



3-Tier Architecture on AWS

06.04.2024

K. DUSHYANT REDDY



Overview

To build the 3 tier architecture for wordpress website in AWS console

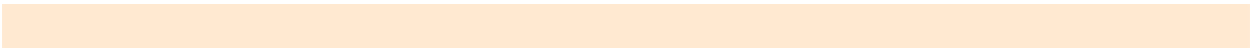
Description

To build a three-tier architecture for a WordPress website on AWS, start by setting up a VPC with public and private subnets. In the public subnet, deploy an EC2 instance for WordPress and configure it with a security group allowing HTTP and HTTPS traffic. Use Route 53 for DNS management to point your domain to the WordPress instance. In the private subnet, set up Amazon RDS to host the MySQL database and configure security groups to allow communication with the EC2 instance. Use Elastic Load Balancing (ELB) to distribute traffic across multiple EC2 instances for scalability and reliability.

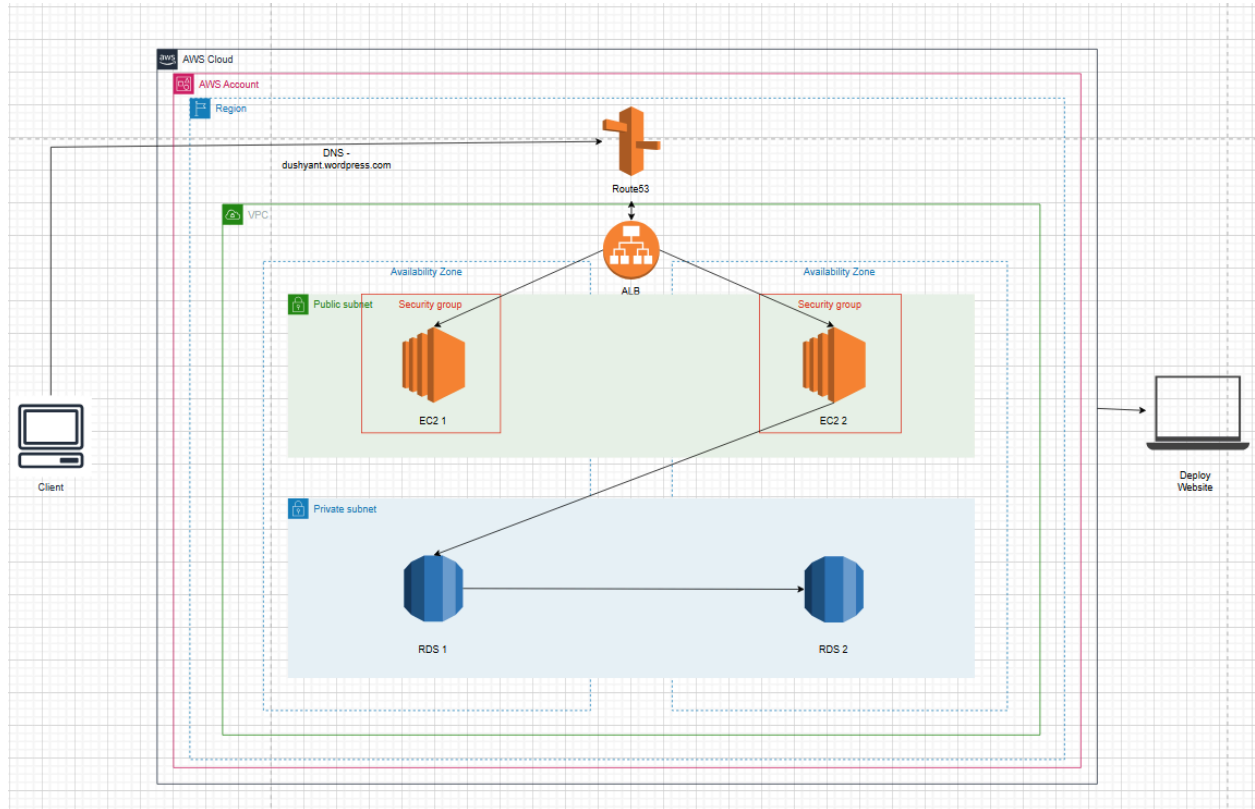
Prerequisites

Basic knowledge on -

1. **VPC** - Isolated virtual network environment within AWS, allowing customization of network settings and security configurations.
Subnet - Segments of a VPC, dividing the network into smaller, manageable parts. Subnets can be public (accessible from the internet) or private (isolated from the internet).
2. **Amazon EC2 (Elastic Compute Cloud)** - Web service which provides resizable compute capacity in the cloud.
Instance - Virtual computing environments
3. **AWS RDS (Relational Database Service)** - Managed relational database service, providing easy setup, scaling, and maintenance of SQL databases like MySQL, PostgreSQL, etc.
4. **Elastic Load Balancing** - manages the workload on the instances and distributes them to other instances in case of an instance failure
Application Load Balancers - Ideal for routing HTTP/HTTPS traffic and performing advanced traffic routing and content-based routing.

- 
5. **Route53** - a highly available and scalable Domain Name System (DNS) web service. You can use Route 53 to perform three main functions in any combination: domain registration, DNS routing, and health checking.

