-3tier-priyanshu)
10.0.0.0/16



Subnet [Info](#)

subnet-0e5a9b907a0f3d15b private-subnet-AZ1
VPC: vpc-0d10d036dac169e48 Owner: 840260673873
Availability Zone: ap-south-1a IP addresses available: 251 CIDR: 10.0.2.0/24)



[Create new subnet](#) [

Auto-assign public IP [Info](#)

Disable

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.

[Create security group](#)

[Select existing security group](#)

Common security groups [Info](#)

Select security groups

my-private-instance-sg-3tier sg-09bbca10323a5d565 X
VPC: vpc-0d10d036dac169e48



[Compare security group rules](#)

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

STEP 2: Attach an role we just created in the first part.

▼ Advanced details [Info](#)

Purchasing option [Info](#)

Request Spot Instances

Domain join directory [Info](#)

Select

[Create new directory](#)

IAM instance profile [Info](#)

my-ec2-trustedrole

arn:aws:iam::840260673873:instance-profile/my-ec2-trustedrole



[Create new IAM prof](#)

Hostname type [Info](#)

IP name

We'll be connecting this instance using the session manager, for better management of the instance accessibility without using the keys, we can use the session manager.

Connect to instance Info

Connect to your instance i-04e3e8b60119c0a79 (WebAPP-3Tier) using any of these options

EC2 Instance Connect **Session Manager** SSH client EC2 serial console

Session Manager usage:

- Connect to your instance without SSH keys, a bastion host, or opening any inbound ports.
- Sessions are secured using an AWS Key Management Service key.
- You can log session commands and details in an Amazon S3 bucket or CloudWatch Logs log group.
- Configure sessions on the Session Manager [Preferences](#) page.

Cancel

Connect

STEP 3: For Testing purpose, Download the mysql dependencies for testing your RDS Database accessibility.

```
[root@ip-10-0-2-134 ~]# sudo yum install https://dev.mysql.com/get/mysql57-community-release-el7-11.noarch.rpm
Last metadata expiration check: 0:14:53 ago on Tue Oct 17 11:54:53 2023.
mysql57-community-release-el7-11.noarch.rpm          66 kB/s |  25 kB     00:00
Dependencies resolved.
=====
 Package           Architecture Version       Repository      Size
=====
Installing:
 mysql57-community-release      noarch        el7-11        @commandline   25 k
Transaction Summary
=====
Install 1 Package

Total size: 25 k
Installed size: 31 k
Is this ok [y/N]: y
Downloading Packages:
Running transaction check
Transaction check succeeded.
Running transaction test
Transaction test succeeded.
Running transaction
Preparing :                                         1/1
```

STEP 4: Copy the writer endpoint from the RDS service page.

The screenshot shows the AWS RDS console. At the top, there's a navigation bar with tabs for Connectivity & security, Monitoring, Logs & events, and Config. The Connectivity & security tab is selected. Below the tabs, there's a table with two rows. The first row has a blue dot icon, the text 'database-1-instance-1', a green checkmark icon labeled 'Available', and the word 'Writer'. The second row has a grey circle icon, the text 'database-1-instance-1-ap-south-1b', a green checkmark icon labeled 'Available', and the word 'Reader'. Below the table, the 'Connectivity & security' tab is highlighted in blue. The main content area is titled 'Connectivity & security'. It contains two sections: 'Endpoint & port' and 'Networking'. In the 'Endpoint & port' section, under 'Endpoint', the text 'database-1-instance-1.c9inyukwf1e8.ap-south-1.rds.amazonaws.com' is displayed, with the entire URL highlighted in yellow. In the 'Networking' section, under 'Availability Zone', the text 'ap-south-1a' is displayed. Under 'VPC', the text 'VPC' is displayed.

●	database-1-instance-1	Available	Writer
○	database-1-instance-1-ap-south-1b	Available	Reader

Connectivity & security Monitoring Logs & events Config

Connectivity & security

Endpoint & port	Networking
Endpoint database-1-instance-1.c9inyukwf1e8.ap-south-1.rds.amazonaws.com	Availability Zone ap-south-1a VPC

STEP 5: login the database using the endpoint. If you are unable to login then you must check your database security group settings.

```
[root@ip-10-0-2-134 ~]# mysql -h database-1-instance-1.c9inyukwf1e8.ap-south-1.rds.amazonaws.com -u admin -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 163
Server version: 8.0.28 Source distribution

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

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

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

mysql> 
```

STEP 6 Adding data in the database.

```

mysql> create DATABASE webappdb;
Query OK, 1 row affected (0.00 sec)

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

mysql> use webappdb;
Database changed
mysql> CREATE TABLE IF NOT EXISTS transactions(id INT NOT NULL
    -> AUTO_INCREMENT, amount DECIMAL(10,2), description
    -> VARCHAR(100), PRIMARY KEY(id));
Query OK, 0 rows affected (0.02 sec)

mysql> show tables
-> ^C
mysql> show tables;
+-----+
| Tables_in_webappdb |
+-----+
| transactions       |
+-----+
1 row in set (0.00 sec)

mysql> INSERT INTO transactions (amount,description) VALUES ('400','groceries');
Query OK, 1 row affected (0.01 sec)

mysql> SELECT * FROM transactions;
+---+-----+-----+
| id | amount | description |
+---+-----+-----+
| 1  | 400.00 | groceries   |
+---+-----+-----+
1 row in set (0.00 sec)

mysql> █

```

Refer to this code repository: [git clone <https://github.com/aws-samples/aws-three-tier-web-architecture-workshop.git>](https://github.com/aws-samples/aws-three-tier-web-architecture-workshop.git)

Setting up S3 bucket Code Section

In this section we will be adding the code repository in the s3 bucket storage.

