**Student: Borhan Uddin**

**Student ID:270651714**

**RESEARCH AND IMPLEMENT A PRACTICAL DEMONSTRATION DCE-03 | ASSESSMENT-3.1**

Contents

[RESEARCH AND IMPLEMENT A PRACTICAL DEMONSTRATION DCE-03 | ASSESSMENT-3.1 0](#_Toc203836860)

[Task 1: Research and implement open-source infrastructure to facilitate a cloud-ready environment 2](#_Toc203836861)

[Design Topology: 3](#_Toc203836862)

[Submission Image: Yoobee\_VPC\_1 in Sydney Region 4](#_Toc203836863)

[Submission Image: Yoobee\_VPC\_2 in Singapore Region 5](#_Toc203836864)

[Submission Image: Yoobee\_VPC \_Singapore Subnet 6](#_Toc203836865)

[Yoobee Server EC2 Linux instance at the Sydney region 6](#_Toc203836866)

[Submission Image-: Yoobee Server\_2 sin EC2 Linux instance at the Singapore region 7](#_Toc203836867)

[Submission Image: Yoobee Server EC2 Linux instance accessible using SSH port. 8](#_Toc203836868)

[Submission Image: Yoobee Server2 EC2 Linux instance not accessible using SSH port. 9](#_Toc203836869)

[Creating VPC Peering 11](#_Toc203836870)

[Submission Image: vpc1 -vpc2 peering is 12](#_Toc203836871)

[Submission Image: Yoobee\_Web\_Server\_2 EC2 Linux instance accessible using SSH port from EC2 Linux Yoobee Server located in Sydney VPC. 13](#_Toc203836872)

[Task 2: Discuss and develop strategies to mitigate security risks of open source and cloud-based components. Practical Demonstration-2 14](#_Toc203836873)

[Design topology: 14](#_Toc203836874)

[Submission Image: EC2 Linux instances in common security group 14](#_Toc203836875)

[Submission Image: EC2 Linux instances in common security group 15](#_Toc203836876)

[Server1 Is accessible 15](#_Toc203836877)

[Server2 is Also Accessible 16](#_Toc203836878)

[Now sharing file with server 2 is accessible as you can see the message “This is the message from linux1 which accessible in Linux 2” 16](#_Toc203836879)

[Submission Image: Sample file access with SAMBA Server 17](#_Toc203836880)

[From Linux to windows 17](#_Toc203836881)

[Submission Image: Sample file access in Windows with samba client 18](#_Toc203836882)

# Task 1: Research and implement open-source infrastructure to facilitate a cloud-ready environment

The activity requires to setup two VPCs in two different regions with entirely different CIDR blocks. Within these VPCs we create two VMs, one of them in a public subnet and another in a private subnet. We then set up a VPC peering between the two subnets and access the VM in the private subnet.

Stage 1

Setup

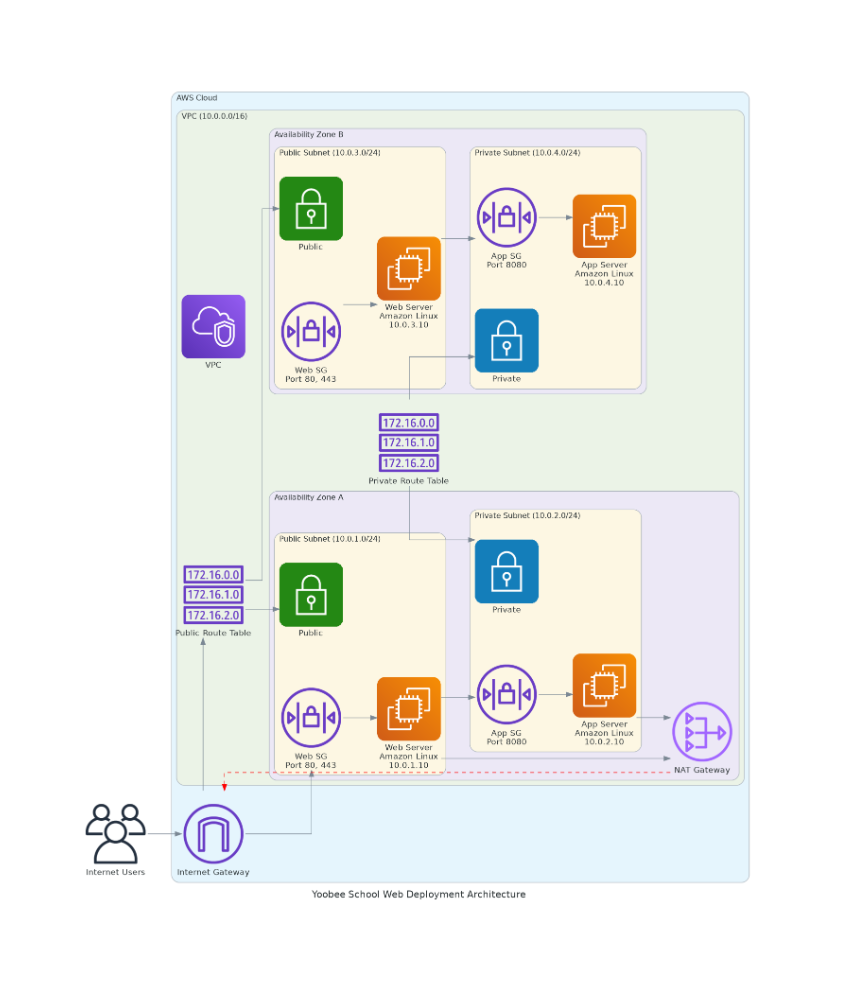
For the first stage, we will create two VPCs in Sydney and Singapore regions. In the Sydney region, we will create a public subnet with a connection to the internet gateway. For the Singapore region, we will create a private subnet. We create two EC2 machines in each of the subnets, with Amazon Linux.

