

# Lab 3: Creating a Database Layer in Your Amazon VPC Infrastructure

© 2025 Amazon Web Services, Inc. or its affiliates. All rights reserved. This work may not be reproduced or redistributed, in whole or in part, without prior written permission from Amazon Web Services, Inc. Commercial copying, lending, or selling is prohibited. All trademarks are the property of their owners.

Note: Do not include any personal, identifying, or confidential information into the lab environment. Information entered may be visible to others.

Corrections, feedback, or other questions? Contact us at [AWS Training and Certification](#).

## Lab overview

A backend database plays an important role in any environment, and the security and access control to this critical resource is vital to any architecture. In this lab, you create an Amazon Aurora database (DB) cluster to manage a MySQL database and an Application Load Balancer (ALB). The Amazon Web Services (AWS) Security pillar of the Well-Architected Framework recommends keeping people away from data; as such, the database is separated from the front end using the Application Load Balancer. The Application Load Balancer routes traffic to healthy Amazon Elastic Compute Cloud (Amazon EC2) instances that hosts the front-end application. This provides high availability and allow communication to the database to happen behind the Application Load Balancer in a private subnet.

## Objectives

By the end of this lab, you will be able to do the following:

- Create an Amazon Relational Database Service (Amazon RDS) database instance.
- Create an Application Load Balancer.
- Create an HTTP listener for the Application Load Balancer.
- Create a target group.
- Register targets with a target group.
- Test the load balancer and the application connectivity to the database.
- Review the Amazon RDS DB instance metadata using the console.
- Optional Task: Create an Amazon RDS read replica in a different AWS Region.

## Prerequisites

This lab requires the following:

- Access to a notebook computer with Wi-Fi and Microsoft Windows, macOS, or Linux (Ubuntu, SuSE, or Red Hat)
- An internet browser, such as Chrome, Firefox, or Microsoft Edge
- A plaintext editor

## Start lab

1. To launch the lab, at the top of the page, choose **Start Lab**.

**Caution:** You must wait for the provisioned AWS services to be ready before you can continue.

2. To open the lab, choose **Open Console**.

You are automatically signed in to the AWS Management Console in a new web browser tab.

**Warning:** Do not change the **Region** unless instructed.

The screenshot shows the AWS Training and Certification interface for a lab titled "Architecting on AWS - Lab 3 - Create a database layer in your Amazon VPC infrastructure". The left sidebar displays "Lab Information" (Expires Apr 18 at 11:21 PM, 1 hour 30 minutes, Available languages: English), "Resources" (DBClusterParameterGroup: labstack-bf420498-e8ed-4, LabPassword: Q4Kn7dGTIOR, Region: us-west-2, RemoteRegion: US East (N. Virginia)), and "Lab Content" (Privacy, Site terms, Cookie preferences). A green banner at the top right says "Lab is ready. Open the console to begin. Keep the default region. Your lab will be active until Apr 18 at 11:21 PM. Tip: open the console in a new window to see it side-by-side with these instructions." The main content area is titled "Lab 3: Creating a Database Layer in Your Amazon VPC Infrastructure" and includes sections for "Objectives" (Create an Amazon Relational Database Service (Amazon RDS) database instance) and "Lab overview" (A backend database plays an important role in any environment, and the security and access control to this critical resource is vital to any architecture. In this lab, you create an Amazon Aurora database (DB) cluster to manage a MySQL database and an Application Load Balancer (ALB). The Amazon Web Services (AWS) Security pillar of the Well-Architected Framework recommends keeping people away from data; as such, the database is separated from the front end using the Application Load Balancer. The Application Load Balancer routes traffic to healthy Amazon Elastic Compute Cloud (Amazon EC2) instances that hosts the front-end application. This provides high availability and allow communication to the database to happen behind the Application Load Balancer in a private subnet.)

## Common sign-in errors

**Error: You must first sign out**

The screenshot shows the "Amazon Web Services Sign In" page. It features a large orange header and a central message: "You must first log out before logging into a different AWS account." Below the message is a link: "To logout, click here".

If you see the message, **You must first log out before logging into a different AWS account**:

- Choose the **click here** link.
- Close your **Amazon Web Services Sign In** web browser tab and return to your initial lab page.
- Choose **Open Console** again.

**Error: Choosing Start Lab has no effect**

In some cases, certain pop-up or script blocker web browser extensions might prevent the **Start Lab** button from working as intended. If you experience an issue starting the lab:

- Add the lab domain name to your pop-up or script blocker's allow list or turn it off.
- Refresh the page and try again.

## Resources

**DBClusterParameterGroup:** labstack-bf420498-e8ed-48d3-8727-cd8a9792f173-9qaiwkxz56u4hbnz8cqrbo-0-rdsdbclusterparametergroup-opdcurm5zhnd

**LabPassword:** 85A4XUadNq4I

**Region:** us-west-2

**RemoteRegion:** US East (N. Virginia)

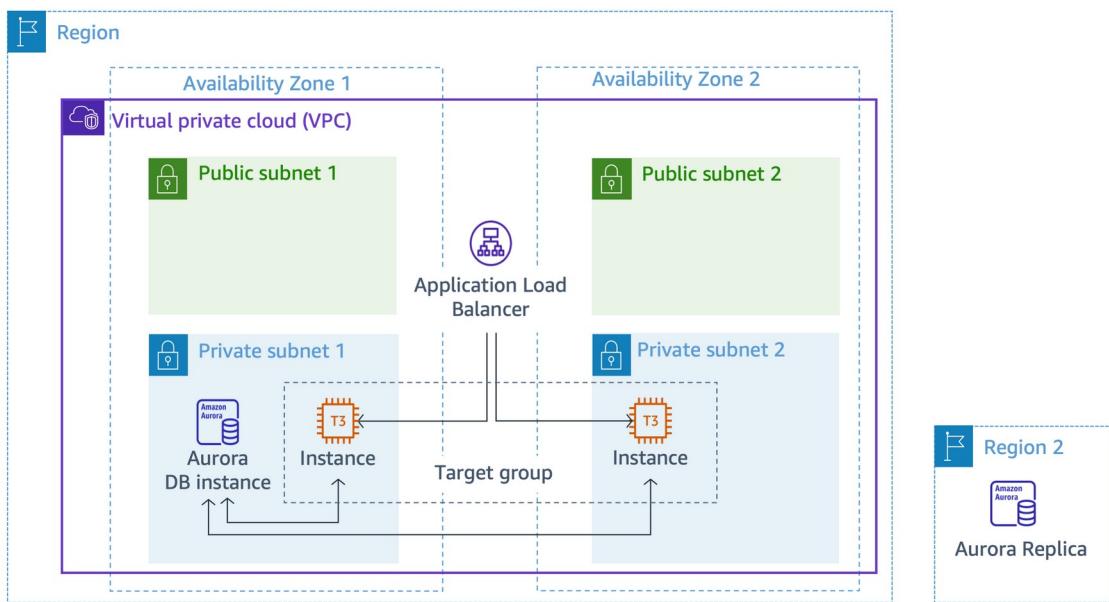
## Scenario

Your team has been tasked with prototyping an architecture for a new web-based application. To define your architecture, you need to have a better understanding of load balancers and managed databases, such as Amazon RDS.

## Lab environment

The lab environment provides you with the following resources to get started: an Amazon Virtual Private Cloud (Amazon VPC), underlying necessary network structure, three security groups to control inbound and outbound traffic, two EC2 instances in a private subnet, and an associated EC2 instance profile. The instance profile contains the permissions necessary to allow the AWS Systems Manager Session Manager feature to access the EC2 instance.

The following diagram shows the expected architecture of the important lab resources you build and how they should be connected at the end of the lab.



## AWS services not used in this lab

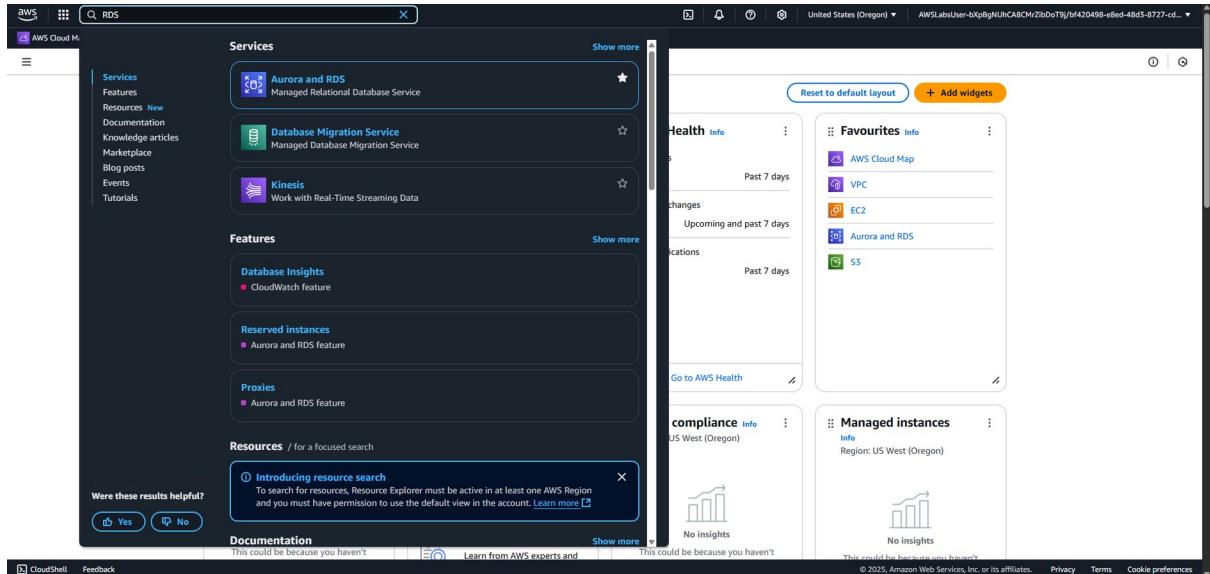
AWS services not used in this lab are turned off in the lab environment. In addition, the capabilities of the services used in this lab are limited to only what the lab requires. Expect to receive errors when accessing other services or performing actions beyond those provided in this lab guide.

## Task 1: Create an Amazon RDS database

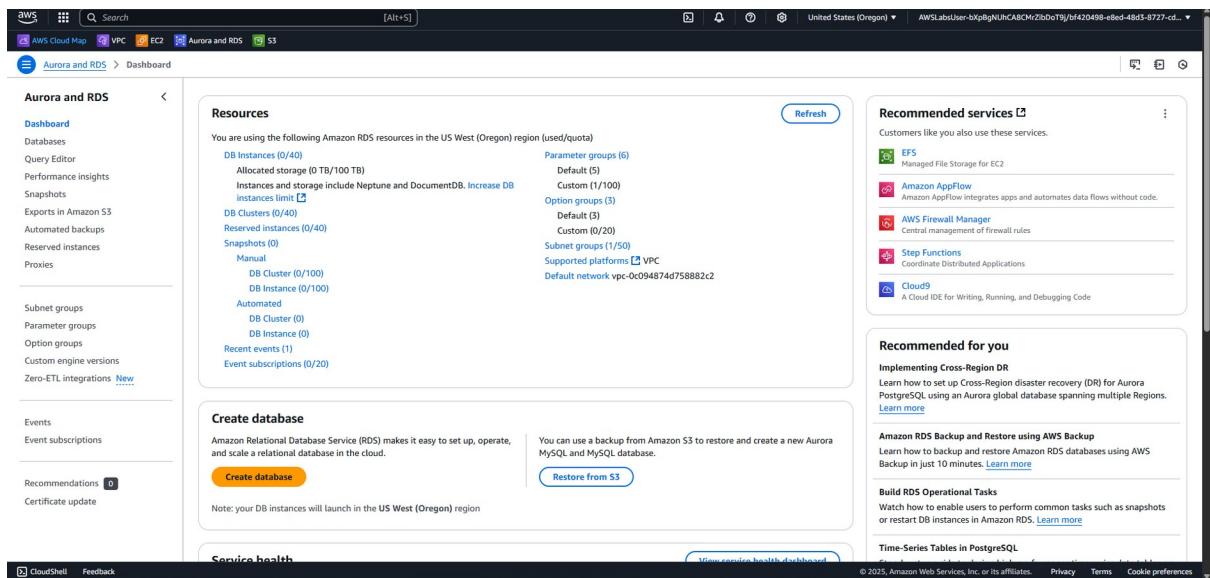
In this task, you create an Aurora DB cluster that is compatible with MySQL. An Aurora DB cluster consists of one or more DB instances and a cluster volume that manages the data for those DB instances.

**Learn more:** Amazon Aurora is a fully managed relational database engine that is compatible with MySQL and PostgreSQL. Aurora is part of the managed database service, Amazon RDS. Amazon RDS is a web service that makes it easier to set up, operate, and scale a relational database in the cloud. For more information, see [What is Amazon Aurora?](#).

