

Deploying a Two-Tier Application Architecture and Making it Highly Available

1. Create VPC

The screenshot shows the AWS VPC dashboard. A green success message at the top says "You successfully created vpc-0fb64b2f1b4a16f3 / VPC-EpicReads". The main card displays details for the VPC, including its ID (vpc-0fb64b2f1b4a16f3), state (Available), and various network settings like DNS resolution, Main network ACL, and IPv4 CIDR. Below the main card is a "Resource map" section showing the VPC's connections to Subnets, Route tables, and Network Connections.

2. Create Subnet

The screenshot shows the AWS Subnets page. A green success message at the top says "You have successfully created 2 subnets: subnet-068c97297d2a8b5d8, subnet-07fba0299ab7689c0". The main table lists two subnets: "PrivateSubnet1-EpicReads" and "PublicSubnet1-EpicReads". Both subnets are in the "Available" state, belong to the "VPC-EpicReads" VPC, and have their "Block Public Access" setting set to "Off". Their respective IPv4 CIDRs are 10.0.1.0/24 and 10.0.0.0/24.

3. Create Internet gateway and attach to VPC.

The screenshot shows the AWS CloudFormation console with the following details:

- Account ID:** 3111-4154-2115
- Region:** United States (N. Virginia)
- Owner:** Adrina Colaco
- Internet gateway ID:** igw-0dc7cf900db9abc68
- State:** Attached
- VPC ID:** vpc-0fb64b2f1b4a16f3 | VPC-EpicReads
- Tags:** Name = IGW-EpicReads
- Notifications:** 0 errors, 0 warnings, 2 changes, 0 info, 0 pending.

4. Create Private Route Table and Private Subnet in subnet associations

The screenshot shows the AWS CloudFormation console with the following details:

- Route table ID:** rtb-009782fcbe6a69866
- Main:** No
- Owner ID:** 311141542113
- Explicit subnet associations:** -
- Edge associations:** -
- Routes (1):**

Destination	Target	Status	Propagated	Route Origin
10.0.0.0/16	local	Active	No	Create Route Table
- Subnet associations:**
 - Explicit subnet associations (1):**

Name	Subnet ID	IPv4 CIDR	IPv6 CIDR
PrivateSubnet1-EpicReads	subnet-07fba0299ab7689c0	10.0.1.0/24	-

5. Create Public Route Table, add Internet Gateway in the Routes, and Public Subnet in subnet associations.

Updated routes for rtb-0bddaeb5b816c74f7 / Public-RT-EpicReads successfully

rtb-0bddaeb5b816c74f7 / Public-RT-EpicReads

[Actions](#)

Details		Info	
Route table ID	rtb-0bddaeb5b816c74f7	Main	<input checked="" type="checkbox"/> No
VPC	vpc-0fb64b2f1b4a16f3 VPC-EpicReads	Owner ID	311141542113
Explicit subnet associations			
Edge associations			

[Routes](#) [Subnet associations](#) [Edge associations](#) [Route propagation](#) [Tags](#)

Routes (2)

Destination	Target	Status	Propagated	Route Origin
0.0.0.0/0	igw-0dc7cf900db9abc68	Active	No	Create Route
10.0.0.16	local	Active	No	Create Route Table

[Both](#) [Edit routes](#)

[rtb-0bddaeb5b816c74f7 / Public-RT-EpicReads](#)

[Details](#) [Routes](#) [Subnet associations](#) [Edge associations](#) [Route propagation](#) [Tags](#)

Explicit subnet associations (1)

Name	Subnet ID	IPv4 CIDR	IPv6 CIDR
PublicSubnet1-EpicReads	subnet-068c97297d2a8b5d8	10.0.0.0/24	-

[Edit subnet associations](#)

6. Create Security Group

sg-0d680468fe4d9859f - WebServer-SG

Security group (sg-0d680468fe4d9859f | WebServer-SG) was created successfully

sg-0d680468fe4d9859f - WebServer-SG

[Actions](#)

Details		Description	VPC ID
Security group name	WebServer-SG	Security group ID	sg-0d680468fe4d9859f
Owner	311141542113	Inbound rules count	2 Permission entries
		Outbound rules count	1 Permission entry

[Inbound rules](#) [Outbound rules](#) [Sharing - new](#) [VPC associations - new](#) [Tags](#)

Inbound rules (2)

Name	Security group rule ID	IP version	Type	Protocol	Port range	Source
-	sgr-080d350934da736a4	IPv4	SSH	TCP	22	0.0.0.0/0
-	sgr-002b31ffb1a86e207	IPv4	HTTP	TCP	80	0.0.0.0/0

[Manage tags](#) [Edit inbound rules](#)

7. Launch EC2 instance

Instances (1/1) Info

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Publ
EpicReadsWebserver	i-026fec5852e62358b	Running	t2.micro	2/2 checks passed	View alarms +	us-east-1a	-	44.20

i-026fec5852e62358b (EpicReadsWebserver)

Details | Status and alarms | Monitoring | Security | Networking | Storage | Tags

Instance summary

Instance ID	Public IPv4 address	Private IPv4 addresses
i-026fec5852e62358b	44.202.136.227 [open address]	10.0.0.232
IPv6 address	Instance state	Public DNS
-	Running	-
Hostname type	Private IP DNS name (IPv4 only)	Elastic IP addresses
IP name: ip-10-0-0-232.ec2.internal	ip-10-0-0-232.ec2.internal	-
Answer private resource DNS name	Instance type	AWS Compute Optimizer finding
-	t2.micro	Opt-in to AWS Compute Optimizer for recommendations. Learn more
Auto-assigned IP address	VPC ID	Auto Scaling Group name
44.202.136.227 [Public IP]	vpc-0fbcc64b2f1b4a16f3 (VPC-EpicReads) [open]	-
IAM Role	Subnet ID	Managed
-	subnet-068c97297d2a8b5d8 (PublicSubnet1-EpicReads) [open]	false
IMDSv2	Instance ARN	
Required	arn:aws:ec2:us-east-1:311141542113:instance/i-026fec5852e62358b	
Operator		
-		
Instance details		
AMI ID	Monitoring	Platform details
ami-08982fc5bf93d976	disabled	Linux/UNIX
AMI name	Allowed image	Termination protection
al2023-ami-2023.8.20250915.0-kernel-6.1-x86_64	-	Disabled
Stop protection	Launch time	AMI location
Disabled	Sun Sep 21 2025 11:58:14 GMT+0530 (India Standard Time) (5 minutes)	amazon/al2023-ami-2023.8.20250915.0-kernel-6.1-x86_64
Instance reboot migration	Instance auto-recovery	Lifecycle
Default (On)	Default	normal
Stop-hibernate behavior	AMI Launch index	Key pair assigned at launch
Disabled	0	DevopsAppHosting
State transition reason	Credit specification	Kernel ID
-	standard	-
State transition message	Usage operation	RAM disk ID
-	RunInstances	-
Owner	Enclaves Support	Boot mode
311141542113	-	uefi-preferred
Virtualization type	Reservation	Partition number
hvm	r-09a53bb372855bf1a	-
Number of vCPUs	Capacity Reservation setting	
1	open	
Capacity reservation		
Capacity Reservation ID		
-		

