

<b>Declaration</b>	
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	Rik KISNAH
	<Enter your name above this line to indicate that you are in agreement>

<b>Instructions</b>	
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 program office will not extend the project submission deadline under any circumstances.	
The file should be renamed in the format BATCH_FIRSTNAME_LASTNAME_PROJECT1. For example: PGPCCMAY18_VIJAY_DWIVEDI_PROJECT1.pdf	

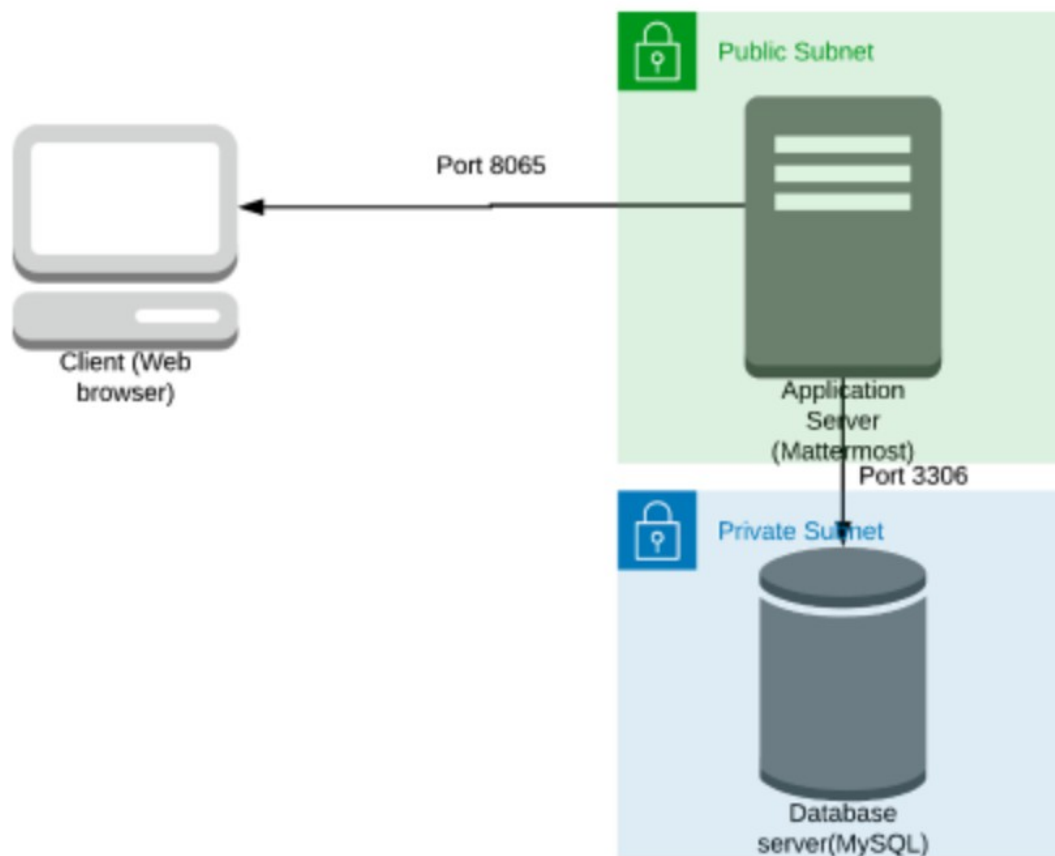
<b>Resource Clean Up</b>	
Cloud is always pay per use model and all resources/services that we consume are chargeable. Cleaning up when you've completed your lab or project is always necessary. This is true whether you're doing a lab or implementing a project at your workplace.	
After completing the 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



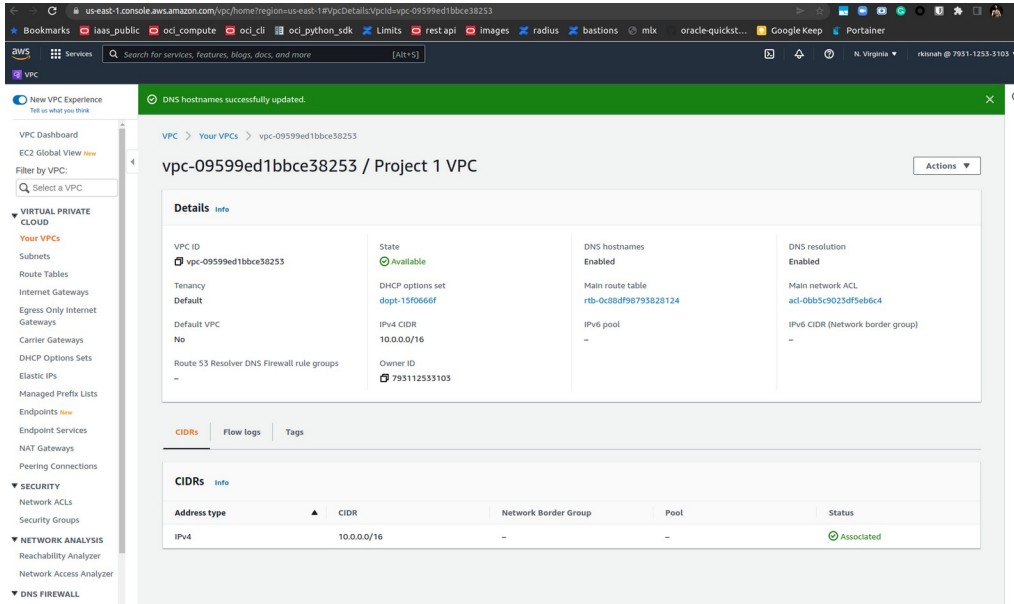
Architecture Implementation	
1	Implement 2 different subnets (one public and the other private) in a custom VPC
2	Install and configure MySQL on an Amazon Linux 2 instance on the private subnet using the instructions provided. (Hint: Use a bastion host and a NAT gateway)
3	Install and configure Mattermost 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	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	done
Expected screenshots	1) Created VPC with properties visible	

<Insert Screenshot a(1) here>



Step number	b
Step name	Creation of public subnet
Instructions	<ol style="list-style-type: none"> <li>1) Navigate to VPC-&gt;Subnets</li> <li>2) Click on "Create Subnet"</li> <li>3) Enter the following fields Name tag : Public Subnet VPC : Select the Project 1 VPC IPv4 CIDR block : 10.0.1.0/24 The other options can be ignored</li> <li>4) Click on Create</li> <li>5) Once the subnet has been created, select the subnet and click on Actions-&gt;Modify Auto-assign IP settings</li> <li>6) Enable the option "Auto assign IPv4" and select Save</li> </ol>
Expected screenshots	1) Subnet Creation screen

<Insert Screenshot b(1) here>

The screenshot shows the AWS Management Console interface for a subnet. At the top, a green notification bar states: "You have successfully changed subnet settings: Enable auto-assign public IPv4 address". The main heading is "subnet-05d1de24b5dac9007 / Public Subnet".

**Details**

Subnet ID subnet-05d1de24b5dac9007	Subnet ARN arn:aws:ec2:us-east-1:793112533103:subnet/subnet-05d1de24b5dac9007	State Available	IPv4 CIDR 10.0.1.0/24
Available IPv4 addresses 251	IPv6 CIDR -	Availability Zone us-east-1a	Availability Zone ID use1-az6
Network border group us-east-1	VPC vpc-09599ed1bbce38253   Project 1 VPC	Route table rtb-0c88df98793828124	Network ACL acl-0bb5c9023df5eb6c4
Default subnet No	Auto-assign public IPv4 address Yes	Auto-assign IPv6 address No	Auto-assign customer-owned IPv4 address No
Customer-owned IPv4 pool -	Outpost ID -	IPv4 CIDR reservations -	IPv6 CIDR reservations -
IPv6-only No	Hostname type IP name	Resource name DNS A record Disabled	Resource name DNS AAAA record Disabled
DNS64 Disabled	Owner 793112533103		

**Flow logs**

Filter flow logs

Name	Flow log ID	Filter	Destination type	Destination name	IAM role ARN
No flow logs found in this Region					

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>

The screenshot displays the AWS Management Console interface for a private subnet. The browser address bar shows the URL: `us-east-1.console.aws.amazon.com/vpc/home?region=us-east-1#SubnetDetails:subnetid=subnet-04291e0039aee9c0e`. The console header includes the AWS logo, a search bar, and navigation links for various services. The left-hand navigation pane is expanded to show the 'VIRTUAL PRIVATE CLOUD' section, with 'Subnets' selected. The main content area is titled 'subnet-04291e0039aee9c0e / Private Subnet' and contains a 'Details' tab. The details are organized into a grid of key-value pairs:

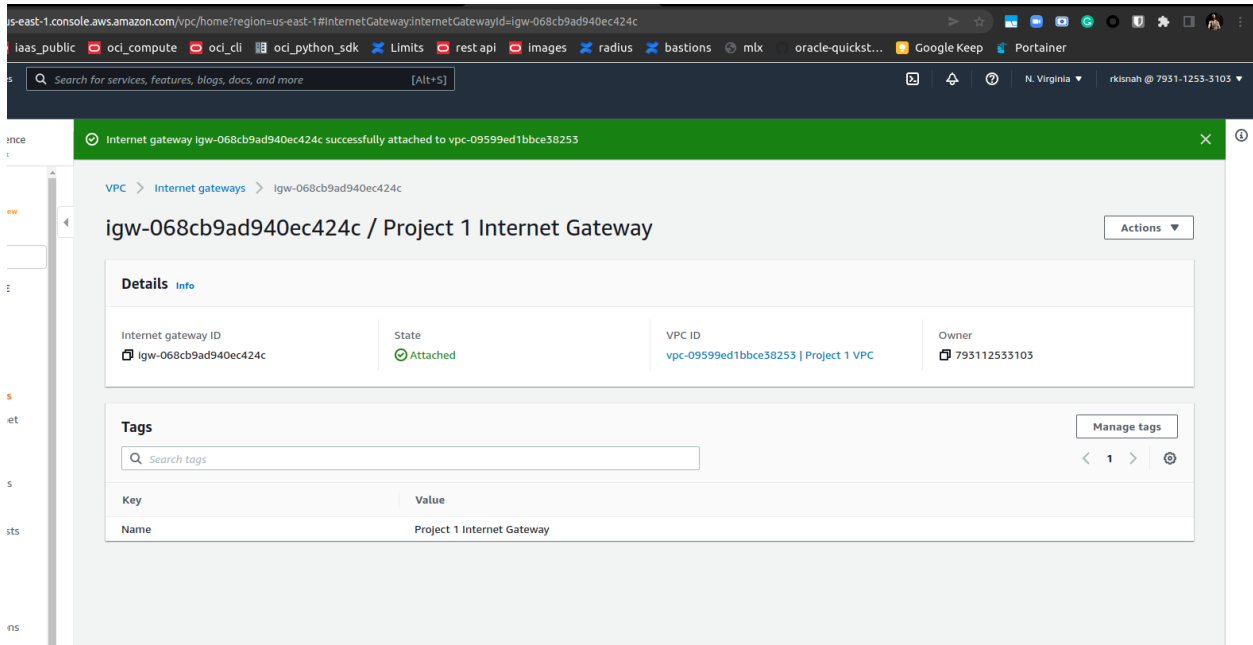
Details			
Subnet ID	Subnet ARN	State	IPv4 CIDR
subnet-04291e0039aee9c0e	arn:aws:ec2:us-east-1:793112533103:subnet/subnet-04291e0039aee9c0e	Available	10.0.2.0/24
Available IPv4 addresses	IPv6 CIDR	Availability Zone	Availability Zone ID
251	-	us-east-1a	use1-az6
Network border group	VPC	Route table	Network ACL
us-east-1	vpc-09599ed1bbce38253   Project 1 VPC	rtb-0c80df98793628124	acl-0bb5c9023df5eb6c4
Default subnet	Auto-assign public IPv4 address	Auto-assign IPv6 address	Auto-assign customer-owned IPv4 address
No	No	No	No
Customer-owned IPv4 pool	Outpost ID	IPv4 CIDR reservations	IPv6 CIDR reservations
-	-	-	-
IPv6-only	Hostname type	Resource name DNS A record	Resource name DNS AAAA record
No	IP name	Disabled	Disabled
DNS64	Owner		
Disabled	793112533103		

Below the details grid, there are tabs for 'Flow logs', 'Route table', 'Network ACL', 'CIDR reservations', 'Sharing', and 'Tags'. The 'Flow logs' tab is active, showing a search bar with the placeholder 'Filter flow logs' and a table with columns: Name, Flow log ID, Filter, Destination type, Destination name, and IAM role ARN. The table is currently empty, with a message at the bottom stating 'No flow logs found in this Region'.

## Step 2 : Internet Gateway and VPC

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

<Insert Screenshot a(1) here >



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

### <Insert Screenshot b(1) here>

The screenshot displays the AWS Management Console interface for a VPC. The main content area shows the details for the route table 'rtb-0f7a46ba45504b048 / Public Route Table'. The 'Routes' tab is active, showing a list of two routes. The first route has a destination of 10.0.0.0/16 and a target of 'local', with a status of 'Active'. The second route has a destination of 0.0.0.0/0 and a target of 'lgw-068cb9ad940ec424c', also with a status of 'Active'. The left sidebar contains the navigation menu, with 'Route Tables' highlighted under the 'VIRTUAL PRIVATE CLOUD' section.

<Insert Screenshot b(2) here>

The screenshot shows the AWS Management Console interface for a VPC route table. The breadcrumb navigation indicates the path: VPC > Route tables > rtb-0f7a46ba45504b048. The main heading is "rtb-0f7a46ba45504b048 / Public Route Table".

A notification banner at the top states: "You can now check network connectivity with Reachability Analyzer" with a "Run Reachability Analyzer" button.

The "Details" section provides the following information:

- Route table ID: rtb-0f7a46ba45504b048
- Main: No
- Explicit subnet associations: subnet-05d1de24b5dac9007 / Public Subnet
- Edge associations: -
- VPC: vpc-09599ed1bbce38253 | Project 1 VPC
- Owner ID: 793112533103

Below the details, there are tabs for "Routes", "Subnet associations", "Edge associations", "Route propagation", and "Tags". The "Subnet associations" tab is active.

The "Explicit subnet associations (1)" section shows a table with one entry:

Subnet ID	IPv4 CIDR	IPv6 CIDR
subnet-05d1de24b5dac9007 / Public Subnet	10.0.1.0/24	-

The "Subnets without explicit associations (1)" section includes a note: "The following subnets have not been explicitly associated with any route tables and are therefore associated with the main route table:". It shows a table with one entry:

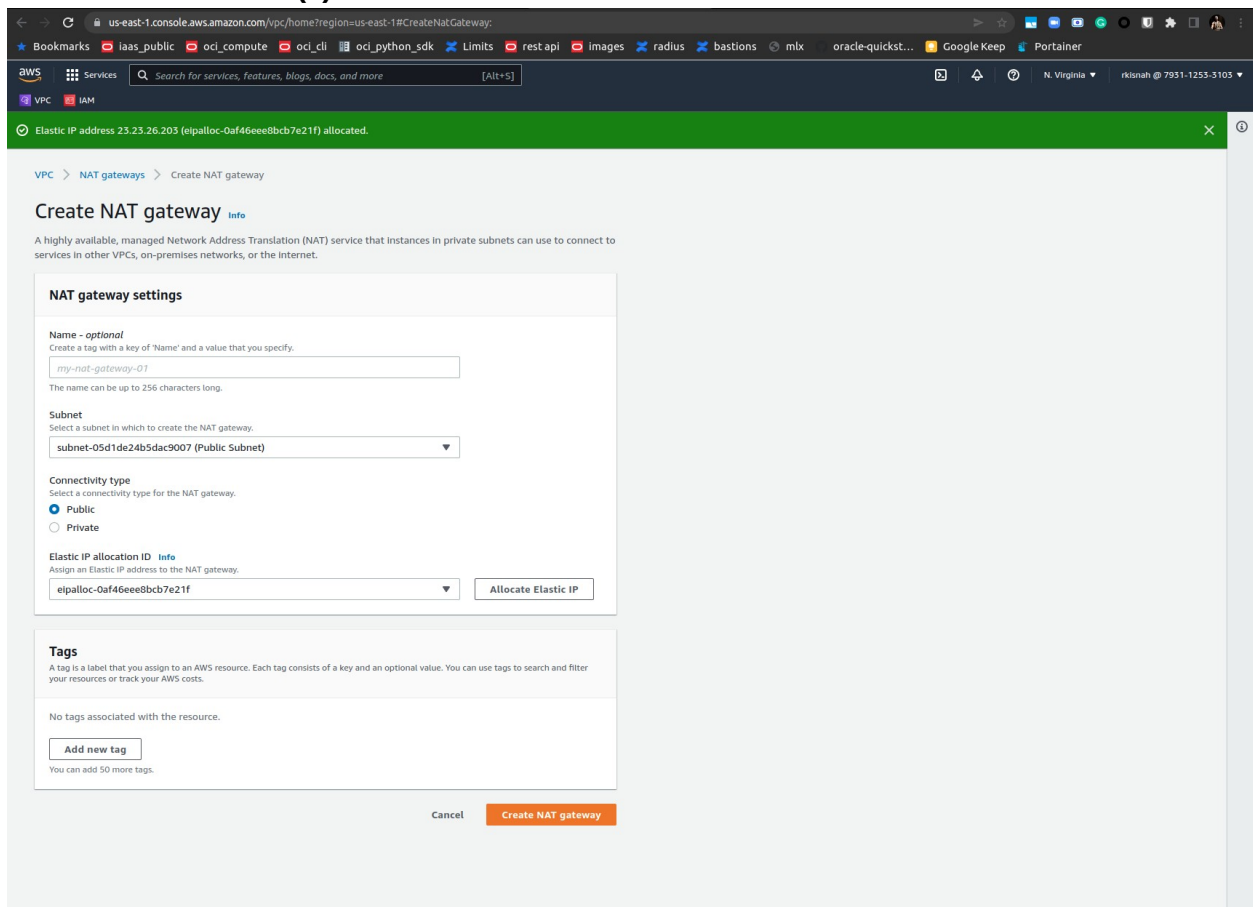
Subnet ID	IPv4 CIDR	IPv6 CIDR
subnet-04291e0039aee9c0e / Private Subnet	10.0.2.0/24	-