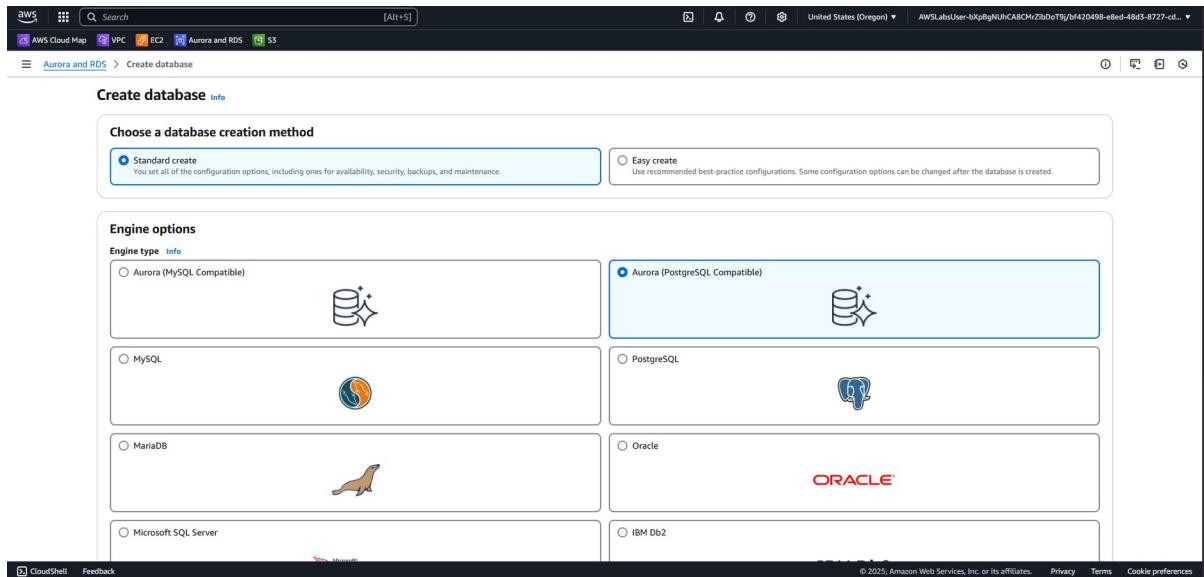
- At the top of the AWS Management Console, in the search bar, search for and choose **RDS**.



- In the left navigation pane, choose **Databases**.

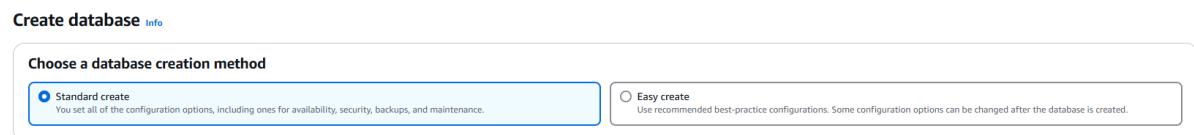


- Choose **Create database**.



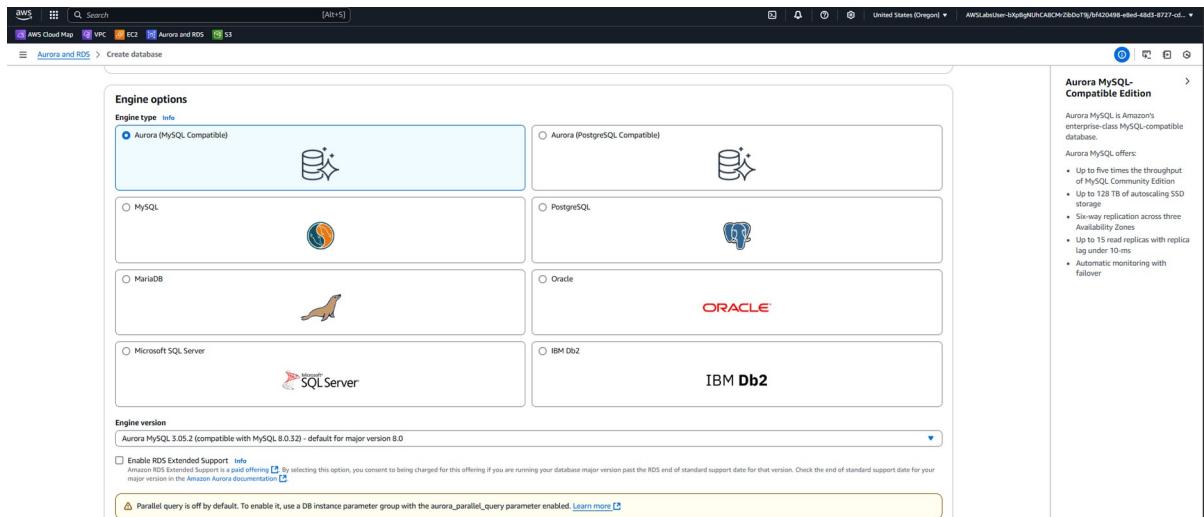
The **Create database** page is displayed.

- In the **Choose a database creation method** section, select **Standard create**.



- In the **Engine options** section, configure the following:

- Engine type:** Select **Aurora (MySQL Compatible)**.



- In the **Templates** section, select **Dev/Test**.



9. In the **Settings** section, configure the following:

- **DB cluster identifier:** Enter **aurora**.
- **Master username:** Enter **dbadmin**.
- Under **Credentials management** select **Self managed**
  - **Master password:** Paste the **LabPassword (85A4XUadNq4I)** value from the left side of these lab instructions.
  - **Confirm master password:** Paste the **LabPassword (85A4XUadNq4I)** value from the left side of these lab instructions.

**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.

The DB cluster identifier is case-insensitive, but is stored as all lowercase (as in "mydbcluster"). Constraints: 1 to 63 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.

1 to 32 alphanumeric characters. The first character must be a letter.

**Credentials management**  
You can use AWS Secrets Manager or manage your master user credentials.

**Managed in AWS Secrets Manager - most secure**  
RDS generates a password for you and manages it throughout its lifecycle using AWS Secrets Manager.

**Self managed**  
Create your own password or have RDS create a password that you manage.

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

**Master password** [Info](#)  
  
Password strength **Very strong**  
Minimum constraints: At least 8 printable ASCII characters. Can't contain any of the following symbols: / /\* @

**Confirm master password** [Info](#)

10. In the **Instance configuration** section, configure the following:

- **DB instance class:** Select **Burstable classes (includes t classes)**.
- From the dropdown menu, choose the **db.t3.medium** instance type.

**Cluster storage configuration** [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 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.

**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.

**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**  
 **Include previous generation classes**

**Serverless v2**  
 **Memory optimized classes (includes r classes)**  
 **Burstable classes (includes t classes)**

**db.t3.medium**  
2 vCPUs 4 GiB RAM Network: Up to 2,085 Mbps

11. In the **Availability & durability** section, for **Multi-AZ deployment**, select **Don't create an Aurora Replica**.

**Availability & durability**

**Multi-AZ deployment** [Info](#)

**Create an Aurora Replica or Reader node in a different AZ (recommended for scaled availability)**  
Creates an Aurora Replica for fast failover and high availability.

**Don't create an Aurora Replica**

**Learn more:** AWS recommends that you distribute the primary instance and reader instances in your DB cluster over multiple Availability Zones to improve the availability of your DB cluster. That way, an issue that affects an entire Availability Zone doesn't cause an outage for your cluster. You can set up a Multi-AZ DB cluster by making a simple choice when you create the cluster. You can also convert an existing Aurora DB cluster into a Multi-AZ DB cluster by adding a new reader DB instance and specifying a different Availability Zone. For more information see [High availability for Amazon Aurora](#).

**Note:** Since this lab is about knowing the resources required to build a multi-tier architecture, you do not need to perform a Multi-AZ deployment. You learn how to deploy a Multi-AZ architecture in the next lab.

12. In the **Connectivity** section, configure the following:

- **Virtual private cloud (VPC):** Select **LabVPC** from the dropdown menu.
- **DB subnet group:** Select **labdbsubnetgroup** from the dropdown menu.
- **Public access:** Select **No**.
- **VPC security group (firewall):** Select **Choose existing**.
- **Existing VPC security groups:**
  - To remove the **default** security group from the **Existing VPC security groups** field, select the **X**.
  - In the **Existing VPC security groups** dropdown menu, enter **LabDBSecurityGroup** to choose this option.

**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.

**Network type**

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)**

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

**LabVPC (vpc-0385cb2f75d62617b)**  
4 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**

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.

**labdbsubnetgroup**  
2 Subnets, 2 Availability Zones

**Public access**

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

**No**  
RDS doesn't assign a public IP address to the cluster. Only Amazon EC2 instances and other resources inside the VPC can connect to your cluster. Choose one or more VPC security groups that specify which resources can connect to the cluster.

**VPC security group (firewall)**

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

**LabDBSecurityGroup**

**Availability Zone**

No preference

**RDS Proxy**

RDS Proxy is a fully managed, highly available database proxy that improves application scalability, resiliency, and security.

**Create an RDS Proxy** Info  
RDS automatically creates an IAM role and a Secrets Manager secret for the proxy. RDS Proxy has additional costs. For more information, see [Amazon RDS Proxy pricing](#).

**Certificate authority - optional**

Using a server certificate provides an extra layer of security by validating that the connection is being made to an Amazon database. It does so by checking the server certificate that is automatically installed on all databases that you provision.

rds-ca-rsa2048-g1 (default)  
Expiry: May 25, 2061

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

**Read replica write forwarding**

Turn on local write forwarding [Info](#)  
Issues write operations from reader DB instances within the same DB cluster.

**Tags - optional**  
A tag consists of a case-sensitive key-value pair.  
No tags associated with the resource.

[Add new tag](#)  
You can add up to 50 more tags.

**Database authentication** [Info](#)  
Password authentication is always active for your database engine. You can also turn on additional authentication methods for your database below.

IAM database authentication  
Authenticates using IAM database authentication.

Kerberos authentication  
Authenticates using Kerberos authentication through an AWS Directory Service for Microsoft Active Directory.

**Learn more:** Subnets are segments of an IP address range in an Amazon VPC that you designate to group your resources based on security and operational needs. A DB subnet group is a collection of subnets (typically private) that you create in an Amazon VPC and then designate for your DB instances. With a DB subnet group, you can specify an Amazon VPC when creating DB instances using the command line interface or API. If you use the console, you can just select the Amazon VPC and subnets you want to use. For more information, see [Working with DB subnet groups](#).

**Learn more:** With Amazon VPC, you can launch AWS resources into a virtual network that you have defined. This virtual network closely resembles a traditional network that you would operate in your own data center, with the benefits of using the scalable infrastructure of AWS. For more information, see [Amazon VPC VPCs and Amazon RDS](#).

### 13. In the **Monitoring** section, de-select **Enable Enhanced monitoring**

**Monitoring** [Info](#)  
Choose monitoring tools for this database. Database Insights provides a combined view of Performance Insights and Enhanced Monitoring for your fleet of databases. Database Insights pricing is separate from RDS monthly estimates. See [Amazon CloudWatch pricing](#).

Database Insights - Advanced  
• Retains 15 months of performance history  
• Fleet-level monitoring  
• Integration with CloudWatch Application Signals

Database Insights - Standard

**▼ Additional monitoring settings**  
Enhanced Monitoring, CloudWatch Logs and DevOps Guru

**Enhanced Monitoring**  
 Enable Enhanced monitoring  
Enabling Enhanced Monitoring metrics are useful when you want to see how different processes or threads use the CPU.

**Log exports**  
Select the log types to publish to Amazon CloudWatch Logs

- Audit log
- Error log
- General log
- iam-db-auth-error log
- instance log
- Slow query log

**IAM role**  
The following service-linked role is used for publishing logs to CloudWatch Logs.

RDS service-linked role

### 14. Expand the **Additional configuration** main section at the end of the page.

### 15. In the **Database options** section, configure the following:

- **Initial database name:** Enter **inventory**

**DB cluster parameter group:** Choose the value from the dropdown menu that matches the **DBClusterParameterGroup (labstack-bf420498-e8ed-48d3-8727-cd8a9792f173-9qaiwkxz56u4hbnz8cqrbo-0-rdsdbclusterparametergroup-opdcurm5znhd)** value from the left side of this page.

**▼ Additional configuration**

Database options, encryption turned on, failover, backup turned on, backtrack turned off, maintenance, CloudWatch Logs, delete protection turned off.

**Database options**

Initial database name [Info](#)  
  
If you do not specify a database name, Amazon RDS does not create a database.

DB cluster parameter group [Info](#)

DB parameter group [Info](#)

Option group [Info](#)

Failover priority

**Backup**

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

Copy tags to snapshots

**Caution:** Ensure the correct value for **DB cluster parameter group** is selected from the dropdown menu. An incorrect value results in errors when building the database replicas.

#### 16. In the **Encryption** section, unselect **Enable encryption**.

**Encryption**

Enable encryption  
Choose to encrypt the given instance. Master key IDs and aliases appear in the list after they have been created using the AWS Key Management Service console. [Info](#)

**Backtrack**

Backtrack lets you quickly rewind the DB cluster to a specific point in time, without having to create another DB cluster. [Info](#)

Enable Backtrack  
Enabling Backtrack will charge you for storing the changes you make for backtracking.

**Learn more:** You can encrypt your Amazon RDS instances and snapshots at rest by activating the encryption option for your Amazon RDS DB instance. Data that is encrypted at rest includes the underlying storage for a DB instance, its automated backups, read replicas, and snapshots. For more information, see [Encrypting Amazon RDS resources](#).

#### 17. In the **Maintenance** section, unselect **Enable auto minor version upgrade**.

**Maintenance**

Auto minor version upgrade [Info](#)

Enable auto minor version upgrade  
Enabling auto minor version upgrade will automatically upgrade your database minor version. For limitations and more details, see [Automatically upgrading the minor engine version documentation](#).

**Maintenance window** [Info](#)  
Select the period you want pending modifications or maintenance applied to the database by Amazon RDS.

Choose a window  
 No preference

**Deletion protection**

Enable deletion protection  
Protects the database from being deleted accidentally. While this option is enabled, you can't delete the database.

**Note:** Because the nature of this lab is short lived there is no need to set up a maintenance schedule for the database.

#### 18. Scroll to the bottom of the screen, then choose **Create database**.

**Estimated monthly costs**

|             |                  |
|-------------|------------------|
| DB instance | 59.86 USD        |
| Total       | <b>59.86 USD</b> |

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, I/Os, 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**

19. On the **Suggested add-ons for aurora** pop-up window, choose **Close**.

**Suggested add-ons for aurora**

Simplify the configuration of the following suggested add-ons by using settings from your new database.

