



General Immersion Day

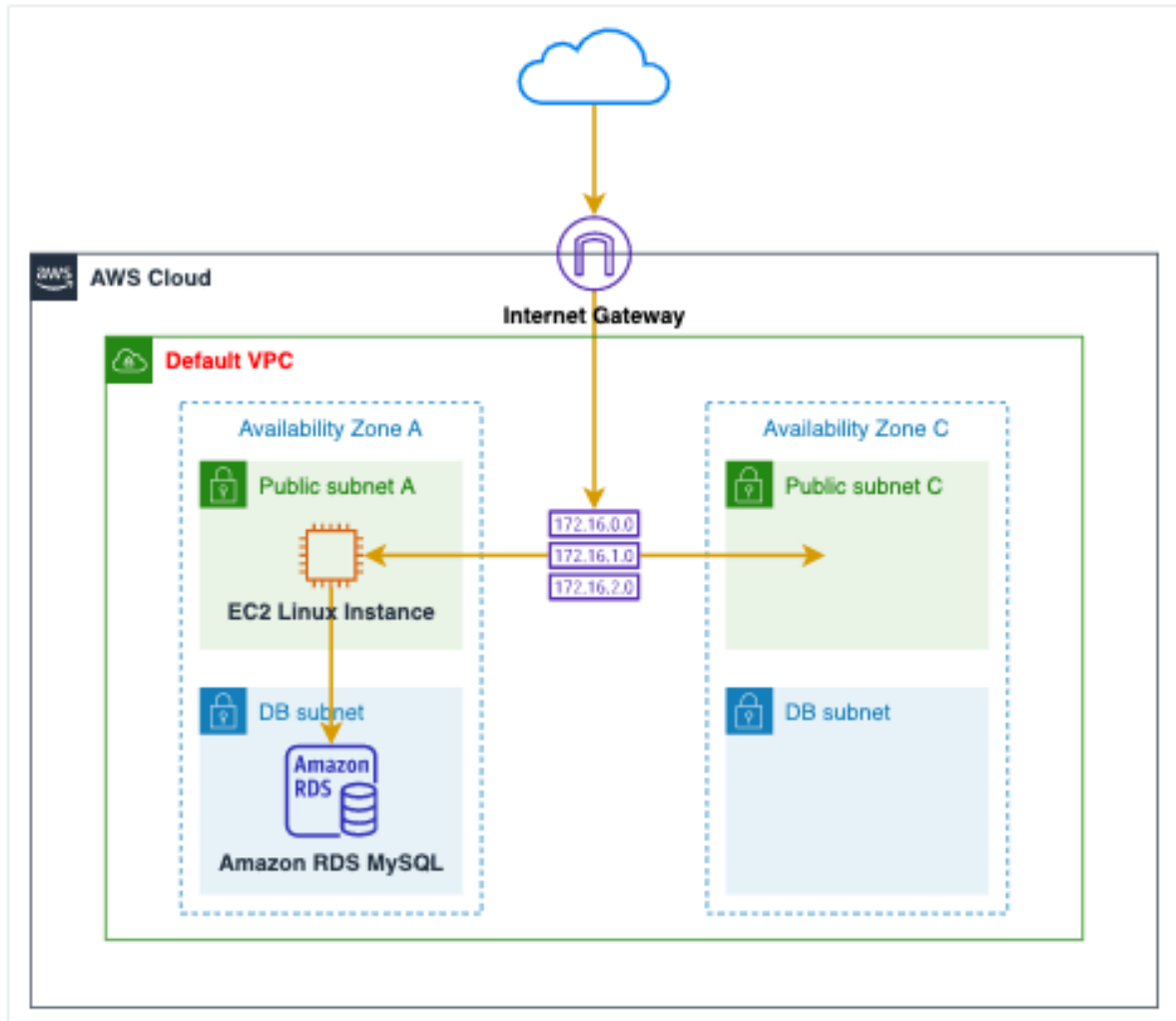
Lab 5

Amazon RDS MySQL Hands on Lab

Amazon RDS Overview

Amazon RDS is a web service that makes it easy to set up, operate, and scale a relational database in the cloud. It provides cost-efficient and resizable capacity while managing time-consuming database administration tasks, freeing you up to focus on your applications and business.

Note: This lab requires [EC2 Linux Hands-On Lab](#) in advance to complete. This lab will make use of the web server previously created in EC2 lab to connect RDS MySQL.



This lab will walk you through the following:

1. [Create VPC Security Group](#)
2. [Launch an RDS Instance](#)
3. [Save RDS Credentials](#)
4. [Access RDS from EC2](#)
5. [Create an RDS Snapshot \(Optional\)](#)
6. [Modify RDS Instance Size \(Optional\)](#)