The left sidebar contains the navigation menu with categories like "VIRTUAL PRIVATE CLOUD", "SECURITY", "NETWORK ANALYSIS", "DNS FIREWALL", and "NETWORK FIREWALL".



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

<Insert Screenshot c(1) here>



The screenshot shows the AWS Management Console interface for creating a NAT gateway. At the top, a green notification bar states: "Elastic IP address 23.23.26.203 (elipalloc-0af46eee8bc7e21f) allocated." The breadcrumb navigation shows "VPC > NAT gateways > Create NAT gateway". The main heading is "Create NAT gateway" with an "info" link. Below this, a description states: "A highly available, managed Network Address Translation (NAT) service that instances in private subnets can use to connect to services in other VPCs, on-premises networks, or the Internet."

The "NAT gateway settings" section contains the following fields:

- Name - optional:** A text input field containing "my-nat-gateway-01". Below it, a note says: "The name can be up to 256 characters long."
- Subnet:** A dropdown menu showing "subnet-05d1de24b5dac9007 (Public Subnet)".
- Connectivity type:** Two radio buttons: "Public" (selected) and "Private".
- Elastic IP allocation ID:** A dropdown menu showing "elipalloc-0af46eee8bc7e21f". To its right is a button labeled "Allocate Elastic IP".

The "Tags" section at the bottom states: "No tags associated with the resource." and includes an "Add new tag" button. At the very bottom of the form are two buttons: "Cancel" and "Create NAT gateway".

<Insert Screenshot c(2) here>

The screenshot displays the AWS Management Console interface for a NAT gateway. The browser address bar shows the URL: `us-east-1.console.aws.amazon.com/vpc/home?region=us-east-1#NatGatewayDetails:natGatewayId=nat-0ba3792f5d1b91c11`. The console header includes the AWS logo, a search bar, and navigation links for VPC and IAM. The left sidebar lists various AWS services, with 'NAT Gateways' highlighted under the 'VPC' section. The main content area shows the details of the NAT gateway `nat-0ba3792f5d1b91c11` under the 'nat-gw-project-1' VPC. The 'Details' tab is active, displaying a table with the following information:

Property	Value
NAT gateway ID	<code>nat-0ba3792f5d1b91c11</code>
Connectivity type	Public
State	Pending
State message	-
Elastic IP address	-
Private IP address	<code>10.0.1.229</code>
Network interface ID	<code>eni-0475282ad332922cf</code>
Subnet	<code>subnet-05d1de24b5dac9007</code> / Public Subnet
Created	Sunday, April 3, 2022, 10:21:37 PDT
Deleted	-

Below the details, the 'Monitoring' tab is active, showing six graphs for the NAT gateway's performance. The graphs are arranged in a 2x3 grid. The top row contains 'Packets out to destination (Count)', 'Packets out to source (Count)', and 'Bytes out to destination (Bytes)'. The bottom row contains 'Bytes out to source (Bytes)', 'Error port allocation (Count)', and 'Active connection (Count)'. Each graph shows a time series of data points over a 1-hour period, with the x-axis labeled from 16:30 to 17:15. The y-axis for all graphs ranges from 0 to 1. The graphs show that the NAT gateway is currently in a 'Pending' state, and no data is available for the monitoring charts.

Step  
number

Step name Creation of private route tables

Instructions

- 1) Navigate to VPC -> Route Tables and click on Create Route table
- 2) Enter the name tag "Private Route Table", select the Project 1 VPC from the dropdown and click on Create
- 3) Once the route table is created, select it and select the Routes tab below the list of route tables
- 4) Click in Edit Routes and add the following route (Don't edit the existing one)
  - Destination : 0.0.0.0/0
  - Target: Select NAT Gateway and select the NAT Gateway created in the previous stepClick on Save Routes
- 5) Select the Subnet Associations tab and click on Edit Subnet Associations
- 6) Select the private Subnet from the list and click on Save

Expected  
screenshot

- 1) Route list of the route table
- 2) Subnet association of the route table

<Insert Screenshot for d(1) here >

The screenshot shows the AWS Management Console interface for a Private Route Table. The breadcrumb navigation is VPC > Route tables > rtb-047c4dd8fcc0d8973. The main heading is 'rtb-047c4dd8fcc0d8973 / Private Route Table'. A green notification bar at the top states 'Updated routes for rtb-047c4dd8fcc0d8973 / Private Route Table successfully'. Below this, a blue box suggests using the Reachability Analyzer. The 'Details' section shows the Route table ID (rtb-047c4dd8fcc0d8973), VPC (vpc-09599ed1bbce38253 | Project 1 VPC), Main flag (No), Owner ID (793112533103), and empty fields for Explicit subnet associations and Edge associations. The 'Routes' tab is selected, showing a table with 2 routes. The table has columns for Destination, Target, Status, and Propagated. The first route has Destination 10.0.0.0/16, Target local, Status Active, and Propagated No. The second route has Destination 0.0.0.0/0, Target nat-0ba3792f5d1b91c11, Status Active, and Propagated No. The left sidebar shows the navigation menu with categories like VIRTUAL PRIVATE CLOUD, SECURITY, NETWORK ANALYSIS, DNS FIREWALL, and NETWORK FIREWALL.

Destination	Target	Status	Propagated
10.0.0.0/16	local	Active	No
0.0.0.0/0	nat-0ba3792f5d1b91c11	Active	No

<Insert Screenshot for d(2) here>

us-east-1.console.aws.amazon.com/vpc/home?region=us-east-1#RouteTableDetails:RouteTableId=rtb-047c4dd8fcc0d8973

Bookmarks: iaas\_public, oci\_compute, oci\_cli, oci\_python\_sdk, Limits, rest api, Images, radius, bastions, mlx, oracle-quickst..., Google Keep, Portainer

Services: Search for services, features, blogs, docs, and more [Alt+S]

VPC IAM

New VPC Experience Tell us what you think

Filter by VPC: Select a VPC

VIRTUAL PRIVATE CLOUD

- Your VPCs
- Subnets
- Route Tables**
- Internet Gateways
- Egress Only Internet Gateways
- Carrier Gateways
- DHCP Options Sets
- Elastic IPs
- Managed Prefix Lists
- Endpoints **New**
- Endpoint Services
- NAT Gateways
- Peering Connections

SECURITY

- Network ACLs
- Security Groups

NETWORK ANALYSIS

- Reachability Analyzer
- Network Access Analyzer

DNS FIREWALL

- Rule Groups **New**
- Domain Lists **New**

NETWORK FIREWALL

- Firewalls
- Firewall Policies
- Network Firewall Rule Groups

VIRTUAL PRIVATE NETWORK (VPN)

- Customer Gateways **New**

You have successfully updated subnet associations for rtb-047c4dd8fcc0d8973 / Private Route Table.