8. Connect to EC2 and install HTTPD

```
sudo yum update
sudo yum install httpd -y
sudo systemctl start httpd
sudo systemctl status httpd
```

```
aws [Alt+S] United States (N. Virginia) Adrina Colaco Account ID: 3111-4154-2115

[ec2-user@ip-10-0-0-232 ~]$ sudo yum update
Amazon Linux 2023 Kernel Livepatch repository
Dependencies resolved.
Nothing to do.
Complete!
[ec2-user@ip-10-0-0-232 ~]$ sudo yum install httpd -y
Last metadata expiration check: 0:01:11 ago on Sun Sep 21 06:35:43 2025.
Dependencies resolved.

Package           Architecture      Version       Repository      Size
Installing:
httpd             x86_64          2.4.65-1.amzn2023.0.1
Installing dependencies:
apr               x86_64          1.7.5-1.amzn2023.0.4
aprutil            x86_64          1.6.3-1.amzn2023.0.1
generic-logos-httd x86_64          18.0.0-0.12.amzn2023.0.3
httpd-core         x86_64          2.4.65-1.amzn2023.0.1
httpd-filesystem  noarch          2.4.65-1.amzn2023.0.1
httpd-tools        x86_64          2.4.65-1.amzn2023.0.1
libbrotli          x86_64          1.0.9-4.amzn2023.0.2
mailcap            noarch          2.1.49-3.amzn2023.0.3
Installing weak dependencies:
apr-util-openssl x86_64          1.6.3-1.amzn2023.0.1
mod_http2          x86_64          2.0.27-1.amzn2023.0.3
mod_lua            x86_64          2.4.65-1.amzn2023.0.1

Transaction Summary
Install 12 Packages

Total download size: 2.3 M
Installed size: 6.9 M
Downloading Packages:
(1/12): apr-util-openssl-1.6.3-1.amzn2023.0.1.x86_64.rpm 517 kB/s | 17 kB 00:00
(2/12): apr-util-1.6.3-1.amzn2023.0.1.x86_64.rpm 2.1 MB/s | 98 kB 00:00
(3/12): apr-1.7.5-1.amzn2023.0.4.x86_64.rpm 2.6 MB/s | 129 kB 00:00
(4/12): generic-logos-httd-18.0.0-0.12.amzn2023.0.3.noarch.rpm 1.9 MB/s | 11 kB 00:00
(5/12): httpd-2.4.65-1.amzn2023.0.1.x86_64.rpm 803 kB/s | 19 kB 00:00
(6/12): httpd-filesystem-2.4.65-1.amzn2023.0.1.noarch.rpm 2.2 MB/s | 47 kB 00:00
(7/12): httpd-tools-2.4.65-1.amzn2023.0.1.x86_64.rpm 593 kB/s | 13 kB 00:00
(8/12): httpd-2.4.65-1.amzn2023.0.1.x86_64.rpm 3.7 MB/s | 81 kB 00:00
(9/12): libbrotli-2.1.49-3.amzn2023.0.3.noarch.rpm 2.7 MB/s | 1 kB 00:00
(10/12): libbrotli-1.0.9-4.amzn2023.0.2.x86_64.rpm 1.4 MB/s | 33 kB 00:00
(11/12): mod_http2-2.0.27-1.amzn2023.0.3.x86_64.rpm 8.7 MB/s | 315 kB 00:00
(12/12): mod_lua-2.4.65-1.amzn2023.0.1.x86_64.rpm 5.6 MB/s | 166 kB 00:00
                                                               2.9 MB/s | 60 kB 00:00

Total
Running transaction check
Transaction check succeeded.
Running transaction test
Transaction test succeeded.
Running transaction
Preparing : 1/1
  Installing : apr-1.7.5-1.amzn2023.0.4.x86_64
  Installing : apr-util-openssl-1.6.3-1.amzn2023.0.1.x86_64
  Installing : apr-util-1.6.3-1.amzn2023.0.1.x86_64
  Installing : mailcap-2.1.49-3.amzn2023.0.3.noarch
  Installing : httpd-tools-2.4.65-1.amzn2023.0.1.x86_64
  Installing : libbrotli-1.0.9-4.amzn2023.0.2.x86_64
  Running scriptlet: httpd-filesystem-2.4.65-1.amzn2023.0.1.noarch
  Installing : httpd-filesystem-2.4.65-1.amzn2023.0.1.noarch
  Installing : httpd-core-2.4.65-1.amzn2023.0.1.x86_64
  Installing : mod_http2-2.0.27-1.amzn2023.0.3.x86_64
  Installing : mod_lua-2.4.65-1.amzn2023.0.1.x86_64
  Installing : generic-logos-httd-18.0.0-0.12.amzn2023.0.3.noarch
  Running scriptlet: httpd-2.4.65-1.amzn2023.0.1.x86_64
  Installing : httpd-2.4.65-1.amzn2023.0.1.x86_64
  Verifying : apr-1.7.5-1.amzn2023.0.4.x86_64
  Verifying : apr-util-1.6.3-1.amzn2023.0.1.x86_64
  Verifying : apr-util-openssl-1.6.3-1.amzn2023.0.1.x86_64
  Verifying : generic-logos-httd-18.0.0-0.12.amzn2023.0.3.noarch
  Verifying : httpd-2.4.65-1.amzn2023.0.1.x86_64
  Verifying : httpd-core-2.4.65-1.amzn2023.0.1.x86_64
  Verifying : httpd-filesystem-2.4.65-1.amzn2023.0.1.noarch
  Verifying : httpd-tools-2.4.65-1.amzn2023.0.1.x86_64
  Verifying : libbrotli-1.0.9-4.amzn2023.0.2.x86_64
  Verifying : mailcap-2.1.49-3.amzn2023.0.3.noarch
  Verifying : mod_http2-2.0.27-1.amzn2023.0.3.x86_64
  Verifying : mod_lua-2.4.65-1.amzn2023.0.1.x86_64
  Installed: apr-1.7.5-1.amzn2023.0.4.x86_64           apr-util-openssl-1.6.3-1.amzn2023.0.1.x86_64
  httpd-2.4.65-1.amzn2023.0.1.x86_64           httpd-core-2.4.65-1.amzn2023.0.1.x86_64
  libbrotli-1.0.9-4.amzn2023.0.2.x86_64          mailcap-2.1.49-3.amzn2023.0.3.noarch
                                                               generic-logos-httd-18.0.0-0.12.amzn2023.0.3.noarch
                                                               httpd-tools-2.4.65-1.amzn2023.0.1.x86_64
                                                               mod_http2-2.0.27-1.amzn2023.0.3.x86_64
                                                               mod_lua-2.4.65-1.amzn2023.0.1.x86_64

Complete!
[ec2-user@ip-10-0-0-232 ~]$ sudo systemctl start httpd
[ec2-user@ip-10-0-0-232 ~]$ sudo systemctl status httpd
● httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; disabled; preset: disabled)
   Active: active (running) since Sun 2025-09-21 06:37:08 UTC; 11s ago
     Docs: man:systemctl(1)
Main PID: 21835 (httpd)
   Status: "Total requests: 0; Idle/Busy workers 100/0;Requests/sec: 0; Bytes served/sec: 0 B/sec"
   Tasks: 177 (limit: 1111)
  Memory: 13.0M
    CPU: 72ms
   CGroup: /system.slice/httpd.service
           └─21835 /usr/sbin/httpd -DFOREGROUND
[21925 /usr/sbin/httpd -DFOREGROUND
[21929 /usr/sbin/httpd -DFOREGROUND
[21930 /usr/sbin/httpd -DFOREGROUND
[21931 /usr/sbin/httpd -DFOREGROUND

Sep 21 06:37:08 ip-10-0-0-232.ec2.internal systemd[1]: Starting httpd.service - The Apache HTTP Server...
Sep 21 06:37:08 ip-10-0-0-232.ec2.internal systemd[1]: Started httpd.service - The Apache HTTP Server.
Sep 21 06:37:08 ip-10-0-0-232.ec2.internal httpd[21835]: Server configured, listening on: port 80
[ec2-user@ip-10-0-0-232 ~]$ i-026fec5852e62358b (EpicReadsWebserver)
```