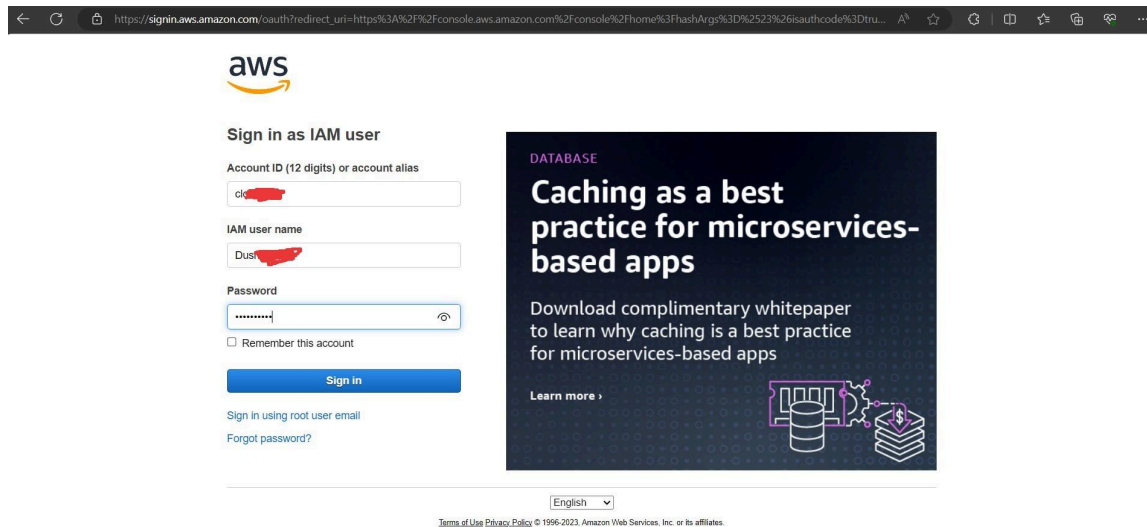
Architecture



Steps

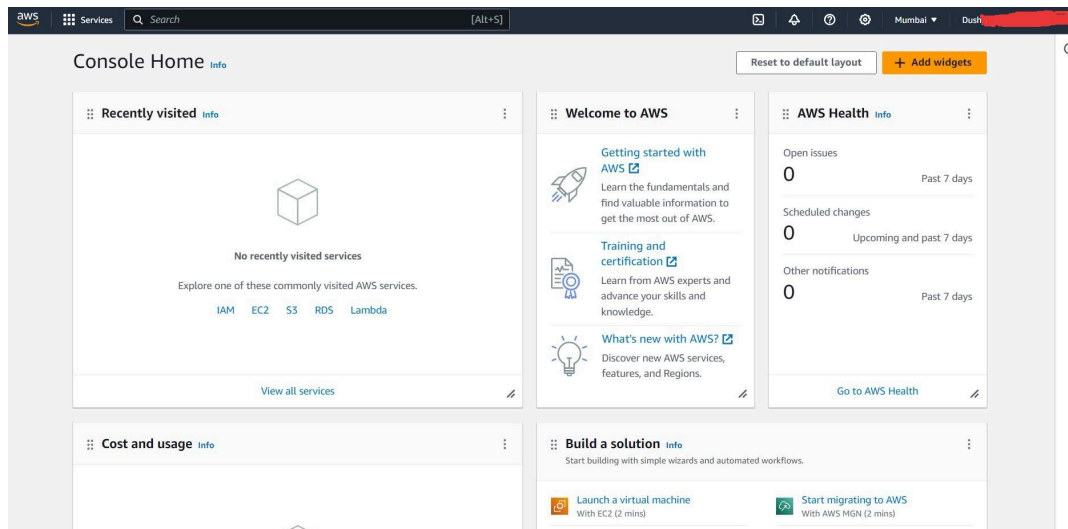
Step 1: Login into AWS Console

1.1 Create an account and login into it by providing the credentials.



The screenshot shows the AWS IAM user login page. The browser address bar displays the URL: `https://signin.aws.amazon.com/oauth?redirect_uri=https%3A%2F%2Fconsole.aws.amazon.com%2Fconsole%2Fhome%3FhashArgs%3D%2523%26isauthcode%3Dtrue...`. The page features the AWS logo at the top left. Below it, the heading "Sign in as IAM user" is followed by a form with the following fields: "Account ID (12 digits) or account alias" (containing "cl-"), "IAM user name" (containing "Dust"), and "Password" (masked with dots). There is a "Remember this account" checkbox and a "Sign in" button. Below the button are links for "Sign in using root user email" and "Forgot password?". On the right side, there is a promotional banner for "Caching as a best practice for microservices-based apps" with a "Learn more" link. At the bottom, there is a language selector set to "English" and a footer with "Terms of Use Privacy Policy" and copyright information.

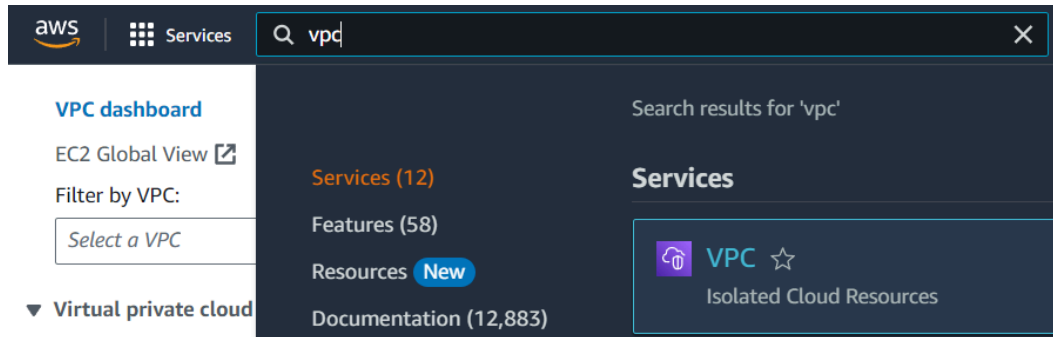
1.2 Give access to AWS Dashboard



The screenshot shows the AWS Console Home dashboard. The top navigation bar includes the AWS logo, "Services", a search bar, and a user profile dropdown showing "Mumbai" and "Dust". The main content area is titled "Console Home" and includes a "Reset to default layout" button and an "Add widgets" button. The dashboard is divided into several sections: "Recently visited" (showing "No recently visited services" with links to IAM, EC2, S3, RDS, and Lambda), "Welcome to AWS" (with links for "Getting started with AWS", "Training and certification", and "What's new with AWS?"), "AWS Health" (showing "Open issues", "Scheduled changes", and "Other notifications" all at 0), "Cost and usage", and "Build a solution" (with links for "Launch a virtual machine" and "Start migrating to AWS").

Step 2: Launch VPC in Console

2.1 In Dashboard, search for VPC service, click **VPC** to open it.



2.2 Click **Create VPC** to configure.

Your VPCs (2) [Info](#)

<input type="checkbox"/>	Name	VPC ID	State	IPv4 CIDR
<input type="checkbox"/>	-	vpc-07cb237232d06e04c	Available	172.31.0.0/16
<input type="checkbox"/>	tiervpc	vpc-0ccc3fb6f698d2464	Available	10.0.0.0/24

2.3 Type the **Name** which you prefer for naming the VPC and allocate the IPv4 CIDR block as **10.0.0.0/24**.

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.

IPv4 CIDR block [Info](#)

☒ IPv4 CIDR manual input ☐ IPAM-allocated IPv4 CIDR block

IPv4 CIDR

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

IPv6 CIDR block [Info](#)

☒ No IPv6 CIDR block ☐ IPAM-allocated IPv6 CIDR block ☐ Amazon-provided IPv6 CIDR block ☐ IPv6 CIDR owned by me

Tenancy [Info](#)

2.4 Now, select the subnets in the left pane and create **4 Subnets**, two Public for EC2 and other two Private for RDS with CIDR as **10.0.0.0/26**, **10.0.0.64/26**, **10.0.0.128/26** and **10.0.0.192/26** resp..

Subnets (4) [Info](#)

Find resources by attribute or tag

Name : sub X Clear filters

< 1 > ⚙

<input type="checkbox"/>	Name	Subnet ID	State	VPC	IPv4 CIDR
<input type="checkbox"/>	Prvsub1	subnet-0811d6ddc1d1607a4	Available	vpc-0f9329e90338753a9 tiervpc	10.0.0.64/26
<input type="checkbox"/>	Pubsub1	subnet-0761341a7fee23ec7	Available	vpc-0f9329e90338753a9 tiervpc	10.0.0.0/26
<input type="checkbox"/>	Prvsub2	subnet-0dd4cae8b88f230ab	Available	vpc-0f9329e90338753a9 tiervpc	10.0.0.192/26
<input type="checkbox"/>	Pubsub2	subnet-082b12e931b78231a	Available	vpc-0f9329e90338753a9 tiervpc	10.0.0.128/26

2.5 After creating the subnets now create an internet gateway, NAT gateway and Route table.

Internet gateways (1) [Info](#)

Search

tier X Clear filters

< 1 > ⚙

<input type="checkbox"/>	Name	Internet gateway ID	State	VPC ID	Owner
<input type="checkbox"/>	tier-ig	igw-0eb9d7c03c958dc8b	Attached	vpc-0f9329e90338753a9 tiervpc	654654193066