VPC > Route tables > rtb-047c4dd8fcc0d8973

### rtb-047c4dd8fcc0d8973 / Private Route Table

Actions

You can now check network connectivity with Reachability Analyzer [Run Reachability Analyzer](#)

**Details** info

Route table ID rtb-047c4dd8fcc0d8973	Main No	Explicit subnet associations <a href="#">subnet-04291e0039aee9c0e / Private Subnet</a>	Edge associations -
VPC vpc-09599ed1bbce38253   Project 1 VPC	Owner ID 793112533103		

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

**Explicit subnet associations (1)** [Edit subnet associations](#)

Find subnet association

Subnet ID	IPv4 CIDR	IPv6 CIDR
<a href="#">subnet-04291e0039aee9c0e / Private Subnet</a>	10.0.2.0/24	-

**Subnets without explicit associations (0)** [Edit subnet associations](#)

The following subnets have not been explicitly associated with any route tables and are therefore associated with the main route table:

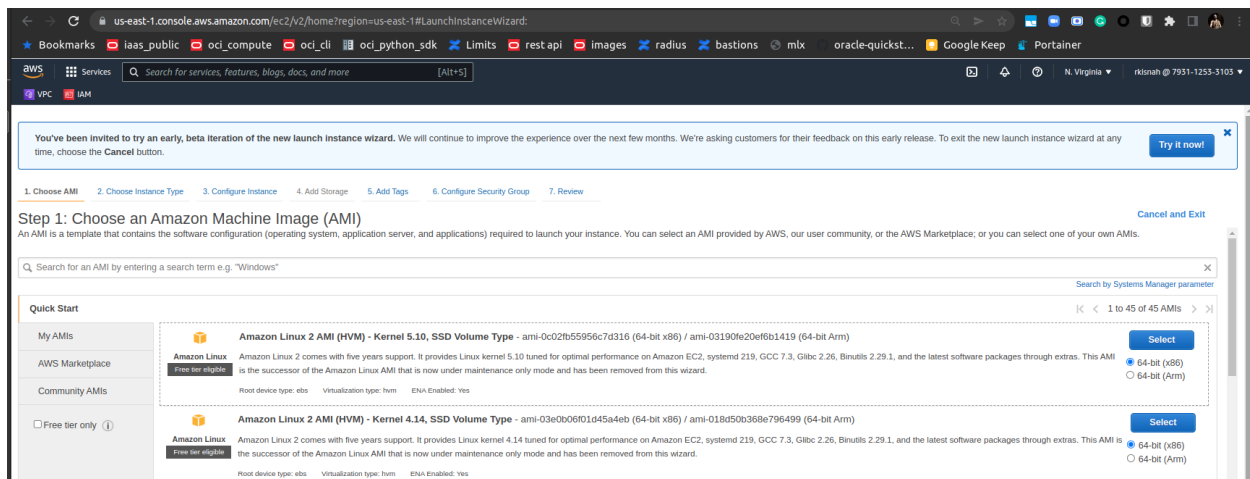
Find subnet association

No subnets without explicit associations  
All your subnets are associated with a route table.

## Step 3 : Creation of database and application servers

Step number	a
Step name	Creation of application server
Instructions	<ol style="list-style-type: none"><li>1) Navigate to EC2 using the Services button at the top of the screen</li><li>2) Select Instances at the left side of the screen</li><li>3) Click on Launch Instance<ul style="list-style-type: none"><li>- Select the AMI Amazon 2 Linux</li><li>- Select the instance type t2.micro</li><li>- Select Network as "Project 1 VPC" and subnet as "Public Subnet"</li><li>- For the security group, open the ports 80,443, 22 and 8065 for source set to "Anywhere"</li></ul></li><li>4) Launch the instance after creating a new pem file and downloading it</li></ol>
Expected screenshots	<ol style="list-style-type: none"><li>1) AMI used</li><li>2) Instance configuration screen</li><li>3) Security group rules</li><li>4) Instance after creation</li></ol>

<Insert screenshot a(1) here>



<Insert screenshot a(2) here>

us-east-1.console.aws.amazon.com/ec2/v2/home?region=us-east-1#LaunchInstanceWizard.

Bookmarks iaas\_public oci\_compute oci\_cli oci\_python\_sdk Limits rest api images radius bastions mix oracle-quickst... Google Keep Portainer

BWS Services Search for services, features, blogs, docs, and more [Alt+S]

VPC IAM

1. Choose AMI 2. Choose Instance Type 3. Configure Instance 4. Add Storage 5. Add Tags 6. Configure Security Group 7. Review

### Step 3: Configure Instance Details

**No default subnet found**  
Please choose another subnet in your default VPC, or choose another VPC.

Configure the instance to suit your requirements. You can launch multiple instances from the same AMI, request Spot instances to take advantage of the lower pricing, assign an access management role to the instance, and more.

**Number of instances** 1 [Launch into Auto Scaling Group](#)

**Purchasing option** ☐ Request Spot Instances

**Network** vpc-09699ed11b0c38253 | Project 1 VPC [Create new VPC](#)

**Subnet** subnet-0c1de24b5d4c9007 | Public Subnet | us-east-1 [Create new subnet](#)

**Auto-assign Public IP** Use subnet setting (Enable)

**Hostname type** Use subnet setting (IP name)

**DNS Hostname** ☐ Enable IP name (IPv4 (A record) DNS requests)  
☒ Enable resource-based IPv4 (A record) DNS requests  
☐ Enable resource-based IPv6 (AAAA record) DNS requests

**Placement group** ☐ Add instance to placement group

**Capacity Reservation** Open

**Domain join directory** No directory [Create new directory](#)

**IAM role** None [Create new IAM role](#)

**Shutdown behavior** Stop

**Stop - Hibernate behavior** ☐ Enable hibernation as an additional stop behavior

**Enable termination protection** ☐ Protect against accidental termination

**Monitoring** ☐ Enable CloudWatch detailed monitoring  
Additional charges apply

**Tenancy** Shared - Run a shared hardware instance  
Additional charges will apply for dedicated tenancy.

**Elastic Inference** ☐ Add an Elastic Inference accelerator  
Additional charges apply

**Credit specification** ☐ Unlimited  
Additional charges may apply

**File systems** [Add file system](#) [Create new file system](#)

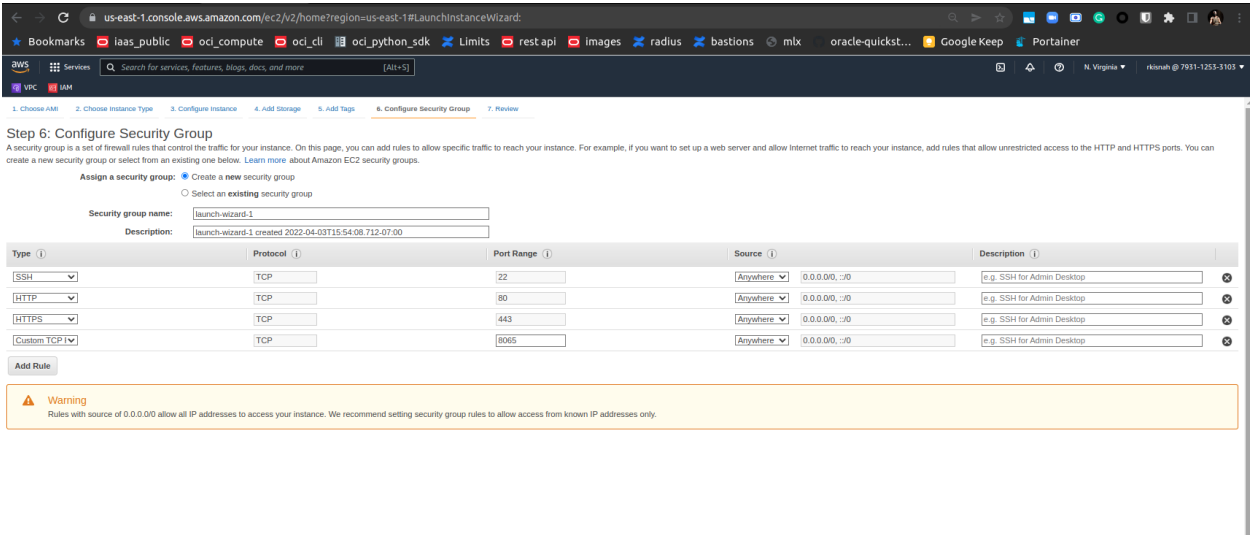
▼ **Network interfaces**

Device	Network Interface	Subnet	Primary IP	Secondary IP addresses	IPv6 IPs
eth0	New network interface	subnet-05f1de24	Auto-assign	<a href="#">Add IP</a>	The selected subnet does not support IPv6 because it does not have an IPv6 CIDR.