9. Open Public URL of EC2 and see if HTTPD works.



10. Install PHP and MySQL

```
sudo dnf update -y  
sudo dnf install -y php php-mysqlnd  
sudo dnf install -y mariadb105-server
```


11. Goto /var/www/html folder, download WordPress and unzip it.

```
cd /var/www/html/  
sudo wget https://wordpress.org/latest.tar.gz  
sudo tar -xvf latest.tar.gz
```

```
[ec2-user@ip-10-0-0-232 ~]$ cd /var/www/html/  
[ec2-user@ip-10-0-0-232 html]$ sudo wget https://wordpress.org/latest.tar.gz  
--2025-09-21 06:45:36 -- https://wordpress.org/latest.tar.gz  
Resolving wordpress.org (wordpress.org)... 198.143.164.252, 2607:f970:5:8002::c60f:a4fc  
Connecting to wordpress.org (wordpress.org)|198.143.164.252|:443... connected.  
HTTP request sent, awaiting response... 200 OK  
Length: 26925441 (26M) [application/octet-stream]  
Saving to: 'latest.tar.gz'  
  
latest.tar.gz          100%[=====] 25.68M 26.1MB/s   in 1.0s  
  
2025-09-21 06:45:38 (26.1 MB/s) - 'latest.tar.gz' saved [26925441/26925441]  
[ec2-user@ip-10-0-0-232 html]$
```

i-026fec5852e62358b (EpicReadsWebserver)

PublicIPs: 44.202.136.227 PrivateIPs: 10.0.0.232

12. Go into wordpress directory,

```
cd wordpress/  
# Create configuration file(Using sample config file)  
sudo cp wp-config-sample.php wp-config.php
```

#Adjust permissions for wordpress files

```
sudo chown -R apache:apache /var/www/html/wordpress/  
sudo chmod -R 755 /var/www/html/wordpress/
```

```
[ec2-user@ip-10-0-0-232 html]$ cd wordpress/  
[ec2-user@ip-10-0-0-232 wordpress]$ sudo cp wp-config-sample.php wp-config.php  
[ec2-user@ip-10-0-0-232 wordpress]$ sudo chown -R apache:apache /var/www/html/wordpress/  
[ec2-user@ip-10-0-0-232 wordpress]$ sudo chmod -R 755 /var/www/html/wordpress/  
[ec2-user@ip-10-0-0-232 wordpress]$
```

i-026fec5852e62358b (EpicReadsWebserver)

PublicIPs: 44.202.136.227 PrivateIPs: 10.0.0.232

13. Create another Private Subnet and associate it in the Private Route table.

<input checked="" type="checkbox"/> PrivateSubnet2-EpicReads	subnet-01e5d19a8fa97092a	<input checked="" type="checkbox"/> Available	vpc-0fbcb64b2f1b4a16f3 VPC-...	<input type="radio"/> Off	10.0.2.0/24	-
<input type="checkbox"/> PrivateSubnet1-EpicReads	subnet-07fba0299ab7689c0	<input checked="" type="checkbox"/> Available	vpc-0fbcb64b2f1b4a16f3 VPC-...	<input type="radio"/> Off	10.0.1.0/24	-
<input type="checkbox"/> -	subnet-0a87caa5987a358cd	<input checked="" type="checkbox"/> Available	vpc-00d70ddc6828a5629 DoN...	<input type="radio"/> Off	172.31.0.0/20	-
<input type="checkbox"/> -	subnet-0258f65b700f32cb7	<input checked="" type="checkbox"/> Available	vpc-00d70ddc6828a5629 DoN...	<input type="radio"/> Off	172.31.16.0/20	-
<input type="checkbox"/> -	subnet-0f71aef3ccffrfrdr4d	<input checked="" type="checkbox"/> Available	vnr-00r70ddr6828a5629 DoN	<input type="radio"/> Off	172.31.32.0/20	-

subnet-01e5d19a8fa97092a / PrivateSubnet2-EpicReads

Details Flow logs Route table Network ACL CIDR reservations Sharing Tags

Details	Subnet ID subnet-01e5d19a8fa97092a	Subnet ARN arn:aws:ec2:us-east-1:311141542113:subnet/subnet-01e5d19a8fa97092a	State <input checked="" type="checkbox"/> Available	Block Public Access <input type="radio"/> Off
IPv4 CIDR 10.0.2.0/24	Available IPv4 addresses 251	IPv6 CIDR -	IPv6 CIDR association ID -	Route table rtb-009782fcbe6a69866 Private-RT-EpicReads
Availability Zone use1-az4 (us-east-1b)	Network border group us-east-1	VPC vpc-0fbcb64b2f1b4a16f3 VPC-EpicReads	Auto-assign public IPv4 address No	Auto-assign IPv6 address No
Network ACL acl-09993ca117e789ae7	Default subnet			

Private-RT-EpicReads rtb-009782fcbe6a69866 2 subnets No vpc-0fb64b2f1b4a16f3 | VPC... 31114154...

rtb-009782fcbe6a69866 / Private-RT-EpicReads

Details | Routes | **Subnet associations** | Edge associations | Route propagation | Tags

Explicit subnet associations (2)

Name	Subnet ID	IPv4 CIDR	IPv6 CIDR
PrivateSubnet1-EpicReads	subnet-07fba0299ab7689c0	10.0.1.0/24	-
PrivateSubnet2-EpicReads	subnet-01e5d19a8fa97092a	10.0.2.0/24	-

Edit subnet associations

14. Create an RDS subnet group

Aurora & RDS > Subnet groups > Create DB subnet group

Create DB subnet group

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

Subnet group details

Name
EpicReads-SubnetGroup

Description
This is Epic Reads Subnet Group

VPC
VPC-EpicReads (vpc-0fb64b2f1b4a16f3)
3 Subnets, 2 Availability Zones

Add subnets

Availability Zones
Choose the Availability Zones that include the subnets you want to add.
Choose an availability zone
us-east-1a X us-east-1b X

Subnets
Choose the subnets that you want to add. The list includes the subnets in the selected Availability Zones.
Select subnets
PrivateSubnet1-EpicReads Subnet ID: subnet-07fba0299ab7689c0 CIDR: 10.0.1.0/24 PrivateSubnet2-EpicReads Subnet ID: subnet-01e5d19a8fa97092a CIDR: 10.0.2.0/24

(?) For Multi-AZ DB clusters, you must select 3 subnets in 3 different Availability Zones.

