

Declaration	
Questions in this exercise are intentionally complex and could be convoluted or confusing. This is by design and to simulate real life situations where customers seldom give crystal clear requirements and ask unambiguous questions.	

I have read the above statement and agree to these conditions	
I AGREE	Nivedita
<Enter your name above this line to indicate that you are in agreement>	

Instructions	
Every screenshot requested in this workbook is compulsory and carries 0.5 marks	
Your AWS account ID must be clearly visible in every screenshot using the AWS console; missing id or using someone else's id is not permitted. Such cases will be considered as plagiarism and severe penalty will be imposed.	
All screenshots must be in the order mentioned under "Expected Screenshots" for every step	
DO NOT WAIT UNTIL THE LAST MINUTE.	
The file should be renamed in the format BATCH_FIRSTNAME_LASTNAME_PROJECT1. For example: IITR_FSD_VIJAY_DWIVEDI_PROJECT1.docx	

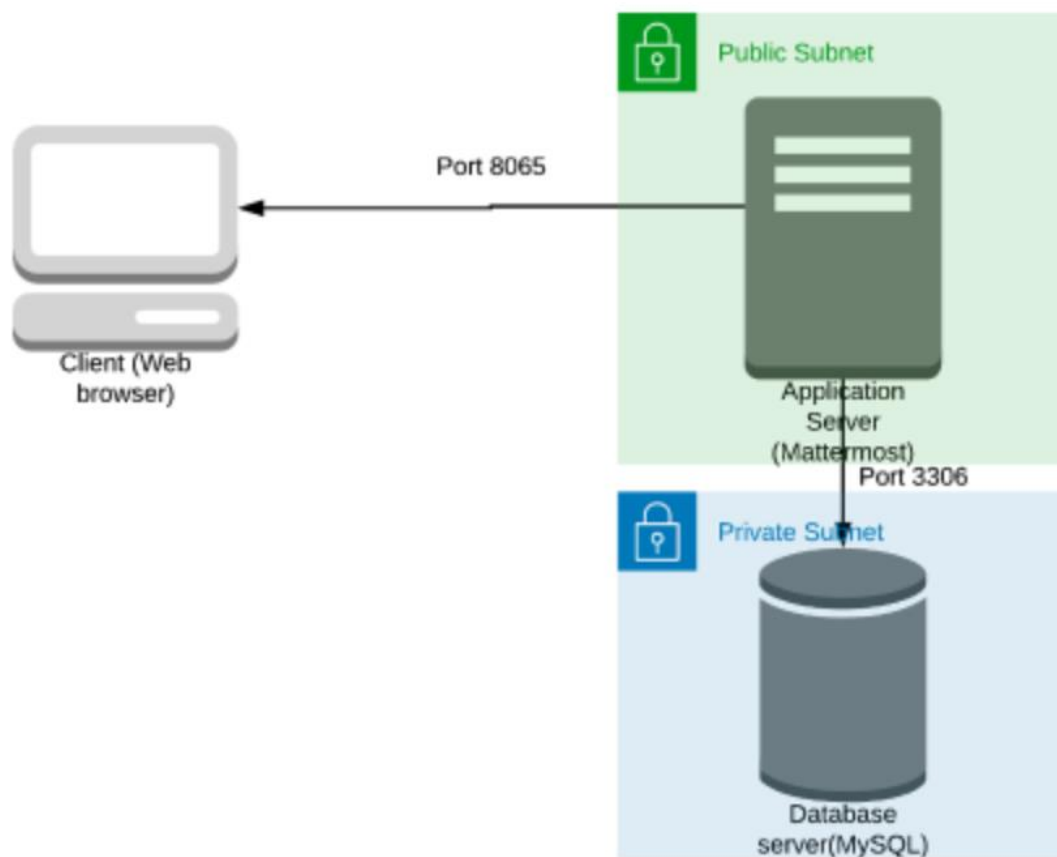
Resource Clean Up	
Cloud is always pay per use model and all resources/services that we consume are chargeable. Cleaning up when you've completed your lab or project is always necessary. This is true whether you're doing a lab or implementing a project at your workplace.	
After completing the lab, make sure to delete each resource created in reverse chronological order.	

Scenario

Team communication and instant messaging solutions are an integral part of any business environment today. As of 2020, the total number of users of Slack and Microsoft Teams exceeded 20 million.

Some organizations might have compliance policies in place which do not allow them to use services managed by third parties. They will prefer solutions that can be managed and hosted on servers controlled by them. The same will extend to communication solutions as well.

Architecture diagram



Architecture Implementation

- | | |
|---|---|
| 1 | Implement 2 different subnets (one public and the other private) in a custom VPC |
| 2 | Install and configure MySQL on an Amazon Linux 2 instance on the private subnet using the instructions provided. (Hint: Use a bastion host and a NAT gateway) |

3	Install and configure Mattermost on an Amazon Linux 2 instance on the public subnet using the provided instructions.
4	Configure the security groups to allow the ports as shown in the architecture.
5	Test the installation by accessing the IP of the public instance in a browser via the port 8065.