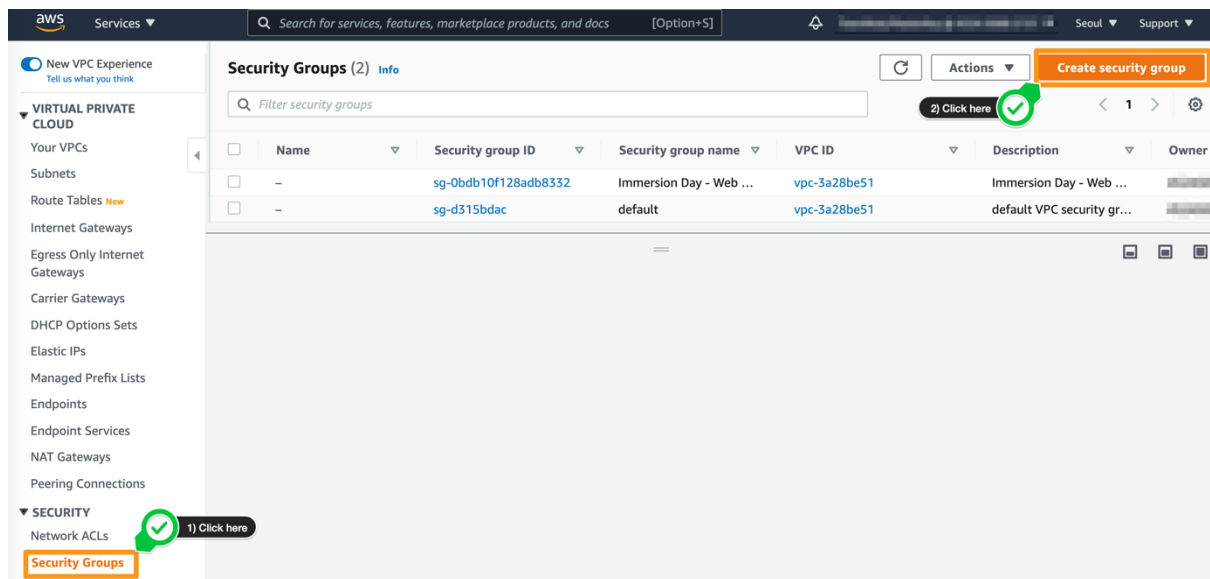
5-1 Create VPC Security Group

Prerequisite: [EC2 Linux Hands-On Lab](#)

In [EC2 Linux Hands-On Lab](#), we launched a web server EC2 instance with the security group, **Immersion Day - Web Server**, that allows TCP 80 for the web server.

First, we will create a new VPC security group, **Immersion Day - DB Tier**, for our database tier that only allows traffic from our web tier.

1. In the VPC dashboard, click **Security Groups**, then the **Create Security Group** button.



2. Type Security group name and Description as below and keep the VPC setting to the same VPC you've launched your EC2 instance in.

VPC > Security Groups > Create security group

Create security group Info

A security group acts as a virtual firewall for your instance to control inbound and outbound traffic. To create a new security group, complete the fields below.

Basic details

Security group name Info
Immersion Day DB Tier
Name cannot be edited after creation.

Description Info
Immersion Day DB Tier

VPC Info
vpc-3a28be51

KEY	VALUE
Security group name	Immersion Day DB Tier
Description	Immersion Day DB Tier
VPC	VPC-xxxxxx (default)

- Under **Inbound Rules**, click **Add rule** button.
- Add a new inbound rule for the EC2 server(s) in our web tier. The type should be **MySQL/Aurora (3306)**, the protocol **TCP (6)**, and in the source box, type the name of the security group to which your EC2 instance belongs. While you're typing, a list of security group(s) that match that name should be presented to you. Select your security group.

VPC > Security Groups > Create security group

Create security group Info

A security group acts as a virtual firewall for your instance to control inbound and outbound traffic. To create a new security group, you must specify a VPC and a security group name.

Basic details

Security group name Info
Immersion Day DB Tier
Name cannot be edited after creation.

Description Info
Immersion Day DB Tier

VPC Info
vpc-3a28be51

Inbound rules Info

Type Info **MySQL/Aurora** 1) Click here Port range Info 3306 Source Info Custom 2) Click here

CIDR blocks

- 0.0.0.0/0
- 0.0.0.0/8
- 0.0.0.0/16
- 0.0.0.0/24
- 0.0.0.0/32
- ::/0
- ::/16
- ::/32
- ::/48
- ::/64

Security Groups

- Immersion Day - Web ... | sg-0bda10f128adb8332
- default | sg-d315bdac

Prefix lists

- com.amazonaws.ap-nor... | pl-48a54021

Description - optional Info

Add rule

Delete

- Set Name tag and group name to Immersion Day DB Tier
- Then, scroll down and click on Click **Create security group** button. This will create the Security group for your RDS instance.