NAT gateways (1) [Info](#)

Find resources by attribute or tag

prv X Clear filters

< 1 > ⚙

<input type="radio"/>	Name	NAT gateway ID	Connectivity...	State	State message	Primary public I...	Primary priva
<input type="radio"/>	Prv-nat1	nat-017176988f5369ae3	Public	Available	-	13.232.21.196	10.0.0.12

2.6 Under Route Tables created associate the subnets based upon the resource enlisted such as public or private subnets.

Route tables (2) [Info](#)

Find resources by attribute or tag

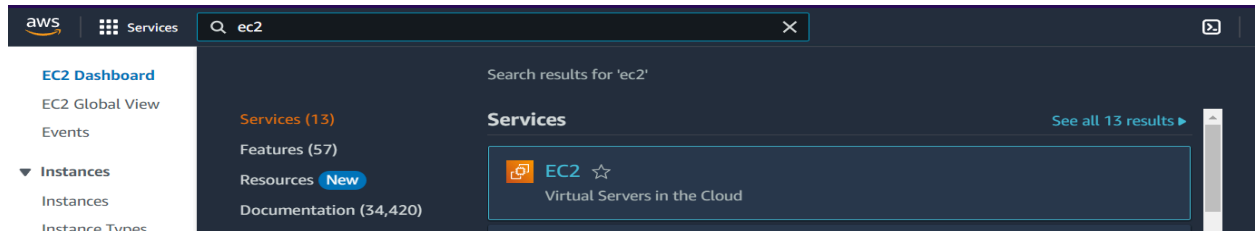
Name : rt X Clear filters

< 1 > ⚙

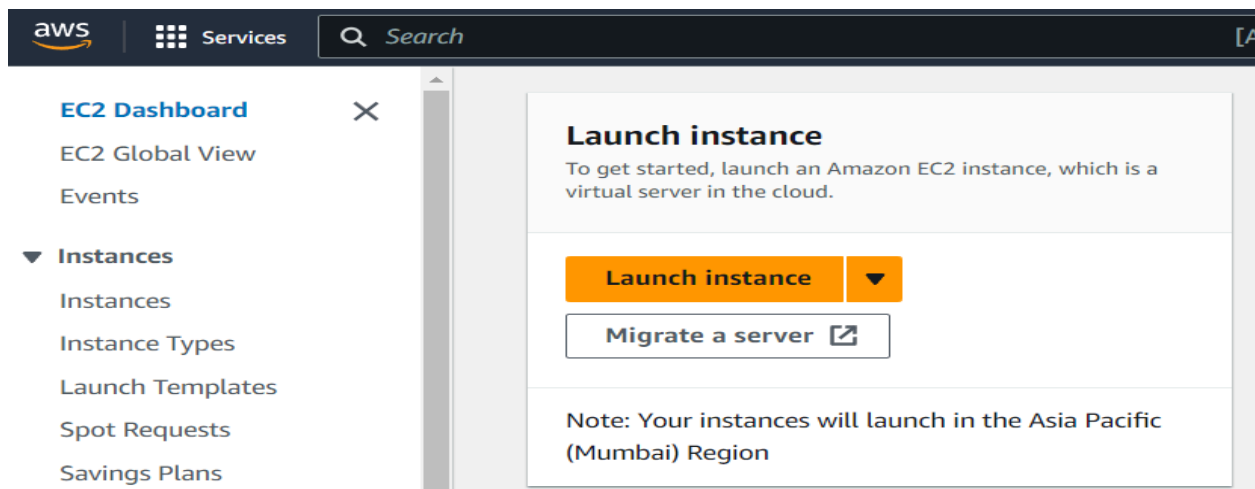
<input type="checkbox"/>	Name	Route table ID	Explicit subnet associ...	Edge associations	Main	VPC
<input type="checkbox"/>	Pub-rt	rtb-0a14d088961d87330	2 subnets	-	No	vpc-0f9329e90338753a9
<input type="checkbox"/>	Prv-rt	rtb-031be39231dc5212b	2 subnets	-	No	vpc-0f9329e90338753a9

Step 3: Launch EC2 in Console

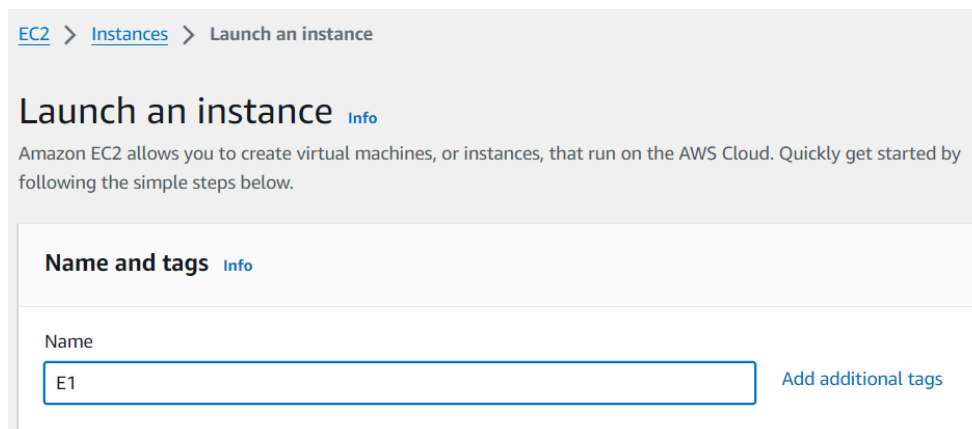
3.1 In Dashboard, search for EC2 service, click **EC2** to open it.



3.2 Click **Launch instance** to configure.



3.3 Type the **Name** which you prefer for naming the instance and Choose the **AMI**, so here we are using **Linux**.



3.4 Select the instance type **t2.micro** and then create the pair to do that click on **Create new pair key** for secure connection.

The screenshot shows the AWS console interface for creating an instance. Under the 'Instance type' section, 't2.micro' is selected, showing details like 1 vCPU and 1 GiB Memory. It is marked as 'Free tier eligible'. To the right, the 'All generations' radio button is selected, and a 'Compare instance types' link is present. Below this, the 'Key pair (login)' section is expanded, explaining that a key pair is needed for secure connection. A dropdown menu for 'Key pair name - required' shows 'Select' as the current choice. To the right of the dropdown is a 'Create new key pair' button with a circular arrow icon.

3.5 By clicking the **Create new pair key** a modal will pop up write the **key pair name** as you wish then under **key pair type** choose **RSA** radio button and after that **private key file format** should be considered **.pem** for OpenSSH. and then click on **Create key pair**.

The screenshot shows the 'Create key pair' modal window. The 'Key pair name' field contains 'dushtier'. Below it, a note states: 'The name can include up to 255 ASCII characters. It can't include leading or trailing spaces.' Under 'Key pair type', the 'RSA' option is selected with a radio button, with the description 'RSA encrypted private and public key pair'. The 'ED25519' option is unselected. Under 'Private key file format', the '.pem' option is selected with a radio button, with the description 'For use with OpenSSH'. The '.ppk' option is unselected, with the description 'For use with PuTTY'. A warning box at the bottom states: 'When prompted, store the private key in a secure and accessible location on your computer. You will need it later to connect to your instance. Learn more'. At the bottom right, there are 'Cancel' and 'Create key pair' buttons.