The screenshot shows the AWS S3 service page. At the top, there are tabs for Services (8), Features (26), Resources (New), Documentation (23,408), and Knowledge Articles. Below the tabs, there are two main sections: "S3" (Scalable Storage in the Cloud) and "S3 Glacier" (Archive Storage in the Cloud). The "S3" section is expanded, showing its details. At the bottom, there is a "Create folder" button and an "Upload" button.

STEP 1: edit the DB config file with the db specific argument, like writer endpoints.

STEP 2: Upload the files of the app tier code in this S3 bucket.

The screenshot shows the AWS S3 Objects (6) page. It displays a list of six files: DbConfig.js, index.js, package-lock.json, package.json, README.md, and TransactionService.js. Each file entry includes a checkbox, the file name, type (js or json), last modified date (October 17, 2023, 17:57:47 UTC+05:30), and size.

	Name	Type	Last modified	Size
<input type="checkbox"/>	DbConfig.js	js	October 17, 2023, 17:59:24 (UTC+05:30)	
<input type="checkbox"/>	index.js	js	October 17, 2023, 17:57:47 (UTC+05:30)	
<input type="checkbox"/>	package-lock.json	json	October 17, 2023, 17:57:47 (UTC+05:30)	
<input type="checkbox"/>	package.json	json	October 17, 2023, 17:57:48 (UTC+05:30)	
<input type="checkbox"/>	README.md	md	October 17, 2023, 17:57:50 (UTC+05:30)	
<input type="checkbox"/>	TransactionService.js	js	October 17, 2023, 17:57:51 (UTC+05:30)	

STEP 3: Testing this code Successfully running node application using pm2.

```
[root@ip-10-0-2-134 app-tier]# pm2 start index.js
[PM2] Applying action restartProcessId on app [index] (ids: [ 0 ])
[PM2] [index] (0) ✓
[PM2] Process successfully started

id | name          | mode | ⚙ | status | cpu | memory
--|---|---|---|---|---|---|
0 | index         | fork | 15 | online | 0% | 12.4mb

]

[root@ip-10-0-2-134 app-tier]# pm2 list

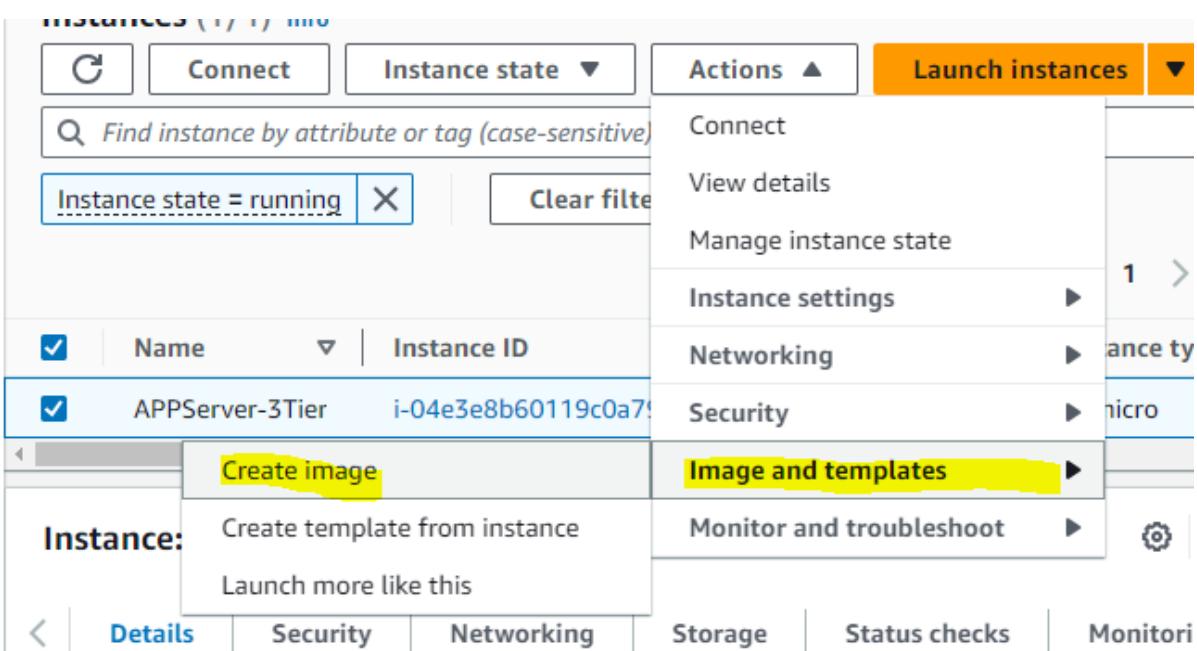
id | name          | mode | ⚙ | status | cpu | memory
--|---|---|---|---|---|---|
0 | index         | fork | 15 | online | 0% | 56.8mb
```