Subnets selected (2)

Availability zone	Subnet name	Subnet ID	CIDR block
us-east-1a	PrivateSubnet1-EpicReads	subnet-07fba0299ab7689c0	10.0.1.0/24
us-east-1b	PrivateSubnet2-EpicReads	subnet-01e5d19a8fa97092a	10.0.2.0/24

Create

15. Create MySQL database

Adriana Colaco

database-1

Successfully created database database-1

RDS has generated your database master password during the database creation and it will be displayed in the connection details. The only way to view your master password is to choose [View connection details](#) during database creation. You can modify your DB instance to create a new password at any time.

You can use settings from database-1 to simplify configuration of suggested database add-ons while we finish creating your DB for you.

[View connection details](#) [X](#)

database-1

[C](#) [Modify](#) [Actions ▾](#)

Summary				
DB identifier database-1	Status Available	Role Instance	Engine MySQL Community	Recommendations
CPU -	Class db.t4g.micro	Current activity	Region & AZ us-east-1b	

< [Connectivity & security](#) Monitoring Logs & events Configuration Zero-ETL integrations Maintenance & backups Data migrations - new >

Connectivity & security

Endpoint & port	Networking	Security
Endpoint database-1.c4xi6s2qytn.us-east-1.rds.amazonaws.com	Availability Zone us-east-1b	VPC security groups EpicReads-DBSG (sg-0a9dbbb9a50062d30) Active
Port 3306	VPC VPC-EpicReads (vpc-0fb64b2f1b4a16f3)	Publicly accessible No
	Subnet group epicreads-subnetgroup	Certificate authority Info rds-ca-rsa2048-g1
	Subnets subnet-01e5d19a8fa97092a subnet-07fba0299ab7689c0	Certificate authority date May 26, 2061, 05:04 (UTC+05:30)
	Network type IPv4	DB instance certificate expiration date September 21, 2026, 12:55 (UTC+05:30)

Connected compute resources (0) [Info](#)

Connections to compute resources that were created automatically by RDS are shown here. Connections to compute resources that were created manually aren't shown.

[Filter by compute resources](#)

< 1 > [Actions ▾](#)

Resource identifier	Resource type	Availability Zone	VPC security group	Compute resource security group	Connected proxy
No connected compute resources					

No connected compute resources that were created automatically to display.

[Set up EC2 connection](#) [Set up Lambda connection](#)

Proxies (0) [Create proxy](#)

[Filter by proxies](#)

Proxy identifier	Status	Engine family
No proxies		

You don't have any proxies.

[Create proxy](#)

Security group rules (2) [Create rule](#)

[Filter by Security group rules](#)

Security group	Type	Rule
EpicReads-DBSG (sg-0a9dbbb9a50062d30)	CIDR/IP - Inbound	223.177.210.62/32
EpicReads-DBSG (sg-0a9dbbb9a50062d30)	CIDR/IP - Outbound	0.0.0.0/0

Replication (1) [Create replication](#)

[Filter by Replication](#)

DB identifier	Role	Region & AZ	Replication source	Replication state	Lag
database-1	Instance	us-east-1b	-	-	-

16. Edit the Database security group to give access to web server security group.

Inbound security group rules successfully modified on security group (sg-0a9dbbb9a50062d30 | EpicReads-DBSG)

sg-0a9dbbb9a50062d30 - EpicReads-DBSG

Details

Security group name EpicReads-DBSG	Security group ID sg-0a9dbbb9a50062d30	Description Created by RDS management console	VPC ID vpc-0fb64b2f1b4a16f3
Owner 311141542113	Inbound rules count 1 Permission entry	Outbound rules count 1 Permission entry	

Inbound rules | Outbound rules | Sharing - new | VPC associations - new | Tags

Inbound rules (1)

Name	Security group rule ID	IP version	Type	Protocol	Port range	Source
-	sgr-00bcd7d1dccf8868d	-	MySQL/Aurora	TCP	3306	sg-0d680468fe4d9859f...

17. Test if you can access the DB from webserver

Successfully created database database-1

RDS has generated your database master password during the database creation and it will be displayed in the connection details. The only way to view your master password is to choose View connection details during database creation. You can modify your DB instance to create a new password at any time.

You can use settings from database-1

Connection details to your database database-1

This is the only time you can view this password. Copy and save the password for your reference. If you lose the password, you must modify your database to change it. You can use a SQL client application or utility to connect to your database.

Learn about connecting to your database [\[?\]](#)

Master username

Master password copied

Endpoint

database-1.c4xii6s2qytn.us-east-1.rds.amazonaws.com

Engine MySQL Community
Region & AZ us-east-1b

Recommendations

Connectivity & security

Endpoint & port

Endpoint database-1.c4xii6s2qytn.us-east-1.rds.amazonaws.com

Networking

Availability Zone us-east-1b

Security

VPC security groups EpicReads-DBSG (sg-0a9dbbb9a50062d30)
Active

Option 1:

```
mysql -h database-1.c4xii6s2qytn.us-east-1.rds.amazonaws.com -u admin -p
```

```
[ec2-user@ip-10-0-0-232 wordpress]$ mysql -h database-1.c4xii6s2qytn.us-east-1.rds.amazonaws.com -u admin --password=***** wordpress
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MySQL connection id is 33
Server version: 8.0.42 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 [wordpress]>
```

i-026fec5852e62358b (EpicReadsWebserver)

Public IPs: 44.202.136.227 Private IPs: 10.0.0.232

Option 2:

```
mysql -h database-1.c4xii6s2qytn.us-east-1.rds.amazonaws.com -u admin --
password=***** wordpress
```

```
[ec2-user@ip-10-0-0-232 wordpress]$ mysql -h database-1.c4xiis2qytn.us-east-1.rds.amazonaws.com -u admin --password=***** wordpress
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MySQL connection id is 33
Server version: 8.0.42 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 [wordpress]> [REDACTED]
i-026fec5852e62358b (EpicReadsWebserver)
PublicIPs: 44.202.136.227 PrivateIPs: 10.0.0.232
```

18. Configure WordPress

Add database details

Navigate to WordPress secret key generator available at

<https://api.wordpress.org/secret-key/1.1/salt/> and copy the content generated on page.