3.6 Under Network Settings, click **Edit** and under Firewall (security groups) choose **Create security group** option and then check in inbound security group rules that Type field is

▼ Network settings

info

VPC - required

info

vpv-0376063a4cc6a1d50
172.31.0.0/16

(default) ▼

↻

Subnet - info

subnet-0db8d381153e142cc

▼

↻ Create new subnet

VPC: vpv-0376063a4cc6a1d50
Availability Zone: ap-south-1a
Owner: 482788221413
IP addresses available: 4091
CIDR: 172.31.32.0/20

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

Security group name - required

SG-CZ

This security group will be added to all network interfaces. The name can't be edited after the security group is created. Max length is 255 characters. Valid characters: a-z, A-Z, 0-9, spaces, and _-./[]@!+=&:[]\$*

Description - required

info

launch-wizard-1 created 2024-02-16T10:41:01.636Z

Inbound Security Group Rules

▼

Security group rule 1 (TCP, 22, 0.0.0.0/0)

Remove

Type

Info

ssh

▼

Protocol

Info

TCP

Port range

Info

22

Source type

Info

Anywhere

▼

Source

Info

🔍 Add CIDR, prefix list or security

0.0.0.0/0 ✕

Description - optional

Info

e.g. SSH for admin desktop

▼

Security group rule 2 (TCP, 80, 0.0.0.0/0)

Remove

Type

Info

HTTP

▼

Protocol

Info

TCP

Port range

Info

80

Source type

Info

Anywhere

▼

Source

Info

🔍 Add CIDR, prefix list or security

0.0.0.0/0 ✕

Description - optional

Info

e.g. SSH for admin desktop

3.7 Leave rest configuration as default will come back later. Click on **Launch Instance**. Similarly setup one more instance with subnet private

▼ Summary

Number of instances

Info

1

Software Image (AMI)

Amazon Linux 2023 AMI 2023.3.2...[read more](#)

ami-0d63de463e6604d0a

Virtual server type (instance type)

t2.micro

Firewall (security group)

New 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. 30 GiB of EBS storage. 2

×

Cancel

Launch instance

3.8 Instance is successfully created.

EC2 > Instances > i-06a9a609aa02ea846

Instance summary for i-06a9a609aa02ea846 (EC1) Info

Updated less than a minute ago

Connect Instance state Action

Instance ID i-06a9a609aa02ea846 (EC1)	Public IPv4 address 13.232.193.188 open address	Private IPv4 addresses 10.0.0.56
IPv6 address -	Instance state Running	Public IPv4 DNS ec2-13-232-193-188.ap-south-1.compute.amazonaws.com open address
Hostname type IP name: ip-10-0-0-56.ap-south-1.compute.internal	Private IP DNS name (IPv4 only) ip-10-0-0-56.ap-south-1.compute.internal	Elastic IP addresses -
Answer private resource DNS name -	Instance type t2.micro	AWS Compute Optimizer finding Opt-in to AWS Compute Optimizer for recommendations. Learn more
Auto-assigned IP address 13.232.193.188 [Public IP]	VPC ID vpc-0f9329e90338753a9 (tiervpc) open address	Auto Scaling Group name -
IAM Role -	Subnet ID subnet-0761341a7fee23ec7 (Pubs1) open address	
IMDSv2 Required		

EC2 > Instances > i-0d567e3b59102c006

Instance summary for i-0d567e3b59102c006 (EC2) Info

Refreshing instance data

Connect Instance state Action

Instance ID i-0d567e3b59102c006 (EC2)	Public IPv4 address 43.205.144.81 open address	Private IPv4 addresses 10.0.0.104
IPv6 address -	Instance state Running	Public IPv4 DNS ec2-43-205-144-81.ap-south-1.compute.amazonaws.com open address
Hostname type IP name: ip-10-0-0-104.ap-south-1.compute.internal	Private IP DNS name (IPv4 only) ip-10-0-0-104.ap-south-1.compute.internal	Elastic IP addresses -
Answer private resource DNS name -	Instance type t2.micro	AWS Compute Optimizer finding Opt-in to AWS Compute Optimizer for recommendations. Learn more
Auto-assigned IP address 43.205.144.81 [Public IP]	VPC ID vpc-0f9329e90338753a9 (tiervpc) open address	Auto Scaling Group name -
IAM Role -	Subnet ID subnet-0811d6ddc1d1607a4 (Prvs1) open address	

3.9 Now, open E1 instance and connect via EC2 instance connect.

Connect to instance Info

Connect to your instance i-06a9a609aa02ea846 (EC1) using any of these options

EC2 Instance Connect Session Manager SSH client EC2 serial console

Instance ID
i-06a9a609aa02ea846 (EC1)

Connection Type

☒ Connect using EC2 Instance Connect
Connect using the EC2 Instance Connect browser-based client, with a public IPv4 address.

☐ Connect using EC2 Instance Connect Endpoint
Connect using the EC2 Instance Connect browser-based client, with a private IPv4 address and a VPC endpoint.

Public IP address
13.232.193.188

Username
Enter the username defined in the AMI used to launch the instance. If you didn't define a custom username, use the default username, ec2-user.

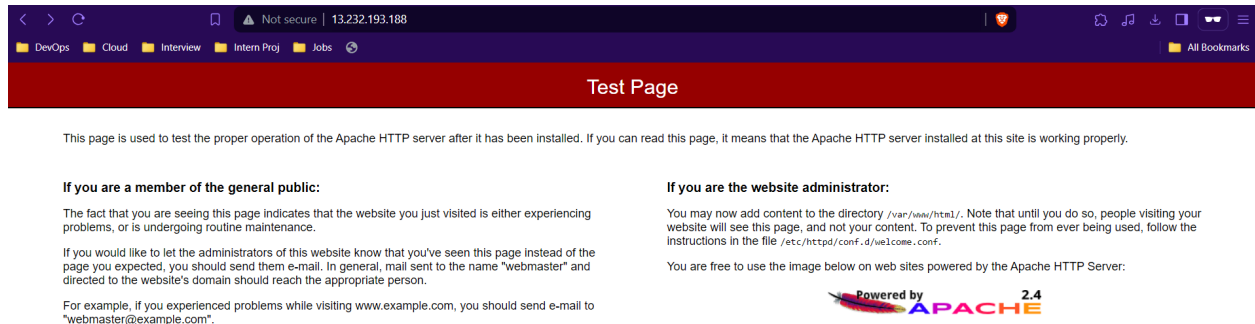
ec2-user

Note: In most cases, the default username, ec2-user, is correct. However, read your AMI usage instructions to check if the AMI owner has changed the default AMI username.