STEP 4: Testing Endpoints.

```
[root@ip-10-0-2-134 app-tier]# curl http://localhost:4000/health
"This is the health check"[root@ip-10-0-2-134 app-tier]#
```

```
[root@ip-10-0-2-134 app-tier]# curl http://localhost:4000/transaction
{"result": [{"id": 1, "amount": 400, "description": "groceries"}}][root@ip-10-0-2-134 app-tier]#
```

STEP 5: AS this is an multi AZ setup hence, We have to do the same setup in the second AZ, Hence we'll be creating an Image out of this, And then Creating the Instance in the second AZ.



Create image Info

An image (also referred to as an AMI) defines the programs and settings that are applied when you launch an EC2 instance.

Instance ID

i-04e3e8b60119c0a79 (WebAPP-3Tier)

Image name

AppTier-image

Maximum 127 characters. Can't be modified after creation.

Image description - *optional*

AppTier-image

Maximum 255 characters

No reboot

Enable

Instance volumes

Security group for private instance and as its app tier

Subnet Info

Don't include in launch template

 Create new subnet Info

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

 Compare security group rules

my-private-instance-sg-3tier sg-09bbca10323a5d565 
VPC: vpc-0d10d036dac169e48

 Advanced network configuration

choose the instance profile from the advance section

Purchasing option [Info](#)
 Request Spot Instances

IAM instance profile [Info](#)

my-ec2-trustedrole
arn:aws:iam::840260673873:instance-profile/my-ec2-trustedrole

[Create new IA](#)

Hostname type [Info](#)

Don't include in launch template

DNS Hostname [Info](#)

Success
Successfully created **app-tier-launch-template(lt-0739c111e82a869b4)**.

STEP 6: Creating Target Groups, for load balancing of the app tier.

Target groups [Info](#)

Filter target groups

Actions [Create target group](#)

< 1 >

Choosing the custom VPC we created in the earlier steps, 4000 is the target port.

Target group name

A maximum of 32 alphanumeric characters including hyphens are allowed, but the name must not begin or end with a hyphen.

Protocol Port

1-65535

IP address type

Only targets with the indicated IP address type can be registered to this target group.

 IPv4

Each instance has a default network interface (eth0) that is assigned the primary private IPv4 address. The instance's primary private IPv4 address is the one that will be applied to the target.

 IPv6

Each instance you register must have an assigned primary IPv6 address. This is configured on the instance's default network interface (eth0). [Learn more](#)

VPC

Select the VPC with the instances that you want to include in the target group. Only VPCs that support the IP address type selected above are available in this list.

Protocol version

 HTTP1

/health as the health check endpoint.

Health checks

The associated load balancer periodically sends requests, per the settings below, to the registered targets to test their status.

Health check protocol

Health check path

Use the default path of "/" to ping the root, or specify a custom path if preferred.

Up to 1024 characters allowed.

Target groups (1) Info

Filter target groups

<input type="checkbox"/>	Name	ARN	Port
<input type="checkbox"/>	App-Tier-TG	arn:aws:elasticloadbalancing:us-east-1:123456789012:targetgroup/App-Tier-TG/53f34a2e4a5d4a2a	4000

STEP 7: Now create an Load Balancer and attach this Target group to this LB.

Load balancers

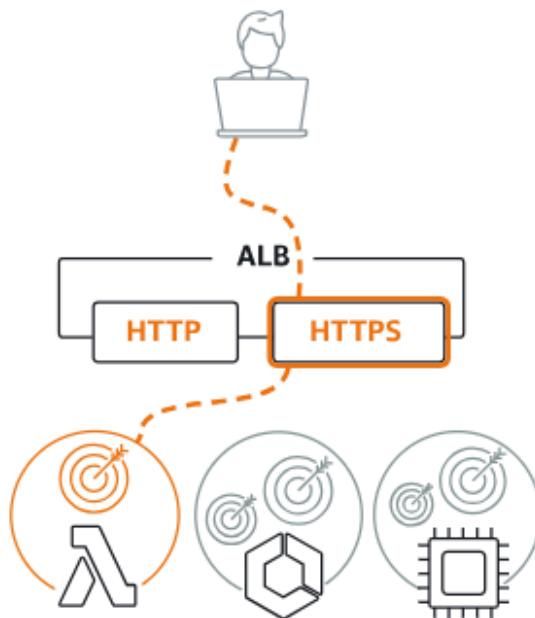
Elastic Load Balancing scales your load balancer capacity automatically in response to changes in incoming traffic.