[Add Device](#)

▼ **Advanced Details**

<Insert screenshot a(3) here>



<Insert screenshot a(4) here>

<Insert screenshot a(4) here>



us-east-1.console.aws.amazon.com/ec2/v2/home?region=us-east-1#InstanceDetails:instanceId=i-048cd5470af59e85c

Bookmarks

laas\_public

oci\_compute

oci\_cli

oci\_python\_sdk

Limits

rest api

images

radius

bastions

mlx

oracle-quickst...

Google Keep

Portainer

DWS

Services

VPC

IAM

Search for services, features, blogs, docs, and more

(All+5)

N. Virginia

rkronah @ 7931-1253-3103

New EC2 Experience

Tell us what you think

EC2 Dashboard

EC2 Global View

Events

Tags

Limits

Instances

Instances

Instance Types

Launch Templates

Spot Requests

Savings Plans

Reserved Instances

Dedicated Hosts

Scheduled Instances

Capacity Reservations

Images

AMIs

AMI Catalog

Elastic Block Store

Volumes

Snapshots

Lifecycle Manager

Network & Security

Security Groups

Elastic IPs

Placement Groups

Key Pairs

Network Interfaces

Load Balancing

Load Balancers

Target Groups

Auto Scaling

Launch Configurations

Auto Scaling Groups

EC2 > Instances > i-048cd5470af59e85c

Instance summary for i-048cd5470af59e85c

Updated less than a minute ago

Info

Connect

Instance state

Actions

Instance ID

i-048cd5470af59e85c

Public IPv4 address

52.3.11.171 | open address

Private IPv4 addresses

10.0.1.247

IPv6 address

-

Instance state

Running

Public IPv4 DNS

ec2-52-3-11-171.compute-1.amazonaws.com | open address

Hostname type

IP name: ip-10-0-1-247.ec2.internal

Private IP DNS name (IPv4 only)

ip-10-0-1-247.ec2.internal

Answer private resource DNS name

IP v4 (A)

Instance type

t2.micro

Elastic IP addresses

-

VPC ID

vpc-09599ed1bce38253 (Project 1 VPC)

AWS Compute Optimizer finding

Opt-in to AWS Compute Optimizer for recommendations. | Learn more

IAM Role

-

Subnet ID

subnet-05d1de24b5dac9007 (Public Subnet)

Details

Security

Networking

Storage

Status checks

Monitoring

Tags

Instance details

Info

Platform

Amazon Linux (Inferred)

AMI ID

ami-0c2fb55956c7d316

Monitoring

disabled

Platform details

Linux/UNIX

AMI name

amzn2-ami-kernel-5.10-hvm-2.0.20220316.0-x86\_64-gp2

Termination protection

Disabled

Launch time

Sun Apr 03 2022 15:56:50 GMT-0700 (Pacific Daylight Time) (2 minutes)

AMI location

amazon/amzn2-ami-kernel-5.10-hvm-2.0.20220316.0-x86\_64-gp2

Instance auto-recovery

Default

Lifecycle

normal

Stop-hibernate behavior

disabled

AMI Launch index

0

Key pair name

rk\_key\_pair\_aws

State transition reason

-

Credit specification

standard

Kernel ID

-

State transition message

-

Usage operation

RunInstances

RAM disk ID

-

Owner

793112533103

ClassicLink

-

Enclaves Support

-

Boot mode

-

Allow tags in instance metadata

Disabled

Use RBN as guest OS hostname

Disabled

Answer RBN DNS hostname IPv4

Enabled

Host and placement group

Info

Host ID

-

Affinity

-

Placement group

-

Host resource group name

-

Tenancy

default

Virtualization type

hvm

Reservation

r-0cf139809bf134358

Partition number

-

Number of vCPUs

1

Step number	b
Step name	Creation of database server
Instructions	<ol style="list-style-type: none"> <li>1) Navigate to EC2 using the Services button at the top of the screen</li> <li>2) Select Instances at the left side of the screen</li> <li>3) Click on Launch Instance <ul style="list-style-type: none"> <li>- Select the AMI Amazon 2 Linux</li> <li>- Select the instance type t2.micro</li> <li>- Select Network as "Project 1 VPC" and subnet as "Private Subnet"</li> <li>- For the security group, open the ports 80, 443,22 and 3306 for source set to "Anywhere"</li> </ul> </li> <li>4) Launch the instance by selecting the same pem file created in the previous step</li> </ol>
Expected screenshots	<ol style="list-style-type: none"> <li>1) AMI used</li> <li>2) Instance configuration screen</li> <li>3) Security group rules</li> <li>4) Instance after creation</li> </ol>

**<Insert screenshot b(1) here>**

**<Insert screenshot b(2) here>**

**<Insert screenshot b(3) here>**

**<Insert screenshot b(4) here>**

## 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 <i>scp -i &lt;application server pem file&gt; &lt;database server pem file&gt; ec2-user@&lt;application server public IP&gt;:/home/ec2-user</i></p> <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 <i>ssh -i &lt;database server pem file&gt; ec2-user@&lt;private IP of database server&gt;</i></p> <p>4) Enter the following commands to install and configure MySQL on the database server <i>sudo yum update</i> <i>wget http://dev.mysql.com/get/mysql57-community-release-el7-9.noarch.rpm</i> <i>sudo yum localinstall mysql57-community-release-el7-9.noarch.rpm -y</i> <i>sudo yum install mysql-community-server -y --nogpgcheck</i> <i>sudo systemctl start mysqld.service</i></p> <p>Run the below command to retrieve a temporary password for MySQL <i>sudo grep 'temporary password' /var/log/mysqld.log   rev   cut -d" " -f1   rev   tr -d "."</i></p> <p>Log in to MySQL with the below command and enter the above password when prompted <i>mysql -u root -p</i></p> <p>Enter the below command after you login to MySQL <i>ALTER USER 'root'@'localhost' IDENTIFIED BY 'Password42!';</i></p> <p>Type 'exit' into the MySQL prompt and press Enter to exit out of the MySQL environment. Enter the below commands to complete the setup. Ignore any warning messages you receive. <i>wget https://d60pu47qoi4ee.cloudfront.net/install_mysql_linux.sh</i> <i>chmod 777 install_mysql_linux.sh</i> <i>sudo ./install_mysql_linux.sh</i></p> <p>5) Type <i>exit</i> to exit the database server and go back to the application server</p>
Expected screenshots	<p>1) Installation of MySQL</p> <p>2) Retrieving the temporary password</p> <p>3) Executing the provided script</p>

<Insert screenshot a(1) here> - install mysql

the instructions are wrong outdated. Attached is the terminal text output.

```
ec2-user@ip-10-0-2-215 ~]$ mkdir downloads
```

```
[ec2-user@ip-10-0-2-215 ~]$ cd downloads/
```

```
[ec2-user@ip-10-0-2-215 downloads]$ 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-10-0-2-215 downloads]$ #wtf right??? instructions are dud
```

```
# Using dr google good ref: https://tecadmin.net/install-mysql-5-7-centos-rhel/
```

```
[ec2-user@ip-10-0-2-215 downloads]$ sudo yum localinstall
```

```
https://dev.mysql.com/get/mysql57-community-release-el7-9.noarch.rpm
```

```
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
```

```
mysql57-community-release-el7-9.noarch.rpm | 9.0 kB  
00:00:00
```

```
Examining /var/tmp/yum-root-jwDpAh/mysql57-community-release-el7-9.noarch.rpm:  
mysql57-community-release-el7-9.noarch
```

```
Marking /var/tmp/yum-root-jwDpAh/mysql57-community-release-el7-9.noarch.rpm to  
be installed
```

```
Resolving Dependencies
```

```
--> Running transaction check
```

```
---> Package mysql57-community-release.noarch 0:el7-9 will be installed
```

```
--> Finished Dependency Resolution
```

```
Dependencies Resolved
```

```
=====
```

Package	Arch	Version	Repository
mysql57-community-release	noarch	el7-9	/mysql57-community-release-el7-9.noarch

```
=====
```

```
Installing:
```

```
mysql57-community-release 8.6 k
```

```
Transaction Summary
```

```
Install 1 Package
```

```
Total size: 8.6 k
```

```
Installed size: 8.6 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-el7-9.noarch  
1/1  
Verifying : mysql57-community-release-el7-9.noarch  
1/1

Installed:  
mysql57-community-release.noarch 0:el7-9

Complete!