Cancel Connect

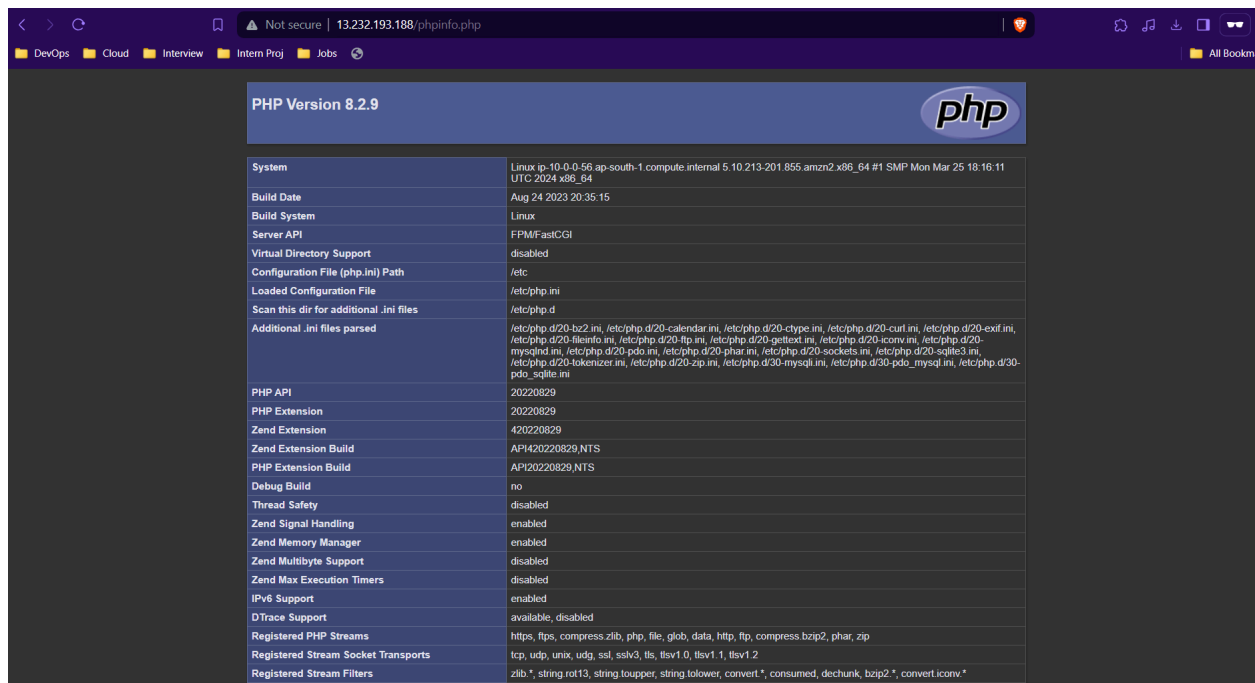
3.10 Follow the commands to install Apache

<https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/ec2-lamp-amazon-linux-2.html>

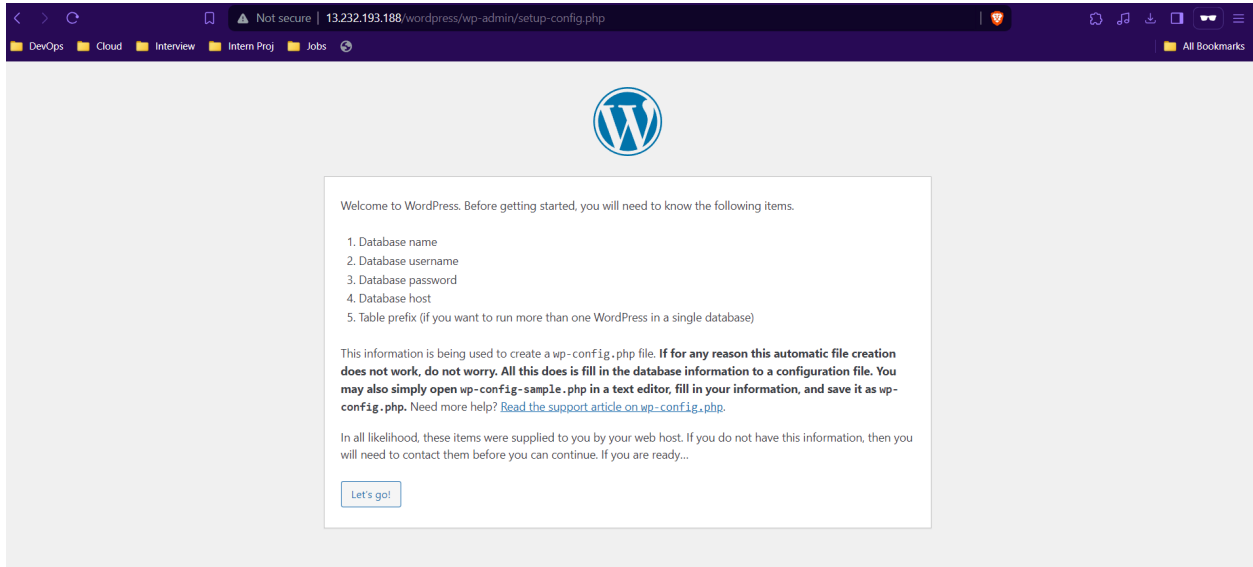


3.11 Then integrate Wordpress via PHP by

<https://developer.wordpress.org/advanced-administration/before-install/howto-install/>

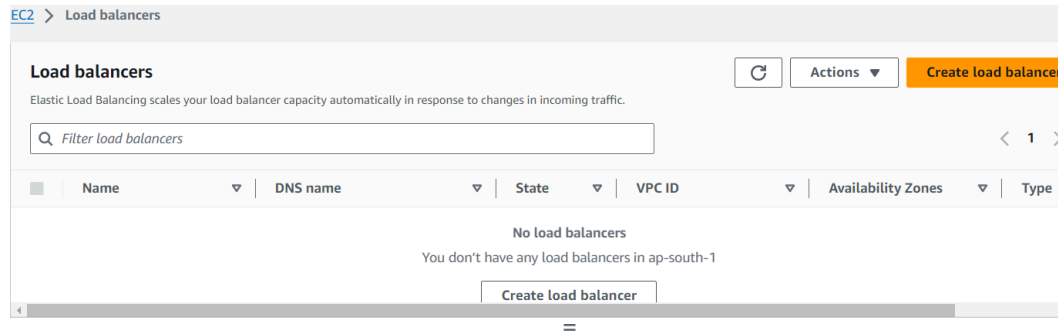


```
[ec2-user@ip-10-0-0-56 ~]$ groups
ec2-user adm wheel apache systemd-journal
[ec2-user@ip-10-0-0-56 ~]$ sudo chown -R ec2-user:apache /var/www
[ec2-user@ip-10-0-0-56 ~]$ sudo chmod 2775 /var/www && find /var/www -type d -exec sudo chmod 2775 {} \;
[ec2-user@ip-10-0-0-56 ~]$ find /var/www -type f -exec sudo chmod 0664 {} \;
[ec2-user@ip-10-0-0-56 ~]$ echo "<?php phpinfo(); ?>" > /var/www/html/phpinfo.php
[ec2-user@ip-10-0-0-56 ~]$ cd /var/www/html/
[ec2-user@ip-10-0-0-56 html]$ wget https://wordpress.org/latest.tar.gz
--2024-04-07 08:41:37-- https://wordpress.org/latest.tar.gz
Resolving wordpress.org (wordpress.org)... 198.143.164.252
Connecting to wordpress.org (wordpress.org)|198.143.164.252|:443... connected.
HTTP request sent, awaiting response...
```



Step 4: Launch ELB to distribute load

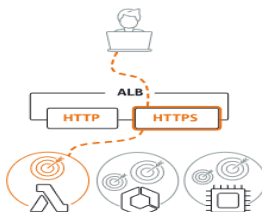
4.1 In the EC2 left pane scroll down to find **Load Balancer** click on it and create load balancer.