Tags - optional

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

Key Info **Name** Value - optional Info **Immersion Day DB Tier** Remove

Add new tag 1) Click here 2) Enter 'Name' 3) Enter the Value 4) Click to create

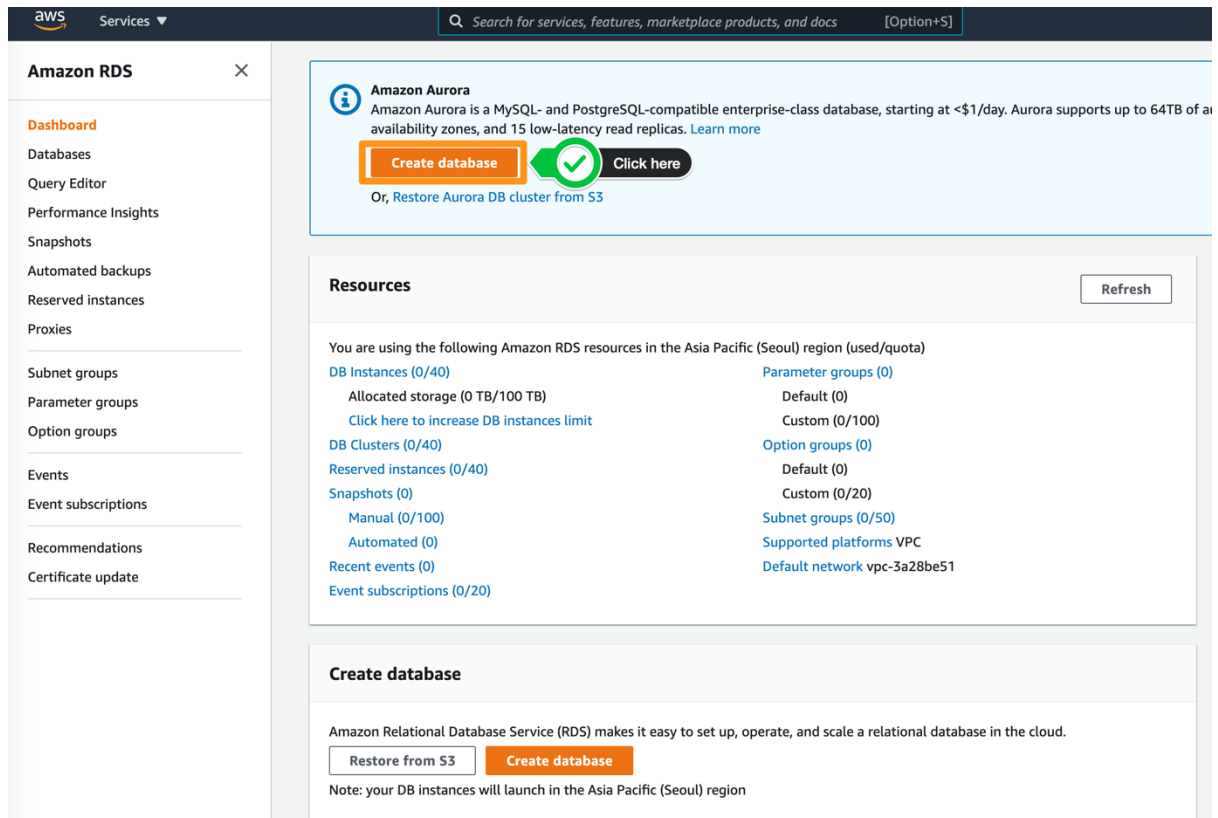
Create security group

Cancel

5-2 Launch an RDS Instance

Now that our VPC security group for Database is ready, let's configure and launch a MySQL RDS Instance.

1. Sign into the AWS Management Console and open the [Amazon RDS console](#)
2. Click on **Create database**



3. For **Choose a database creation method**, select Standard option. With Standard Create, you setup the configurations for your database.

Note: Easy Create option provides recommended best-practices configurations to get started with deploying databases.

4. Select **MySQL** in **Engine Options**.

Create database


Choose a database creation method [Info](#)


☒ **Standard create**
You set all of the configuration options, including ones for availability, security, backups, and maintenance.


☐ **Easy create**
Use recommended best-practice configurations. Some configuration options can be changed after the database is created.


Engine options


Engine type [Info](#)


☐ Amazon Aurora


☒ **MySQL**


☐ MariaDB



☐ PostgreSQL


☐ Oracle


☐ Microsoft SQL Server


Edition

☒ **MySQL Community**

 **Known issues/limitations**
Review the [Known issues/limitations](#) to learn about potential compatibility issues with specific database versions.

Version

MySQL 5.7.33

5. When you select MySQL as your database engine, the latest version will be automatically selected for you. For this lab, select **MySQL version 5.7.X**.

6. For Template, there are three options available: Production, Dev/Test and Free Tier. For the lab purpose, we will select **Free Tier**.

Templates

Choose a sample template to meet your use case.

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

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

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

7. In **Settings** section, fill in the following for each field

Parameter	Value
DB Instance Identifier	awsdb
Master Username	awsuser
Master Password	awspassword

Settings

DB instance identifier [Info](#)
Type a name for your DB instance. The name must be unique across all DB instances owned by your AWS account in the current AWS Region.

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

▼ Credentials Settings

Master username [Info](#)
Type a login ID for the master user of your DB instance.

1 to 16 alphanumeric characters. First character must be a letter

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

Master password [Info](#)

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

Confirm password [Info](#)

8. In **DB Instance size** section, for **DB instance class**, select **burstable classes-db.t2.micro**. This option will be automatically selected for you.

Instance configuration

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

DB instance class [Info](#)

- Standard classes (includes m classes)
- Memory optimized classes (includes r and x classes)
- ☒ Burstable classes (includes t classes)

db.t2.micro

1 vCPUs 1 GiB RAM Not EBS Optimized

▼

☐ Include previous generation classes

9. In the **Storage** section, select the **Storage Type** as **General Purpose SSD**. You can select or deselect the option for Auto Scaling for the lab purposes.

Storage

Storage type [Info](#)


General Purpose SSD (gp2)
Baseline performance determined by volume size


Allocated storage

20

GIB

(Minimum: 20 GiB. Maximum: 16,384 GiB) Higher allocated storage **may improve** IOPS performance.

 You might see better baseline performance with your selected volume size by specifying General Purpose SSD storage. [Learn more about using Provisioned IOPS storage for consistent performance.](#)



Storage autoscaling [Info](#)

Provides dynamic scaling support for your database's storage based on your application's needs.

☒ **Enable storage autoscaling**

Enabling this feature will allow the storage to increase once the specified threshold is exceeded.

Maximum storage threshold [Info](#)

Charges will apply when your database autoscales to the specified threshold

1000

GIB

Minimum: 21 GiB. Maximum: 16,384 GiB

10. Since we selected the Template option as Free Tier-used only for doing hands-on or testing the applications, Multi-AZ deployment is not required and hence, the Availability and Durability section will be disabled for you.

Note: For a database used in Production and Dev/Test, we recommend using a **Multi-AZ Deployment**.

Availability & durability

Multi-AZ deployment [Info](#)

☐ Do not create a standby instance

☒ Create a standby instance (recommended for production usage)

Creates a standby in a different Availability Zone (AZ) to provide data redundancy, eliminate I/O freezes, and minimize latency spikes during system backups.