```
[ec2-user@ip-10-0-2-215 downloads]$ yum install mysql-community-server
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
You need to be root to perform this command.
[ec2-user@ip-10-0-2-215 downloads]$ sudo yum install mysql-community-server
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
amzn2-core | 3.7 kB
00:00:00
49 packages excluded due to repository priority protections
Resolving Dependencies
--> Running transaction check
---> Package mysql-community-server.x86_64 0:5.7.37-1.el7 will be installed
--> Processing Dependency: mysql-community-common(x86-64) = 5.7.37-1.el7 for
package: mysql-community-server-5.7.37-1.el7.x86_64
--> Processing Dependency: mysql-community-client(x86-64) >= 5.7.9 for
package: mysql-community-server-5.7.37-1.el7.x86_64
--> Running transaction check
---> Package mysql-community-client.x86_64 0:5.7.37-1.el7 will be installed
--> Processing Dependency: mysql-community-libs(x86-64) >= 5.7.9 for package:
mysql-community-client-5.7.37-1.el7.x86_64
--> Processing Dependency: libncurses.so.5()(64bit) for package: mysql-
community-client-5.7.37-1.el7.x86_64
--> Processing Dependency: libtinfo.so.5()(64bit) for package: mysql-
community-client-5.7.37-1.el7.x86_64
---> Package mysql-community-common.x86_64 0:5.7.37-1.el7 will be installed
--> Running transaction check
---> Package mariadb-libs.x86_64 1:5.5.68-1.amzn2 will be obsoleted
--> Processing Dependency: libmysqlclient.so.18()(64bit) for package:
2:postfix-2.10.1-6.amzn2.0.3.x86_64
--> Processing Dependency: libmysqlclient.so.18(libmysqlclient_18)(64bit) for
package: 2:postfix-2.10.1-6.amzn2.0.3.x86_64
---> Package mysql-community-libs.x86_64 0:5.7.37-1.el7 will be obsoleting
---> Package ncurses-compat-libs.x86_64 0:6.0-8.20170212.amzn2.1.3 will be
installed
--> Running transaction check
---> Package mysql-community-libs-compat.x86_64 0:5.7.37-1.el7 will be
obsoleting
```

--> Finished Dependency Resolution

## Dependencies Resolved

=====			
=====			
Package Size	Arch	Version	Repository
=====			
Installing:			
mysql-community-libs community 2.4 M	x86_64	5.7.37-1.el7	mysql57-
replacing mariadb-libs.x86_64 1:5.5.68-1.amzn2			
mysql-community-libs-compat community 1.2 M	x86_64	5.7.37-1.el7	mysql57-
replacing mariadb-libs.x86_64 1:5.5.68-1.amzn2			
mysql-community-server community 174 M	x86_64	5.7.37-1.el7	mysql57-
Installing for dependencies:			
mysql-community-client community 25 M	x86_64	5.7.37-1.el7	mysql57-
mysql-community-common community 311 k	x86_64	5.7.37-1.el7	mysql57-
ncurses-compat-libs 308 k	x86_64	6.0-8.20170212.amzn2.1.3	amzn2-core

## Transaction Summary

=====

Install 3 Packages (+3 Dependent packages)

Total download size: 203 M

Is this ok [y/d/N]: y

Downloading packages:

warning: /var/cache/yum/x86\_64/2/mysql57-community/packages/mysql-community-common-5.7.37-1.el7.x86\_64.rpm: Header V4 RSA/SHA256 Signature, key ID 3a79bd29: NOKEY

Public key for mysql-community-common-5.7.37-1.el7.x86\_64.rpm is not installed

(1/6): mysql-community-common-5.7.37-1.el7.x86\_64.rpm | 311 kB  
00:00:00

(2/6): mysql-community-libs-5.7.37-1.el7.x86\_64.rpm | 2.4 MB  
00:00:00

(3/6): mysql-community-libs-compat-5.7.37-1.el7.x86\_64.rpm | 1.2 MB  
00:00:00

(4/6): mysql-community-client-5.7.37-1.el7.x86\_64.rpm | 25 MB  
00:00:00

(5/6): ncurses-compat-libs-6.0-8.20170212.amzn2.1.3.x86\_64.rpm | 308 kB  
00:00:01

(6/6): mysql-community-server-5.7.37-1.el7.x86\_64.rpm | 174 MB  
00:00:02

-----  
-----  
Total | 70 MB/s | 203 MB  
00:00:02

Retrieving key from file:///etc/pki/rpm-gpg/RPM-GPG-KEY-mysql

Importing GPG key 0x5072E1F5:

Userid : "MySQL Release Engineering <mysql-build@oss.oracle.com>"

Fingerprint: a4a9 4068 76fc bd3c 4567 70c8 8c71 8d3b 5072 e1f5

Package : mysql57-community-release-el7-9.noarch (installed)

From : /etc/pki/rpm-gpg/RPM-GPG-KEY-mysql

Is this ok [y/N]: y

Public key for mysql-community-libs-compat-5.7.37-1.el7.x86\_64.rpm is not installed

Failing package is: mysql-community-libs-compat-5.7.37-1.el7.x86\_64

GPG Keys are configured as: file:///etc/pki/rpm-gpg/RPM-GPG-KEY-mysql

[ec2-user@ip-10-0-2-215 downloads]\$ sudo yum install mysql-community-server -y  
--nogpgcheck

-bash: sudo yum install mysql-community-server: command not found

[ec2-user@ip-10-0-2-215 downloads]\$ ^Cdo yum install mysql-community-server -y  
--nogpgcheck

[ec2-user@ip-10-0-2-215 downloads]\$ mysql^C

[ec2-user@ip-10-0-2-215 downloads]\$ yum install mysql-community-server

Loaded plugins: extras\_suggestions, langpacks, priorities, update-motd

You need to be root to perform this command.

[ec2-user@ip-10-0-2-215 downloads]\$ sudo yum install mysql-community-server -y  
--nogpgcheck

Loaded plugins: extras\_suggestions, langpacks, priorities, update-motd

49 packages excluded due to repository priority protections

Resolving Dependencies

--> Running transaction check

---> Package mysql-community-server.x86\_64 0:5.7.37-1.el7 will be installed

--> Processing Dependency: mysql-community-common(x86-64) = 5.7.37-1.el7 for  
package: mysql-community-server-5.7.37-1.el7.x86\_64

--> Processing Dependency: mysql-community-client(x86-64) >= 5.7.9 for  
package: mysql-community-server-5.7.37-1.el7.x86\_64

--> Running transaction check

---> Package mysql-community-client.x86\_64 0:5.7.37-1.el7 will be installed

--> Processing Dependency: mysql-community-libs(x86-64) >= 5.7.9 for package:  
mysql-community-client-5.7.37-1.el7.x86\_64

--> Processing Dependency: libncurses.so.5()(64bit) for package: mysql-  
community-client-5.7.37-1.el7.x86\_64

```
--> Processing Dependency: libtinfo.so.5()(64bit) for package: mysql-
community-client-5.7.37-1.el7.x86_64
---> Package mysql-community-common.x86_64 0:5.7.37-1.el7 will be installed
--> Running transaction check
---> Package mariadb-libs.x86_64 1:5.5.68-1.amzn2 will be obsoleted
--> Processing Dependency: libmysqlclient.so.18()(64bit) for package:
2:postfix-2.10.1-6.amzn2.0.3.x86_64
--> Processing Dependency: libmysqlclient.so.18(libmysqlclient_18)(64bit) for
package: 2:postfix-2.10.1-6.amzn2.0.3.x86_64
---> Package mysql-community-libs.x86_64 0:5.7.37-1.el7 will be obsoleting
---> Package ncurses-compat-libs.x86_64 0:6.0-8.20170212.amzn2.1.3 will be
installed
--> Running transaction check
---> Package mysql-community-libs-compat.x86_64 0:5.7.37-1.el7 will be
obsoleting
--> Finished Dependency Resolution
```

#### Dependencies Resolved

Package Size	Arch	Version	Repository
Installing:			
mysql-community-libs community 2.4 M	x86_64	5.7.37-1.el7	mysql57-
replacing mariadb-libs.x86_64 1:5.5.68-1.amzn2			
mysql-community-libs-compat community 1.2 M	x86_64	5.7.37-1.el7	mysql57-
replacing mariadb-libs.x86_64 1:5.5.68-1.amzn2			
mysql-community-server community 174 M	x86_64	5.7.37-1.el7	mysql57-
Installing for dependencies:			
mysql-community-client community 25 M	x86_64	5.7.37-1.el7	mysql57-
mysql-community-common community 311 k	x86_64	5.7.37-1.el7	mysql57-
ncurses-compat-libs 308 k	x86_64	6.0-8.20170212.amzn2.1.3	amzn2-core

#### Transaction Summary