4.2 So here, we will choose **Application Load Balancer** to request HTTP traffic.

Load balancer types

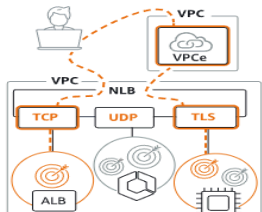
Application Load Balancer [Info](#)



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](#)


Network Load Balancer [Info](#)



Choose a Network Load Balancer when you need ultra-high performance, TLS offloading at scale, centralized certificate deployment, support for UDP, and static IP addresses for your applications. Operating at the connection level, Network Load Balancers are capable of handling millions of requests per second securely while maintaining ultra-low latencies.

[Create](#)

Gateway Load Balancer [Info](#)



Choose a Gateway Load Balancer when you need to deploy and manage a fleet of third-party virtual appliances that support GENEVE. These appliances enable you to improve security, compliance, and policy controls.

[Create](#)

Create Application Load Balancer [Info](#)

The Application Load Balancer distributes incoming HTTP and HTTPS traffic across multiple targets such as Amazon EC2 instances, microservices, and containers, based on request attributes. When the load balancer receives a connection request, it evaluates the listener rules in priority order to determine which rule to apply, and if applicable, it selects a target from the target group for the rule action.

► How Application Load Balancers work

Basic configuration

Load balancer name

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

tier-alb

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.

4.3 Add the VPC and the mappings for Availability zones with the existing security group

Network mapping [Info](#)

The load balancer routes traffic to targets in the selected subnets, and in accordance with your IP address settings.

VPC [Info](#)

Select the virtual private cloud (VPC) for your targets or you can [create a new VPC](#). Only VPCs with an internet gateway are enabled for selection. The selected VPC can't be changed after the load balancer is created. To confirm the VPC for your targets, view your [target groups](#).

tiervpc

vpc-0f9329e90338753a9
IPv4 VPC CIDR: 10.0.0.0/24



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 that are not supported by the load balancer or the VPC are not available for selection.

☒ ap-south-1a (aps1-az1)

Subnet

subnet-0761341a7fee23ec7

Pubsub1 ▼

IPv4 address

Assigned by AWS

☒ ap-south-1b (aps1-az3)

Subnet

subnet-082b12e931b78231a

Pubsub2 ▼

IPv4 address

Assigned by AWS

[EC2](#) > [Target groups](#) > Create target group

Step 1

Specify group details

Specify group details


Your load balancer routes requests to the targets in a target group and performs health checks on the targets.

Basic configuration

Settings in this section can't be changed after the target group is created.

Choose a target type

☒ Instances

- Supports load balancing to instances within a specific VPC.
- Facilitates the use of [Amazon EC2 Auto Scaling](#)  to manage and scale your EC2 capacity.

Register targets

This is an optional step to create a target group. However, to ensure that your load balancer routes traffic to this target group you must register your targets.

Available instances (2/2)

<input checked="" type="checkbox"/>	Instance ID	Name	State	Security groups
<input checked="" type="checkbox"/>	i-0d567e3b59102c006	EC2	Running	ec-sg
<input checked="" type="checkbox"/>	i-06a9a609aa02ea846	EC1	Running	ec-sg

2 selected

Ports for the selected instances

Ports for routing traffic to the selected instances.

1-65535 (separate multiple ports with commas)

Successfully created load balancer: tier-alb

It might take a few minutes for your load balancer to fully set up and route traffic. Targets will also take a few minutes to complete the registration process and pass initial health checks.