Step 1: VPC and Subnet Creation

Step number	A
Step name	Creation of VPC
Instructions	<ol style="list-style-type: none"> 1) Navigate to VPC using the Services button at the top of the screen 2) Select "Your VPCs" on the left side of the screen 3) Click on "Create VPC" 4) Enter the following fields : Name: Project 1 VPC IPv4 CIDR Block : 10.0.0.0/16 The rest of the options can be ignored 5) Select "Create VPC" 6) Select the VPC and click on Actions->Edit DNS hostnames 7) Enable DNS hostnames and click on Save
Expected screenshots	<ol style="list-style-type: none"> 1) Created VPC with properties visible

<Insert Screenshot a(1) here>

VPC > Your VPCs > vpc-ob179486e850dcbcb

You have successfully modified the settings for vpc-ob179486e850dcbcb / Project 1 VPC.

vpc-ob179486e850dcbcb / Project 1 VPC

Details Info

VPC ID

vpc-ob179486e850dcbcb

DNS resolution

Enabled

Main network ACL

acl-0dc88e21241c79edd

IPv6 CIDR (Network border group)

-

State

Available

Tenancy

default

Default VPC

No

Network Address Usage metrics

Disabled

Block Public Access

Off

DHCP option set

dopt-0515b130ceaa98519

IPv4 CIDR

10.0.0.0/16

Route 53 Resolver DNS Firewall rule groups

Failed to load rule groups

DNS hostnames

Enabled

Main route table

rtb-Ob210acc2930b4a1f

IPv6 pool

-

Owner ID

070904952432

Resource map

CIDRs

Flow logs

Tags

Integrations

Resource map Info

VPC Show details

Your AWS virtual network

Project 1 VPC

Subnets (0)

Subnets within this VPC

Route tables (1)

Route network traffic to resources

rtb-Ob210acc2930b4a1f

Network c

Connections to c

Step number	b
Step name	Creation of public subnet
Instructions	<div>1) Navigate to VPC->Subnets</div> <div>2) Click on "Create Subnet"</div> <div>3) Enter the following fields</div> <div>Name tag : Public Subnet</div> <div>VPC : Select the Project 1 VPC</div> <div>IPv4 CIDR block : 10.0.1.0/24</div> <div>The other options can be ignored</div> <div>4) Click on Create</div> <div>5) Once the subnet has been created, select the subnet and click on Actions->Modify Auto-assign IP settings</div> <div>6) Enable the option "Auto assign IPv4" and select Save</div>
Expected screenshots	<div>1) Subnet Creation screen</div>

<Insert Screenshot b(1) here>

Subnet 1 of 1

Subnet name
Create a tag with a key of 'Name' and a value that you specify.
Public Subnet
The name can be up to 256 characters long.

Availability Zone [Info](#)
Choose the zone in which your subnet will reside, or let Amazon choose one for you.
No preference

IPv4 VPC CIDR block [Info](#)
Choose the VPC's IPv4 CIDR block for the subnet. The subnet's IPv4 CIDR must lie within this block.
10.0.0.0/16

IPv4 subnet CIDR block
10.0.1.0/24 256 IPs

Tags - optional

Key	Value - optional	
Q Name	Public Subnet	Remove

[Add new tag](#)
You can add 49 more tags.

[Remove](#)

[Add new subnet](#)

[Cancel](#) [Create subnet](#)

Step number	C
Step name	Creation of private subnet
Instructions	1) Navigate to VPC->Subnets 2) Click on "Create Subnet" 3) Enter the following fields Name tag : Private Subnet VPC : Select the Project 1 VPC IPv4 CIDR block : 10.0.2.0/24 The other options can be ignored 4) Click on Create
Expected screenshots	1) Subnet Creation screen

<Insert Screenshot c(1) here>

Subnet name
Create a tag with a key of 'Name' and a value that you specify.
Private Subnet
The name can be up to 256 characters long.

Availability Zone [Info](#)
Choose the zone in which your subnet will reside, or let Amazon choose one for you.
No preference

IPv4 VPC CIDR block [Info](#)
Choose the VPC's IPv4 CIDR block for the subnet. The subnet's IPv4 CIDR must lie within this block.
10.0.0.0/16

IPv4 subnet CIDR block
10.0.2.0/24 256 IPs

Tags - optional

Key	Value - optional	
Name	Private Subnet	Remove

[Add new tag](#)
You can add 49 more tags.
[Remove](#)
[Add new subnet](#)

[Cancel](#) [Create subnet](#)

Step 2 : Internet Gateway and VPC

Step number	a
Step name	Creation and Configuration of Internet Gateway
Instructions	<ol style="list-style-type: none"> 1) Navigate to VPCs->Internet Gateway 2) Click on "Create Internet Gateway" 3) Enter the name tag "Project 1 Internet Gateway" and click on "Create Internet Gateway" 4) After the gateway is created, select it and click on Actions->Attach to VPC 5) Select the Project 1 VPC and click on "Attach Internet Gateway"
Expected screenshots	1) Creation of Internet Gateway

<Insert Screenshot a(1) here >

aws

Search

[Option+S]

Oregon

voclabs/user3470877=niveditam10@gmail.com @ 0709-0495-2432

[VPC](#) > [Internet gateways](#) > Create Internet gateway

Create internet gateway

Info

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

Internet gateway settings

Name tag

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

Project 1 Internet Gateway

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

Q Name X

Value - optional

Q Project 1 Internet Gateway X

Remove

Add new tag

You can add 49 more tags.

Cancel

Create Internet gateway

<Insert Screenshot b(1) here>

Step number	c
Step name	Creation of NAT gateway
Instructions	<ol style="list-style-type: none"> 1) Navigate to VPC using the Services button at the top of the screen 2) Select NAT Gateway at the left side of the screen 3) Click on Create NAT Gateway <ul style="list-style-type: none"> - Deploy it in the public subnet - Connectivity type : Public - Allocate an elastic IP by clicking on "Allocate Elastic IP" 4) Click on "Create NAT Gateway" to create the gateway
Expected screenshots	<ol style="list-style-type: none"> 1) NAT gateway creation details 2) Gateway after creation

<Insert Screenshot c(1) here>

aws

Search

[Option+S]

Oregon

voclabs/user3470877=niveditarn10@gmail.com @ 0709-0495-2432

VPC > NAT gateways > Create NAT gateway

NAT gateway settings

Name - optional

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

my-nat-gateway-01

The name can be up to 256 characters long.