```
Install 3 Packages (+3 Dependent packages)
```

Total size: 203 M

Downloading packages:



Running transaction check  
Running transaction test  
Transaction test succeeded  
Running transaction

Installing : mysql-community-common-5.7.37-1.el7.x86\_64  
1/7  
Installing : mysql-community-libs-5.7.37-1.el7.x86\_64  
2/7  
Installing : ncurses-compat-libs-6.0-8.20170212.amzn2.1.3.x86\_64  
3/7  
Installing : mysql-community-client-5.7.37-1.el7.x86\_64  
4/7  
Installing : mysql-community-server-5.7.37-1.el7.x86\_64  
5/7  
Installing : mysql-community-libs-compat-5.7.37-1.el7.x86\_64  
6/7  
Erasing : 1:mariadb-libs-5.5.68-1.amzn2.x86\_64  
7/7  
Verifying : ncurses-compat-libs-6.0-8.20170212.amzn2.1.3.x86\_64  
1/7  
Verifying : mysql-community-libs-compat-5.7.37-1.el7.x86\_64  
2/7  
Verifying : mysql-community-libs-5.7.37-1.el7.x86\_64  
3/7  
Verifying : mysql-community-common-5.7.37-1.el7.x86\_64  
4/7  
Verifying : mysql-community-server-5.7.37-1.el7.x86\_64  
5/7  
Verifying : mysql-community-client-5.7.37-1.el7.x86\_64  
6/7  
Verifying : 1:mariadb-libs-5.5.68-1.amzn2.x86\_64  
7/7

Installed:

mysql-community-libs.x86\_64 0:5.7.37-1.el7  
mysql-community-libs-compat.x86\_64 0:5.7.37-1.el7  
mysql-community-server.x86\_64 0:5.7.37-1.el7

Dependency Installed:

mysql-community-client.x86\_64 0:5.7.37-1.el7  
mysql-community-common.x86\_64 0:5.7.37-1.el7  
ncurses-compat-libs.x86\_64 0:6.0-8.20170212.amzn2.1.3

Replaced:

mariadb-libs.x86\_64 1:5.5.68-1.amzn2

Complete!

```
[root@ip-10-0-2-215 log]# systemctl start mysqld
```

```
[root@ip-10-0-2-215 downloads]# sudo systemctl status mysqld
```

• **mysqld.service - MySQL Server**

Loaded: loaded (/usr/lib/systemd/system/mysqld.service; enabled; vendor preset: disabled)

Active: active (running) since Sun 2022-04-03 23:19:17 UTC; 9min ago

Docs: man:mysqld(8)

<http://dev.mysql.com/doc/refman/en/using-systemd.html>

Process: 3713 ExecStart=/usr/sbin/mysqld --daemonize  
--pid-file=/var/run/mysqld/mysqld.pid \$MYSQLD\_OPTS (code=exited,  
status=0/SUCCESS)

Process: 3664 ExecStartPre=/usr/bin/mysqld\_pre\_systemd (code=exited,  
status=0/SUCCESS)

Main PID: 3717 (mysqld)

CGroup: /system.slice/mysqld.service

└─3717 /usr/sbin/mysqld --daemonize  
--pid-file=/var/run/mysqld/mysqld.pid

Apr 03 23:19:11 ip-10-0-2-215.ec2.internal systemd[1]: Starting MySQL  
Server...

Apr 03 23:19:17 ip-10-0-2-215.ec2.internal systemd[1]: Started MySQL Server.

```
[root@ip-10-0-2-215 downloads]#
```

## <Insert screenshot a(2) here> - retrieve temp pwd

```
root@ip-10-0-2-215 log]# vim mysqld.log
```

```
[root@ip-10-0-2-215 log]# grep 'A temporary password' /var/log/mysqld.log |tail -1
```

```
2022-04-03T23:19:14.591543Z 1 [Note] A temporary password is generated for root@localhost: lH%*,#jwy70_
```

```
[root@ip-10-0-2-215 log]# ^C
```

```
[root@ip-10-0-2-215 log]# mysql -u root -p
```

Enter password:

Welcome to the MySQL monitor. Commands end with ; or \g.

Your MySQL connection id is 2

Server version: 5.7.37

Copyright (c) 2000, 2022, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its  
affiliates. Other names may be trademarks of their respective  
owners.

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

```
mysql> ALTER USER 'root'@'localhost' IDENTIFIED BY 'Password42!';
```

```
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near  
'ALTER USER 'root'@'localhost' IDENTIFIED BY 'Password42!'" at line 1
```

```
mysql> ^DBye
```

```
[root@ip-10-0-2-215 log]# /usr/bin/mysql_secure_installation
```

Securing the MySQL server deployment.

Enter password for user root:

The existing password for the user account root has expired. Please set a new password.

New password:

Re-enter new password:

The 'validate\_password' plugin is installed on the server.

The subsequent steps will run with the existing configuration  
of the plugin.

Using existing password for root.

Estimated strength of the password: 100

Change the password for root ? ((Press y|Y for Yes, any other key for No) : y

New password:

Re-enter new password:

Estimated strength of the password: 100

Do you wish to continue with the password provided?(Press y|Y for Yes, any other key for No) : Y

By default, a MySQL installation has an anonymous user, allowing anyone to log into MySQL without having to have a user account created for them. This is intended only for testing, and to make the installation go a bit smoother. You should remove them before moving into a production environment.

Remove anonymous users? (Press y|Y for Yes, any other key for No) :

... skipping.

Normally, root should only be allowed to connect from 'localhost'. This ensures that someone cannot guess at the root password from the network.

Disallow root login remotely? (Press y|Y for Yes, any other key for No) :

... skipping.

By default, MySQL comes with a database named 'test' that anyone can access. This is also intended only for testing, and should be removed before moving into a production environment.

Remove test database and access to it? (Press y|Y for Yes, any other key for No) :

... skipping.

Reloading the privilege tables will ensure that all changes made so far will take effect immediately.

Reload privilege tables now? (Press y|Y for Yes, any other key for No) :

... skipping.

All done!

[root@ip-10-0-2-215 log]# mysql -h localhost -u root -p

Enter password:

Welcome to the MySQL monitor. Commands end with ; or \g.

Your MySQL connection id is 6

Server version: 5.7.37 MySQL Community Server (GPL)

Copyright (c) 2000, 2022, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.

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

mysql> exit

Bye

[root@ip-10-0-2-215 log]# cd

## <Insert screenshot a(3) here> - execute provided script

```
[ec2-user@ip-10-0-2-215 downloads]$ wget https://d60pu47qoi4ee.cloudfront.net/install_mysql_linux.sh
--2022-04-03 23:23:09-- http://wget/
Resolving wget (wget)... failed: Name or service not known.
wget: unable to resolve host address 'wget'
--2022-04-03 23:23:09-- https://d60pu47qoi4ee.cloudfront.net/install_mysql_linux.sh
Resolving d60pu47qoi4ee.cloudfront.net (d60pu47qoi4ee.cloudfront.net)... 13.32.204.92, 13.32.204.110, 13.32.204.150, ...
Connecting to d60pu47qoi4ee.cloudfront.net (d60pu47qoi4ee.cloudfront.net)[13.32.204.92]:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 189 [text/x-sh]
Saving to: 'install_mysql_linux.sh'

100%[=====>] 189      --.-K/s  in 0s

2022-04-03 23:23:09 (27.1 MB/s) - 'install_mysql_linux.sh' saved [189/189]