[EC2](#) > [Load balancers](#) > tier-alb

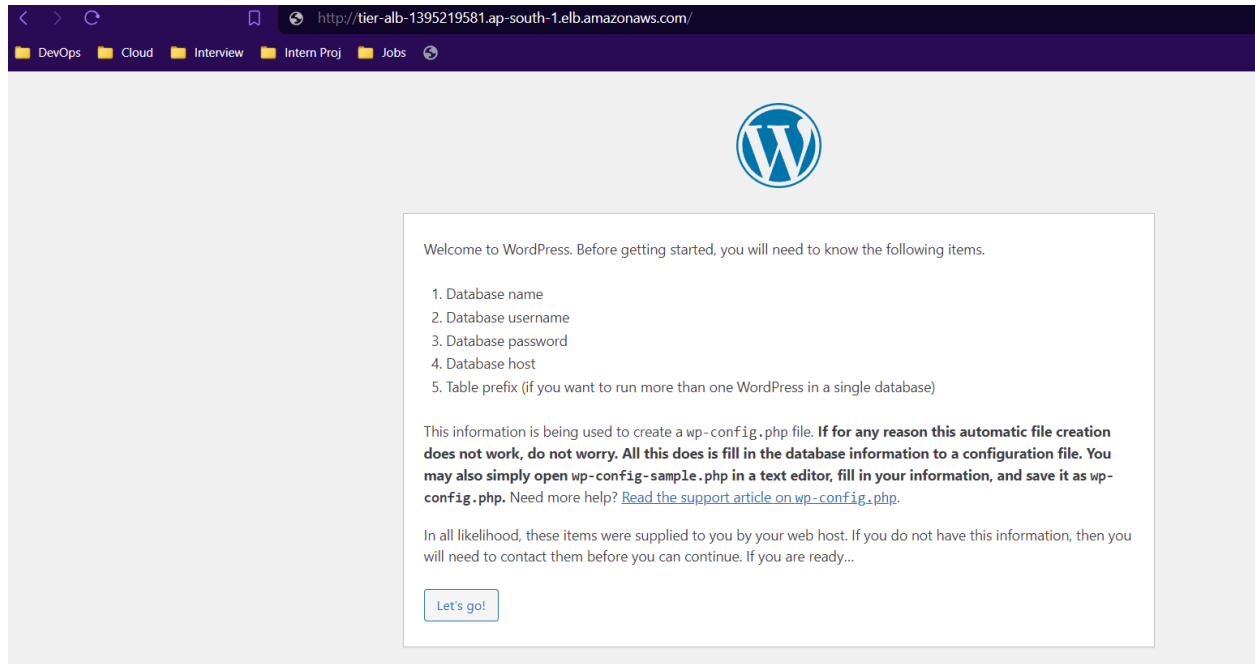
tier-alb

Actions ▾

▼ Details

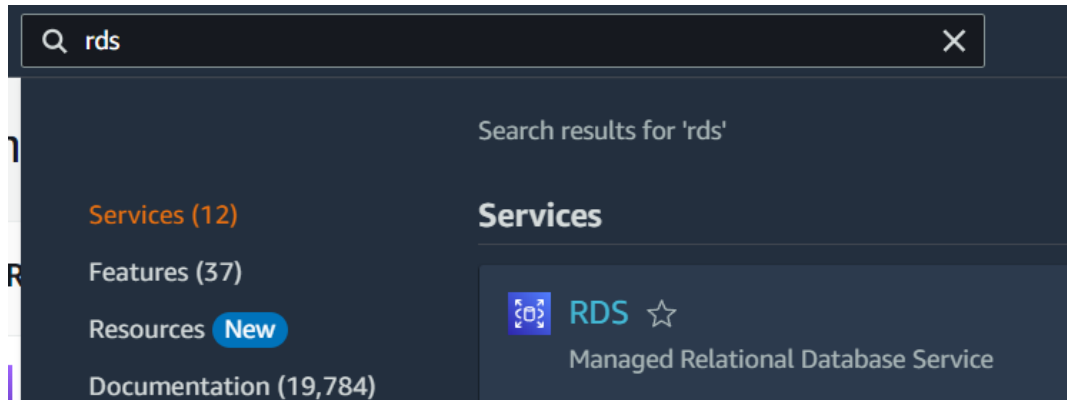
Load balancer type Application	Status Provisioning	VPC vpc-0f9329e90338753a9	IP address type IPv4
Scheme Internet-facing	Hosted zone ZP97RAFLXTNZK	Availability Zones subnet-0761341a7fee23ec7 ap-south-1a (aps1-az1) subnet-082b12e931b78231a ap-south-1b (aps1-az3)	Date created April 7, 2024, 14:25 (UTC+05:30)
Load balancer ARN arn:aws:elasticloadbalancing:ap-south-1:654654193066:loadbalancer/app/tier-alb/ca70353d43876823		DNS name info tier-alb-1395219581.ap-south-1.elb.amazonaws.com (A Record)	

4.7 Now copy the **DNS name** and paste in the New tab to check the load balancing of two instances.

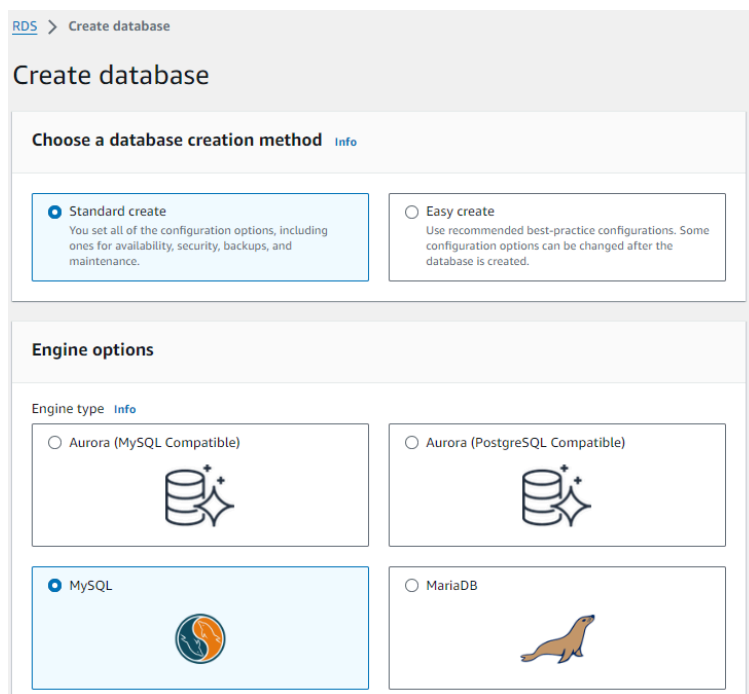


Step 5: Launch RDS to build the database

5.1 In Dashboard, search for RDS service, click **RDS** to open it.



5.2 Keep the database creation **Standard create** and select engine as **MySQL**.



5.3 Now, under the templates select **Free tier**, with the db instance as **database-1**.



Templates

Choose a sample template to meet your use case.

☐ **Production**
Use defaults for high availability and fast, consistent performance.

☐ **Dev/Test**
This instance is intended for development use outside of a production environment.

☒ **Free tier**
Use RDS Free Tier to develop new applications, test existing applications, or gain hands-on experience with Amazon RDS.
[Info](#)

Availability and durability

Deployment options [Info](#)

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

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

Settings

DB instance identifier [Info](#)

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

database-1

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

5.4 Add the credentials for database creation with password

▼ Credentials Settings

Master username [Info](#)

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

admin

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

☐ **Manage master credentials in AWS Secrets Manager**
Manage master user credentials in Secrets Manager. RDS can generate a password for you and manage it throughout its lifecycle.

ⓘ If you manage the master user credentials in Secrets Manager, some RDS features aren't supported.
[Learn more](#)

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

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](#)

5.5 Make the instance configuration to default **Bustable classes**.



Instance configuration

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

DB instance class [Info](#)

▼ Hide filters

☒ Show instance classes that support Amazon RDS Optimized Writes [Info](#)

Amazon RDS Optimized Writes improves write throughput by up to 2x at no additional cost.

☒ Include previous generation classes

☐ Standard classes (includes m classes)

☐ Memory optimized classes (includes r and x classes)

☒ Burstable classes (includes t classes)

db.t3.micro

2 vCPUs 1 GiB RAM Network: 2,085 Mbps

▼

5.6 Add the storage as per requirement and choose the EC2 instance which built prior.

Storage

Storage type [Info](#)

General Purpose SSD (gp2)

Baseline performance determined by volume size

▼

Allocated storage [Info](#)

20

GiB

The minimum value is 20 GiB and the maximum value is 6,144 GiB

ⓘ After you modify the storage for a DB instance, the status of the DB instance will be in storage-optimization. Your instance will remain available as the storage-optimization operation completes. [Learn more](#)

► Storage autoscaling

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.

Choose an EC2 instance

▼

↻

5.7 Similarly, attach the VPC made before here as well and rest leave it default and click **create a database**.

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

Amazon RDS will add a new VPC security group `rds-ec2-1` to allow connectivity with your compute resource.

Availability Zone [Info](#)

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.

Expiry: May 20, 2061

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

5.8 Successfully created the database.

Creating database database-1 [View credential details](#)
Your database might take a few minutes to launch. You can use settings from database-1 to simplify configuration of [suggested database add-ons](#) while we finish creating your DB for you.

Introducing Aurora I/O-Optimized
Aurora's I/O-Optimized [🔗](#) is a new cluster storage configuration that offers predictable pricing for all applications and improved price-performance, with up to 40% costs savings for I/O-intensive applications.

RDS > Databases

Consider creating a Blue/Green Deployment to minimize downtime during upgrades [X](#)
You may want to consider using Amazon RDS Blue/Green Deployments and minimize your downtime during upgrades. A Blue/Green Deployment provides a staging environment for changes to production databases. [RDS User Guide](#) [Aurora User Guide](#)

Databases (1) ☒ Group resources

DB identifier ▲	Status ▼	Role ▼	Engine ▼	Region & AZ ▼	Size ▼	Recommendations ▼	CPU ▼	Current activity ▼	Maintenance ▼	VPC
database-1	Creating	Instance	MySQL Community	-	db.t3.micro	-	-	-	none	vpc-0f93...

5.9 Wait for some time then the endpoint will also get updated.

Successfully created database database-1 [View connection details](#)
You can use settings from database-1 to simplify configuration of [suggested database add-ons](#) while we finish creating your DB for you.