Subnet

Select a subnet in which to create the NAT gateway.

subnet-0087f21a3bfbace86 (Public Subnet)

Connectivity type

Select a connectivity type for the NAT gateway.

☒ Public

☐ Private

Elastic IP allocation ID

Assign an Elastic IP address to the NAT gateway.

elpalloc-0b03fdc54f05ce86b

Allocate Elastic IP

Additional settings

Tags

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

No tags associated with the resource.

Add new tag

You can add 50 more tags.

Cancel>Create NAT gateway

<Insert Screenshot c(2) here>

aws

Search

[Option+S]

Oregon

voclabs/user3470877=niveditarn10@gmail.com @ 0709-0495-2432

VPC > NAT gateways > nat-0b078cbdc87be923e

VPC dashboard

EC2 Global View

Filter by VPC

Virtual private cloud

Your VPCs

Subnets

Route tables

Internet gateways

Egress-only internet gateways

Carrier gateways

DHCP option sets

Elastic IPs

Managed prefix lists

NAT gateways

Peering connections

Security

Network ACLs

Security groups

PrivateLink and Lattice

Getting started

Endpoints

Endpoint services

NAT gateway nat-0b078cbdc87be923e was created successfully.

Actions

Details

NAT gateway ID

nat-0b078cbdc87be923e

NAT gateway ARN

arn:aws:ec2:us-west-2:070904952432:natgateway/nat-0b078cbdc87be923e

VPC

vpc-0b179486e850dcbcb / Project 1 VPC

Connectivity type

Public

Primary public IPv4 address

-

Subnet

subnet-0087f21a3bfbace86 / Public Subnet

State

Pending

Primary private IPv4 address

-

Created

Monday, December 23, 2024 at 22:10:25 GMT+5:30

State message

-

Primary network interface ID

-

Deleted

-

Secondary IPv4 addresses

Monitoring

Tags

Secondary IPv4 addresses

Search

Private IPv4 address

Network interface ID

Status

Failure message

Secondary IPv4 addresses are not available for this nat gateway.

Step number	d
Step name	Creation of private route tables
Instructions	<p>1) Navigate to VPC -> Route Tables and click on Create Route table</p> <p>2) Enter the name tag "Private Route Table", select the Project 1 VPC from the dropdown and click on Create</p> <p>3) Once the route table is created, select it and select the Routes tab below the list of route tables</p> <p>4) Click in Edit Routes and add the following route (Don't edit the existing one)</p> <ul style="list-style-type: none"> - Destination : 0.0.0.0/0 - Target: Select NAT Gateway and select the NAT Gateway created in the previous step <p>Click on Save Routes</p> <p>5) Select the Subnet Associations tab and click on Edit Subnet Associations</p> <p>6) Select the private Subnet from the list and click on Save</p>
Expected screenshots	<p>1) Route list of the route table</p> <p>2) Subnet association of the route table</p>

<Insert Screenshot for d(1) here >

aws

Search

[Option+S]

Oregon

voclabs/user:3470877=niveditarn10@gmail.com @ 0709-0495-2432

VPC > Route tables > rtb-06aa49734ab56e3af > Edit subnet associations

Edit subnet associations

Change which subnets are associated with this route table.

Available subnets (1/2)

Filter subnet associations

	Name	Subnet ID	IPv4 CIDR	IPv6 CIDR	Route table ID
<input type="checkbox"/>	Public Subnet	subnet-0087f21a3bfabce86	10.0.1.0/24	-	rtb-0a77482b7141aab3f / Public Rout
<input checked="" type="checkbox"/>	Private Subnet	subnet-0f72ea8c179f91c18	10.0.2.0/24	-	Main (rtb-0b210acc2930b4a1f)

Selected subnets

subnet-0f72ea8c179f91c18 / Private Subnet

Cancel

Save associations

<Insert Screenshot for d(2) here>

aws

Search

[Option+S]

Oregon

voclabs/user:3470877=niveditarn10@gmail.com @ 0709-0495-2432

VPC > Route tables > rtb-06aa49734ab56e3af

VPC dashboard

EC2 Global View

Filter by VPC

Virtual private cloud

Your VPCs

Subnets

Route tables

Internet gateways

Egress-only Internet gateways

Carrier gateways

DHCP option sets

Elastic IPs

Managed prefix lists

NAT gateways

Peering connections

Security

Network ACLs

Security groups

PrivateLink and Lattice

Getting started

Endpoints

Endpoint services

Updated routes for rtb-06aa49734ab56e3af / Private Route Table successfully

Details

rtb-06aa49734ab56e3af / Private Route Table

Actions

Details

Route table ID

rtb-06aa49734ab56e3af

VPC

vpc-0b179486e850dcbcb | Project 1 VPC

Main

No

Owner ID

070904952432

Explicit subnet associations

-

Edge associations

-

Routes

Subnet associations

Edge associations

Route propagation

Tags

Routes (2)

Filter routes

Destination	Target	Status	Propagated
0.0.0.0/0	igw-0c3cdbfead7f840c9	Active	No
10.0.0.0/16	local	Active	No