Paste into your config file.

Add this line at the end "define('W3TC_CONFIG_DATABASE', true);"

sudo vi wp-config.php

```
/*
** Database settings - You can get this info from your web host */
/** The name of the database for WordPress */
define( 'DB_NAME', 'wordpress' ); [REDACTED]

/** Database username */
define( 'DB_USER', 'admin' ); [REDACTED]

/** Database password */
define( 'DB_PASSWORD', '*****' ); [REDACTED]

/** Database hostname */
define( 'DB_HOST', 'database-1.c4xiis2qytn.us-east-1.rds.amazonaws.com' ); [REDACTED]

/** Database charset to use in creating database tables. */
define( 'DB_CHARSET', 'utf8' );

/** The database collate type. Don't change this if in doubt. */
define( 'DB_COLLATE', '' );

/**#@+
 * Authentication unique keys and salts.
 *
 * Change these to different unique phrases! You can generate these using
 * the {@link https://api.wordpress.org/secret-key/1.1/salt/ WordPress.org secret-key service}.
 *
 * You can change these at any point in time to invalidate all existing cookies.
 * This will force all users to have to log in again.
 *
 * @since 2.6.0
 */
define('AUTH_KEY',      'g-#-04yw{9jo.8]x_1|>|:#(ljaVw<e)4PKHyb1c`WVcq?iR7SX]CO2M%Q|i5H5G');
define('SECURE_AUTH_KEY', 'o/J79.1733F1~~~;7@Z)id2X&8*!L1stQ|N uGXKfe+.18D);|?eM.TFl*RJjw(');
define('LOGGED_IN_KEY',   ')|-|V0Kt]k-Xy[A5DLk-@8=$vJi*wX#i@,5$+-X<`U7.(qjsm),G667Z9%It&-qn0');
define('NONCE_KEY',       '!p=-ztf.%x2g(9?]i=(Rh~5q2*.1z->u`6p5UeQ=r0#Te`eh`A$F+SheBn(');
define('AUTH_SALT',        '$Yh4(n JU;nTK*JZfeo8$U_h_eLIG9PEl~rZ@0 +> #Y.b;z>`VU)-z|Z_hrzj;');
define('SECURE_AUTH_SALT', 'e9tP1[NH|)gtQcd2B;.=aYL*O1?6pu[ML@A5,nsD/O,LzBOZH@|#E8aIPKzE/');
define('LOGGED_IN_SALT',   '3(<1nCmlxw.Khtbjd-Rqv ~ I #dB!<)ehc8JF<p|)Onsj c/p0<HVU+[B6t9;j');
define('NONCE_SALT',       'cn2F,owj<(q/.!+_m58<%90x:n{$bgYQo-o-(PKZ|8+:7s&LNr?D-BLEV@{1~R3');

/**#@-*/
[REDACTED]
```

19. Commands to deploy WordPress.

sudo yum install php-xml -y

cd ..

sudo cp -r wordpress/* /var/www/html

sudo chown -R apache:apache /var/www/html

sudo systemctl start httpd.service

```
sudo systemctl restart php-fpm
```

```
[ec2-user@ip-10-0-0-232 wordpress]$ sudo yum install php-xml -y
Last metadata expiration check: 1:39:12 ago on Sun Sep 21 06:35:43 2025.
Package php8.4-xml-8.4.10-1.amzn2023.0.1.x86_64 is already installed.
Dependencies resolved.
Nothing to do.
Complete!
[ec2-user@ip-10-0-0-232 wordpress]$ cd ..
[ec2-user@ip-10-0-0-232 html]$ sudo cp -r wordpress/* /var/www/html
[ec2-user@ip-10-0-0-232 html]$ sudo chown -R apache:apache /var/www/html
[ec2-user@ip-10-0-0-232 html]$ sudo systemctl start httpd.service
[ec2-user@ip-10-0-0-232 html]$ sudo systemctl restart php-fpm
[ec2-user@ip-10-0-0-232 html]$ █
```

i-026fec5852e62358b (EpicReadsWebserver)