RDS > Databases > database-1

database-1

Summary

DB identifier database-1	Status ✔ Available	Role Instance	Engine MySQL Community	Recommendations
CPU <div><div></div> 5.20%</div>	Class db.t3.micro	Current activity <div><div></div> 0 Connections</div>	Region & AZ ap-south-1a	

Connectivity & security

Endpoint & port	Networking	Security
Endpoint database-1.cncaakmqzcb.ap-south-1.rds.amazonaws.com	Availability Zone ap-south-1a	VPC security groups default (sg-02d1b1bf97bafa3c7)
Port 3306	VPC tiervpc (vpc-0f9329e90338753a9)	✔ Active ec-sg (sg-0d29dabe0f8f5617c)

5.10 Now, open EC2 instance and connect it and type the commands to connect “**mysql -h (endpoint of database) -P 3306 -u admin -p**”

```
[ec2-user@ip-172-31-90-181 wordpress]$ mysql -h database-1.cvasjorjdkkc.us-east-1.rds.amazonaws.com -u admin -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MySQL connection id is 15
Server version: 8.0.23 Source distribution

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

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

MySQL [(none)]>
MySQL [(none)]> create database wordpress;
Query OK, 1 row affected (0.01 sec)

MySQL [(none)]> exit
Bye
[ec2-user@ip-172-31-90-181 wordpress]$ pwd
/var/www/html/wordpress
[ec2-user@ip-172-31-90-181 wordpress]$
```

5.11 Now, create a **Replica DB instance** for processing the primary DB data into it.

RDS > Databases > Create read replica

Create read replica

You are creating a replica DB instance from a source DB instance. This new DB instance will have the source DB instance's DB security groups and DB parameter groups.

Settings

Replica source

Source DB instance identifier

database-1
Role: Instance

DB instance identifier

This is the unique key that identifies a DB instance. This parameter is stored as a lowercase string (for example, mydbinstance).

dush-db

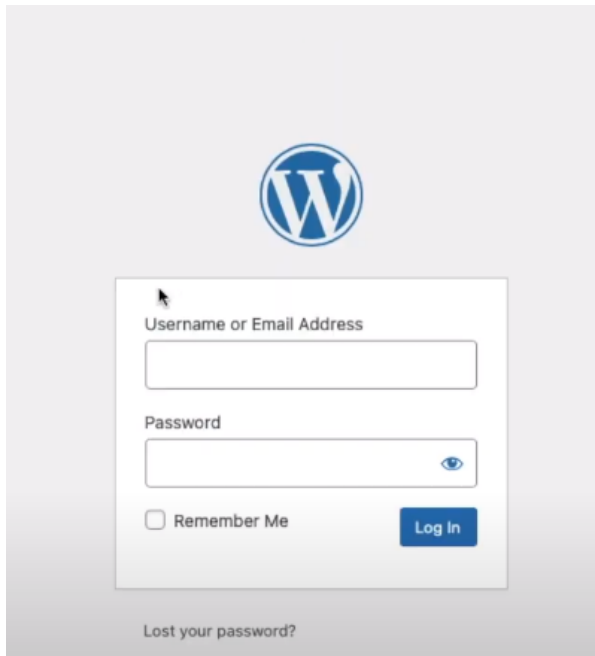
5.12 Successfully created the replica database ‘**dush-db**’.

Databases (2)									
<div>Group resources</div> <div>Filter by databases</div>									
DB identifier	Status	Role	Engine	Region & AZ	Size	Recommendations	CPU	Current activ	
database-1	Modifying	Primary	MySQL Community	ap-south-1a	db.t3.micro		3.16%	0 Co	
dush-db	Creating	Replica	MySQL Community	-	db.t3.micro		-		

5.13 Now, configure the wordpress file in the EC2 instance.

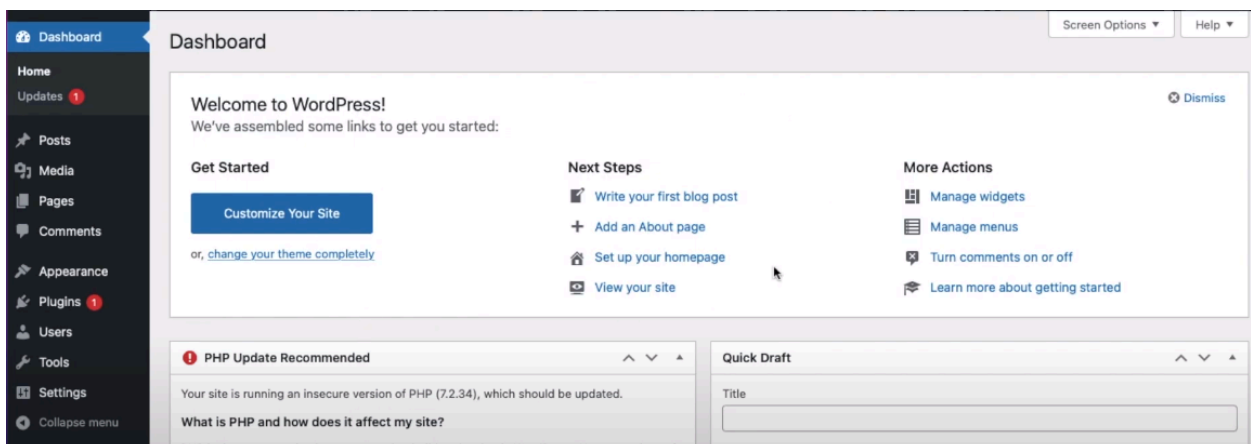
```
[ec2-user@ip-172-31-90-181 wordpress]$ pwd
/var/www/html/wordpress
[ec2-user@ip-172-31-90-181 wordpress]$ ls
index.php      wp-activate.php  wp-comments-post.php  wp-cron.php      wp-load.php  wp-settings.php  xmlrpc.php
license.txt    wp-admin         wp-config-sample.php  wp-includes      wp-login.php  wp-signup.php
readme.html    wp-blog-header.php  wp-content            wp-links-opml.php  wp-mail.php  wp-trackback.php
[ec2-user@ip-172-31-90-181 wordpress]$ vim wp-config.php
```

5.14 After running the installation, provide the information for creating the credentials.



The image shows the WordPress login page. At the top center is the WordPress logo. Below it is a white box containing the login form. The form has two input fields: "Username or Email Address" and "Password". Below the password field is a checkbox labeled "Remember Me" and a blue "Log In" button. At the bottom of the box is a link that says "Lost your password?".

5.15 Successfully launched the wordpress site.



The image shows the WordPress dashboard. On the left is a dark sidebar with a menu containing links to Dashboard, Home, Updates (with a red notification badge), Posts, Media, Pages, Comments, Appearance, Plugins (with a red notification badge), Users, Tools, Settings, and a Collapse menu option. The main content area has a header with "Dashboard" and "Screen Options" and "Help" buttons. Below the header is a "Welcome to WordPress!" message with a "Dismiss" link. The main area is divided into three columns: "Get Started" with a "Customize Your Site" button and a link to "change your theme completely"; "Next Steps" with links to "Write your first blog post", "Add an About page", "Set up your homepage", and "View your site"; and "More Actions" with links to "Manage widgets", "Manage menus", "Turn comments on or off", and "Learn more about getting started". At the bottom, there is a "PHP Update Recommended" notification stating that the site is running an insecure version of PHP (7.2.34) and a "Quick Draft" section with fields for "Title" and "Content".

Result

Task on to build the 3 tier architecture for wordpress website in AWS console successfully accomplished.