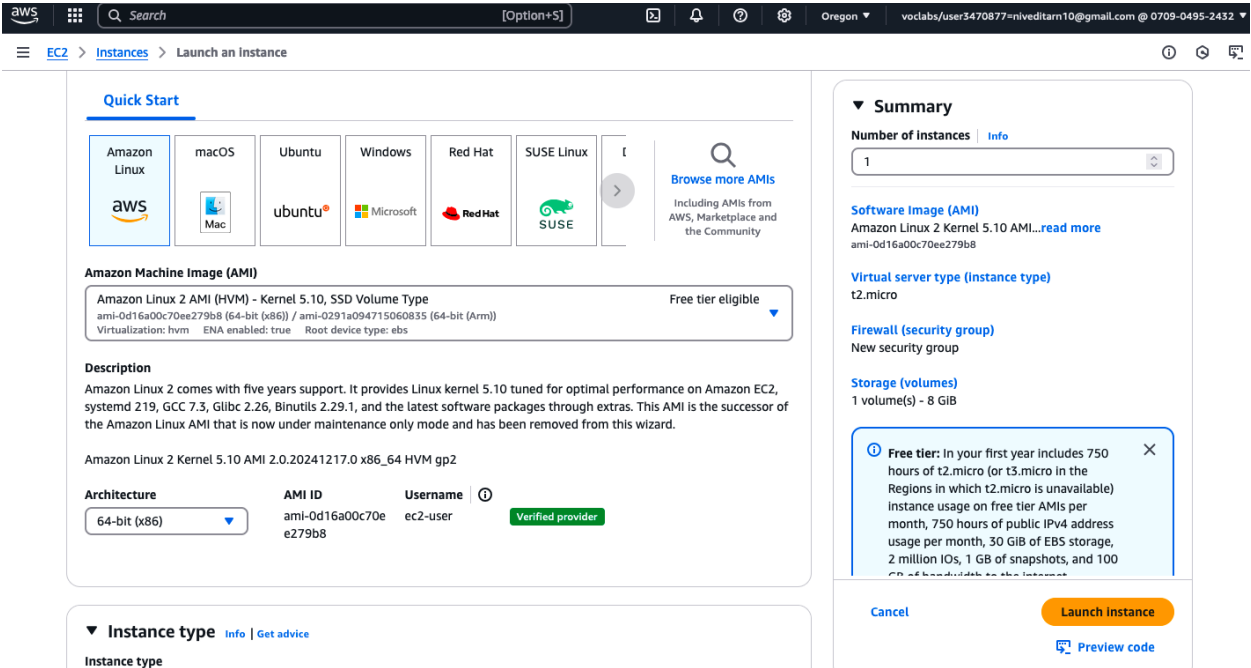
Both

Edit routes

Step 3 : Creation of database and application servers

Step number	a
Step name	Creation of application server
Instructions	1) Navigate to EC2 using the Services button at the top of the screen 2) Select Instances at the left side of the screen 3) Click on Launch Instance - Select the AMI Amazon 2 Linux - Select the instance type t2.micro - Select Network as "Project 1 VPC" and subnet as "Public Subnet" - For the security group, open the ports 80,443, 22 and 8065 for source set to "Anywhere" 4) Launch the instance after creating a new pem file and downloading it
Expected screenshots	1) AMI used 2) Instance configuration screen 3) Security group rules 4) Instance after creation

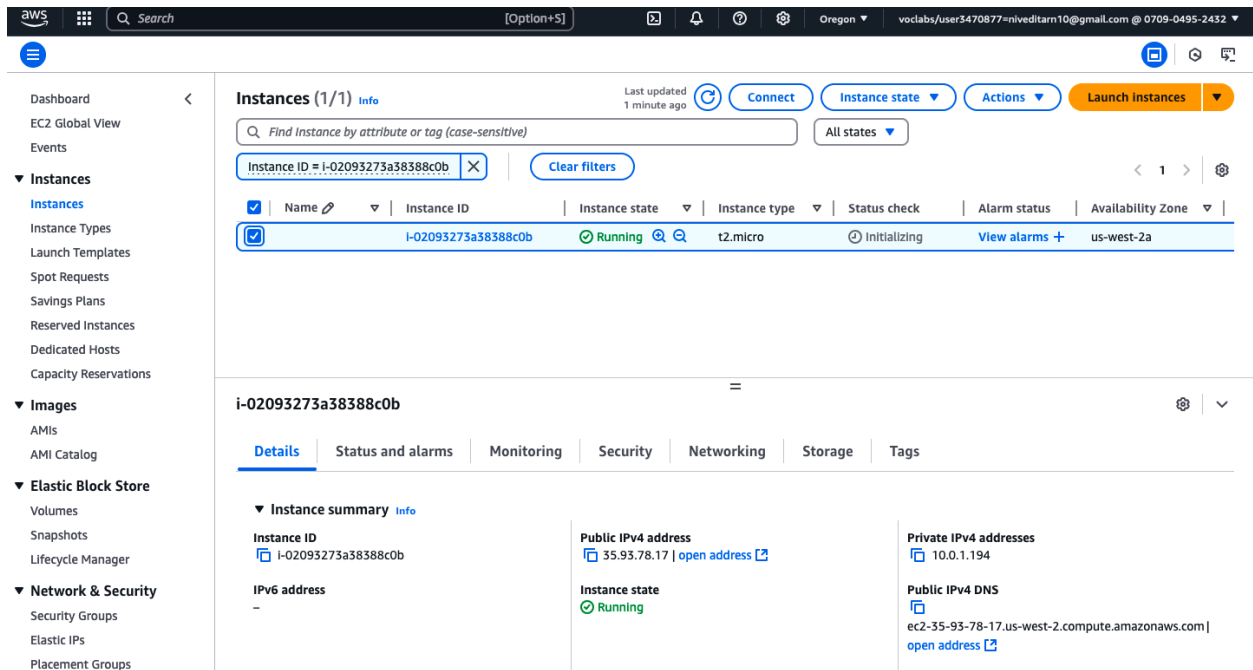
<Insert screenshot a(1) here>



<Insert screenshot a(2) here>

<Insert screenshot a(3) here>

<Insert screenshot a(4) here>



Instances (1/1) Info Last updated 1 minute ago [Connect](#) [Instance state](#) [Actions](#) [Launch instances](#)