Actions ▾

Create load balancer ▾

Application Load Balancer Info



Choose an Application Load Balancer when you need a flexible feature set for your applications with HTTP and HTTPS traffic. Operating at the request level, Application Load Balancers provide advanced routing and visibility features targeted at application architectures, including microservices and containers.

[Create](#)

STEP 8: Select internal as its not an internet facing load balancer. Use the private subnet as its internal.

Load balancer name

Name must be unique within your AWS account and can't be changed after the load balancer is created.

app-tier-internal-lb

A maximum of 32 alphanumeric characters including hyphens are allowed, but the name must not begin or end with a hyphen.

Scheme | Info

Scheme can't be changed after the load balancer is created.

Internet-facing

An internet-facing load balancer routes requests from clients over the internet to targets. Requires a public subnet. [Learn more](#)

Internal

An internal load balancer routes requests from clients to targets using private IP addresses.

IP address type | Info

Select the type of IP addresses that your subnets use.

IPv4

Recommended for internal load balancers.

Dualstack

Includes IPv4 and IPv6 addresses.

... ▾

Select the virtual private cloud (VPC) for your targets or you can [create a new VPC](#). The selected VPC can't be changed after the load balancer is created. targets, view your [target groups](#).

my-3tier-priyanshu

vpc-0d10d036dac169e48

IPv4: 10.0.0.0/16



Mappings | Info

Select at least two Availability Zones and one subnet per zone. The load balancer routes traffic to targets in these Availability Zones only. Availability Zones t balancer or the VPC are not available for selection.

ap-south-1a (aps1-az1)

Subnet

subnet-0e5a9b907a0f3d15b

private-subnet-AZ1 ▾

IPv4 address

Assigned from CIDR 10.0.2.0/24

ap-south-1b (aps1-az3)

Subnet

subnet-071856a3d56840d6e

private-subnet-AZ2 ▾

STEP 9: Attach the internal ALB sg which we created to this ALB.

Security groups [Info](#)

A security group is a set of firewall rules that control the traffic to your load balancer.

Security groups

Select up to 5 security groups

internal-lb-3tier

sg-00b2a1902de18ebdd VPC: vpc-0d10d036dac169e48



▼ Listener HTTP:80

Remove

Protocol

Port

Default action

[Info](#)

HTTP

: 80

1-65535

Forward to

App-Tier-TG

Target type: Instance, IPv4

HTTP



[Create target group](#)

STEP 10: Creating Launch template for auto scaling for the app tier.

Launch template [Info](#)

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.

app-tier-launch-template



[Create a launch template](#)

STEP 11: Select Subnet for the app tier private.

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

VPC

Choose the VPC that defines the virtual network for your Auto Scaling group.

▼ C

[Create a VPC](#)

Availability Zones and subnets

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

▼ C

ap-south-1a | subnet-0e5a9b907a0f3d15b X
(private-subnet-AZ1)
10.0.2.0/24

ap-south-1b | subnet-071856a3d56840d6e X
(private-subnet-AZ2)
10.0.3.0/24

[Create a subnet](#)

STEP 12: Select the existing Load Balancer for the internal routing , and the target group we created above.

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 an existing load balancer

Select the load balancers that you want to attach to your Auto Scaling group.

Choose from your load balancer target groups
This option allows you to attach Application, Network, or Gateway Load Balancers.

Choose from Classic Load Balancers

Existing load balancer target groups

Only instance target groups that belong to the same VPC as your Auto Scaling group are available for selection.

▼ C

App-Tier-TG | HTTP X
Application Load Balancer: app-tier-internal-lb

STEP 13: Setup Dynamic scaling , keeping desired and all as 2 for maximum fault tolerance and availability.

Configure group size and

Set the desired, minimum, and maximum capacity dynamically scale the number of instances in the g

Group size - optional [Info](#)

Specify the size of the Auto Scaling group by ch maximum capacity limits. Your desired capacity

Desired capacity

2

Minimum capacity

2

Maximum capacity

2

Autoscaling group for the app tier will be created successfully.

Auto Scaling groups (1) Info		C	Launch configurations	Launch templates	Actions	Create Auto Sc
<input type="text"/> Search your Auto Scaling groups						
Name	Launch template/configuration	Instances	Status	Desired capacity		
app-tier-asg	app-tier-launch-template Version Defa	2	-	2		

So the summary for the app tier was that we did all the setup in the ec2 from which we created an image and then put that image as the ami image for the other instance in the launch template , and this launch template will be used by the autoscaling group.. The load balancer wil contact to this instance.

Name	Instance ID	Instance state	Instance type	Status check	Ali
-	i-0cdd7f052f9721fd9	Running	t2.micro	2/2 checks passed	Nc
APPServer-3Tier	i-04e3e8b60119c0a79	Running	t2.micro	2/2 checks passed	Nc
-	i-05c18f345cb6dafce	Running	t2.micro	2/2 checks passed	Nc

Setting up the web tier

This Tier will be holding our frontend Tier.