FINISHED --2022-04-03 23:23:09--
Total wall clock time: 0.06s
Downloaded: 1 files, 189 in 0s (27.1 MB/s)
[ec2-user@ip-10-0-2-215 downloads]$ ls
install_mysql_linux.sh
[ec2-user@ip-10-0-2-215 downloads]$ chmod +x install_mysql_linux.sh
[ec2-user@ip-10-0-2-215 downloads]$ chmod 777 install_mysql_linux.sh
[ec2-user@ip-10-0-2-215 downloads]$ sudo ./install_mysql_linux.sh
mysql: [Warning] Using a password on the command line interface can be insecure.
[ec2-user@ip-10-0-2-215 downloads]$ cat install_mysql_linux.sh
mysql -u root -pPassword42! <<-EOF
CREATE USER 'mmuser'@'%' IDENTIFIED BY 'Mostest42!';
CREATE DATABASE mattermost_test;
GRANT ALL PRIVILEGES ON mattermost_test.* TO 'mmuser'@'%';
EOF
[ec2-user@ip-10-0-2-215 downloads]$ ^C
[ec2-user@ip-10-0-2-215 downloads]$ ^C
[ec2-user@ip-10-0-2-215 downloads]$ ls
```

Step number	b
Step name	Installation and configuration of Mattermost
Instructions	<p>1) Enter the following commands after logging into the application server via SSH to install and configure Mattermost</p> <pre>wget https://d6opu47qoi4ee.cloudfront.net/install_mattermost_linux.sh</pre> <pre>sudo yum install dos2unix -y</pre> <pre>sudo dos2unix install_mattermost_linux.sh</pre> <pre>chmod 700 install_mattermost_linux.sh</pre> <pre>sudo ./install_mattermost_linux.sh &lt;private IP of MySQL server&gt;</pre> <p>Example : <code>sudo ./install_mattermost_linux 173.65.34.7</code></p> <pre>sudo chown -R mattermost:mattermost /opt/mattermost</pre> <pre>sudo chmod -R g+w /opt/mattermost</pre> <pre>cd /opt/mattermost</pre> <pre>sudo -u mattermost ./bin/mattermost</pre> <p>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.</p> <p>&lt;public IP of the application server&gt;:8065</p>
Expected screenshots	<p>1) Executing the script</p> <p>2) Starting the Mattermost server</p> <p>3) Accessing the application via web browser</p>

<Insert screenshot b(1) here> execute script

```
[ec2-user@ip-10-0-1-247 .ssh]$ mkdir download
[ec2-user@ip-10-0-1-247 .ssh]$ cd download/
[ec2-user@ip-10-0-1-247 download]$ wget https://d6opu47qoi4ee.cloudfront.net/install_mattermost_linux.sh
--2022-04-03 23:52:39-- https://d6opu47qoi4ee.cloudfront.net/install_mattermost_linux.sh
Resolving d6opu47qoi4ee.cloudfront.net (d6opu47qoi4ee.cloudfront.net)... 99.84.218.168, 99.84.218.37, 99.84.218.70, ...
Connecting to d6opu47qoi4ee.cloudfront.net (d6opu47qoi4ee.cloudfront.net)[99.84.218.168]:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 592 [text/x-sh]
Saving to: 'install_mattermost_linux.sh'

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

2022-04-03 23:52:39 (77.9 MB/s) - 'install_mattermost_linux.sh' saved [592/592]

[ec2-user@ip-10-0-1-247 download]$ sudo yum -y install dos2unix -y
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
amzn2-core                                     | 3.7 kB  00:00:00
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  00:00:00
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-10-0-1-247 download]$ sudo dos2unix install_mattermost_linux.sh
dos2unix: converting file install_mattermost_linux.sh to Unix format ...
[ec2-user@ip-10-0-1-247 download]$ chmod 700 install_mattermost_linux.sh
[ec2-user@ip-10-0-1-247 download]$ sudo ./install_mattermost_linux.sh 10.0.2.215
rm: cannot remove '/opt/mattermost': No such file or directory
--2022-04-03 23:54:06-- https://releases.mattermost.com/5.19.0/mattermost-5.19.0-linux-amd64.tar.gz
Resolving releases.mattermost.com (releases.mattermost.com)... 13.32.207.77, 13.32.207.97, 13.32.207.65, ...
Connecting to releases.mattermost.com (releases.mattermost.com)[13.32.207.77]: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 34.9MB/s  in 4.2s

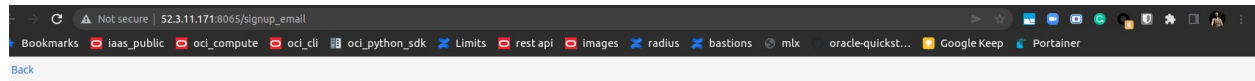
2022-04-03 23:54:10 (35.5 MB/s) - 'mattermost-5.19.0-linux-amd64.tar.gz' saved [155314485/155314485]

Downloaded Mattermost
mattermost/
mattermost/client/
mattermost/client/18.11f0f217b22217f7cd67.js
mattermost/client/icon_16x16.png
```

## <Insert screenshot b(2) here> - start mattermost server

```
ec2-user@ip-10-0-1-247 download]$ sudo chown -R mattermost:mattermost /opt/mattermost
[ec2-user@ip-10-0-1-247 download]$ sudo chmod -R g+w /opt/mattermost
[ec2-user@ip-10-0-1-247 download]$ cd /opt/mattermost
[ec2-user@ip-10-0-1-247 mattermost]$ sudo -u mattermost ./bin/mattermost
{"level":"info","ts":1649030086.0054836,"caller":"utils/i18n.go:83","msg":"Loaded system translations","for locale":"en","from locale":"/opt/mattermost/i18n/en.json"}
{"level":"info","ts":1649030086.0057468,"caller":"app/server_app_adapters.go:58","msg":"Server is initializing..."}
{"level":"info","ts":1649030086.0143862,"caller":"sqlstore/supplier.go:212","msg":"Pinging SQL","database":"master"}
{"level":"info","ts":1649030086.5916622,"caller":"sqlstore/upgrade.go:110","msg":"The database schema version has been set","version":"5.19.0"}
{"level":"error","ts":1649030088.25941,"caller":"app/server_app_adapters.go:125","msg":"SiteURL must be set. Some features will operate incorrectly if the SiteURL is not set. See documentation for details: http://about.mattermost.com/default-site-url"}
{"level":"info","ts":1649030088.262597,"caller":"app/license.go:39","msg":"License key from https://mattermost.com required to unlock enterprise features."}
{"level":"info","ts":1649030088.2635503,"caller":"app/migrations.go:26","msg":"Migrating roles to database."}
{"level":"info","ts":1649030088.3410423,"caller":"sqlstore/post_store.go:1351","msg":"Post.Message has size restrictions","max_characters":16383,"max_bytes":65535}
{"level":"info","ts":1649030088.345663,"caller":"app/migrations.go:102","msg":"Migrating emojis config t
```

<Insert screenshot b(3) here> - access app via web browser



## Mattermost

All team communication in one place,  
searchable and accessible anywhere

Let's create your account

Already have an account? [Click here to sign in.](#)

What's your email address?

Valid email required for sign-up

Choose your username

You can use lowercase letters, numbers, periods, dashes, and  
underscores.

Choose your password

Create Account

By proceeding to create your account and use Mattermost,  
you agree to our [Terms of Service](#) and [Privacy Policy](#). If you  
do not agree, you cannot use Mattermost.



## Step 5: Answer the following questions

### Answer the following questions

Q1 What is the default setting for DNS hostnames when a new VPC is created?

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

Enter your answer here

b)

Q2 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)

Q3 The database server security group in this exercise has to keep port 3306 open. Which protocol uses this port to communicate?

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

Enter your answer here

c

Q4 Which port is being used by Mattermost to communicate with the client application

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

Enter your answer here

b

Q5 Which of the following is a reason why we cannot set the CIDR block for the public 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

Q6 Assume that you have been asked to create 3 EC2 instances - application server, 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

Q7 Describe the steps you would take to increase security of the servers you have deployed so that they are not reachable from external sources

Option 1 Add a WAF in front the application server and use ACLS or country zone filtering

Option 2 Uses AWS Cognito to authenticate connection pools of allowed users

Q8 Describe the steps required to deploy the given application in an autoscaling environment

---

AS – can only apply the application server ; for the db it is more involved as we need to sync the data. (as the question was open-ended – i went with application server)

Use this blueprint

<https://docs.aws.amazon.com/autoscaling/ec2/userguide/GettingStartedTutorial.html>

Step 1. Make a snapshot of the current app server

Step 2. Create a Launch template from Step 1 snapshot

Step 3 Create a single instance ASG (Auto scaling group)

Step 4. Scale size of ASG via CPU or memory – go with horizontal scaling i.e add more hosts of the same type

**Max marks**

**15**

Grades distribution	
MCQs	6 (1 mark each)
Subjective questions	10 marks (5+5)
Implementation screenshots	24 marks (1 marks each)
Total	40 marks