11. In the **Connectivity** section:

Parameter	Value
Virtual private cloud (VPC)	Default VPC
Subnet Group	default
Publicly accessible	No
VPC Security Group(s)	Select Choose existing VPC security groups, then pick Immersion Day DB Tier
Availability Zone	No preference
Database port	3306
(Click on "Additional configuration")	

Connectivity

Virtual private cloud (VPC) [Info](#)
VPC that defines the virtual networking environment for this DB instance.

Default VPC (vpc-3a28be51) ▼

Only VPCs with a corresponding DB subnet group are listed.

After a database is created, you can't change the VPC selection.

Subnet group [Info](#)
DB subnet group that defines which subnets and IP ranges the DB instance can use in the VPC you selected.

default ▼

Public access [Info](#)

☐ Yes
Amazon EC2 instances and devices outside the VPC can connect to your database. Choose one or more VPC security groups that specify which EC2 instances and devices inside the VPC can connect to the database.

☒ No
RDS will not assign a public IP address to the database. Only Amazon EC2 instances and devices inside the VPC can connect to your database.

VPC security group
Choose a VPC security group to allow access to your database. Ensure 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 VPC security groups ▼

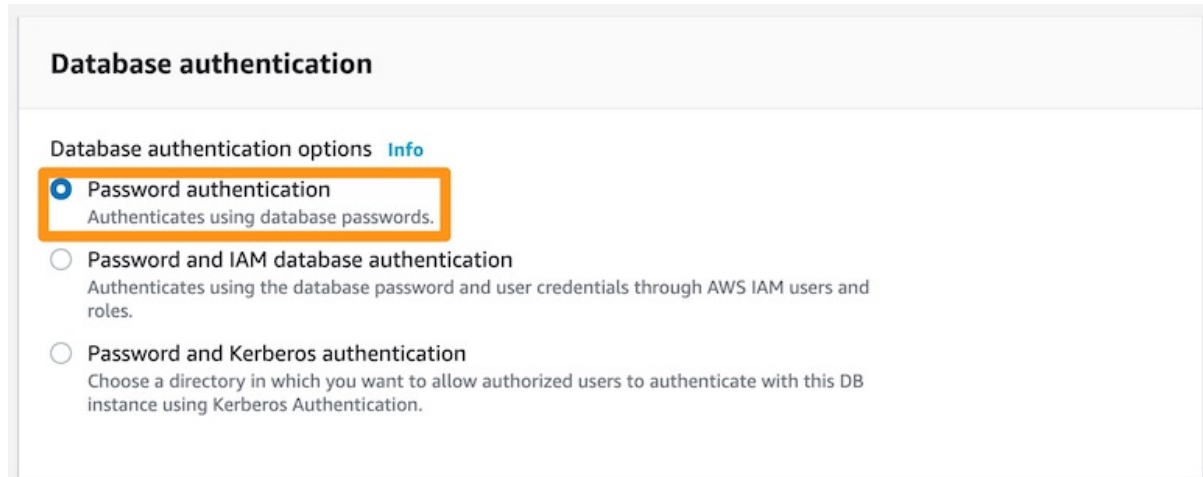
Immersion Day DB Tier ✕

Availability Zone [Info](#)

No preference ▼

12. For **Database authentication**, there are two options to select from. **Password Authentication** will authenticate the user only with the database password. With **Password and IAM Database authentication**, the user will be authenticated with the database password and also with the user credentials through IAM roles and policies.

For this lab, we will select: Password Authentication.



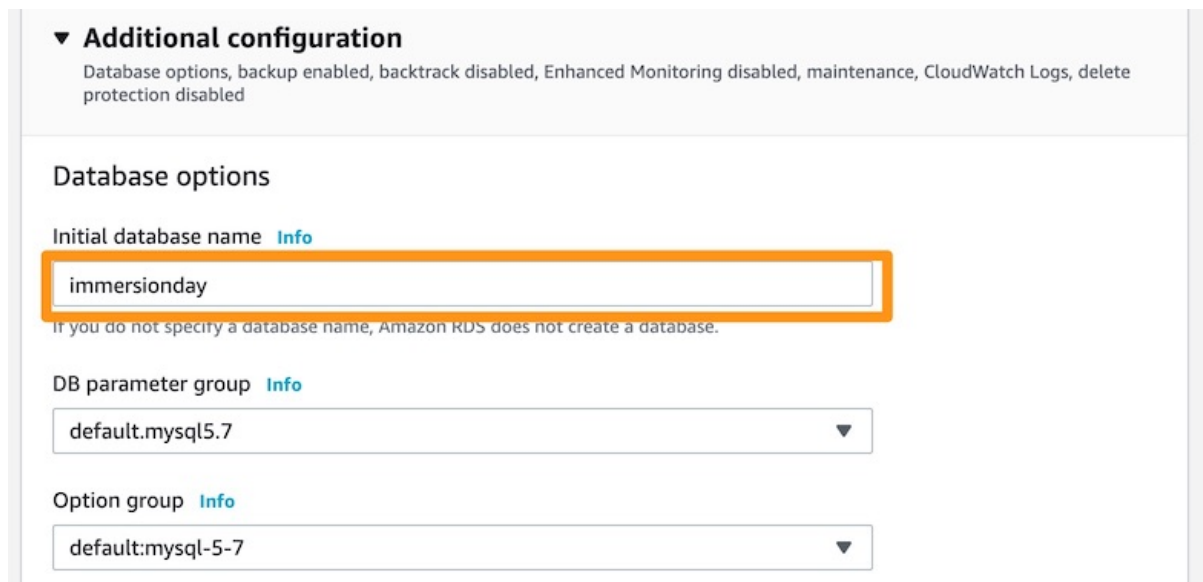
Database authentication

Database authentication options [Info](#)

- ☒ **Password authentication**
Authenticates using database passwords.
- ☐ **Password and IAM database authentication**
Authenticates using the database password and user credentials through AWS IAM users and roles.
- ☐ **Password and Kerberos authentication**
Choose a directory in which you want to allow authorized users to authenticate with this DB instance using Kerberos Authentication.

