 

# Placement Empowerment Program

***Cloud Computing and DevOps Centre***

***Set a private network in cloud – Create a VPC with subnets for your instances. Configure routing for internal communication between subnets***

Name :Subiksha R Department: AML



**Introduction**

## Networking in the cloud is an essential component for securely managing and deploying applications. In Microsoft Azure, a Virtual Network (VNet) is used to create a secure, isolated network environment where resources like Virtual Machines (VMs) can communicate. Within a VNet, subnets allow segmentation of resources, enabling better security, performance, and control over traffic flow.

## In this guide, we will set up a Virtual Network (VNet) with multiple subnets, configure a route table, deploy Virtual Machines (VMs) in different subnets, and test their internal communication using Remote Desktop Protocol (RDP) or SSH.

## **Overview**

## This hands-on exercise involves:

## 1. Creating a Virtual Network (VNet) with a defined IP address space.

## 2. Defining multiple subnets within the VNet for logical segmentation.

## 3. Configuring a route table to control how traffic moves between subnets.

## 4. Deploying Virtual Machines (VMs) in different subnets.

## 5. Testing internal communication between VMs using RDP (for Windows) or SSH (for Linux).

## This setup is useful for simulating a real-world cloud environment, where applications are divided into front-end, back-end, and database layers, each in its own subnet for security and management purposes.

## **Objective**

## The main goal is to:

## Understand Azure Virtual Networks (VNet) and Subnets.

## Learn how to segregate workloads using multiple subnets.

## Configure a route table to manage network traffic.

## Deploy and connect Virtual Machines across different subnets.

## Validate internal communication using RDP (Windows) or SSH (Linux).

## By the end of this guide, you will have a functional, multi-subnet network with VMs communicating securely.

## **Step-by-Step Procedure**

## **Step 1:** Create a Virtual Network (VNet)

## Log in to Azure Portal at portal.azure.com.

## 

## **Step 2 :**In the Search bar, type Virtual Networks and select it.

## **Step 3 :**Click + Create to start creating a new VNet.

## 

## **Step 4 :** Enter basic details

## Subscription: Select your active subscription.

## Resource Group: Create a new one or use an existing one.

## Virtual Network Name: Example – MyVNet.

## Region: Choose a region close to your users (e.g., East US).

## Define Address Space:

## Example: 10.0.0.0/16 (This allows a large number of subnets within it).

## **Step 5 :** Click Next: Security, keep default settings, then Review + Create → Create.

## 

## 

## **Step 6:** Create Multiple Subnets

## Once the VNet is created, go to Virtual Networks and select MyVNet.

## In the left panel, click Subnets → + Add Subnet.

## Create Subnet 1 (Frontend)

## Name: Frontend-Subnet

## Address Range: 10.0.1.0/24

## Network Security Group (NSG): None (will configure later)

## Click Save.

## Create Subnet 2 (Backend):

## Name: Backend-Subnet

## Address Range: 10.0.2.0/24

## Click Save.

## 

## This step creates two isolated subnets within the VNet for better traffic control.

## 

## **Step 7:** Create and Configure a Route Table

## Go to Route Tables in Azure Portal.

## 

## **Step 8:** Click + Create and enter details

## 

## Name: MyRouteTable

## Resource Group: Same as VNet

## Region: Same as VNet

## Click Review + Create → Create.

## 

## 

## **Step 9 :** Associate Route Table with Subnets

## Open MyRouteTable → Click Subnets → + Associate.

## 

## 

## **Step 10 :** Select MyVNet → Choose Frontend-Subnet → Click OK.

## Repeat for Backend-Subnet.

## 

## This route table will help control how traffic moves between subnets.

## **Step 11:** Deploy Virtual Machines in Different Subnets

## Go to Virtual Machines → Click + Create.

## Create the first VM (Frontend VM):

## Name: VM-Frontend

## Image: Windows Server (for RDP) or Ubuntu (for SSH).

## Size: Standard B2s (good for testing).

## Virtual Network: Select MyVNet.

## Subnet: Select Frontend-Subnet.

## Public IP: Choose None (for private networking).

## Click Review + Create → Create.

## 

## **Step 12 :** Create the second VM (Backend VM):

## Name: VM-Backend

## Virtual Network: MyVNet.

## Subnet: Backend-Subnet.

## Public IP: None.

## Click Create.

## 

## Now you have two VMs in different subnets.

## **Step 13 :** Test Internal Communication

## Using RDP for Windows VMs

## Find the Private IP of Backend VM:

## Go to Virtual Machines → VM-Backend → Networking.

## Note the Private IP (e.g., 10.0.2.4).

## 2. Connect to Frontend VM using RDP:

## Open Remote Desktop Connection (RDP).

## Enter VM-Frontend’s Public IP (if available) and connect

## 3. From VM-Frontend, RDP into VM-Backend using Private IP:

## Open RDP inside the VM and connect to 10.0.2.4.

## 

## **Outcome**

## At the end of this exercise, you will have:

## ✅ A Virtual Network (VNet) with multiple subnets.

## ✅ VMs deployed in different subnets.

## ✅ Configured internal communication using RDP or SSH.

## ✅ Understood how Azure networking, routing, and security groups work.

## This setup is useful for real-world applications, where different services (frontend, backend, databases) need to communicate securely in a cloud environment.