[Clear filters](#) All states

Instance ID = i-02093273a38388c0b

<input checked="" type="checkbox"/>	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone
<input checked="" type="checkbox"/>		i-02093273a38388c0b	Running	t2.micro	Initializing	View alarms	us-west-2a

i-02093273a38388c0b

[Details](#) [Status and alarms](#) [Monitoring](#) [Security](#) [Networking](#) [Storage](#) [Tags](#)

Instance summary Info

Instance ID i-02093273a38388c0b	Public IPv4 address 35.93.78.17 open address	Private IPv4 addresses 10.0.1.194
IPv6 address -	Instance state Running	Public IPv4 DNS ec2-35-93-78-17.us-west-2.compute.amazonaws.com open address

Step number b

Step name Creation of database server

Instructions

- 1) Navigate to EC2 using the Services button at the top of the screen
- 2) Select Instances at the left side of the screen
- 3) Click on Launch Instance
 - Select the AMI Amazon 2 Linux
 - Select the instance type t2.micro
 - Select Network as "Project 1 VPC" and subnet as "Private Subnet"
 - For the security group, open the ports 80, 443,22 and 3306 for source set to "Anywhere"
- 4) Launch the instance by selecting the same pem file created in the previous step

- Expected screenshots
- 1) AMI used

2) Instance configuration screen

3) Security group rules

4) Instance after creation

<Insert screenshot b(1) here>

aws

Search

[Option+S]

Oregon

voclabs/user3470877=niveditam10@gmail.com @ 0709-0495-2432

EC2

Instances

Launch an Instance

▼ Application and OS Images (Amazon Machine Image) Info

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below

Search our full catalog including 1000s of application and OS images

Recents

Quick Start

Amazon Linux

macOS

Ubuntu

Windows

Red Hat

SUSE Linux

aws

Mac

ubuntu

Microsoft

Red Hat

SUSE

Browse more AMIs

Including AMIs from AWS, Marketplace and the Community

Amazon Machine Image (AMI)

Amazon Linux 2 AMI (HVM) - Kernel 5.10, SSD Volume Type

ami-0d16a00c70ee279b8 (64-bit (x86)) / ami-0291a094715060835 (64-bit (Arm))

Virtualization: hvm ENA enabled: true Root device type: ebs

Free tier eligible

Description

Amazon Linux 2 comes with five years support. It provides Linux kernel 5.10 tuned for optimal performance on Amazon EC2, systemd 219, GCC 7.3, Glibc 2.26, Binutils 2.29.1, and the latest software packages through extras. This AMI is the successor of the Amazon Linux AMI that is now under maintenance only mode and has been removed from this wizard.

Amazon Linux 2 Kernel 5.10 AMI 2.0.20241217.0 x86_64 HVM gp2

Architecture

64-bit (x86)

AMI ID

ami-0d16a00c70ee279b8

Username

ec2-user

Verified provider

▼ Summary

Number of instances Info

1

Software Image (AMI)

Amazon Linux 2 Kernel 5.10 AMI...read more

ami-0d16a00c70ee279b8

Virtual server type (instance type)

t2.micro

Firewall (security group)

New security group

Storage (volumes)

1 volume(s) - 8 GiB

Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 750 hours of public IPv4 address usage per month, 30 GiB of EBS storage, 2 million I/Os, 1 GB of snapshots, and 100 GB of bandwidth to the Internet

Cancel

Launch Instance

Preview code

<Insert screenshot b(2) here>

aws

Search

[Option+S]

Oregon

voclabs/user3470877=niveditam10@gmail.com @ 0709-0495-2432

EC2

Instances

Launch an Instance

Launch an instance Info

Amazon EC2 allows you to create virtual machines, or instances, that run on the AWS Cloud. Quickly get started by following the simple steps below.

Name and tags Info

Name

Database Server

Add additional tags

► Application and OS Images (Amazon Machine Image) Info

▼ Instance type Info | Get advice

Instance type

t2.micro

Family: t2 1 vCPU 1 GiB Memory Current generation: true

On-Demand Linux base pricing: 0.0116 USD per Hour

On-Demand SUSE base pricing: 0.0116 USD per Hour

On-Demand Windows base pricing: 0.0162 USD per Hour

On-Demand RHEL base pricing: 0.026 USD per Hour

On-Demand Ubuntu Pro base pricing: 0.0134 USD per Hour

Free tier eligible

All generations

Compare instance types

Additional costs apply for AMIs with pre-installed software

► Key pair (login) Info

▼ Summary

Number of instances Info

1

Software Image (AMI)

Amazon Linux 2 Kernel 5.10 AMI...read more

ami-0d16a00c70ee279b8

Virtual server type (instance type)

t2.micro

Firewall (security group)

New security group

Storage (volumes)

1 volume(s) - 8 GiB

Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 750 hours of public IPv4 address usage per month, 30 GiB of EBS storage, 2 million I/Os, 1 GB of snapshots, and 100 GB of bandwidth to the Internet

Cancel

Launch Instance

Preview code

<Insert screenshot b(3) here>

aws

Search

[Option+S]

Oregon

voclabs/user3470877=niveditarn10@gmail.com @ 0709-0495-2432

EC2

Instances

Launch an Instance

Firewall (security groups)