**Create an ElastiCache cluster from RDS using your DB settings - new**

You can save costs and improve read performance by using ElastiCache with RDS versus running on RDS alone.  
\*For example: you can save up to 55% in cost and gain up to 80x faster read performance using ElastiCache with RDS for MySQL (vs. RDS for MySQL alone).

[Learn more](#) [Create ElastiCache cluster](#)

**Use RDS Proxy**

Using a proxy allows your applications to pool and share database connections to help them scale. A proxy simplifies connection management and makes applications more resilient to database failures.

[Learn more](#) [Create proxy](#)

You can hide these suggestions so they don't appear after database creation. All these actions can be taken from the database list page or database details page.

Hide add-ons for 30 days [Close](#)

**Successfully created database aurora**

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

**Databases (2)**

| DB identifier     | Status    | Role           | Engine       | Region ... | Size         | Recommendations | CPU | Current activity | Mainten... |
|-------------------|-----------|----------------|--------------|------------|--------------|-----------------|-----|------------------|------------|
| aurora            | Available | Regional cl... | Aurora My... | us-west-2  | 1 instance   | -               | -   | -                | none       |
| aurora-instance-1 | Creating  | Reader ins...  | Aurora My... | -          | db.t3.medium | -               | -   | -                | none       |

**aurora**

**Related**

| DB identifier     | Status    | Role           | Engine       | Region ... | Size          | Recommendations | CPU      | Current ... | Mainten... | VPC |
|-------------------|-----------|----------------|--------------|------------|---------------|-----------------|----------|-------------|------------|-----|
| aurora            | Available | Regional cl... | Aurora My... | us-west-2  | 1 instance    | -               | -        | -           | none       | -   |
| aurora-instance-1 | Available | Writer inst... | Aurora My... | us-west-2b | db.t3.medi... | 8.85%           | 2 Select | none        | vpc-0385   | -   |

**Connectivity & security** **Monitoring** **Logs & events** **Configuration** **Zero-ETL integrations** **Maintenance & backups** **Data migrations - new** **Tags** **Recommendations**

**Endpoints (2)**

| Endpoint name  | Status    | Type   | Port |
|--|-----------|--------|------|
| aurora.cluster-csijgp9emsgh.us-west-2.rds.amazonaws.com    | Available | Writer | 3306 |
| aurora.cluster-ro-csijgp9emsgh.us-west-2.rds.amazonaws.com | Available | Reader | 3306 |

**Manage IAM roles**

- Select IAM roles to add to this cluster  
Add an existing IAM role to this cluster.
- Select a service to connect to this cluster  
Connect a service to this cluster by creating a new IAM role with permissions to access the service.

[Choose an IAM role to add](#) [Add role](#)

A **(Successfully created database aurora)** message is displayed on top of the screen.

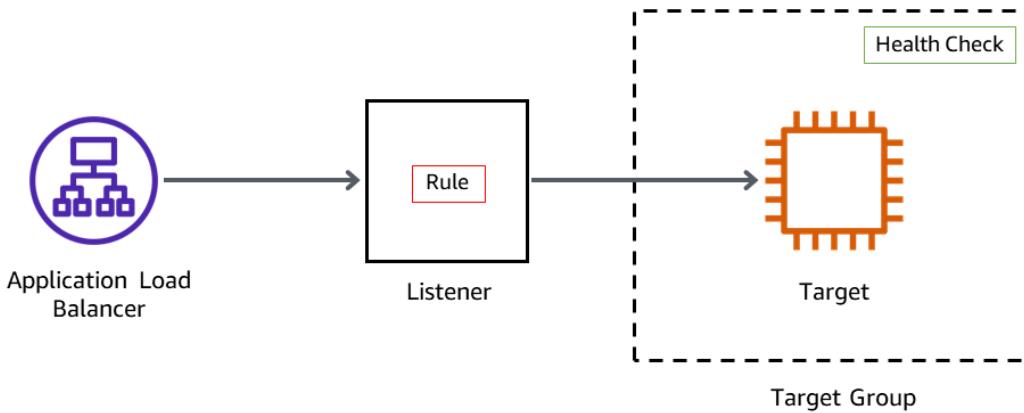
Your Aurora MySQL DB cluster is in the process of launching. The Amazon RDS database can take up to 5 minutes to launch. However, you can continue to the next task.

**Congratulations!** You have successfully created an Amazon RDS database.

## Task 2: Create and configure an Application Load Balancer

In this task, you create an Application Load Balancer in the public subnets to access the application from a browser. You navigate to the Amazon EC2 console and create an Application Load Balancer into the existing Amazon VPC infrastructure and add the private EC2 instances as a target.

A load balancer serves as the single point of contact for clients. Clients send requests to the load balancer, and the load balancer sends them to targets, such as EC2 instances. To configure your load balancer, you create target groups and then register targets with your target groups.



### Task 2.1: Create a target group

In this task, you create a target group and register your targets with the target group. By default, the load balancer sends requests to registered targets using the port and protocol that you specified for the target group.

20. At the top of the console, in the search bar, search for and choose [EC2](#).

The screenshot shows the AWS Cloud Explorer interface. On the left, the navigation pane includes sections for Aurora and Databases, Subnet groups, Option groups, Custom engines, Events, and Recommendations. The main area displays the EC2 service details, featuring a 'Services' section with EC2, EC2 Image Builder, and EC2 Global View; a 'Features' section with Dashboard, EC2 Instances, and AMIs; and a 'Resources' section for a focused search. A modal window titled 'Introducing resource search' provides instructions on how to use the search feature. The bottom right corner contains links for Actions, Create custom endpoint, and a status table.

21. In the left navigation pane, expand the **Load Balancing** section and choose **Target Groups**.

The screenshot shows the Target groups page under the Load Balancing section. The left navigation pane highlights the 'Target Groups' option. The main content area shows a table for 'Target groups info' with columns for Name, ARN, Port, Protocol, Target type, Load balancer, and VPC ID. A message indicates 'No target groups' found in the us-west-2 region. A 'Create target group' button is visible. The bottom of the screen shows the URL https://us-west-2.console.aws.amazon.com/ec2/home?region=us-west-2#TargetGroups.

22. Choose **Create target group**.

The screenshot shows the 'Specify group details' step of the 'Create target group' wizard. The left sidebar shows 'Step 1 Specify group details' selected. The main content area is titled 'Specify group details' and describes the purpose of a target group. It includes a 'Basic configuration' section with a note that settings can't be changed after creation. A 'Choose a target type' section has 'Instances' selected, with a note about supporting EC2 Auto Scaling. Other options like 'IP addresses', 'Lambda function', and 'Application Load Balancer' are also listed. A 'Target group name' input field is at the bottom. The bottom of the screen shows the URL https://us-west-2.console.aws.amazon.com/ec2/home?region=us-west-2#CreateTargetGroup.

The **Specify group details** page is displayed.

23. In the **Basic configuration** section, configure the following:

- **Choose a target type:** Select **Instances**.
- **Target group name:** Enter **ALBTargetGroup**.
- **VPC:** Select **LabVPC** from the dropdown menu.

**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.

**IP addresses**  
Supports load balancing to VPC and on-premises resources.  
Facilitates routing to multiple IP addresses and network interfaces on the same instance.  
Offers flexibility with microservice based architectures, simplifying inter-application communication.  
Supports IPv6 targets, enabling end-to-end IPv6 communication, and IPv4-to-IPv6 NAT.

**Lambda function**  
Facilitates routing to a single Lambda function.  
Accessible to Application Load Balancers only.

**Application Load Balancer**  
Offers the flexibility for a Network Load Balancer to accept and route TCP requests within a specific VPC.  
Facilitates using static IP addresses and PrivateLink with an Application Load Balancer.

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

**Protocol : Port**  
Choose a protocol for your target group that corresponds to the Load Balancer type that will route traffic to it. Some protocols now include anomaly detection for the targets and you can set mitigation options once your target group is created. This choice cannot be changed after creation.

HTTP  80

**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.

LabVPC  
vpc-0385c82b75d62617b  
IPv4 VPC CIDR: 10.0.0.0/20

The remaining settings on the page can be left at their default values.

**Protocol version**

**HTTP1**  
Send requests to targets using HTTP/1.1. Supported when the request protocol is HTTP/1.1 or HTTP/2.

**HTTP2**  
Send requests to targets using HTTP/2. Supported when the request protocol is HTTP/2 or gRPC, but gRPC-specific features are not available.

**gRPC**  
Send requests to targets using gRPC. Supported when the request protocol is gRPC.

**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 perform health checks on the root, or specify a custom path if preferred.  
/   
Up to 1024 characters allowed.

**Advanced health check settings**

▼ Advanced health check settings

[Restore defaults](#)

**Health check port**

The port the load balancer uses when performing health checks on targets. By default, the health check port is the same as the target group's traffic port. However, you can specify a different port as an override.

- Traffic port  
 Override

**Healthy threshold**

The number of consecutive health checks successes required before considering an unhealthy target healthy.

5

2-10

**Unhealthy threshold**

The number of consecutive health check failures required before considering a target unhealthy.

2

2-10

**Timeout**

The amount of time, in seconds, during which no response means a failed health check.

5

seconds

2-120

**Interval**

The approximate amount of time between health checks of an individual target

30

seconds

5-300

**Success codes**

The HTTP codes to use when checking for a successful response from a target. You can specify multiple values (for example, "200,202") or a range of values (for example, "200-299").

200

**Attributes**

Certain default attributes will be applied to your target group. You can view and edit them after creating the target group.

▼ Tags - optional

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

No tags associated with this resource.

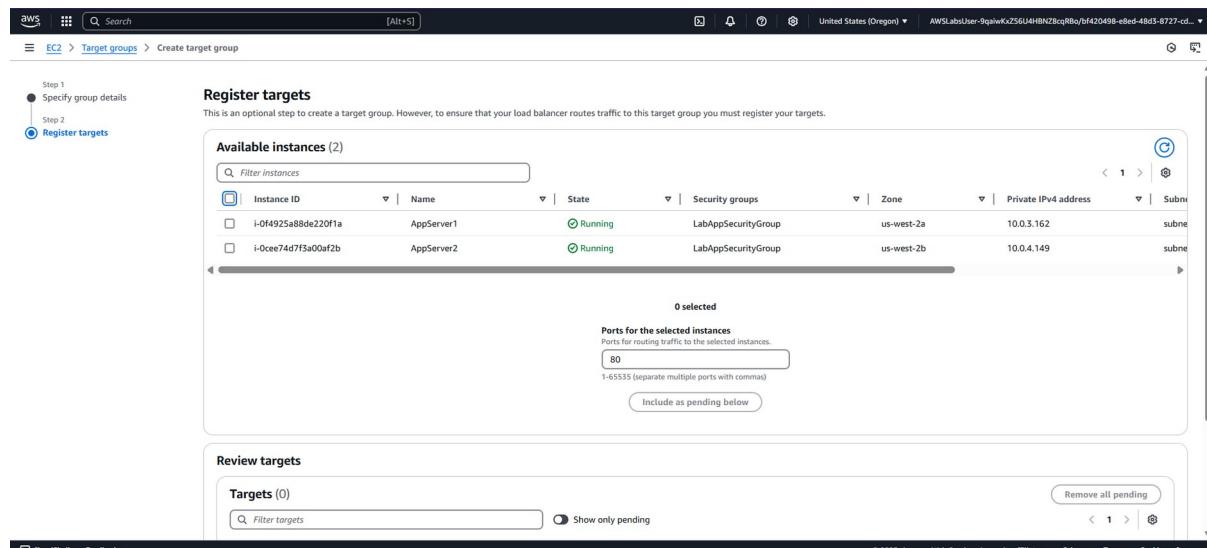
[Add new tag](#)

You can add up to 50 tags.

[Cancel](#)

[Next](#)

24. Choose **Next**.



Step 1  
Specify group details  
Step 2  
Register targets

**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)**

| Instance ID         | Name       | State   | Security groups     | Zone       | Private IPv4 address | Subnet |
|---------------------|------------|---------|---------------------|------------|----------------------|--------|
| i-0f4925a88de220f1a | AppServer1 | Running | LabAppSecurityGroup | us-west-2a | 10.0.3.162           | subne  |
| i-0ceef7d7f3a00af2b | AppServer2 | Running | LabAppSecurityGroup | us-west-2b | 10.0.4.149           | subne  |

**Ports for the selected instances**

Ports for routing traffic to the selected instances.

80

1-65535 (separate multiple ports with commas)

[Include as pending below](#)

**Review targets**

Targets (0)

[Remove all pending](#)

[CloudShell](#) [Feedback](#)

The **Register targets** page is displayed.

25. In the **Available instances** section, configure the following:

- Select the EC2 instance named **AppServer1** and **AppServer2**.
- Choose **Include as pending below**.