PublicIPs: 44.202.136.227 PrivateIPs: 10.0.0.232

Restart HTTPD if you don't see the correct WordPress setup from next step.

```
sudo systemctl restart httpd
```

```
sudo systemctl status httpd
```

```
[ec2-user@ip-10-0-0-232 html]$ sudo systemctl restart httpd
[ec2-user@ip-10-0-0-232 html]$ sudo systemctl status httpd
● httpd.service - The Apache HTTP Server
    Loaded: loaded (/usr/lib/systemd/system/httpd.service; enabled; preset: disabled)
    Drop-In: /usr/lib/systemd/system/httpd.service.d
              └─php-fpm.conf
      Active: active (running) since Sun 2025-09-21 08:33:17 UTC; 21s ago
        Docs: man:httpd.service(8)
     Main PID: 34510 (httpd)
       Status: "Total requests: 0; Idle/Busy workers 100/0;Requests/sec: 0; Bytes served/sec: 0 B/sec"
          Tasks: 177 (limit: 1111)
         Memory: 13.0M
            CPU: 76ms
           CGroub: /system.slice/httpd.service
                     ├─34510 /usr/sbin/httpd -DFOREGROUND
                     ├─34511 /usr/sbin/httpd -DFOREGROUND
                     ├─34512 /usr/sbin/httpd -DFOREGROUND
                     ├─34513 /usr/sbin/httpd -DFOREGROUND
                     ├─34514 /usr/sbin/httpd -DFOREGROUND

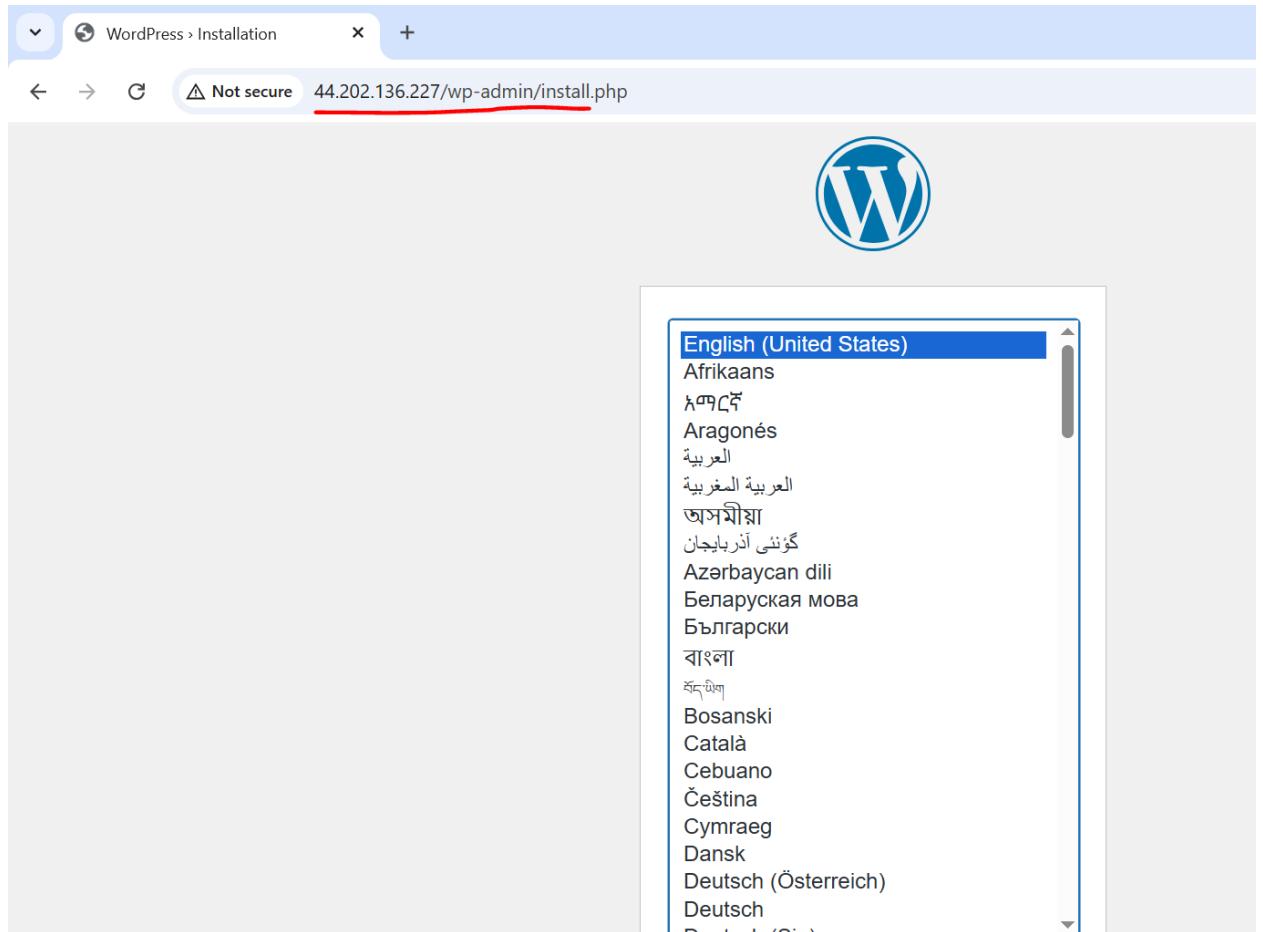
Sep 21 08:33:17 ip-10-0-0-232.ec2.internal systemd[1]: Starting httpd.service - The Apache HTTP Server...
Sep 21 08:33:17 ip-10-0-0-232.ec2.internal systemd[1]: Started httpd.service - The Apache HTTP Server.
Sep 21 08:33:17 ip-10-0-0-232.ec2.internal httpd[34510]: Server configured, listening on: port 80
[ec2-user@ip-10-0-0-232 html]$ █
```

i-026fec5852e62358b (EpicReadsWebserver)

PublicIPs: 44.202.136.227 PrivateIPs: 10.0.0.232

20. Verify and complete WordPress Setup on public IP of EC2 webserver

<http://<PublicIP>/wp-admin/>



Complete the WordPress setup as shown in the video below.



21. Create a blog post.

Add Post < Adrina Colaco's Blog +

⚠ Not secure 44.202.136.227/wp-admin/post-new.php

No title · Post Ctrl+K

Save draft 🖥️ 📁 Publish

Add title

Type / to choose a block

+ []

Post Block

🔗 No title

Set featured image

Add an excerpt...

Last edited a second ago.

Status	Draft
Publish	Immediately
Slug	9
Author	adcola13
Template	Single Posts
Discussion	Open
Format	Standard

Final blog post should look like:

A screenshot of a web browser window. The address bar shows a warning icon followed by "Not secure" and the URL "44.202.136.227/index.php/2025/09/21/my-first-blog/". The main content area displays a blog post with the title "my first blog" and the date "2025/09/21". The browser interface includes standard navigation buttons (back, forward, search), a star icon, and a refresh button. Below the address bar, there are icons for "Edit Site", "1", "New", and "Edit Post". On the right side of the browser, there is a user profile section with the text "Howdy, adcola13" and a small profile picture.

My First Blog

Written by adcola13 in Uncategorized

This Blog is created as a part of a DevOps Micro Internship(DMI) Cohort.
Run by Pravin Mishra.



22. Create another Public Subnet

You have successfully created 1 subnet: subnet-0645b429d9046f316

Subnets (1) Info							Last updated less than a minute ago	Actions	Create subnet
	Name	Subnet ID	State	VPC	Block Public...	IPv4 CIDR	IPv6 CIDR		
<input type="checkbox"/>	PublicSubnet2-EpicReads	subnet-0645b429d9046f316	Available	vpc-0fb64b2f1b4a16f3 VPC...	Off	10.0.3.0/24	-		

Select a subnet

You have successfully created 1 subnet: subnet-0645b429d9046f316

Subnets (15) Info							Last updated 1 minute ago	Actions	Create subnet
	Name	Subnet ID	State	VPC	Block Public...	IPv4 CIDR	IPv6 CIDR		
<input type="checkbox"/>	PublicSubnet2-EpicReads	subnet-0645b429d9046f316	Available	vpc-0fb64b2f1b4a16f3 VPC...	Off	10.0.3.0/24	-		
<input type="checkbox"/>	PublicSubnet1-EpicReads	subnet-068c97297d2a8b5d8	Available	vpc-0fb64b2f1b4a16f3 VPC...	Off	10.0.0.0/24	-		
<input type="checkbox"/>	PrivateSubnet2-EpicReads	subnet-01e5d19a8fa97092a	Available	vpc-0fb64b2f1b4a16f3 VPC...	Off	10.0.2.0/24	-		
<input type="checkbox"/>	PrivateSubnet1-EpicReads	subnet-07fba0299ab7689c0	Available	vpc-0fb64b2f1b4a16f3 VPC...	Off	10.0.1.0/24	-		

Select a subnet

23. Add the newly created Public Subnet to Public Route Table.

You have successfully updated subnet associations for rtb-0bddae5b816c74f7 / Public-RT-EpicReads.