Create security group

Select existing security group

Security group name - required

Database Server SG

Description - required

Allows SSH access to developers

Inbound Security Group Rules

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

Type

ssh

Protocol

TCP

Port range

22

Source type

Anywhere

Source

0.0.0.0/0

Description - optional

e.g. SSH for admin desktop

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

Type

HTTP

Protocol

TCP

Port range

80

Source type

Source

Description - optional

Summary

Number of instances

1

Software Image (AMI)

Amazon Linux 2 Kernel 5.10 AMI...read more

Virtual server type (instance type)

t2.micro

Firewall (security group)

New security group

Storage (volumes)

1 volume(s) - 8 GiB

Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 750 hours of public IPv4 address usage per month, 30 GiB of EBS storage, 2 million I/Os, 1 GB of snapshots, and 100 GB of bandwidth to the Internet

Cancel

Launch Instance

Preview code

<Insert screenshot b(4) here>

aws

Search

[Option+S]

Oregon

voclabs/user3470877=niveditarn10@gmail.com @ 0709-0495-2432

Instances (1/2)

Last updated 1 minute ago

Connect

Instance state

Actions

Launch instances

Find Instance by attribute or tag (case-sensitive)

All states

	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone
<input checked="" type="checkbox"/>	Database Server	i-0842e0fc6af9b3da7	Running	t2.micro	Initializing	View alarms +	us-west-2a
<input type="checkbox"/>	Application Se...	i-02093273a38388c0b	Running	t2.micro	2/2 checks pass	View alarms +	us-west-2a

i-0842e0fc6af9b3da7 (Database Server)

Details

Status and alarms

Monitoring

Security

Networking

Storage

Tags

Instance summary

Instance ID

i-0842e0fc6af9b3da7

Public IPv4 address

34.221.16.207 | open address

Private IPv4 addresses

10.0.2.65

Instance state

Running

Public IPv4 DNS

ec2-34-221-16-207.us-west-2.compute.amazonaws.com | open address

Step 4: Application and Database Installation and Testing

Step number	a
-------------	---

Step name	Installation and configuration of MySQL
Instructions	<p>1) Copy the database pem file into the application server using the below command</p> <pre>scp -i <application server pem file> <database server pem file > ec2-user@<application server public IP>:/home/ec2-user</pre> <p>2) Log into the application server using SSH/Putty</p> <p>3) From the application server, log into the database server using the pem file copied in step 1 and the private IP address of the database server with the following command</p> <pre>ssh -i <database server pem file> ec2-user@<private IP of database server></pre> <p>4) Enter the following commands to install and configure MySQL on the database server</p> <pre>sudo yum update wget http://dev.mysql.com/get/mysql57-community-release-el7-9.noarch.rpm sudo yum localinstall mysql57-community-release-el7-9.noarch.rpm -y sudo yum install mysql-community-server -y --nogpgcheck sudo systemctl start mysqld.service</pre> <p>Run the below command to retrieve a temporary password for MySQL</p> <pre>sudo grep 'temporary password' /var/log/mysqld.log rev cut -d" " -f1 rev tr -d "."</pre> <p>Log in to MySQL with the below command and enter the above password when prompted</p> <pre>mysql -u root -p</pre> <p>Enter the below command after you login to MySQL</p> <pre>ALTER USER 'root'@'localhost' IDENTIFIED BY 'Password42!';</pre> <p>Type 'exit' into the MySQL prompt and press Enter to exit out of the MySQL environment.</p> <p>Enter the below commands to complete the setup. Ignore any warning messages you receive.</p> <pre>wget https://d60pu47qoi4ee.cloudfront.net/install_mysql_linux.sh chmod 777 install_mysql_linux.sh sudo ./install_mysql_linux.sh</pre> <p>5) Type <i>exit</i> to exit the database server and go back to the application server</p>
Expected screenshots	<ol style="list-style-type: none"> 1) Installation of MySQL 2) Retrieving the temporary password 3) Executing the provided script

<Insert screenshot a(1) here>

```
[ec2-user@ip-10-0-2-65 ~]$ wget http://dev.mysql.com/get/mysql57-community-release-el7-9.noarch.rpm
-bash: wget: http://dev.mysql.com/get/mysql57-community-release-el7-9.noarch.rpm: No such file or directory
[ec2-user@ip-10-0-2-65 ~]$ sudo yum localinstall mysql57-community-release-el7-9.noarch.rpm -y
-bash: sudo yum localinstall mysql57-community-release-el7-9.noarch.rpm: command not found
[ec2-user@ip-10-0-2-65 ~]$ sudo wget https://dev.mysql.com/get/mysql57-community-release-el7-11.noarch.rpm
--2024-12-23 18:15:32-- https://dev.mysql.com/get/mysql57-community-release-el7-11.noarch.rpm
Resolving dev.mysql.com (dev.mysql.com)... 23.6.186.166, 2600:1409:9800:98b::2e31, 2600:1409:9800:993::2e31
Connecting to dev.mysql.com (dev.mysql.com)[23.6.186.166]:443... connected.
HTTP request sent, awaiting response... 302 Moved Temporarily
Location: https://repo.mysql.com/mysql57-community-release-el7-11.noarch.rpm [following]
--2024-12-23 18:15:33-- https://repo.mysql.com/mysql57-community-release-el7-11.noarch.rpm
Resolving repo.mysql.com (repo.mysql.com)... 23.198.151.232, 2600:1409:9800:f91::1d68, 2600:1409:9800:f8c::1d68
Connecting to repo.mysql.com (repo.mysql.com)[23.198.151.232]:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 25680 (25k) [application/x-redhat-package-manager]
Saving to: 'mysql57-community-release-el7-11.noarch.rpm'

100%[=====] 25,680  --.-K/s  in 0.008s

2024-12-23 18:15:33 (3.21 MB/s) - 'mysql57-community-release-el7-11.noarch.rpm' saved [25680/25680]

[ec2-user@ip-10-0-2-65 ~]$ sudo yum localinstall mysql57-community-release-el7-9.noarch.rpm -y
-bash: sudo yum localinstall mysql57-community-release-el7-9.noarch.rpm: command not found
[ec2-user@ip-10-0-2-65 ~]$ sudo yum localinstall mysql57-community-release-el7-11.noarch.rpm
Failed to set locale, defaulting to C
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
Examining mysql57-community-release-el7-11.noarch.rpm: mysql57-community-release-el7-11.noarch
Marking mysql57-community-release-el7-11.noarch.rpm to be installed
Resolving Dependencies
--> Running transaction check
--> Package mysql57-community-release.noarch 0:el7-11 will be installed
--> Finished Dependency Resolution

Dependencies Resolved

=====
Package Arch Version Repository Size
=====
Installing:
mysql57-community-release
noarch el7-11 /mysql57-community-release-el7-11.noarch 31 k

Transaction Summary
=====
Install 1 Package

Total size: 31 k
Installed size: 31 k
Is this ok [y/d/N]: y
Downloading packages:
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
Installing : mysql57-community-release-el7-11.noarch 1/1
Verifying : mysql57-community-release-el7-11.noarch 1/1

Installed:
mysql57-community-release.noarch 0:el7-11

Complete!
[ec2-user@ip-10-0-2-65 ~]$ █
```