13. Expand on the **Additional Configuration** section.

- For the Database options, provide the following:
 - Initial Database name: **immersionday**
 - Note: For the purpose of this lab, the database name have to be exactly "immersionday"**
 - DB Parameter group and Option group: default.mysql5.7



▼ **Additional configuration**

Database options, backup enabled, backtrack disabled, Enhanced Monitoring disabled, maintenance, CloudWatch Logs, delete protection disabled

Database options

Initial database name [Info](#)

immersionday

If you do not specify a database name, Amazon RDS does not create a database.

DB parameter group [Info](#)

default.mysql5.7 ▼


Option group [Info](#)

default:mysql-5-7 ▼

- For Backup:
 - Check on **enable automatic backups**.
 - Provide **Backup retention period** as **7 days**.
 - Backup Window: No preference**
 - Leave rest as defaults

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](#)

Choose the number of days that RDS should retain automatic backups for this instance.

7 days ▼

Backup window [Info](#)

Select the period for which you want automated backups of the database to be created by Amazon RDS.

☐ Select window

☒ **No preference**

☒ **Copy tags to snapshots**

Monitoring

☐ **Enable Enhanced monitoring**

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

- For **Log exports**, you can select from various options that which type of logs you would like to analyze in CloudWatch. **Leave as default.**
- For **Maintenance**, leave as defaults. The default options will be auto check on enable auto minor version upgrade and maintenance window will be selected as **No preference.**
- For **Deletion protection**, if checked, it protects your database from accidental deletion and your database cannot be deleted as long as this option is checked. **Leave as default.**

Log exports


Select the log types to publish to Amazon CloudWatch Logs

- ☐ Audit log
- ☐ Error log
- ☐ General log
- ☐ Slow query log

IAM role

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

RDS Service Linked Role

 Ensure that General, Slow Query, and Audit Logs are turned on. Error logs are enabled by default. [Learn more](#)

Maintenance

Auto minor version upgrade [Info](#)

- ☒ **Enable auto minor version upgrade**
Enabling auto minor version upgrade will automatically upgrade to new minor versions as they are released. The automatic upgrades occur during the maintenance window for the database.

Maintenance window [Info](#)

Select the period you want pending modifications or maintenance applied to the database by Amazon RDS.

- ☐ Select 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.

14. At last, it will give you estimated costs for your selected configurations:

Estimated monthly costs

The Amazon RDS Free Tier is available to you for 12 months. Each calendar month, the free tier will allow you to use the Amazon RDS resources listed below for free:

- 750 hrs of Amazon RDS in a Single-AZ db.t2.micro Instance.
- 20 GB of General Purpose Storage (SSD).
- 20 GB for automated backup storage and any user-initiated DB Snapshots.

[Learn more about AWS Free Tier.](#) 

When your free usage expires or if your application use exceeds the free usage tiers, you simply pay standard, pay-as-you-go service rates as described in the [Amazon RDS Pricing page.](#) 

15. Review your settings and click **Create database.**

16. In the RDS Dashboard, monitor your new DB instance until the status changes from "creating" to "backing up" to "available".

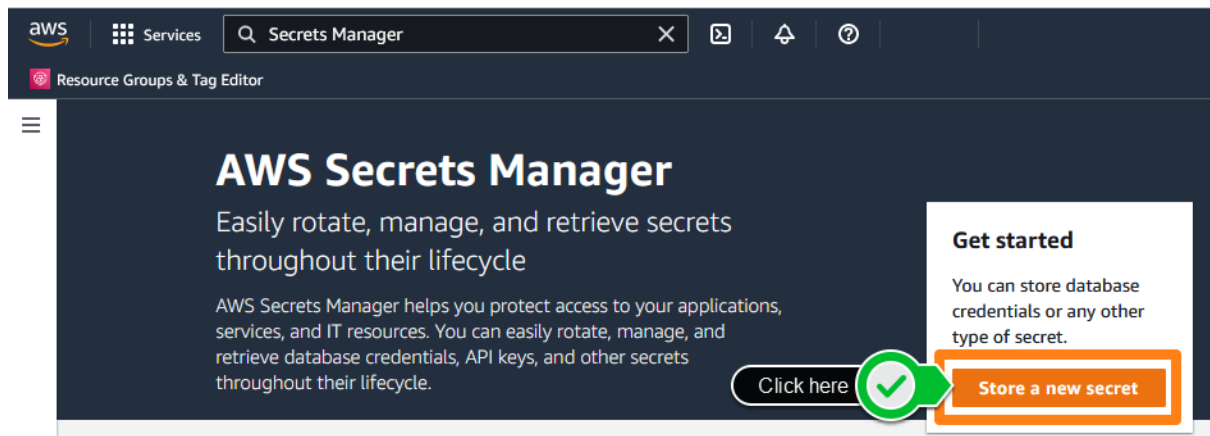
Note: This may take up to 5 minutes as the database is being created and backed up.

5-3 Save RDS Credentials

The web server you created contains sample code for a simple address book. We must tell the sample code how to find the database and connect to it. We will store this information in AWS Secrets Manager.

In this section, we will create a secret containing the database connection information. Later, we will grant permission to the web server to retrieve this secret.