Route tables (1/5) Info							Last updated less than a minute ago	Actions	Create route table
	Name	Route table ID	Explicit subnet associa...	Edge associations	Main	VPC	Owner ID		
<input checked="" type="checkbox"/>	Public-RT-EpicReads	rtb-0bddae5b816c74f7	2 subnets	-	No	vpc-0fb64b2f1b4a16f3 VPC...	31114154...		
<input type="checkbox"/>	Private-RT-EpicReads	rtb-009782fcbe6a69866	2 subnets	-	No	vpc-0fb64b2f1b4a16f3 VPC...	31114154...		

rtb-0bddae5b816c74f7 / Public-RT-EpicReads

Details | Routes | **Subnet associations** | Edge associations | Route propagation | Tags

Explicit subnet associations (2)

Edit subnet associations

Explicit subnet associations (2)						
	Name	Subnet ID	IPv4 CIDR	IPv6 CIDR		
<input type="checkbox"/>	PublicSubnet1-EpicReads	subnet-068c97297d2a8b5d8	10.0.0.0/24	-		
<input checked="" type="checkbox"/>	PublicSubnet2-EpicReads	subnet-0645b429d9046f316	10.0.3.0/24	-		

24. Modify setting for both Public subnets to enable “auto assign” IP address.

The screenshot shows the 'Edit subnet settings' page for a subnet named 'PublicSubnet2-EpicReads'. Under the 'Auto-assign IP settings' section, the checkbox 'Enable auto-assign public IPv4 address' is checked, while 'Enable auto-assign customer-owned IPv4 address' is unchecked. A red arrow points to the checked checkbox. Below this section is the 'Resource-based name (RBN) settings' section, which contains an unchecked checkbox for 'Enable resource name DNS A record on launch'.

25. Create an image of the existing Web Server.

The screenshot shows the 'Amazon Machine Images (AMIs)' page with one item listed: 'WP-Application' (AMI ID: ami-059e547d93eaf3835). The item is selected, indicated by a blue border around its row. The details page for this AMI is open, showing various configuration parameters such as AMI ID, Name, Root device name, Boot mode, and Description. The 'Status' field shows 'Available' with a green checkmark.

26. Launch Template

Launch Templates (1/1) [Info](#)

Launch Template ID	Launch Template Name	Default Version	Latest Version	Create Time	Created By	Manage
lt-0b1181c814a6a2373	EpicReads-App-AMI	1	1	2025-09-21T16:26:58.000Z	arn:aws:iam::311141542113:root	false

EpicReads-App-AMI (lt-0b1181c814a6a2373)

Launch template details

Launch template ID lt-0b1181c814a6a2373	Launch template name EpicReads-App-AMI	Default version 1	Owner arn:aws:iam::311141542113:root
--	---	--------------------------------------	---

Details [Versions](#) [Template tags](#)

Launch template version details

Version 1 (Default)	Description This AMI has working applications and its configurations	Date created 2025-09-21T16:26:58.000Z	Created by arn:aws:iam::311141542113:root
--	---	--	--

[Instance details](#) [Storage](#) [Resource tags](#) [Network interfaces](#) [Advanced details](#)

27. Create Auto Scaling Group

Auto Scaling groups (1/1) [Info](#)

Name	Launch template/configuration	Instances	Status	Desired capacity	Min	Max	Availability Zones	Creation time
EpicReads-ASG	EpicReads-App-AMI Version Default	0	Updating capacity...	2	1	4	2 Availability Zones	Sun Sep 21 2025 22:13:...

Auto Scaling group: EpicReads-ASG

[Details](#) [Integrations - new](#) [Automatic scaling](#) [Instance management](#) [Instance refresh](#) [Activity](#) [Monitoring](#)

EpicReads-ASG Capacity overview

Desired capacity 2	Scaling limits (Min - Max) 1 - 4	Desired capacity type Units (number of instances)	Status Updating capacity
-----------------------	-------------------------------------	--	-----------------------------

Date created
Sun Sep 21 2025 22:13:50 GMT+0530 (India Standard Time)

Launch template [Edit](#)

28. Verify the new instances that are created as a part of autoscaling

Instances (1/3) Info

Last updated less than a minute ago | Connect | Instance state | Actions | Launch instances

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Publ
EpicReadsWebserver	i-026fec5852e62358b	Running	t2.micro	2/2 checks passed	View alarms +	us-east-1a	-	44.20
	i-07c3887b9911a238f	Running	t2.micro	2/2 checks passed	View alarms +	us-east-1a	-	3.84.
	i-029d36d5a93730c9f	Running	t2.micro	2/2 checks passed	View alarms +	us-east-1b	-	54.24

i-029d36d5a93730c9f

Details | Status and alarms | Monitoring | Security | Networking | Storage | Tags

Instance summary

Instance ID: i-029d36d5a93730c9f
Public IPv4 address: 54.242.24.159 [Open address]
Instance state: Running
Private IP DNS name (IPv4 only): ip-10-0-3-132.ec2.internal
Hostname type: IP name: ip-10-0-3-132.ec2.internal
Answer private resource DNS name:
Instance type: t2.micro
VPC ID: vpc-0fb64b2f1b4a16f3 (VPC-EpicReads)
Auto-assigned IP address: 54.242.24.159 [Public IP]

Private IPv4 addresses: 10.0.3.132
Public DNS:
Elastic IP addresses:
AWS Compute Optimizer finding: Opt-in to AWS Compute Optimizer for recommendations.

29. Create Target Group

EpicReads-TG

2 targets registered successfully to EpicReads-TG.

EpicReads-TG

Details

arn:aws:elasticloadbalancing:us-east-1:311141542113:targetgroup/EpicReads-TG/58600c21a671331e

Target type	Protocol	Protocol version	VPC
Instance	HTTP: 80	HTTP1	vpc-0fb64b2f1b4a16f3
IP address type	Load balancer		
IPv4	None associated		

Total targets	Healthy	Unhealthy	Unused	Initial	Draining
2	0	0	2	0	0
	0 Anomalous				

Distribution of targets by Availability Zone (AZ)

Select values in this table to see corresponding filters applied to the Registered targets table below.

Targets | Monitoring | Health checks | Attributes | Tags

Registered targets (2) Info

Anomaly mitigation: Not applicable | Deregister | Register targets

Target groups route requests to individual registered targets using the protocol and port number specified. Health checks are performed on all registered targets according to the target group's health check settings. Anomaly detection is automatically applied to HTTP/HTTPS target groups with at least 3 healthy targets.

Filter targets

Instance ID	Name	Port	Zone	Health status	Health status details	Administrative ove...	Override details
i-07c3887b9911a238f		80	us-east-1a (us...)	Unused	Target group is not co...	-	-
i-029d36d5a93730c9f		80	us-east-1b (us...)	Unused	Target group is not co...	-	-

30. Launch Load balancer

EpicReads-LB

Details

Load balancer type Application	Status Provisioning	VPC vpc-0fb64b2f1b4a16f3	Load balancer IP address type IPv4
Scheme Internet-facing	Hosted zone Z355XDOTRQ7X7K	Availability Zones subnet-0645b429d9046f316 us-east-1b (use1-az2) subnet-068c97297d2a8b5d8 us-east-1a (use1-az2)	Date created September 21, 2025, 22:35 (UTC+05:30)
Load balancer ARN arn:aws:elasticloadbalancing:us-east-1:311141542113:loadbalancer/app/EpicReads-LB/e288496cd b504dd4		DNS name Info EpicReads-LB-1558287041.us-east-1.elb.amazonaws.com (A Record)	

Listeners and rules Network mapping Resource map Security Monitoring Integrations Attributes Capacity Tags

Listeners and rules (1) [Info](#)

A listener checks for connection requests on its configured protocol and port. Traffic received by the listener is routed according to the default action and any additional rules.