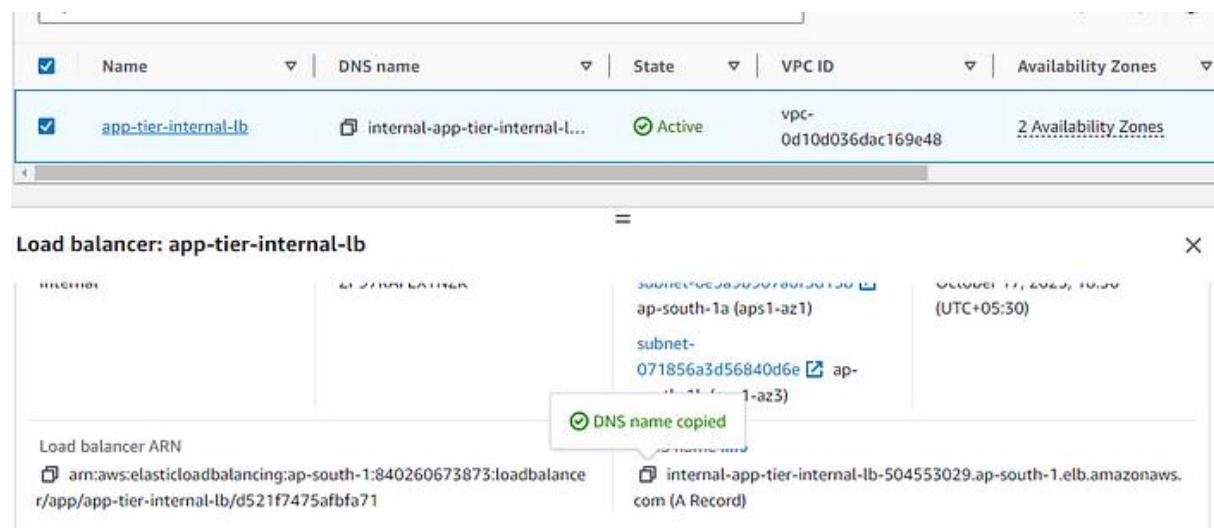
STEP 1: Setup nginx, Edit the nginx conf file, add the proxy pass as the endpoint for the ALB.

```
default_type text/html;
return 200 "<!DOCTYPE html><p>Web Tier Health Check</p>\n";
}

#react app and front end files
location / {
root    /home/ec2-user/web-tier/build;
index index.html index.htm
try_files $uri /index.html;
}

#proxy for internal lb
location /api/{
    proxy_pass http://[REPLACE-WITH-INTERNAL-LB-DNS]:80/;
}
```

Copy the DNS name for the Internal LB.



The screenshot shows the AWS Elastic Load Balancing console. A new internal load balancer named "app-tier-internal-lb" has been created. The "DNS name" field contains "internal-app-tier-internal-lb-504553029.ap-south-1.elb.amazonaws.com (A Record)". A tooltip "DNS name copied" is shown over the copy icon. The ARN of the load balancer is listed as "arn:aws:elasticloadbalancing:ap-south-1:840260673873:loadbalance/r/app/app-tier-internal-lb/d521f7475afbfa71".

```
index index.html index.htm
try_files $uri /index.html;

#proxy for internal lb
location /api/{
    proxy_pass http://internal-app-tier-internal-lb-504553029.ap-south-1.elb.amazonaws.com:80/;
}
```

STEP 2: Edit these files with the endpoint and database connections, and upload this to the S3 bucket under the web-tier folder.

Files and folders (32 Total, 254.7 KB)

All files and folders in this table will be uploaded.

<input type="checkbox"/>	Name	Folder	Type	Size
<input type="checkbox"/>	index.js	web-tier/src/c...	text/javascript	35.0 B
<input type="checkbox"/>	DatabaseDem...	web-tier/src/c...	text/css	655.0 B
<input type="checkbox"/>	DatabaseDem...	web-tier/src/c...	text/javascript	4.0 KB
<input type="checkbox"/>	Home.js	web-tier/src/c...	text/javascript	445.0 B
<input type="checkbox"/>	index.js	web-tier/src/c...	text/javascript	35.0 B
<input type="checkbox"/>	Menu.js	web-tier/src/c...	text/javascript	1.0 KB
<input type="checkbox"/>	Menu.styled.js	web-tier/src/c...	text/javascript	934.0 B
<input type="checkbox"/>	index.js	app-tier/	text/javascript	3.2 KB
<input type="checkbox"/>	package-lock....	app-tier/	application/js...	42.9 KB
<input type="checkbox"/>	package.json	app-tier/	application/js...	682.0 B

<input type="checkbox"/>	Name	Type	Last modified	Size
<input type="checkbox"/>	app-tier/	Folder	-	
<input type="checkbox"/>	web-tier/	Folder	-	

STEP 3: Now setting up instances for the web tier, all the rest configuration is same as for the app tier just change the security groups.