1. In the console, open the [AWS Secrets Manager](#). Click **Store a new secret**.



2. Under **Secret Type**, choose **Credentials for Amazon RDS database**. Provide the user name and password you entered when you created the database.

Store a new secret

Secret type [Info](#)

☒ Credentials for Amazon RDS database

☐ Credentials for Amazon DocumentDB database

☐ Credentials for Amazon Redshift cluster

☐ Credentials for other database

☐ Other type of secret
API key, OAuth token, other.

Credentials [Info](#)

User name

Password

☒ Show password

Encryption key [Info](#)

You can encrypt using the KMS key that Secrets Manager creates or a customer managed KMS key that you create.

[Add new key](#)

- Under **Database**, choose the database you just created. Click **Next**.

Database [Info](#)

Search instances

< 1 >

DB instance	DB engine	Status	Creation date
awsdb	mysql		

Click here

Cancel Next

- Name your secret, "**mysecret**". The sample code is written to ask for the secret by this specific name. Click **Next**.

Note: For the purpose of this lab, the secret has to be named exactly as "**mysecret**"

Store a new secret

Secret name and description [Info](#)

Secret name

A descriptive name that helps you find your secret later.

mysecret


Secret name must contain only alphanumeric characters and the characters / _ + = . @ -

Click here

Cancel Previous Next

- Leave **Secret rotation** at default values. Click **Next**.

Store a new secret

 If you turn on automatic rotation, the first rotation will happen immediately when you store this secret. See [Rotation](#) in the Secrets Manager User Guide.

Secret rotation

Configure automatic rotation - optional [Info](#)

Configure AWS Secrets Manager to rotate this secret automatically.

☐ Automatic rotation

6. Review your choices. Click **Store**.

Sample code

Use these code samples to retrieve the secret in your application.

Java

JavaV2

JavaScript

C#

Python3

Ruby

Go

[illegible]

 [Download AWS SDK for Java](#)

Cancel

[Previous](#)

Store

5-4 Access RDS from EC2

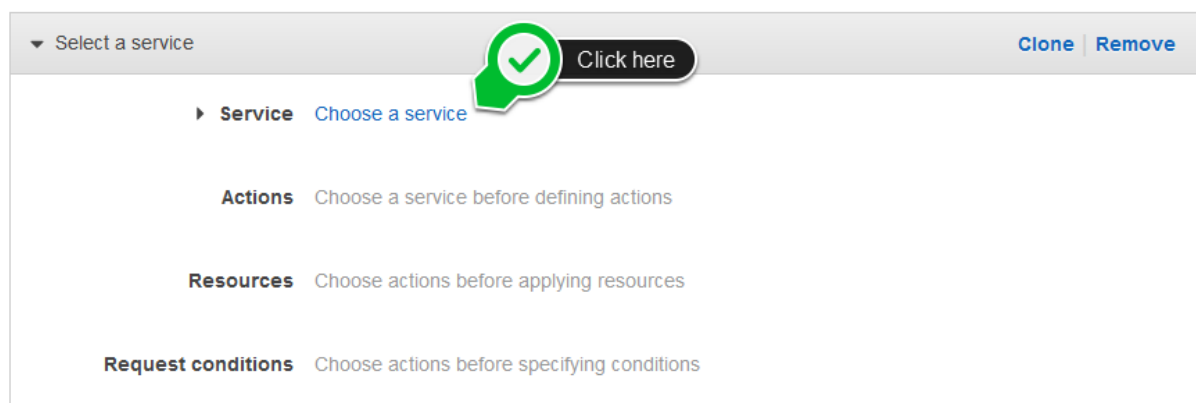
Allow the web server to access the secret

Now that you have created a secret, you must give your web server permission to use it. To do this, we will create a **Policy** that allows the web server to read a secret. We will add this policy to the **Role** you previously assigned to the web server.

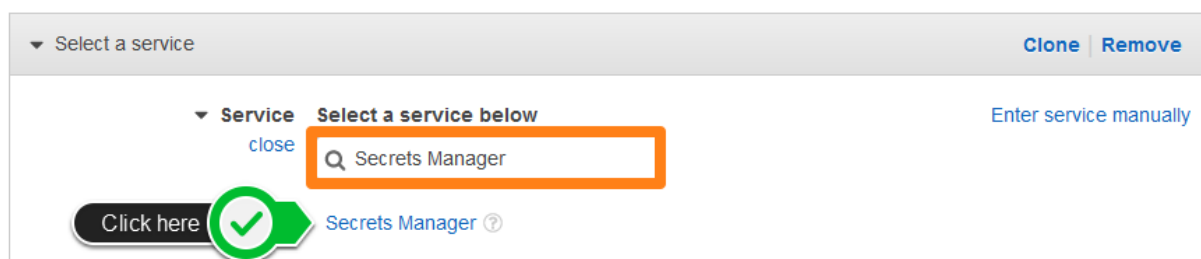
1. If you have not already done so, create an **IAM Instance Profile** as described in [Connect to your Linux instance using Session Manager](#).
2. Sign in to the AWS Management Console and open the [IAM console](#). In the navigation pane, choose **Policies**, and then choose **Create Policy**.