**Step 1**  
Specify group details  
**Step 2**  
Register targets

**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)**

| Instance ID         | Name       | State   | Security groups     | Zone       | Private IPv4 address | Subnet ID                |
|---------------------|------------|---------|---------------------|------------|----------------------|--------------------------|
| i-0f4925a88de220f1a | AppServer1 | Running | LabAppSecurityGroup | us-west-2a | 10.0.3.162           | subnet-0b77eb2ae9ce4b814 |
| i-0cee74d7f3a00af2b | AppServer2 | Running | LabAppSecurityGroup | us-west-2b | 10.0.4.149           | subnet-0422137a24ca0cf9e |

2 selected

Ports for the selected instances  
Ports for routing traffic to the selected instances.  
80  
1-65535 (separate multiple ports with commas)

**Include as pending below**

**Review targets**

**Targets (0)**

**Available instances (2)**

| Instance ID         | Name       | State   | Security groups     | Zone       | Private IPv4 address | Subnet ID                |
|---------------------|------------|---------|---------------------|------------|----------------------|--------------------------|
| i-0f4925a88de220f1a | AppServer1 | Running | LabAppSecurityGroup | us-west-2a | 10.0.3.162           | subnet-0b77eb2ae9ce4b814 |
| i-0cee74d7f3a00af2b | AppServer2 | Running | LabAppSecurityGroup | us-west-2b | 10.0.4.149           | subnet-0422137a24ca0cf9e |

0 selected

Ports for the selected instances  
Ports for routing traffic to the selected instances.  
80  
1-65535 (separate multiple ports with commas)

**Include as pending below**

2 selections are now pending below. Include more or register targets when ready.

**Review targets**

**Targets (2)**

| Instance ID         | Name       | Port | State   | Security groups     | Zone       | Private IPv4 address | Subnet ID                | Launch time                       |
|---------------------|------------|------|---------|---------------------|------------|----------------------|--------------------------|-----------------------------------|
| i-0f4925a88de220f1a | AppServer1 | 80   | Running | LabAppSecurityGroup | us-west-2a | 10.0.3.162           | subnet-0b77eb2ae9ce4b814 | April 18, 2025, 13:08 (UTC+05:30) |
| i-0cee74d7f3a00af2b | AppServer2 | 80   | Running | LabAppSecurityGroup | us-west-2b | 10.0.4.149           | subnet-0422137a24ca0cf9e | April 18, 2025, 13:08 (UTC+05:30) |

The instance appears under the **Targets** section of the page.

## 26. Choose **Create target group**.

**Review targets**

**Targets (2)**

| Instance ID         | Name       | Port | State   | Security groups     | Zone       | Private IPv4 address | Subnet ID                | Launch time                       |
|---------------------|------------|------|---------|---------------------|------------|----------------------|--------------------------|-----------------------------------|
| i-0f4925a88de220f1a | AppServer1 | 80   | Running | LabAppSecurityGroup | us-west-2a | 10.0.3.162           | subnet-0b77eb2ae9ce4b814 | April 18, 2025, 13:08 (UTC+05:30) |
| i-0cee74d7f3a00af2b | AppServer2 | 80   | Running | LabAppSecurityGroup | us-west-2b | 10.0.4.149           | subnet-0422137a24ca0cf9e | April 18, 2025, 13:08 (UTC+05:30) |

2 pending

**Create target group**

A **(Successfully created target group: ALBTarGetGroup)** message is displayed on top of the screen.

## Task 2.2: Create an Application Load Balancer

In this task, you create an Application Load Balancer. To do that, you must first provide basic configuration information for your load balancer, such as a name, scheme, and IP address type. Then, you provide information about your network and one or more listeners.

27. In the left navigation pane, expand the **Load Balancing** section and choose **Load Balancers**.

28. Choose **Create load balancer**.

The screenshot shows the AWS Cloud Map interface with the URL [aws.amazon.com/cloud-map/compare-load-balancer-type](#). The page title is "Compare and select load balancer type". Below it, a sub-header says "A complete feature-by-feature comparison along with detailed highlights is also available. [Learn more](#)".

**Load balancer types**

- Application Load Balancer** [Info](#): Shows a client connecting to an ALB (Application Load Balancer) which then routes traffic to three targets (Lambda function, database, and microservice). It supports HTTP and HTTPS. A "Create" button is present.
- Network Load Balancer** [Info](#): Shows a client connecting to a VPC through a NLB (Network Load Balancer) which then routes traffic via TCP, UDP, or TLS to three targets. It supports VPC and offloads TLS. A "Create" button is present.
- Gateway Load Balancer** [Info](#): Shows a client connecting to a GWLB (Gateway Load Balancer) which then routes traffic to a fleet of third-party virtual appliances. It supports GENEVE and offloads TLS. A "Create" button is present.

At the bottom, there are links for "CloudShell", "Feedback", and copyright information: "© 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences".

Compare and select load balancer type page is displayed.

- In the Load balancer types section, for **Application Load Balancer** card, choose **Create**.

## Compare and select load balancer type

A complete feature-by-feature comparison along with detailed highlights is also available. [Learn more](#)

The screenshot shows the AWS Cloud Map interface with the URL [aws.amazon.com/cloud-map/compare-load-balancer-type](#). The page title is "Compare and select load balancer type". Below it, a sub-header says "A complete feature-by-feature comparison along with detailed highlights is also available. [Learn more](#)".

**Load balancer types**

- Application Load Balancer** [Info](#) (highlighted in green): Shows a client connecting to an ALB (Application Load Balancer) which then routes traffic to three targets (Lambda function, database, and microservice). It supports HTTP and HTTPS. A "Create" button is present.
- Network Load Balancer** [Info](#): Shows a client connecting to a VPC through a NLB (Network Load Balancer) which then routes traffic via TCP, UDP, or TLS to three targets. It supports VPC and offloads TLS. A "Create" button is present.
- Gateway Load Balancer** [Info](#): Shows a client connecting to a GWLB (Gateway Load Balancer) which then routes traffic to a fleet of third-party virtual appliances. It supports GENEVE and offloads TLS. A "Create" button is present.

At the bottom, there is a link for "Classic Load Balancer - previous generation".

[CloudShell](#) [Feedback](#)

Search [Alt+5] United States (Oregon) AWSLabsUser-xXpBgNjihCa8Chr2lbd079j/f420498-e8ed-48d3-8727-cd..

AWS Cloud Map VPC EC2 Aurora and RDS S3

EC2 > Load balancers > Create Application Load Balancer

**Application Load Balancers now support public IPv4 IP Address Management (IPAM)**  
You can get started with this feature by configuring IP pools in the Network mapping section.

## 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.  
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**  
• Servs external traffic.  
• Has public IP addresses.  
• DNS name resolves to public IPs.  
• Requires a public subnet.

**Internal**  
• Servs internal traffic.  
• Has private IP addresses.  
• DNS name resolves to private IPs.  
• Compatible with the IPv4 and Dualstack IP address types.

**Load balancer IP address type | Info**  
Select the front-end IP address type to assign to the load balancer. The VPC and subnets mapped to this load balancer must include the selected IP address types. Public IPv4 addresses have an additional cost.

**IPv4**  
Includes only IPv4 addresses.

**Dualstack**  
Includes IPv4 and IPv6 addresses.

**Dualstack without public IPv4**  
Includes a public IPv6 address, and private IPv4 and IPv6 addresses. Compatible with Internet-facing load balancers only.

The **Create Application Load Balancer** page is displayed.

30. In the **Basic configuration** section, configure the following:

- **Load balancer name:** Enter `LabAppALB`.

**Application Load Balancers now support public IPv4 IP Address Management (IPAM)**

You can get started with this feature by configuring IP pools in the Network mapping section.

## 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.  
 LabAppALB

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**

- Serves internet-facing traffic.
- Has public IP addresses.
- DNS name resolves to public IPs.
- Requires a public subnet.

**Internal**

- Serves internal traffic.
- Has private IP addresses.
- DNS name resolves to private IPs.
- Compatible with the IPv4 and Dualstack IP address types.

**Load balancer IP address type** Info

Select the front-end IP address type to assign to the load balancer. The VPC and subnets mapped to this load balancer must include the selected IP address types. Public IPv4 addresses have an additional cost.

**IPv4**

Includes only IPv4 addresses.

**Dualstack**

Includes IPv4 and IPv6 addresses.

**Dualstack without public IPv4**

Includes a public IPv6 address, and private IPv4 and IPv6 addresses. Compatible with **internet-facing** load balancers only.

31. In the **Network mapping** section, configure the following:

- **VPC:** Select **LabVPC** from the dropdown menu.
  - **Mappings:**
    - Select the check box for the first Availability Zone listed, and select **PublicSubnet1** from the Subnet list dropdown menu.
    - Select the check box for the second Availability Zone listed, and select **PublicSubnet2** from the Subnet list dropdown menu.

**Network mapping** [Info](#)

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

**VPC** [Info](#)

You can optionally choose to configure an IPAM pool as the preferred source for your load balancers IP addresses. Create or view Pools in [Amazon VPC IP Address Manager console](#).

[Create a VPC](#)

**LabVPC**  
vpc-0385c82b75d62617b  
IPv4 VPC CIDR: 10.0.0.0/20

**IP pools - new** [Info](#)

You can optionally choose to configure an IPAM pool as the preferred source for your load balancers IP addresses. Create or view Pools in [Amazon VPC IP Address Manager console](#).

[Use IPAM pool for public IPv4 addresses](#)  
The IPAM pool you choose will be the preferred source of public IPv4 addresses. If the pool is depleted IPv4 addresses will be assigned by AWS.

**Availability Zones and subnets** [Info](#)

Select at least two Availability Zones and a subnet for each zone. A load balancer node will be placed in each selected zone and will automatically scale in response to traffic. The load balancer routes traffic to targets in the selected Availability Zones only.

**us-west-2a (usw2-a2)**  
Subnet  
Only CIDR blocks corresponding to the load balancer IP address type are used. At least 8 available IP addresses are required for your load balancer to scale efficiently.  
**subnet-0f7a4e7a5fd182354**  
IPv4 subnet CIDR: 10.0.0.0/24

**us-west-2b (usw2-a22)**  
Subnet  
Only CIDR blocks corresponding to the load balancer IP address type are used. At least 8 available IP addresses are required for your load balancer to scale efficiently.  
**subnet-09cde285de0f0se1f**  
IPv4 subnet CIDR: 10.0.1.0/24

**Public Subnet 1**

**Public Subnet 2**

### 32. In the **Security groups** section, configure the following:

- Select the **X** to remove the default security group.

**Security groups** [Info](#)

A security group is a set of firewall rules that control the traffic to your load balancer. Select an existing security group, or you can [create a new security group](#).

**Security groups**

[Select up to 5 security groups](#)

**default**  
sg-0420b90c8040730fc VPC: vpc-045ab04e484a53sec

**Security groups** [Info](#)

A security group is a set of firewall rules that control the traffic to your load balancer. Select an existing security group, or you can [create a new security group](#).

**Security groups**

[Select up to 5 security groups](#)

**⚠ Application Load Balancers require at least one security group. If none are selected, the VPC's default security group will be applied.**

- Select **LabALBSecurityGroup** from the dropdown menu.

**Security groups** [Info](#)

A security group is a set of firewall rules that control the traffic to your load balancer. Select an existing security group, or you can [create a new security group](#).

**Security groups**

[Select up to 5 security groups](#)

**LabALBSecurityGroup**  
sg-05bd75546d7cf151 VPC: vpc-045ab04e484a53sec

### 33. In the **Listeners and routing** section, configure the following:

- For **Listener HTTP:80**: From the Default action dropdown menu, select **ALBTargetGroup**.

## 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

Protocol **HTTP**

Port **80**

1-65535

Default action Info

Forward to **ALBTargetGroup**

Target type: Instance, IPv4

HTTP



Remove

[Create target group](#) Info

### Listener tags - optional

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

[Add listener tag](#)

You can add up to 50 more tags.

[Add listener](#)

### ▼ Load balancer tags - optional

Consider adding tags to your load balancer. Tags enable you to categorize your AWS resources so you can more easily manage them. The 'Key' is required, but 'Value' is optional. For example, you can have Key = production-webserver, or Key = webserver, and Value = production.

No tags associated with this load balancer.

[Add new tag](#)

You can add up to 50 tags.

### Optimize with service integrations - optional Info

Optimize your load balancing architecture by integrating AWS services with this load balancer at launch. You can also add these and other services after your load balancer is created by reviewing the load balancer's "Integrations" tab.

#### Amazon CloudFront + AWS Web Application Firewall (WAF) - new Info

Optimizes: Performance, Availability, Security

Apply application layer acceleration and security protections - *in front of the load balancer*

Automatically configures and creates a CloudFront distribution with the basic recommended AWS WAF security protections, and associates it to your load balancer. [Additional charges apply](#) Info

#### ► Benefits and considerations

#### AWS Web Application Firewall (WAF) Info

Optimizes: Security

Apply application layer security protections - *in front of targets*

Your choice of either a pre-defined security configuration with basic recommended AWS WAF security protections, or associate any of your existing WAF configurations for custom protections. [Additional charges apply](#) Info

#### ► Benefits and considerations

### Review

Review the load balancer configurations and make changes if needed. After you finish reviewing the configurations, choose [Create load balancer](#).

#### Summary

Review and confirm your configurations. [Estimate cost](#) Info

Basic configuration [Edit](#)

Name: LabAppALB

Scheme: Internet-facing

IP address type: IPv4

Network mapping [Edit](#)

VPC: [vpc-0385c82b75d62e17b](#)

Public IPv4 IPAM pool:

Availability Zones and subnets:

- us-west-2a  
[subnet-0f7a4e7a5fd182354](#)  
Public Subnet 1
- us-west-2b  
[subnet-09cd2e285de9f03e1f](#)  
Public Subnet 2

Security groups [Edit](#)

LabALBSecurityGroup

[sg-09bb0674491550e48](#)

Listeners and routing [Edit](#)

HTTP:80 | Target group: [ALBTargetGroup](#)

Service integrations [Edit](#)

Amazon CloudFront + AWS Web Application Firewall (WAF): -

AWS WAF: -

AWS Global Accelerator: -

Tags [Edit](#)

-

### Creation workflow and status

#### ► Server-side tasks and status

After completing and submitting the above steps, all server-side tasks and their statuses become available for monitoring.