<Insert screenshot a(2) here>
<Insert screenshot a(3) here>

Step number	b
Step name	Installation and configuration of Mattermost
Instructions	1) Enter the following commands after logging into the application server via SSH to install and

configure Mattermost

wget https://d6opu47qoi4ee.cloudfront.net/install_mattermost_linux.sh

sudo yum install dos2unix -y
sudo dos2unix install_mattermost_linux.sh

chmod 700 install_mattermost_linux.sh
sudo ./install_mattermost_linux.sh <private IP of MySQL server>
Example : sudo ./install_mattermost_linux 173.65.34.7
sudo chown -R mattermost:mattermost /opt/mattermost
sudo chmod -R g+w /opt/mattermost
cd /opt/mattermost
sudo -u mattermost ./bin/mattermost

2) Check whether the server has been successfully deployed by navigating to the following URL in your web browser. The web page might take a couple of minutes to load.
<public IP of the application server>:8065

- Expected screenshots
- 1) Executing the script

2) Starting the Mattermost server

3) Accessing the application via web browser

<Insert screenshot b(1) here>

```
[ec2-user@ip-10-0-2-65 ~]$ wget http://dev.mysql.com/get/mysql57-community-release-el7-9.noarch.rpm
--bash: wget http://dev.mysql.com/get/mysql57-community-release-el7-9.noarch.rpm: No such file or directory
[ec2-user@ip-10-0-2-65 ~]$ sudo yum localinstall mysql57-community-release-el7-9.noarch.rpm -y
--bash: sudo yum localinstall mysql57-community-release-el7-9.noarch.rpm: command not found
[ec2-user@ip-10-0-2-65 ~]$ sudo wget https://dev.mysql.com/get/mysql57-community-release-el7-11.noarch.rpm
--2024-12-23 18:15:32-- https://dev.mysql.com/get/mysql57-community-release-el7-11.noarch.rpm
Resolving dev.mysql.com (dev.mysql.com)... 23.6.106.166, 2600:1409:9800:98b:2e31, 2600:1409:9800:993:2e31
Connecting to dev.mysql.com (dev.mysql.com)|23.6.106.166|:443... connected.
HTTP request sent, awaiting response... 302 Moved Temporarily
Location: https://repo.mysql.com/mysql57-community-release-el7-11.noarch.rpm [following]
--2024-12-23 18:15:33-- https://repo.mysql.com/mysql57-community-release-el7-11.noarch.rpm
Resolving repo.mysql.com (repo.mysql.com)... 23.198.151.232, 2600:1409:9800:f91::1d68, 2600:1409:9800:f8c::1d68
Connecting to repo.mysql.com (repo.mysql.com)|23.198.151.232|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 25680 (25K) [application/x-redhat-package-manager]
Saving to: 'mysql57-community-release-el7-11.noarch.rpm'

100%[=====] 25,680 --.-K/s in 0.008s

2024-12-23 18:15:33 (3.21 MB/s) - 'mysql57-community-release-el7-11.noarch.rpm' saved [25680/25680]

[ec2-user@ip-10-0-2-65 ~]$ sudo yum localinstall mysql57-community-release-el7-9.noarch.rpm -y
--bash: sudo yum localinstall mysql57-community-release-el7-9.noarch.rpm: command not found
[ec2-user@ip-10-0-2-65 ~]$ sudo yum localinstall mysql57-community-release-el7-11.noarch.rpm
Failed to set locale, defaulting to C
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
Examining mysql57-community-release-el7-11.noarch.rpm: mysql57-community-release-el7-11.noarch
Marking mysql57-community-release-el7-11.noarch.rpm to be installed
Resolving Dependencies
--> Running transaction check
--> Package mysql57-community-release.noarch 0:el7-11 will be installed
--> Finished Dependency Resolution

Dependencies Resolved

=====
Package Arch Version Repository Size
=====
Installing:
mysql57-community-release
noarch el7-11 /mysql57-community-release-el7-11.noarch 31 k

Transaction Summary
=====
Install 1 Package

Total size: 31 k
Installed size: 31 k
Is this ok [y/d/N]: y
Downloading packages:
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
Installing : mysql57-community-release-el7-11.noarch 1/1
Verifying : mysql57-community-release-el7-11.noarch 1/1

Installed:
mysql57-community-release.noarch 0:el7-11

Complete!
[ec2-user@ip-10-0-2-65 ~]$ █
```