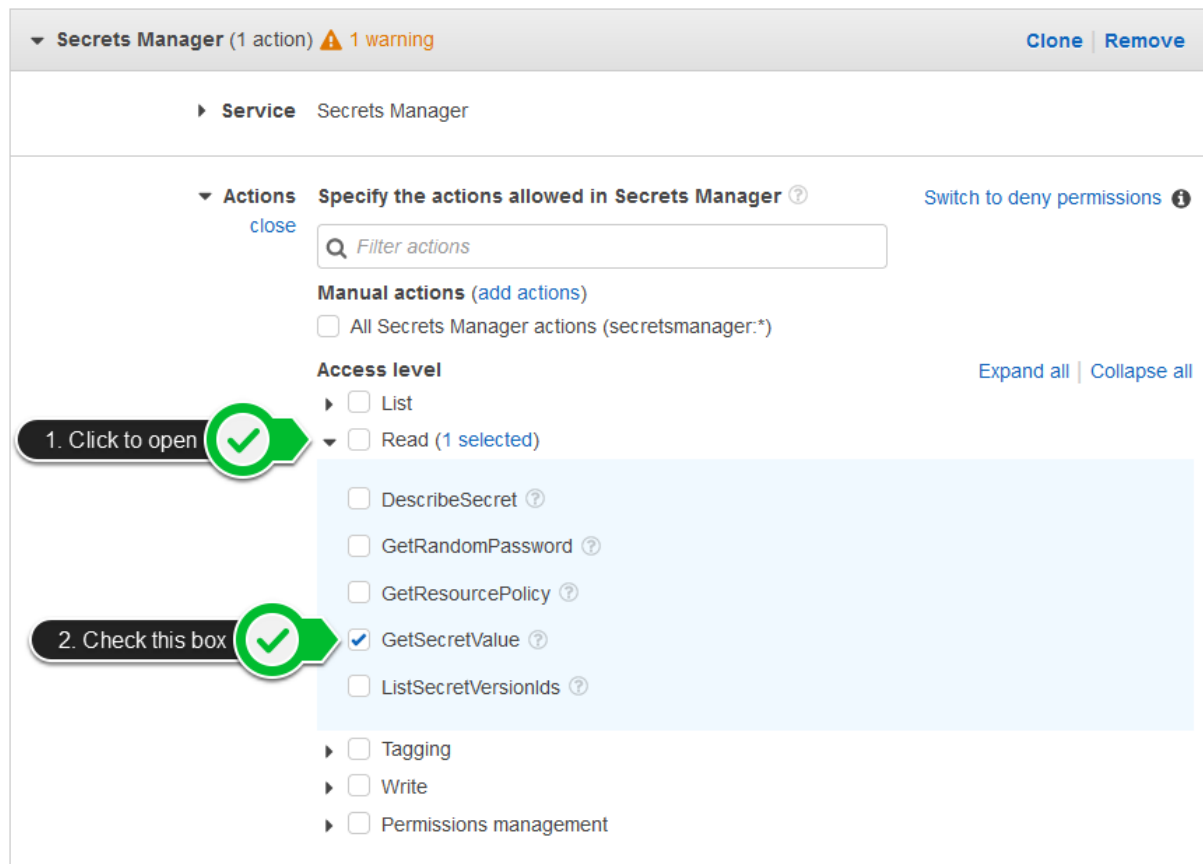
3. Click **Choose a service**.



4. Type **Secrets Manager** into the search box. Click **Secrets Manager**.

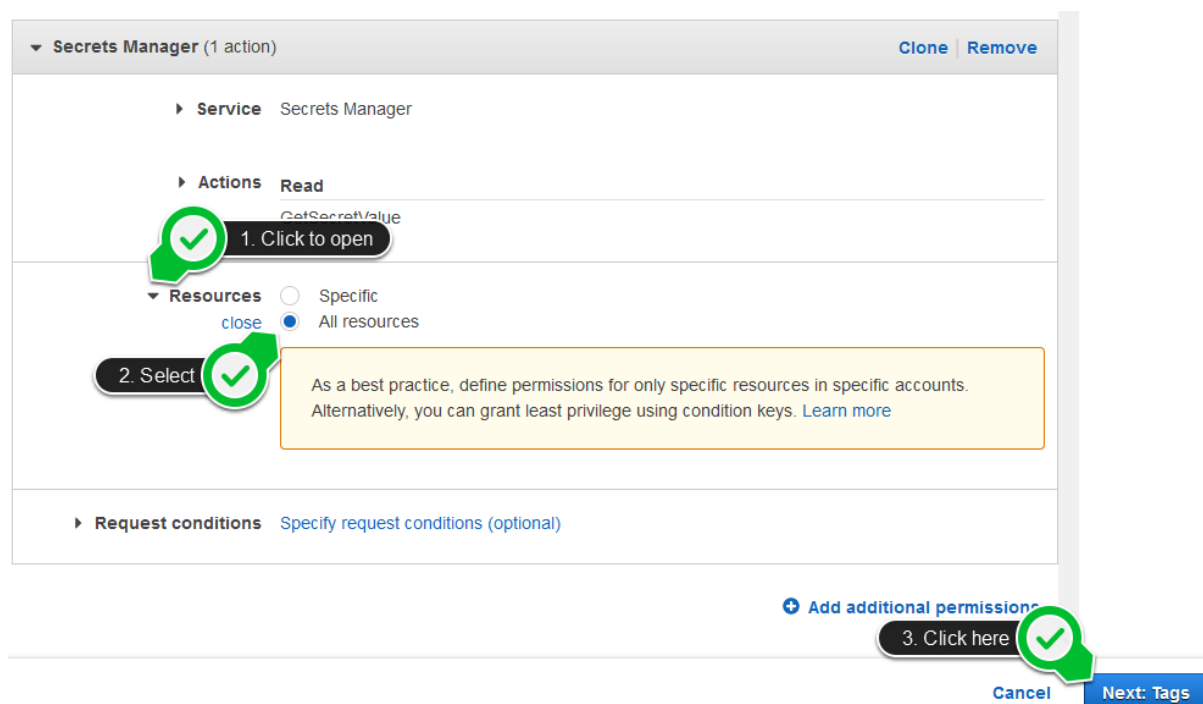


5. Under **Access level**, click on the carat next to **Read** and then check the box by **GetSecretValue**.



6. Click on the carat next to **Resources**. For this lab, select **All resources**. Click **Next: Tags**.

Note: For the lab, we're allowing EC2 to access all secrets. With a real workload, you should consider allowing access to specific secrets.