[Cancel](#)

[Create load balancer](#)

34. Choose **Create load balancer**.

The screenshot shows the AWS Cloud console interface for creating a load balancer. The main message at the top says "Successfully created load balancer: LabAppALB". Below it, another message says "Application Load Balancers now support public IPv4 IP Address Management (IPAM)". The "LabAppALB" page is displayed, showing details like VPC (vpc-0385c82b75d62617b), Hosted zone (Z1H1FLSHABSF5), and DNS name (LabAppALB-1034297522.us-west-2.elb.amazonaws.com). The "Listeners and rules" tab is selected, showing one rule for port 80 forwarding to an ALB target group.

A **(Successfully created load balancer: LabAppALB)** message is displayed on top of the screen.

The load balancer is in the *Provisioning* state for few minutes and then changes to *Active*.

This screenshot is identical to the previous one, showing the successful creation of the load balancer. However, the "Status" field in the "Details" section is now highlighted with a green box and shows the value "Active", indicating the transition from the initial provisioning state.

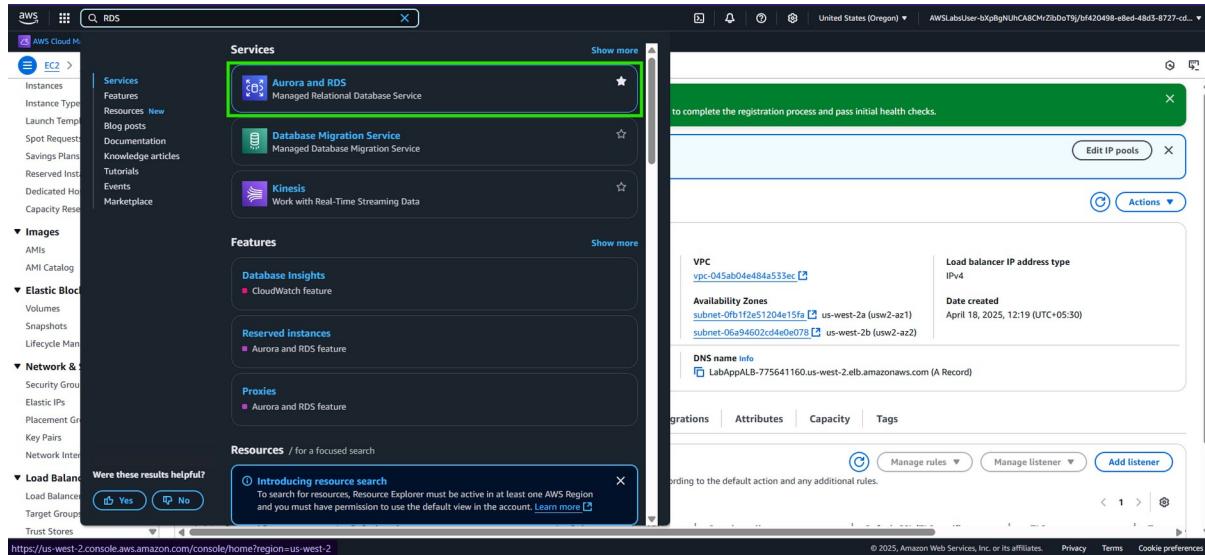
In this task, you created an Application Load Balancer and you added EC2 instances as a target to the load balancer. This task provides a demonstration on how to register a target with a load balancer. In addition to individual EC2 instances, Auto Scaling groups can also be registered as targets for the load balancer. When you use Auto Scaling groups as targets for load balancing, the instances that are launched by the Auto Scaling group are automatically registered with the load balancer. Likewise, EC2 instances that are ended by the Auto Scaling groups are automatically unregistered from the load balancer. Using Auto Scaling groups with a load balancer is demonstrated in the next lab.

**Congratulations!** You have successfully created a load balancer, created target groups, and registered the EC2 instances with the target group.

### Task 3: Review the Amazon RDS DB instance metadata through the console

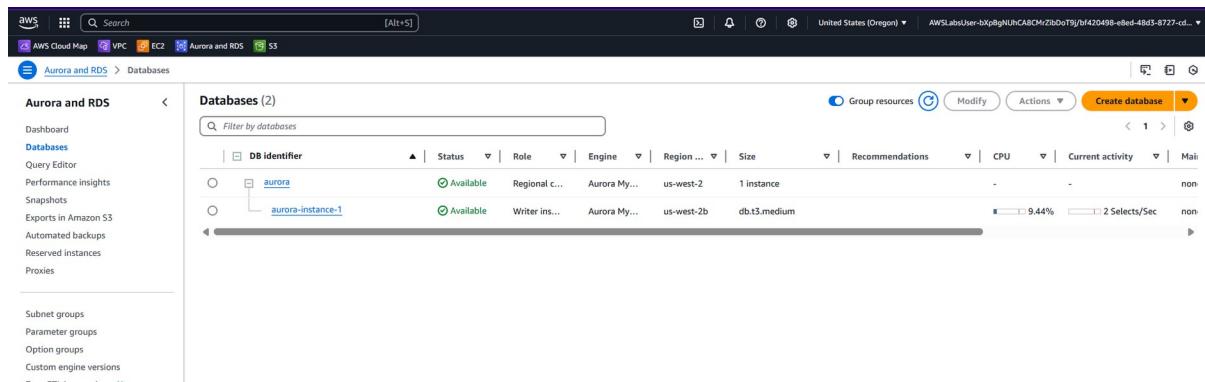
In this task, you navigate through the Amazon RDS console to ensure the instance created in Task 1 has completed and is active. You explore the console to learn how to find the connection information for a DB instance. The connection information for a DB instance includes its endpoint, port, and a valid database user.

- At the top of the console, in the search bar, search for and choose **RDS**.



The screenshot shows the AWS Cloud Map search results for 'RDS'. The 'Aurora and RDS' service is highlighted with a green box. The interface includes a sidebar with various AWS services like EC2, VPC, and S3, and a main content area displaying service details, features, and resources.

- In the navigation pane, choose **Databases**.



The screenshot shows the 'Aurora and RDS' service in the AWS Cloud Map. The 'Databases' section is selected, showing a list of two databases: 'aurora' and 'aurora-instance-1'. The 'aurora' database is selected, indicated by a blue border around its row.

- From the list of DB identifiers, select the hyperlink for the cluster named **aurora**.

The screenshot shows the AWS Aurora RDS console with the 'aurora' database cluster selected. The 'Connectivity & security' tab is active, showing the following endpoint information:

| Endpoint name  | Type   | Port |
|--|--------|------|
| aurora.cluster-csxjgp9emsgh.us-west-2.rds.amazonaws.com    | Writer | 3306 |
| aurora.cluster-ro-csxjgp9emsgh.us-west-2.rds.amazonaws.com | Reader | 3306 |

A page with details about the database is displayed.

38. On the **Connectivity & security** tab, you can find the endpoint and port number for the database cluster. In general, you need the endpoints and the port number to connect to the database.
39. Copy and paste the **Endpoint name** of the **writer instance** value to a notepad. You need this value later in the lab.

The screenshot shows the AWS Aurora RDS console with the 'aurora' database cluster selected. The 'Connectivity & security' tab is active, showing the following endpoint information, with the Writer endpoint highlighted:

| Endpoint name  | Type   | Port |
|--|--------|------|
| aurora.cluster-csxjgp9emsgh.us-west-2.rds.amazonaws.com    | Writer | 3306 |
| aurora.cluster-ro-csxjgp9emsgh.us-west-2.rds.amazonaws.com | Reader | 3306 |

It should look similar to `aurora.cluster-crxwxbgqad61a.us-west-2.rds.amazonaws.com`.

**End point:** `aurora.cluster-csxjgp9emsgh.us-west-2.rds.amazonaws.com`  
**Port:** 3306

**End point:** `aurora.cluster-ro-csxjgp9emsgh.us-west-2.rds.amazonaws.com`  
**Port:** 3306

**Tip:** To copy the **writer instance** endpoint, hover on it and choose the copy icon.

The screenshot shows the AWS Aurora console. In the top navigation bar, there is a 'Related' section with a search bar labeled 'Filter by databases'. Below it is a table for the DB cluster 'aurora'. The table includes columns for DB identifier, Status, Role, Engine, Region, Size, Recommendations, CPU, Current CPU usage, Maintenance mode, and VPC. One instance, 'aurora-instance-1', is listed under the cluster. In the bottom navigation bar, the 'Connectivity & security' tab is selected, along with Monitoring, Logs & events, Configuration, Zero-ETL integrations, Maintenance & backups, Data migrations - new, Tags, and Recommendations.

**Endpoints (2)**

The 'Endpoints' section shows two entries: 'aurora.cluster-rsxjgp9emsgh.us-west-2.rds.amazonaws.com' (Writer) and 'aurora.cluster-ro-rsxjgp9emsgh.us-west-2.rds.amazonaws.com' (Reader). Both are marked as 'Available'.

Notice that the status for the **endpoints** is **Available**.

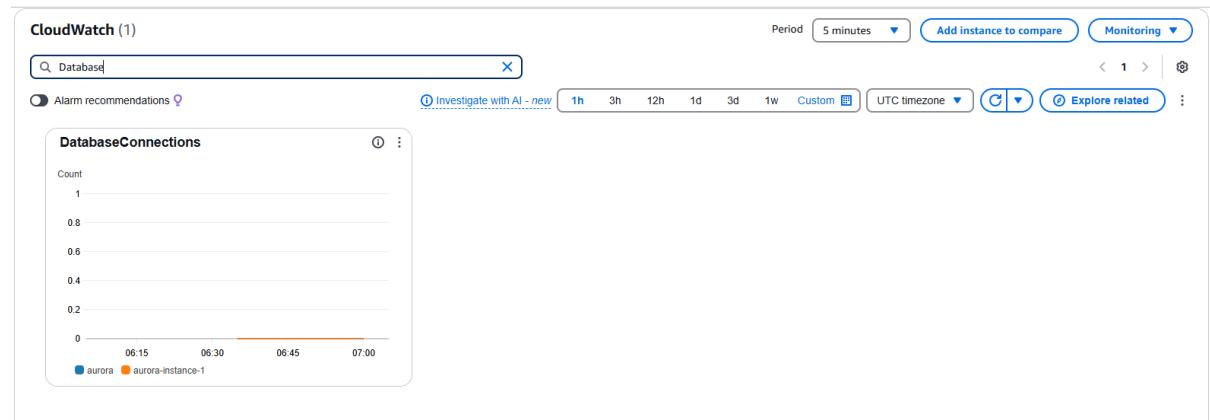
40. On the **Configuration** tab, you can find details regarding how the database is currently configured.

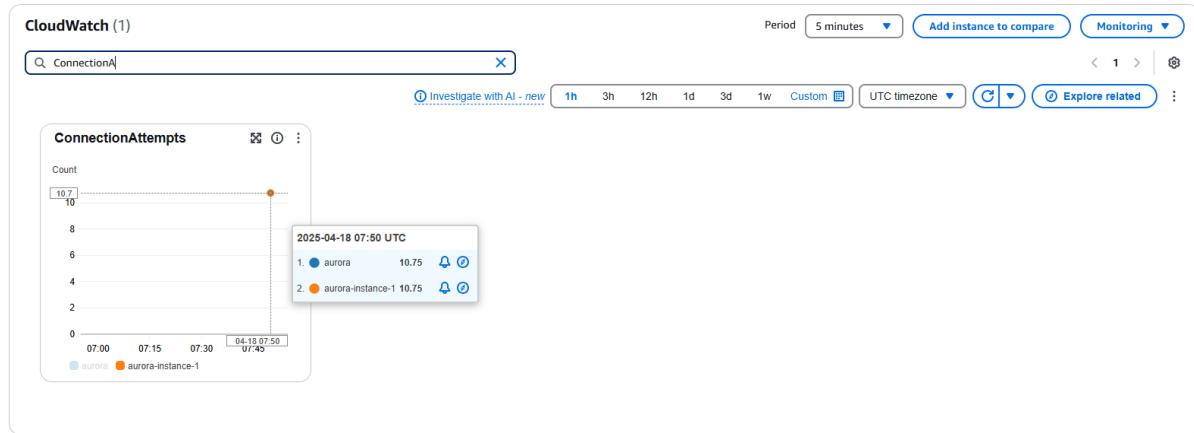
The screenshot shows the 'Configuration' tab for the Aurora database. It is divided into several sections:

- Database**: Contains general configuration like DB cluster role (Regional cluster), Engine version (8.0.mysql\_Aurora.3.05.2), and RDS Extended Support (Disabled).
- Configuration**: Details capacity type (Provisioned), local read replica write forwarding (Disabled), DB cluster ID (aurora), DB cluster parameter group (labstack-bf420498-e8ed-48d3-8727-cd89792f173-9qaiwkxz56u4hbnz8cqrbo-0-rlsdbclusterparametergroup-opdcum5Snhd), deletion protection (Disabled), and limitless database (Disabled).
- Authentication**: Shows IAM DB authentication (Not enabled) and Kerberos authentication (Not enabled).
- Encryption**: Shows encryption (Not enabled).
- Monitoring**: Shows monitoring type (Database Insights - Standard), performance insights (Disabled), enhanced monitoring (Disabled), and devops guru (-).
- Availability**: Shows multi-AZ (No).