Protocol:Port	Default action	Rules	ARN	Security policy	Default SSL/TLS certificate	mTLS
HTTP:80	Forward to target group EpicReads-TG : 1 (100%)	1 rule	ARN	Not applicable	Not applicable	Not applicable

31. Run the DNS from load balancer into your browser.

Load balancers (1/1)

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

Name	State	Type	Scheme	IP address type	VPC ID	Availability Zones	Security groups
EpicReads-LB	Active	application	Internet-facing	IPv4	vpc-0fb64b2f1b4a16f3	2 Availability Zones	2 Security group

Load balancer: EpicReads-LB

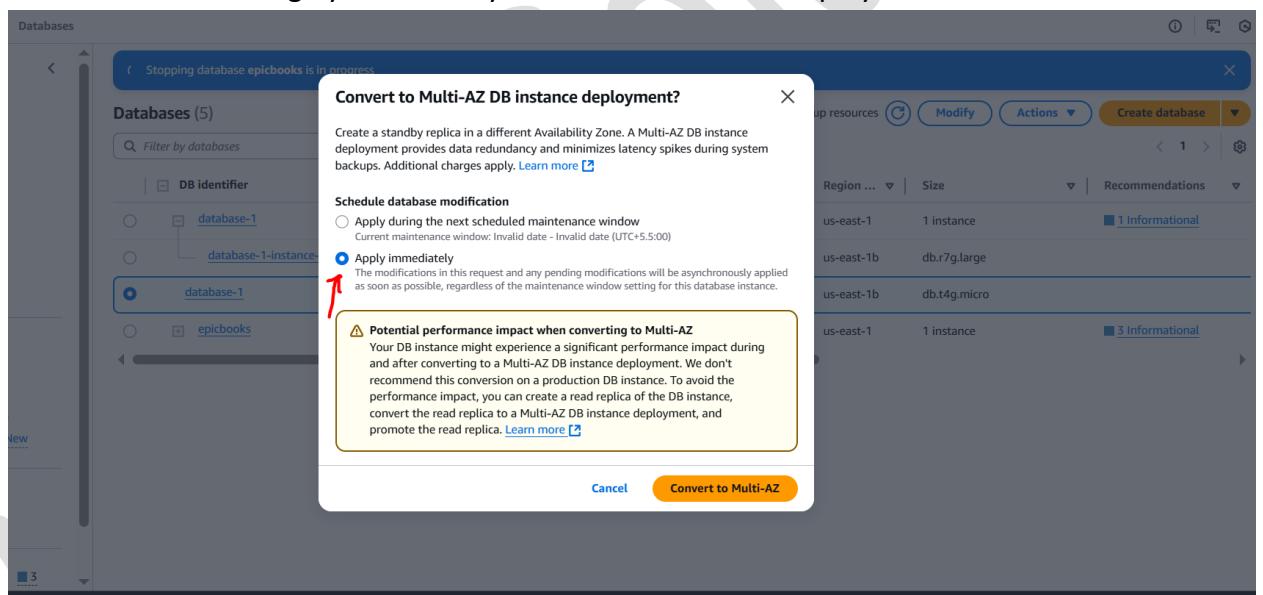
Details Listeners and rules Network mapping Resource map Security Monitoring Integrations Attributes Capacity Tags

Details

Load balancer type Application	Status Active	VPC vpc-0fb64b2f1b4a16f3	Load balancer IP address type IPv4
Scheme Internet-facing	Hosted zone Z355XDOTRQ7X7K	Availability Zones subnet-0645b429d9046f316 us-east-1b (use1-az2) subnet-068c97297d2a8b5d8 us-east-1a (use1-az2)	Date created September 21, 2025, 22:35 (UTC+05:30)
Load balancer ARN arn:aws:elasticloadbalancing:us-east-1:311141542113:loadbalancer/app/EpicReads-LB/e288496cd		DNS name Info EpicReads-LB-1558287041.us-east-1.elb.amazonaws.com (A Record)	

The screenshot shows a web browser window with the title bar "Adrina Colaco's Blog". The address bar contains the URL "epicreads-lb-1558287041.us-east-1.elb.amazonaws.com", which is highlighted with a red underline. The main content area displays the heading "Blog" and "My First Blog". Below the heading, a text block states: "This Blog is created as a part of a DevOps Micro Internship(DMI) Cohort. Run by Pravin Mishra." At the bottom of the page, the date "September 21, 2025" is visible.

32. Make the database highly available by “Convert to Multi AZ Deployment”.



Look out for Availability and Multi-AZ status.

Summary		Status		Role		Engine		Recommendations	
DB identifier database-1	Available	Status	Class	Instance	Current activity	MySQL Community	Region & AZ		
CPU 	2.87%		db.t4g.micro	0 Connections			us-east-1b		

< Connectivity & security | Monitoring | Logs & events | Configuration | Zero-ETL integrations | Maintenance & backups | Data migrations - new | >

Instance			
Configuration	Instance class	Primary storage	Monitoring
DB instance ID database-1	Instance class db.t4g.micro	Encryption Not enabled	Monitoring type Database Insights - Standard
Engine version 8.0.42	vCPU 2	Storage type General Purpose SSD (gp2)	Performance Insights Disabled
RDS Extended Support Disabled	RAM 1 GB	Storage 20 GiB	Enhanced Monitoring Disabled
DB name wordpress	Availability	Provisioned IOPS -	
License model General Public License	Master username admin	Storage throughput -	
Option groups default:mysql-8-0 In sync	Master password *****	Storage autoscaling Enabled	
Amazon Resource Name (ARN) arnaws:rds:us-east-1:311141542113:db:database-1	IAM DB authentication Not enabled	Maximum storage threshold 1000 GiB	
Resource ID db-XFIFHP5YAIWB5WLXEQI6A5S6E	Multi-AZ Yes	Storage file system configuration Current	
Created time September 21, 2025, 12:57 (UTC+05:30)	Secondary Zone -		
DB instance parameter group default.mysql8.0 In sync			
Deletion protection Disabled			
Architecture settings Non-multitenant architecture			

33. Create a Read replica of the database.

Enable By going to Databases → Select database → Modify →

Backup

Enable automated backups
Creates a point-in-time snapshot of your database

⚠ Please note that automated backups are currently supported for InnoDB storage engine only. If you are using MyISAM, refer to details [here](#).

Backup retention period Info
The number of days (1-35) for which automatic backups are kept.
1 day

Actions → Read Replica

34. Run the DNS from load balancer into your browser again.



My First Blog

This Blog is created as a part of a DevOps Micro Internship(DMI) Cohort.
Run by Pravin Mishra.

September 21, 2025



The End