<Insert screenshot b(2) here>

```
[ec2-user@ip-10-0-2-65 ~]$ wget https://d60pu47qoi4ee.cloudfront.net/install_mattermost_linux.sh
--2024-12-23 18:52:58-- https://d60pu47qoi4ee.cloudfront.net/install_mattermost_linux.sh
Resolving d60pu47qoi4ee.cloudfront.net (d60pu47qoi4ee.cloudfront.net)... 3.165.166.122, 3.165.166.43, 3.165.166.103, ...
Connecting to d60pu47qoi4ee.cloudfront.net (d60pu47qoi4ee.cloudfront.net)|3.165.166.122|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 592 [text/x-sh]
Saving to: 'install_mattermost_linux.sh.1'

100%[=====>] 592 --.-K/s in 0s

2024-12-23 18:52:51 (146 MB/s) - 'install_mattermost_linux.sh.1' saved [592/592]

[ec2-user@ip-10-0-2-65 ~]$ sudo yum install dos2unix -y
Failed to set locale, defaulting to C
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
amzn2-core | 3.6 kB 00:00
62 packages excluded due to repository priority protections
Resolving Dependencies
--> Running transaction check
--> Package dos2unix.x86_64 0:6.0.3-7.amzn2.0.3 will be installed
--> Finished Dependency Resolution

Dependencies Resolved

=====
Package Arch Version Repository Size
=====
Installing:
dos2unix x86_64 6.0.3-7.amzn2.0.3 amzn2-core 75 k

Transaction Summary
=====
Install 1 Package

Total download size: 75 k
Installed size: 190 k
Downloading packages:
dos2unix-6.0.3-7.amzn2.0.3.x86_64.rpm | 75 kB 00:00
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
Installing : dos2unix-6.0.3-7.amzn2.0.3.x86_64 1/1
Verifying : dos2unix-6.0.3-7.amzn2.0.3.x86_64 1/1

Installed:
dos2unix.x86_64 0:6.0.3-7.amzn2.0.3

Complete!
[ec2-user@ip-10-0-2-65 ~]$ sudo dos2unix install_mattermost_linux.sh
dos2unix: converting file install_mattermost_linux.sh to Unix format ...
[ec2-user@ip-10-0-2-65 ~]$ chmod 700 install_mattermost_linux.sh
[ec2-user@ip-10-0-2-65 ~]$ sudo ./install_mattermost_linux.sh 10.0.2.65
rm: cannot remove '/opt/mattermost': No such file or directory
--2024-12-23 18:55:12-- https://releases.mattermost.com/5.19.0/mattermost-5.19.0-linux-amd64.tar.gz
Resolving releases.mattermost.com (releases.mattermost.com)... 18.172.170.51, 18.172.170.100, 18.172.170.92, ...
Connecting to releases.mattermost.com (releases.mattermost.com)|18.172.170.51|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 155314485 (148M) [application/x-gzip]
Saving to: 'mattermost-5.19.0-linux-amd64.tar.gz'

100%[=====>] 155,314,485 33.4MB/s in 4.8s
```

<Insert screenshot b(3) here>

Step 5: Answer the following questions

Answer the following questions

Q What is the default setting for DNS hostnames when a new VPC is created?

1

- a) Enabled
- b) Disabled
- c) Can be set during VPC creation
- d) Depends on the region used

Enter your answer here

b

Q What is the term used for the machine when we use it to log into the database server?

- a) Bastion Host

- b) NAT Gateway
- c) Tunnel Interface
- d) SSH Gateway

Enter your answer here

a

Q The database server security group in this exercise has to keep port 3306 open.

3 Which protocol uses this port to communicate?

- a) HTTPS
- b) RDP
- c) TCP
- d) SCP

Enter your answer here

c

Q Which port is being used by Mattermost to communicate with the client application

4

- a) 8080
- b) 80
- c) 443
- d) 8065

Enter your answer here

d

Q Which of the following is a reason why we cannot set the CIDR block for the public
5 subnet to 10.0.2.0/16, assuming the values for the other CIDR blocks are the
same as mentioned in the instructions?

- a) CIDR block overlaps
with existing block
- b) CIDR block is not a
valid CIDR
- c) CIDR block does not fall
within the VPC
- d) There is no reason, this
is a perfectly valid CIDR

Enter your answer here

c

Q Assume that you have been asked to create 3 EC2 instances - application server,
6 the database server and NAT instance. Each of these instances have their own security groups with a set of ports to be kept open. One of those ports is entirely unnecessary for the given architecture to function. Which of the ports given in the option below could it be?

- a) Port 22 on the NAT instances
- b) Port 3306 on the database server
- c) Port 443 on the NAT instance
- d) Port 22 on the application server

Enter your answer here

a

Q Describe the steps you would take to increase security of the servers you have
7 deployed so that they are not reachable from external sources

1. The NAT instance should only accept connections from the private subnet.
2. The database server security group should be set to only receive incoming from the public subnet

Q Describe the steps required to deploy the given application in an autoscaling
8 environment

1. **Create and configure EC2 instance for given application. Details**
 - a. Select the AMI (Amazon Machine Image) for application.

- b. Specify the instance type (e.g., t2.micro, t3.medium).
 - c. Configure storage as per the application's requirements.
2. **Add User Data.**
3. Define Security Groups
4. Now Set Up an Auto Scaling Group using above launch template

Grades distribution	
MCQs	6 (1 mark each)
Subjective questions	20 marks (10+10)
Implementation screenshots	24 marks (1 marks each)
Total	50 marks