VPC - required [Info](#)

vpc-0d10d036dac169e48 (my-3tier-priyanshu)
10.0.0.0/16

Subnet Info

subnet-076df2f16efa400e7 public-web-subnet-AZ1
VPC: vpc-0d10d036dac169e48 Owner: 840260673873
Availability Zone: ap-south-1a IP addresses available: 250 CIDR: 10.0.0.0/24

Create new subnet [\[?\]](#)

Auto-assign public IP [Info](#)

Enable

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.

Create security group Select existing security group

Common security groups [Info](#)

Select security groups

WebTier-sg sg-01f7efe2cc326ac2c X
VPC: vpc-0d10d036dac169e48

Compare security group rules

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

► Advanced network configuration

web-instance-... i-01a4582ef4aa787e0 Running [\[?\]](#) [\[?\]](#) t2.micro Initializing No alarms + ap-south-1a

STEP 4: Connect it using the session manager.

EC2 > Instances > i-01a4582ef4aa787e0 > Connect to instance

Connect to instance [Info](#)

Connect to your instance i-01a4582ef4aa787e0 (web-instance-3tier) using any of these options

EC2 Instance Connect [Session Manager](#) [SSH client](#) [EC2 serial console](#)

Session Manager usage:

- Connect to your instance without SSH keys, a bastion host, or opening any inbound ports.
- Sessions are secured using an AWS Key Management Service key.
- You can log session commands and details in an Amazon S3 bucket or CloudWatch Logs log group.
- Configure sessions on the Session Manager [Preferences](#) [\[?\]](#) page.

Cancel [Connect](#)

STEP 5: Setup the instance web tier application same as we did it in the app tier.

```

Creating an optimized production build...
One of your dependencies, babel-preset-react-app, is importing the
"@babel/plugin-proposal-private-property-in-object" package without
declaring it in its dependencies. This is currently working because
"@babel/plugin-proposal-private-property-in-object" is already in your
node_modules folder for unrelated reasons, but it may break at any time.

babel-preset-react-app is part of the create-react-app project, which
is not maintained anymore. It is thus unlikely that this bug will
ever be fixed. Add "@babel/plugin-proposal-private-property-in-object" to
your devDependencies to work around this error. This will make this message
go away.

Compiled successfully.

File sizes after gzip:

 77.42 kB  build/static/js/main.19a32700.js
  1.79 kB  build/static/js/787.1f63e066.chunk.js
   493 B    build/static/css/main.b20b6ac4.css

The project was built assuming it is hosted at ./.
You can control this with the homepage field in your package.json.

The build folder is ready to be deployed.

Find out more about deployment here:

  https://cra.link/deployment

```

STEP 6: Install nginx.

```

[root@ip-10-0-0-245 web-tier]# yum install nginx -y
Last metadata expiration check: 0:07:17 ago on Tue Oct 17 13:32:49 2023.
Dependencies resolved.
=====
 Package           Architecture      Version
=====
Installing:
nginx              x86_64          1:1.24.0-1.amzn2023.0.1
Installing dependencies:
generic-logos-httd noarch          18.0.0-12.amzn2023.0.3
gperftools-libs   x86_64          2.9.1-1.amzn2023.0.2
libunwind          x86_64          1.4.0-5.amzn2023.0.2
nginx-core         x86_64          1:1.24.0-1.amzn2023.0.1
nginx-filesystem  noarch          1:1.24.0-1.amzn2023.0.1

```

replacing the nginx.conf file from the edit one which we uploaded in the s3 in the /etc/nginx.

```

[root@ip-10-0-0-245 web-tier]# aws s3 cp s3://aws-3tier-priyanshu/nginx.conf .
download: s3://aws-3tier-priyanshu/nginx.conf to ./nginx.conf
[root@ip-10-0-0-245 web-tier]# cd /etc/nginx/
[root@ip-10-0-0-245 nginx]# aws s3 cp s3://aws-3tier-priyanshu/nginx.conf .
download: s3://aws-3tier-priyanshu/nginx.conf to ./nginx.conf
[root@ip-10-0-0-245 nginx]# ls
conf.d      fastcgi.conf.default  koi-utf      mime.types.default  nginx.conf_bkp      uwsgi_params
default.d   fastcgi_params       koi-win      nginx.conf        scgi_params      uwsgi_params.default
fastcgi.conf fastcgi_params.default mime.types  nginx.conf.default scgi_params.default win-utf
[root@ip-10-0-0-245 nginx]# vi nginx.conf
[root@ip-10-0-0-245 nginx]# ls *.conf
fastcgi.conf  nginx.conf