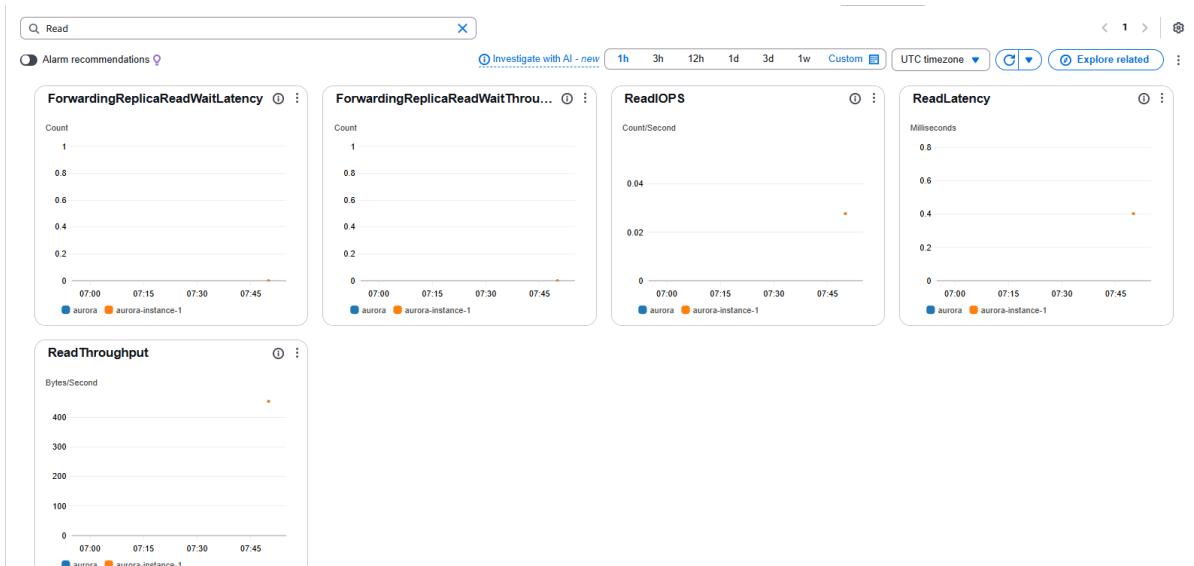
41. On the **Monitoring** tab, you can monitor metrics for the following items of your database:

- The number of connections to a database instance

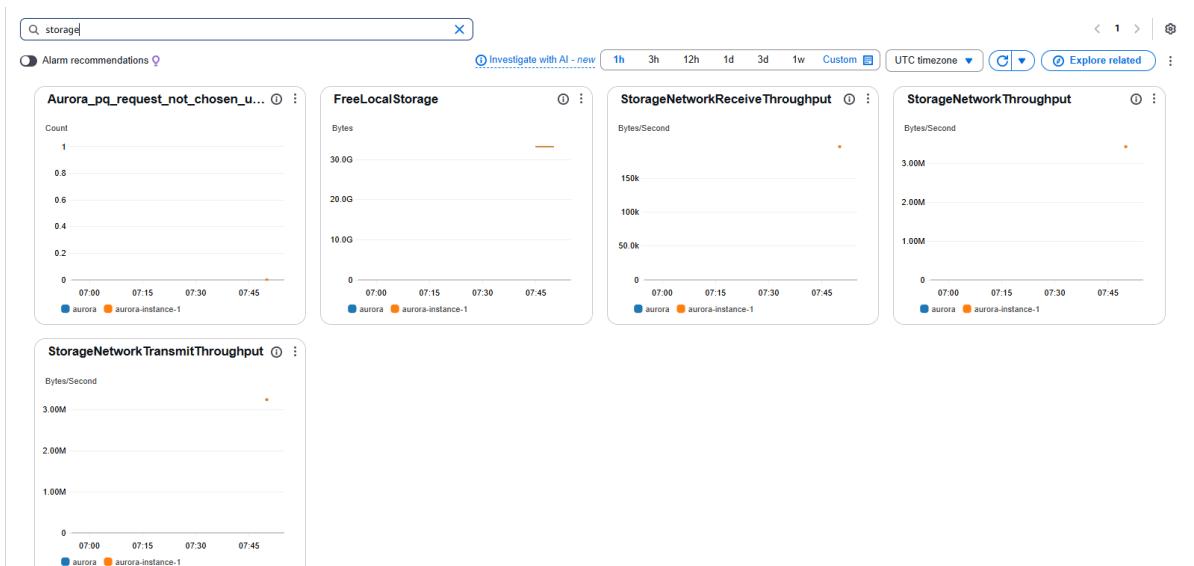




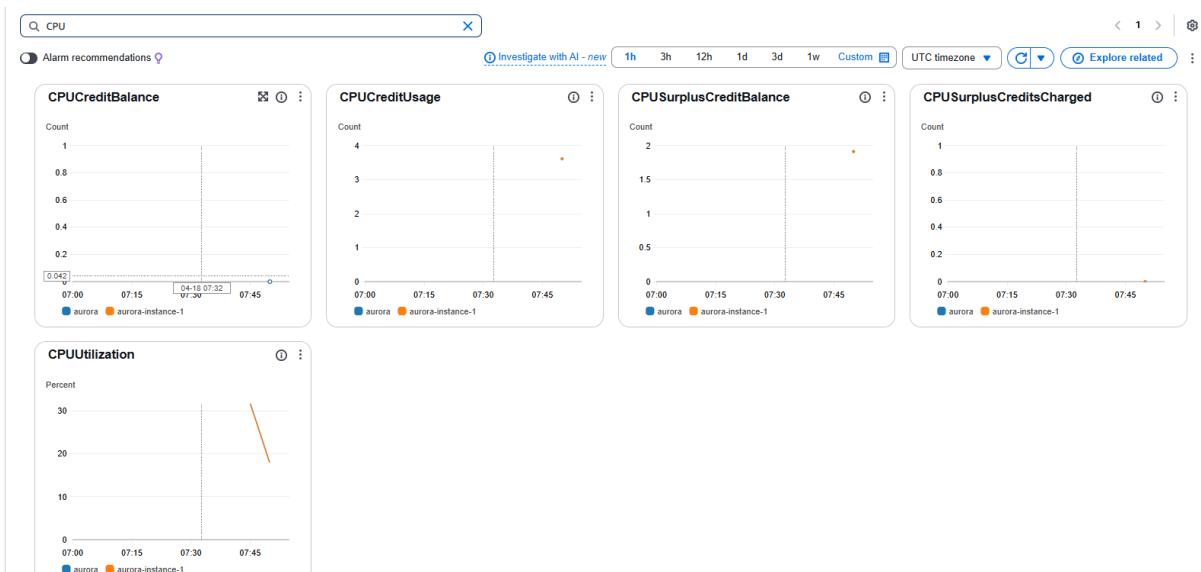
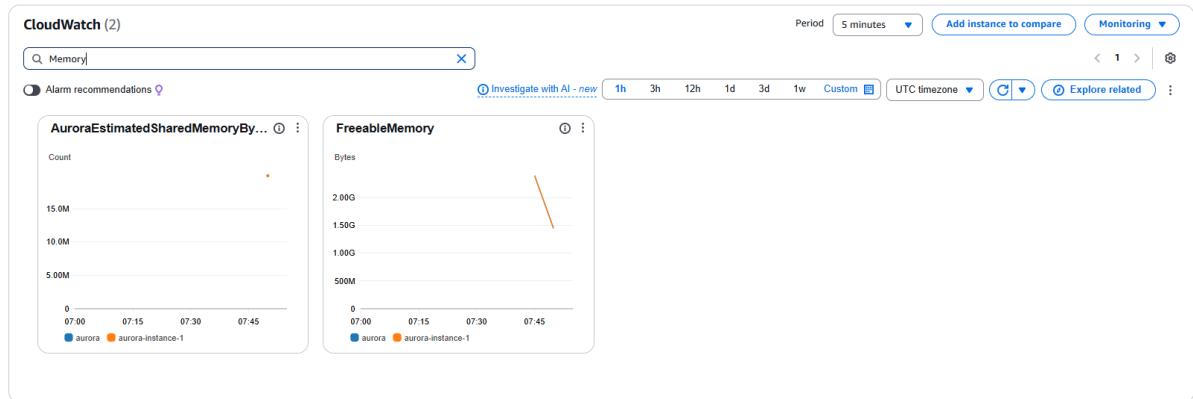
- The amount of read and write operations to a database instance



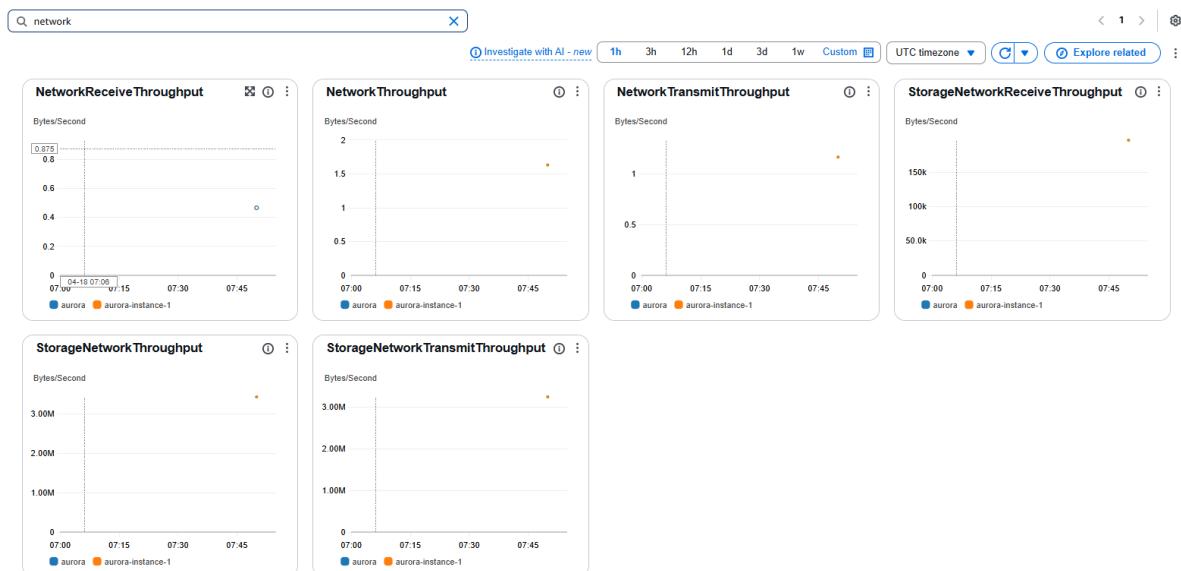
- The amount of storage that a database instance is currently using



- The amount of memory and CPU being used for a database instance



- The amount of network traffic to and from a database instance



**WARNING:** Wait for the **Status** of the **aurora DB instance** to show as **Available** before continuing to the next task.

**Related**

| DB identifier     | Status    | Role           | Engine       | Region ... | Size          | Recommendations | CPU      | Current ... | Mainten... | VPC |
|-------------------|-----------|----------------|--------------|------------|---------------|-----------------|----------|-------------|------------|-----|
| aurora            | Available | Regional cl... | Aurora My... | us-west-2  | 1 instance    | -               | -        | -           | none       | -   |
| aurora-instance-1 | Available | Writer inst... | Aurora My... | us-west-2b | db.t3.medi... | 9.68%           | 2 Select | none        | vpc-0385   |     |

**Connectivity & security** | Monitoring | Logs & events | Configuration | Zero-ETL integrations | Maintenance & backups | Data migrations - new | Tags | Recommendations

**Endpoints (2)**

| Endpoint name  | Status    | Type   | Port |
|--|-----------|--------|------|
| aurora.cluster-csxjgp9emsgh.us-west-2.rds.amazonaws.com    | Available | Writer | 3306 |
| aurora.cluster-ro-csxjgp9emsgh.us-west-2.rds.amazonaws.com | Available | Reader | 3306 |

**Manage IAM roles**

- Select IAM roles to add to this cluster
  - Add an existing IAM role to this cluster.
- Select a service to connect to this cluster
  - Connect a service to this cluster by creating a new IAM role with permissions to access the service.

[Choose an IAM role to add](#) | [Add role](#)

**Congratulations!** You have successfully reviewed the Amazon RDS DB instance metadata through the console.

#### Task 4: Test the application connectivity to the database

In this task, you identify the Application Load Balancer URL and run a basic HTTP request through the load balancer. You launch the web application installed on the EC2 instances and test the application connectivity to the database.

42. At the top of the console, in the search bar, search for and choose **EC2**.

43. In the left navigation pane, choose **Target Groups**.

The screenshot shows the AWS EC2 Target groups page. On the left sidebar, under the Load Balancing section, 'Target Groups' is highlighted with a green box. In the main content area, the 'Target groups (1) Info' table has one row selected, also highlighted with a green box. The row contains the following information:

| Name           | ARN                            | Port | Protocol | Target type | Load balancer | VPC ID                |
|----------------|--------------------------------|------|----------|-------------|---------------|-----------------------|
| ALBTarGetGroup | arn:aws:elasticloadbalancin... | 80   | HTTP     | Instance    | LabAppALB     | vpc-045ab04e484a53sec |

#### 44. Select ALBTarGetGroup.

The screenshot shows the 'Target group: ALBTarGetGroup' details page. The 'Details' tab is selected. The 'Targets' section displays the following data:

| Total targets | Healthy | Unhealthy | Unused | Initial | Draining |
|---------------|---------|-----------|--------|---------|----------|
| 2             | 0       | 0         | 2      | 0       | 0        |

Below this, the 'Distribution of targets by Availability Zone (AZ)' section shows two targets across two availability zones, both marked as healthy.