7. Click **Next: Review**.

Add tags (Optional)

Tags are key-value pairs that you can add to AWS resources to help identify, organize, or search for resources.

No tags associated with the resource.

Add tag

You can add up to 50 more tags

Click here



Cancel

Previous

Next: Review

8. On the **Review Policy** screen, give your new policy the name **ReadSecrets**. Click **Create policy**.

Review policy

Name* ReadSecrets

Use alphanumeric and '+=, @-_' characters. Maximum 128 characters.

Description

Maximum 1000 characters. Use alphanumeric and '+=, @-_' characters.

Summary

Filter		
Service	Access level	Resource
Allow (1 of 314 services) Show remaining 313		
Secrets Manager	Limited: Read	All resources

Tags

Key	Value
-----	-------

No tags associated with the resource.

Click here



Cancel

Previous

Create policy

9. In the navigation pane, choose **Roles** and type **SSMInstanceProfile** into the search box. This is the role you created previously in ["Connect to your Linux instance using Session Manager"](#). Click **SSMInstanceProfile**.

Roles (106) [Info](#)

An IAM role is an identity you can create that has specific permissions with credentials that are valid for short durations. Roles can be assumed by entities that you trust.

1 match < 1 >

<input type="checkbox"/>	Role name	Trusted entities
<input type="checkbox"/>	SSMInstanceProfile	AWS Service: ec2

10. Under **Permissions policies**, click **Attach policies**.

Summary [Edit](#)

Creation date March 21, 2022, 19:15 (UTC+09:00)	ARN arn:aws:iam::[redacted]:role/SSMInstanceProfile	Instance profile ARN arn:aws:iam::[redacted]:instance-profile/SSMInstanceProfile
Last activity None	Maximum session duration 1 hour	

[Permissions](#) | [Trust relationships](#) | [Tags](#) | [Access Advisor](#) | [Revoke sessions](#)

Permissions policies (1) [Simulate](#) [Remove](#) [Add permissions](#)

You can attach up to 10 managed policies.

<input type="checkbox"/>	Policy name	Type	Description
<input type="checkbox"/>	AmazonSSMManagedInstanceCore	AWS managed	The policy for Amazon EC2 Role to enable AWS Systems Manager...

Permissions boundary - (not set)

Set a permissions boundary to control the maximum permissions this role can have. This is not a common setting but can be used to delegate permission management to others.

11. Search for the policy you created called **ReadSecrets**. Check the box and click **Attach policy**.

Attach Permissions [Create policy](#) [Refresh](#)

Filter policies Showing 1 result

<input type="checkbox"/>	Policy name	Type	Used as
<input checked="" type="checkbox"/>	ReadSecrets	Customer managed	None

1. Check box

2. Click here

[Cancel](#) [Attach policy](#)

12. Under **Permissions policies**, verify that **AmazonSSMManagedInstanceCore** and **ReadSecrets** are both listed.

Permissions

Trust relationships

Tags

Access Advisor

Revoke sessions

▼ Permissions policies (2 policies applied)

Attach policies

Add inline policy

Policy name	Policy type	
▶ AmazonSSMManagedInstanceCore	AWS managed policy	✕
▶ ReadSecrets	Managed policy	✕

Try the Address Book

1. Navigate to the [EC2 console](#) and find the web server you launched in the EC2 Linux Hands-On Lab. Note your web server's public IP.

Instances (1/1)

Info

Refresh

Connect

Instance state

Actions

Launch Instances

Search

Instance state = running

Clear filters

✓	Name	Instance ID	Instance state	Instance type	Status check	Alarm status
✓	Web server for IMD	i-089d22683ed0fc34e	Running	t2.micro	2/2 checks passed	No alarms

Instance: i-089d22683ed0fc34e (Web server for IMD)

Details

Security

Networking

Storage

Status checks

Monitoring

Tags

▼ Instance summary

Instance ID

i-089d22683ed0fc34e (Web server for IMD)

IPv6 address

-

Public IPv4 address

3.35.210.158 | open address

Instance state

Running

Private IPv4 addresses

172.31.43.213

Public IPv4 DNS

ec2-3-35-210-158.ap-northeast-2.compute.amazonaws.com | open address

2. Open a new tab and reconnect to your web server's public IP. Click RDS.

Not Secure

powered by aws

LOAD TEST

RDS

Meta-Data

Value

InstanceId

i-0f3f08b5ba0a89bbb

Availability Zone

ap-northeast-2a

Current CPU Load: 0%

3. You should now see a simple page displaying all of the information from the database you just created.



LOAD TEST

RDS

Address Book

Name	Phone	Email	Admin	
			Add Contact	
Alice	571-555-4875	alice@address2.us	Edit	Remove
Bob	630-555-1254	bob@fakeaddress.com	Edit	Remove

This is a very basic example of a simple address book interacting with a MySQL database managed by AWS. RDS can support much more complicated relational database scenarios, but we hope this simple example will suffice to demonstrate the point.

Feel free to play around with the address book and add/edit/remove content from your RDS database by using the **Add Contact**, **Edit**, and **Remove** links in the Address Book.

Great Job: You have successfully deployed and utilized an AWS managed MySQL database!!!