```

STEP 7: starting nginx server and checking the status.

```
[root@ip-10-0-0-245 nginx]# service nginx.service restart
Redirecting to /bin/systemctl restart nginx.service
[root@ip-10-0-0-245 nginx]# systemctl status nginx
● nginx.service - The nginx HTTP and reverse proxy server
   Loaded: loaded (/usr/lib/systemd/system/nginx.service; disabled; preset:
     Active: active (running) since Tue 2023-10-17 13:48:51 UTC; 7s ago
   Process: 26914 ExecStartPre=/usr/bin/rm -f /run/nginx.pid (code=exited, st
   Process: 26916 ExecStartPre=/usr/sbin/nginx -t (code=exited, status=0/SUCC
   Process: 26917 ExecStart=/usr/sbin/nginx (code=exited, status=0/SUCCESS)
 Main PID: 26918 (nginx)
    Tasks: 2 (limit: 1114)
   Memory: 2.2M
      CPU: 58ms
     CGroup: /system.slice/nginx.service
             ├─26918 "nginx: master process /usr/sbin/nginx"
             └─26919 "nginx: worker process"

Oct 17 13:48:51 ip-10-0-0-245.ap-south-1.compute.internal systemd[1]: Starting
Oct 17 13:48:51 ip-10-0-0-245.ap-south-1.compute.internal nginx[26916]: nginx:
Oct 17 13:48:51 ip-10-0-0-245.ap-south-1.compute.internal nginx[26916]: nginx:
Oct 17 13:48:51 ip-10-0-0-245.ap-south-1.compute.internal systemd[1]: Started

ip-10-0-0-245 nginx]# chmod -R 775 /home/ec2-user
ip-10-0-0-245 nginx]# sudo chkconfig nginx on
Forwarding request to 'systemctl enable nginx.service'.
d symlink /etc/systemd/system/multi-user.target.wants/nginx.service → /u
systemd/system/nginx.service.
ip-10-0-0-245 nginx]#
```

Now setting up the load balancer

In this section we'll be creating web Tier autoscaling groups and Load Balancer configurations.

STEP 1: Create ami image for the web instance image as as we did in the app tier.

Create image Info

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

Instance ID

i-01a4582ef4aa787e0 (web-instance-3tier)

Image name

web-tier-image

Maximum 127 characters. Can't be modified after creation.

Image description - optional

web-tier-image

Maximum 255 characters

No reboot

Enable

Instance volumes

STEP 2: After successfullt creating the image,Create Target group for this load balancer.

Target groups (1) Info						
<input type="checkbox"/>	Name	ARN	Port	Protocol	Target type	Load
C Actions Create target group						

STEP 3: As its an web tier , this time the port is 80.

Target group name	<input type="text" value="web-tier-tg"/>
A maximum of 32 alphanumeric characters including hyphens are allowed, but the name must not begin or end with a hyphen.	
Protocol	Port
HTTP	: <input type="text" value="80"/> 1-65535

Health checks
The associated load balancer periodically sends requests, per the settings below, to the registered targets to test their status.
Health check protocol
HTTP
Health check path
Use the default path of "/" to ping the root, or specify a custom path if preferred.
/health
Up to 1024 characters allowed.

Now we have two target group one for app tier and other for web tier

<input type="checkbox"/>	Name	ARN	Port
<input type="checkbox"/>	App-Tier-TG	arn:aws:elasticloadbalancing:us-east-1:123456789012:targetgroup/App-Tier-TG/1234567890123456	4000
<input type="checkbox"/>	web-tier-tg	arn:aws:elasticloadbalancing:us-east-1:123456789012:targetgroup/web-tier-tg/1234567890123456	80

STEP 4: As it is internet facing load balancer enable intenet facing and select the public subnet for both the AZ.

BASIC CONFIGURATION

Load balancer name

Name must be unique within your AWS account and can't be changed after the load balancer is created.

A maximum of 32 alphanumeric characters including hyphens are allowed, but the name must not begin or end with a hyphen.

Scheme | Info

Scheme can't be changed after the load balancer is created.

Internet-facing

An internet-facing load balancer routes requests from clients over the internet to targets. Requires a public subnet. [Learn more](#)

Internal

An internal load balancer routes requests from clients to targets using private IP addresses.

IP address type | Info

Select the type of IP addresses that your subnets use.

IPv4

Recommended for internal load balancers.

Mappings | Info

Select at least two Availability Zones and one subnet per zone. The load balancer routes traffic to targets in these Availability Zones only. Availability Zone balancer or the VPC are not available for selection.

 ap-south-1a (aps1-az1)

Subnet

IPv4 address

Assigned by AWS

 ap-south-1b (aps1-az3)

Subnet

IPv4 address

STEP 5: Select the public facing Load Balancer security group.

Security groups Info

A security group is a set of firewall rules that control the traffic to your load balancer. Se

Security groups

Select up to 5 security groups

InternetFacing-lb-sg

sg-0e4d7dfb095a98e4a VPC: vpc-0d10d036dac169e48



Listeners and routing Info

A listener is a process that checks for connection requests using the port and protocol you configure. The rules that you define for a listener determine how the load balancer routes requests to its registered targets.

▼ Listener HTTP:80

Remove

Protocol

Port

Default action

[Info](#)

HTTP : 80
1-65535

Forward to [web-tier-tg](#)
Target type: Instance, IPv4

HTTP ▾



[Create target group](#)

Now create the launch template similarly to the app tier one just change the images and the security group related to the web tier.