#### 45. In the Targets tab, wait until the instance status is displayed as healthy.

The screenshot shows the 'Target group: ALBTarGetGroup' details page with the 'Targets' tab selected. The 'Registered targets (2) Info' table lists two targets:

| Instance ID         | Name       | Port | Zone               | Health status | Health status details | Administrative... | Override details      | Launch time           | Anomaly detection... |
|---------------------|------------|------|--------------------|---------------|-----------------------|-------------------|-----------------------|-----------------------|----------------------|
| i-0f4925a88de220f1a | AppServer1 | 80   | us-west-2a (us...) | Healthy       | -                     | No override       | No override is cur... | April 18, 2025, 13... | Normal               |
| i-0cee74d7f3a00af2b | AppServer2 | 80   | us-west-2b (us...) | Healthy       | -                     | No override       | No override is cur... | April 18, 2025, 13... | Normal               |

**Learn more:** Elastic Load Balancing periodically tests the ping path on your web server instance to determine health. A 200 HTTP response code indicates a healthy status, and any other response code indicates an unhealthy status. If an instance is unhealthy and continues in that state for a successive number of checks (unhealthy threshold), the load balancer removes it from service until it recovers. For more information, see [Health checks for your target groups](#).

46. In the left navigation pane, choose **Load Balancers**.

The screenshot shows the AWS EC2 Load Balancers page. On the left, the navigation pane is open, and the 'Load Balancers' option under the 'Load Balancing' section is selected, highlighted with a green box. The main content area displays a table titled 'Load balancers (1)'. The table has columns for Name, DNS name, State, VPC ID, Availability Zones, Type, and Date created. One row is listed: 'LabAppALB' with 'LabAppALB-1034297522.us...' as the DNS name, 'Active' state, 'vpc-0385c82b75d62617b' VPC ID, '2 Availability Zones', 'application' type, and 'April 18, 2025, 13:23 (UTC+05:30)' date created. A green box highlights the entire table row.

The **Load balancers** page is displayed.

47. Copy the **DNS name** and paste the value in a new browser tab to invoke the load balancer.

The screenshot shows the AWS EC2 Load Balancer details page for 'LabAppALB'. The left navigation pane is identical to the previous screenshot. The main content area shows the 'Details' tab selected. Under the 'Details' tab, there are several sections: 'Load balancer type' (Application), 'Status' (Active), 'VPC' (vpc-0385c82b75d62617b), 'Load balancer IP address type' (IPv4), 'Scheme' (Internet-facing), 'Hosted zone' (Z1H1FL5HABSF5), 'Availability Zones' (subnet-0f7a4e7a5fd182354 us-west-2a (usw2-az1), subnet-095cd285de9f03e1f us-west-2b (usw2-az2)), 'Date created' (April 18, 2025, 13:23 (UTC+05:30)), and 'DNS name info' (LabAppALB-1034297522.us-west-2.elb.amazonaws.com (A Record)). A green box highlights the 'DNS name info' section.

**Tip:** To copy the *DNS name*, hover on it and select the copy icon.

**DNS name:** *LabAppALB-1034297522.us-west-2.elb.amazonaws.com*

**Expected output:** A web page like this is displayed.

Please configure Settings to connect to database

This page was generated by instance **i-0f4925a88de220f1a** in Availability Zone **us-west-2a**.

48. Choose the **Settings** tab and then configure the following:

- **Endpoint:** Paste the *writer instance endpoint* you copied earlier. (`aurora.cluster-csxjgp9emsg.us-west-2.rds.amazonaws.com`)
- **Database:** Enter `inventory`.
- **Username:** Enter `dbadmin`.
- **Password:** Paste the **LabPassword (85A4XUadNq4I)** value from the left side of these lab instructions.

Endpoint: aurora.cluster-csxjgp9emsg.us-west-2.rds.amazonaws.com

Database: inventory

Username: dbadmin

Password: 85A4XUadNq4I

Save

49. Choose **Save**

| Store       | Item        | Quantity |
|-------------|-------------|----------|
| Puerto Rico | Amazon Echo | 12       |
| Paris       | Amazon Dot  | 3        |
| Detroit     | Amazon Tap  | 5        |

+ Add Inventory

This page was generated by instance **i-0cee74d7f3a00af2b** in Availability Zone **us-west-2b**.

 Inventory 

## Edit Inventory

|           |             |
|-----------|-------------|
| Store:    | Puerto Rico |
| Item:     | Amazon Echo |
| Quantity: | 12          |

**Submit**

| Store   | Item        | Quantity |
|---|-------------|----------|
|  <input checked="" type="checkbox"/> Puerto Rico | Amazon Echo | 12       |
|  <input checked="" type="checkbox"/> Paris       | Amazon Dot  | 3        |
|  <input checked="" type="checkbox"/> Detroit     | Amazon Tap  | 5        |

**+ Add Inventory**

This page was generated by instance i-0f4925a88de220f1a in Availability Zone us-west-2a.

## Edit Inventory

|           |                |
|-----------|----------------|
| Store:    | Rio de janerio |
| Item:     | Amazon batch   |
| Quantity: | 2              |

**Submit**

| Store   | Item        | Quantity |
|---|-------------|----------|
|  <input checked="" type="checkbox"/> Puerto Rico | Amazon Echo | 12       |
|  <input checked="" type="checkbox"/> Paris       | Amazon Dot  | 3        |
|  <input checked="" type="checkbox"/> Detroit     | Amazon Tap  | 5        |

**+ Add Inventory**

This page was generated by instance i-0f4925a88de220f1a in Availability Zone us-west-2a.

 Inventory 

Data Updated!

| Store  | Item         | Quantity |
|--|--------------|----------|
|  <input checked="" type="checkbox"/> Rio de janerio | Amazon batch | 2        |
|  <input checked="" type="checkbox"/> Paris          | Amazon Dot   | 3        |
|  <input checked="" type="checkbox"/> Detroit        | Amazon Tap   | 5        |

**+ Add Inventory**

This page was generated by instance i-0f4925a88de220f1a in Availability Zone us-west-2a.

Inventory    Settings

## Add Inventory

|           |            |
|-----------|------------|
| Store:    | Newyork    |
| Item:     | Amazon VPC |
| Quantity: | 1          |

**Submit**

| Store          | Item         | Quantity |
|----------------|--------------|----------|
| Rio de janerio | Amazon batch | 2        |
| Paris          | Amazon Dot   | 3        |
| Detroit        | Amazon Tap   | 5        |

**+ Add Inventory**

This page was generated by instance **i-0cee74d7f3a00af2b** in Availability Zone **us-west-2b**.

Inventory    Settings

| Store          | Item         | Quantity |
|----------------|--------------|----------|
| Rio de janerio | Amazon batch | 2        |
| Paris          | Amazon Dot   | 3        |
| Detroit        | Amazon Tap   | 5        |
| Newyork        | Amazon VPC   | 1        |

**+ Add Inventory**

This page was generated by instance **i-0f4925a88de220f1a** in Availability Zone **us-west-2a**.

The application connects to the database, loads some initial data, and displays information. With this application, you can add, edit, or delete an item from a store's inventory.

The inventory information is stored in the Amazon RDS MySQL-compatible database you created earlier in the lab. This means that if the web application server fails, the data won't be lost. It also means that multiple application servers can access the same data.

**Congratulations!** You have successfully accessed the web application installed on the EC2 instance through the load balancer.

### Optional Task: Creating an Amazon RDS read replica in a different AWS Region

In this challenge task, you create a cross-Region read replica from the source DB instance. You create a read replica in a different AWS Region to improve your disaster recovery capabilities, scale read operations into an AWS Region closer to your users, and to make it easier to migrate from a data center in one AWS Region to a data center in another AWS Region.

**Note:** This challenge task is optional and is provided in case you have lab time remaining. You can complete this task or skip to the end of the lab [here](#).

50. Switch back to the browser tab open to the AWS Management Console.

The screenshot shows the AWS Management Console Home page. At the top left, there's a 'Recently visited' section with links to 'Aurora and RDS' and 'EC2'. Below it is a 'Welcome to AWS' section with a rocket icon and a link to 'Getting started with AWS'. To the right are sections for 'AWS Health' (0 open issues, 0 scheduled changes) and 'Cost and usage' (Current month costs: Access denied, Forecasted month end costs: Access denied). On the far right, there's an 'Applications' section showing 0 applications, with a 'Create application' button. The bottom of the page includes standard AWS navigation links like 'cloudShell', 'Feedback', and copyright information.

51. At the top of the console, in the search bar, search for and choose **RDS**.

The screenshot shows the AWS Management Console search results for 'RDS'. The search bar at the top left contains 'RDS'. The results are displayed in two main sections: 'Services' and 'Features'. In the 'Services' section, 'Aurora and RDS' is highlighted with a green border. Other services listed include 'Database Migration Service' and 'Kinesis'. In the 'Features' section, 'Database Insights' and 'Reserved instances' are listed. A sidebar on the left shows 'Services' and 'Features' with 'Loading' status. The bottom of the page includes 'Were these results helpful?' buttons ('Yes' and 'No') and standard AWS navigation links.

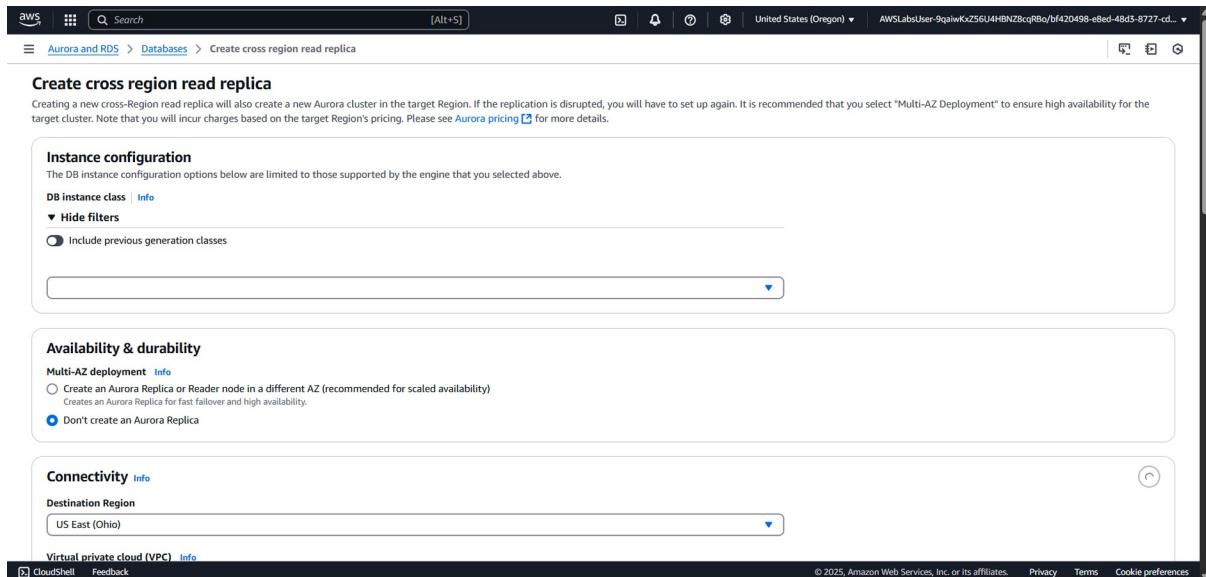
52. In the left navigation pane, choose **Databases**.

A screenshot of the AWS Aurora and RDS Databases page. The left sidebar shows navigation options like Dashboard, Databases (which is selected), Query Editor, Performance insights, Snapshots, Exports in Amazon S3, Automated backups, Reserved instances, Proxies, Subnet groups, Parameter groups, Option groups, Custom engine versions, Zero-ETL integrations, Events, Event subscriptions, Recommendations (0), and Certificate update. The main area displays a table titled 'Databases (2)'. The table has columns: DB identifier, Status, Role, Engine, Region..., Size, Recommendations, and CPU. It lists two entries: 'aurora' (Available, Regional cluster, Aurora MySQL, us-west-2, 1 instance) and 'aurora-instance-1' (Available, Writer instance, Aurora MySQL, us-west-2b, db.t3.medium, 9.02%). A search bar at the top says 'Filter by databases'. At the top right are buttons for 'Group resources', 'Modify', 'Actions', and 'Create database'. Below the table is a horizontal scroll bar.

53. Select **aurora** DB instance as the source for a read replica.

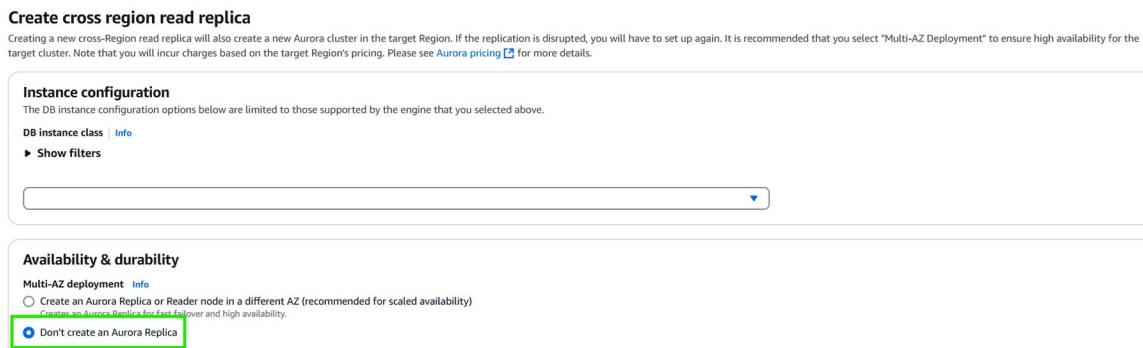
54. Choose **Actions** and select **Create cross-Region read replica**.

A screenshot of the AWS Aurora and RDS Databases page, similar to the first one but with a different focus. The 'Actions' button is highlighted with a green box. A dropdown menu is open over the 'aurora' database row. The menu items include: Stop temporarily, Delete, Set up EC2 connection, Set up Lambda connection, Migrate data from EC2 database - new, Add AWS Region, Add reader, Create cross-Region read replica (which is also highlighted with a green box), Create blue/green deployment, Create clone, Promote, Take snapshot, Restore to point in time, Backtrack, Export to Amazon S3, Add replica auto scaling, Create zero-ETL integration, Create RDS Proxy, and Create ElastiCache cluster. The rest of the interface is identical to the first screenshot.