Launch template name and description

Launch template name - *required*

web-tier-launch-template

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

Template version description

web-tier-launch-template

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

STEP 6: Creating auto scaling group for web tier.

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 [Info](#)

i 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.

C

[Create a launch template](#)

As its an web tier and public hence the Subnet will be Public for both AZ

[vpc-0d10d036dac169e48 \(my-3tier...
10.0.0.0/16](#)

C

[Create a VPC](#)

Availability Zones and subnets
Define which Availability Zones and subnets your Auto Scaling group can

[Select Availability Zones and subnets](#)

C

ap-south-1a | subnet-
076df2f16efa400e7 (public-web-
subnet-AZ1)
10.0.0.0/24

ap-south-1b | subnet-
001a75db22f70898d (public-web-
subnet-az2)
10.0.1.0/24

[Create a subnet](#)

STEP 7: Add the dynamic scaling policy for this web tier auto scaling group.

Group size - optional Info

Specify the size of the Auto Scaling group. You can specify minimum and maximum capacity, and define a capacity limit range.

Desired capacity

2

Minimum capacity

2

Maximum capacity

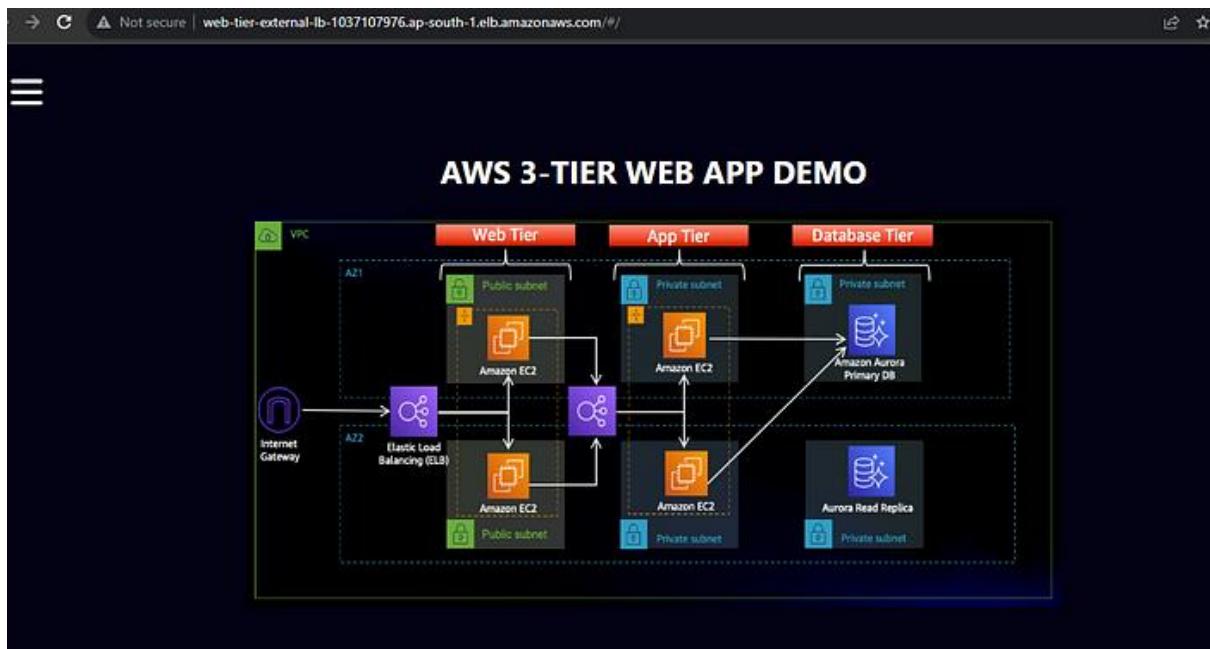
2



Launched the web tier load balancer with the 2 capacity.

Auto Scaling groups (2) <small>Info</small>			Launch configurations	Launch templates	Actions	Create Auto Scaling group
		Search your Auto Scaling groups < 1 >				
<input type="checkbox"/>	Name	Launch template/configuration	Instances	Status	Desired capacity	Min
<input type="checkbox"/>	web-tier-asg	web-tier-launch-template Version Defa	2	-	2	2
<input type="checkbox"/>	app-tier-asg	app-tier-launch-template Version Defa	2	-	2	2

After the complete setup of both the app and web tier, copy the dns of the web tier load balancer and hit it it will display this page.



The database demo can also be done here, and the amount and description can be directly added to the database from the UI.

The screenshot shows the "AURORA DATABASE DEMO PAGE". The page features a table for managing data:

ID	AMOUNT	DESC
<input type="button" value="ADD"/>	<input type="text"/>	<input type="text"/>

Buttons for "DEL" (Delete) and "X" (Close) are visible above the table. On the left side, there is a sidebar with navigation links:

- HOME** (with a house icon)
- DB DEMO** (with a document icon)