The **Create cross region read replica** page is displayed.

For **Multi-AZ deployment:** Select **Don't create an Aurora Replica.**



The remaining settings in this section can be left at their default values.

55. In the **Connectivity** section, configure the following:

- **Destination Region:** From the dropdown menu, select the region that matches the **RemoteRegion (US East (N. Virginia))** value from the lab instructions.
- **Virtual private cloud (VPC):** *LabVPC*
- **Public access:** Select **No**.
- **For Existing VPC security groups:**
  - To remove the *default* security group, select the **X**.
  - From the dropdown menu, enter **LabDBSecurityGroup** to choose this option. The remaining settings in this section can be left at their default values.

**Connectivity** [Info](#)

**Destination Region**  
US East (N. Virginia)

**Virtual private cloud (VPC)** [Info](#)  
Choose the VPC. The VPC defines the virtual networking environment for this DB cluster.

LabVPC (vpc-0eb7d9ef14f41e75)  
0 Subnets, 0 Availability Zones

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.

remotedbsubnetgroup

**Public access** [Info](#)  
 Yes  
RDS assigns a public IP address to the cluster. Amazon EC2 instances and other resources outside of the VPC can connect to your cluster. Resources inside the VPC can also connect to the cluster. Choose one or more VPC security groups that specify which resources can connect to the cluster.  
 No  
RDS doesn't assign a public IP address to the cluster. Only Amazon EC2 instances and other resources inside the VPC can connect to your cluster. Choose one or more VPC security groups that specify which resources can connect to the cluster.

**Existing VPC security groups**  
Choose one or more options  
LabDBSecurityGroup [X](#)

**Availability Zone** [Info](#)  
No preference

**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.

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

**► Additional configuration**

## 56. In the **Settings** section, configure the following:

- DB instance identifier:** Enter **LabDBreplica**.

**Settings**

**Read replica source**  
Source DB instance Identifier  
aurora-instance-1 (DB cluster: aurora)

**DB instance identifier**  
DB instance identifier. This is the unique key that identifies a DB instance. This parameter is stored as a lowercase string (e.g. mydbinstance).  
LabDBreplica

**DB cluster identifier**  
You may optionally specify an identifier for the DB Cluster that will be created along with your instance. If you do not provide one, a default identifier based on the instance identifier will be used. The cluster identifier is used in determining the cluster's endpoint.

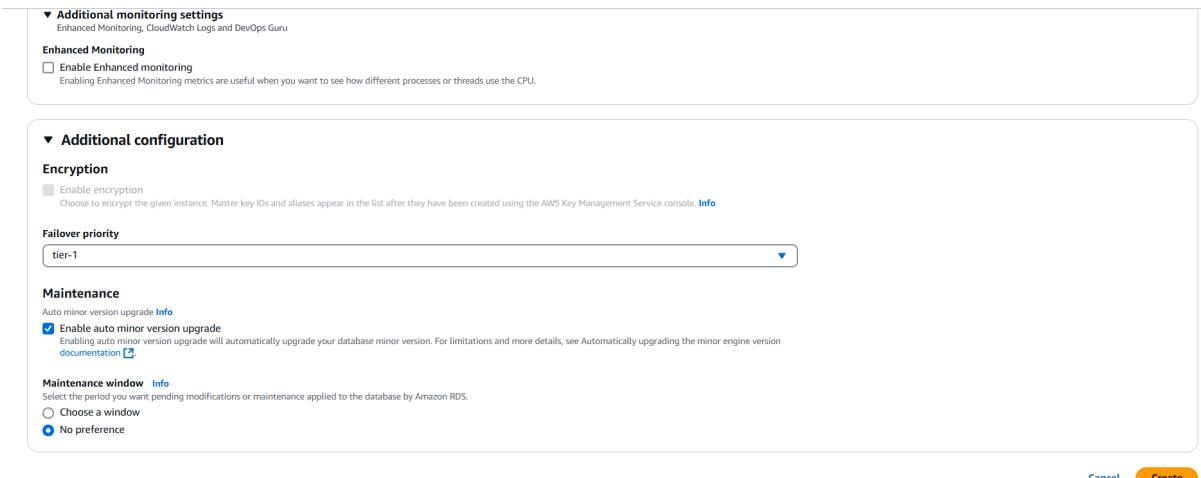
**Monitoring**  
Choose monitoring tools for this database. Database Insights provides a combined view of Performance Insights and Enhanced Monitoring for your fleet of databases. Database Insights pricing is separate from RDS monthly estimates. See [Amazon CloudWatch pricing](#).

Database Insights - Advanced  

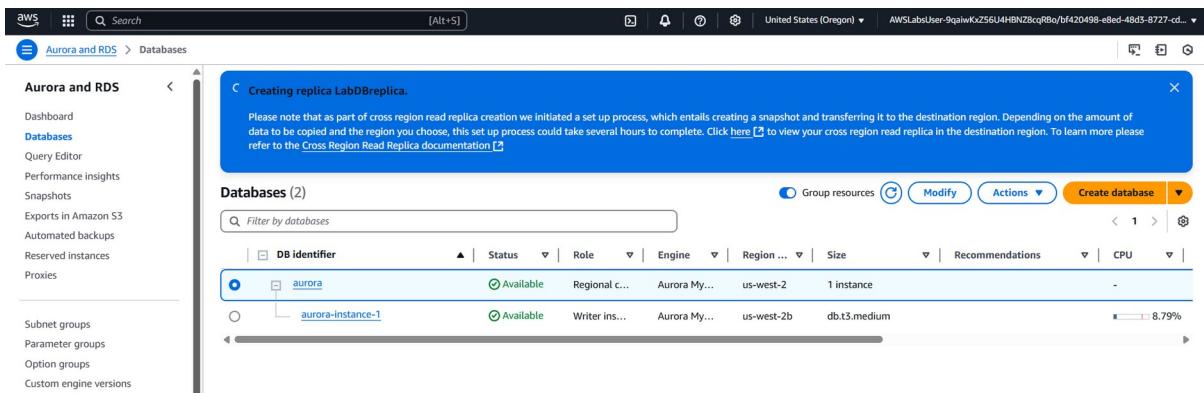
- Retains 15 months of performance history
- Fleet-level monitoring
- Integration with CloudWatch Application Signals

Database Insights - Standard

The remaining settings in this section can be left at their default values.

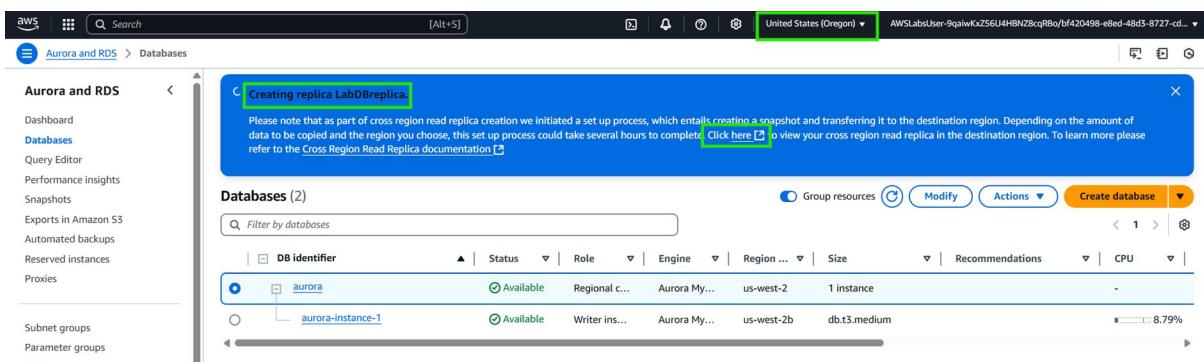


57. Choose **Create**.



A **(Creating replica LabDBreplica.)** message is displayed on the screen.

58. To review the cross-Region read replica in the destination region, choose the hyperlink as present in the below image.



59. Otherwise, choose **Close**.

Aurora and RDS > Databases

Databases (2)

| DB identifier        | Status   | Role          | Engine       | Region ... | Size         | Recommendations |
|----------------------|----------|---------------|--------------|------------|--------------|-----------------|
| labdbreplica-cluster | Creating | Replica cl... | Aurora My... | us-east-1  | 1 instance   |                 |
| labdbreplica         | Creating | Reader ins... | Aurora My... | -          | db.t3.medium |                 |

Aurora and RDS > Databases

Databases (2)

| DB identifier        | Status    | Role          | Engine       | Region ... | Size         | Recommendations |
|----------------------|-----------|---------------|--------------|------------|--------------|-----------------|
| labdbreplica-cluster | Available | Replica cl... | Aurora My... | us-east-1  | 1 instance   |                 |
| labdbreplica         | Available | Writer ins... | Aurora My... | us-east-1a | db.t3.medium |                 |

Aurora and RDS > Databases > labdbreplica-cluster

labdbreplica-cluster

Related

| DB identifier        | Status    | Role          | Engine       | Region ... | Size         | Recommendations | CPU      | Current... | Maintain... |
|----------------------|-----------|---------------|--------------|------------|--------------|-----------------|----------|------------|-------------|
| labdbreplica-cluster | Available | Replica cl... | Aurora My... | us-east-1  | 1 instance   |                 | -        | -          | none        |
| labdbreplica         | Available | Writer ins... | Aurora My... | us-east-1a | db.t3.med... | 9.24%           | 2 Select | none       |             |

Endpoints (2)

| Endpoint name   | Status    | Type   | Port |
|---|-----------|--------|------|
| labdbreplica-cluster.cluster-chc5cmztsrx.us-east-1.rds.amazonaws.com    | Available | Writer | 3306 |
| labdbreplica-cluster.cluster-ro-chc5cmztsrx.us-east-1.rds.amazonaws.com | Available | Reader | 3306 |

Manage IAM roles

- Select IAM roles to add to this cluster
  - Add an existing IAM role to this cluster
- Select a service to connect to this cluster
  - Connect a service to this cluster by creating a new IAM role with permissions to access the service.

**Congratulations!** You have successfully completed the optional task and started the creation of a cross-Region read replica for the Amazon RDS database.

## Conclusion

**Congratulations!** You have now successfully completed the following:

- Created an Amazon RDS DB instance.
- Created an Application Load Balancer.
- Created an HTTP listener for the Application Load Balancer.
- Created a target group.
- Registered targets with a target group.
- Tested the load balancer and the application connectivity to the database.

- Reviewed the Amazon RDS DB instance metadata using the console.

In this lab, you learned how to deploy various resources needed for a prototype web application in your Amazon VPC. However, the architecture that was created in this lab does not meet AWS Cloud best practices because it is not an elastic, durable, highly available design. By relying on only a single Availability Zone in the architecture, there is a single point of failure. You learn how to configure your architecture for redundancy, failover, and high availability in the next lab.

## End lab

Follow these steps to close the console and end your lab.

### 60. Return to the AWS Management Console.

The screenshot shows the AWS Management Console Home page. The top navigation bar includes the AWS logo, a search bar, and a 'Reset to default layout' button. The main content area features several cards: 'Recently visited' (Aurora and RDS, EC), 'Welcome to AWS' (Getting started with AWS), 'AWS Health' (Open issues 0, Past 7 days), and 'Cost and usage' (Current month costs, Forecasted month end costs, both with 'Access denied' status). On the far right, the user 'AWSLabsUser' is logged in, and a 'Sign out' button is highlighted with a yellow box.

### 61. At the upper-right corner of the page, choose AWSLabsUser, and then choose Sign out.

The screenshot shows the AWS Management Console Home page after signing out. The user information ('AWSLabsUser') is no longer visible in the top right corner. A 'Sign out' button is now part of a larger 'Sign out' callout menu, which also includes 'Switch role' and 'Turn on multi-session support'. The rest of the page content remains the same as in the previous screenshot.

### 62. Choose End Lab and then confirm that you want to end your lab.

**Architecting on AWS - Lab 3 - Create a database layer in your Amazon VPC infrastructure**

RemoteRegion  
US East (N. Virginia)

**Lab Content**

- Lab overview
- Start lab
- Task 1: Create an Amazon RDS database
- Task 2: Create and configure an Application Load Balancer
- Task 3: Review the Amazon RDS DB instance metadata through the console
- Task 4: Test the application connectivity to the database
- Optional Task: Creating an Amazon RDS read replica in a different AWS Region
- Conclusion
- End lab

**End Lab** **Open Console**

**Lab is ready.**  
Open the console to begin. Keep the default region. Your lab will be active until Apr 19 at 1:05 AM.  
Tip: open the console in a new window to see it side-by-side with these instructions.

Follow these steps to close the console and end your lab.

60. Return to the [AWS Management Console](#).
61. At the upper-right corner of the page, choose **AWSLabsUser**, and then choose **Sign out**.
62. Choose **End Lab** and then confirm that you want to end your lab.

For more information about AWS Training and Certification, see <https://aws.amazon.com/training/>.  
*Your feedback is welcome and appreciated.*  
If you would like to share any feedback, suggestions, or corrections, please provide the details in our [AWS Training and Certification Contact Form](#).

Nice work! Help us make Builder Labs better in just 2 minutes.  
[Launch Survey](#)

[Privacy](#) [Site terms](#) [Cookie preferences](#) [Feedback](#) [Help](#)

For more information about AWS Training and Certification, see <https://aws.amazon.com/training/>.

*Your feedback is welcome and appreciated.*

If you would like to share any feedback, suggestions, or corrections, please provide the details in our [AWS Training and Certification Contact Form](#).