Once we create the Linux machines, we try to SSH into the EC2 machine in the Sydney region. We also try to SSH into the EC2 machine in the Singapore region. We then try to SSH into the EC2 machine in Singapore region from the EC2 machine in Sydney region.

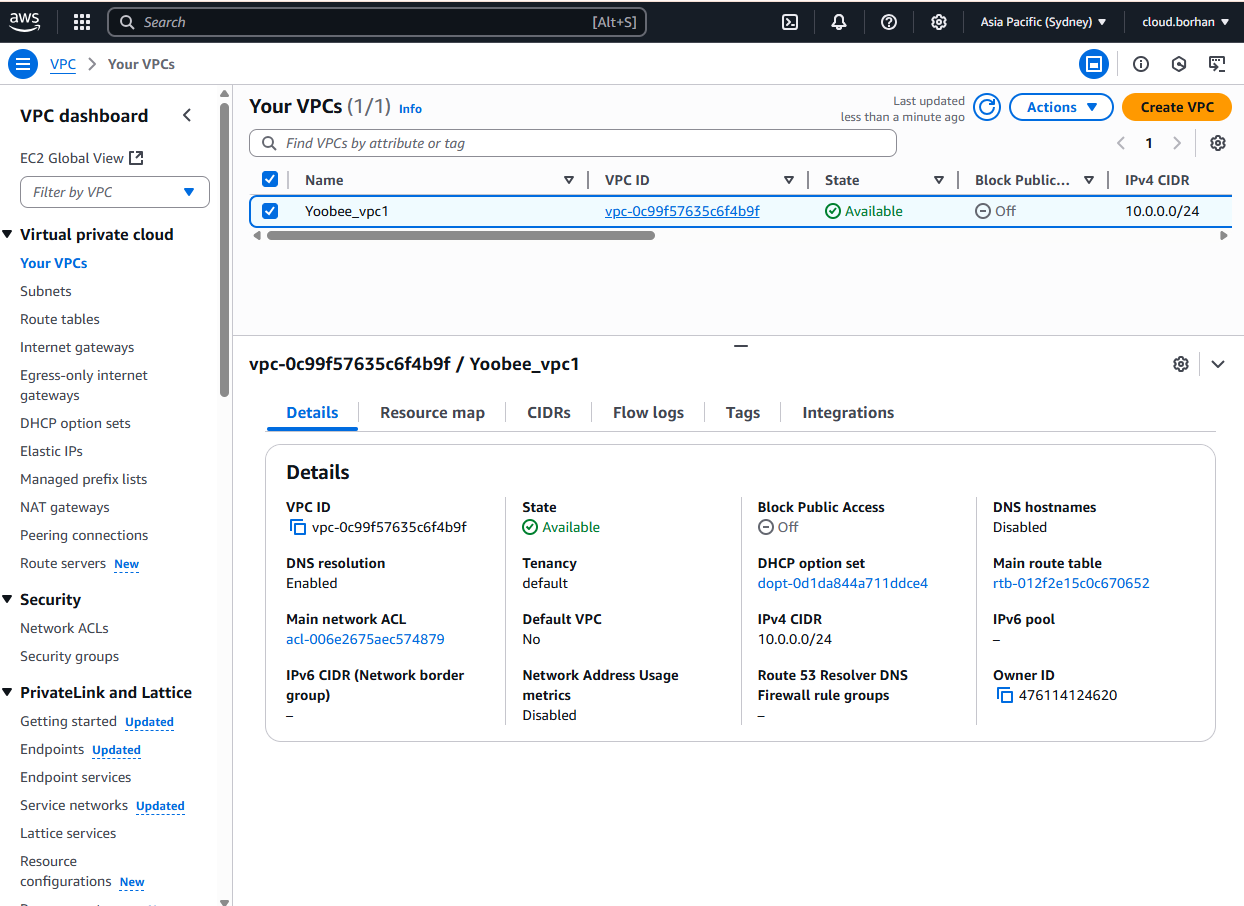
A screen shot of a cloud

AI-generated content may be incorrect.

## Design Topology:



## Submission Image: Yoobee\_VPC\_1 in Sydney Region



Submission Image: Yoobee\_VPC\_2 in Singapore RegionA screenshot of a computer

AI-generated content may be incorrect.Submission Image: Yoobee\_VPC \_Sydney Subnet

## Submission Image: Yoobee\_VPC \_Singapore Subnet

A screenshot of a computer

AI-generated content may be incorrect.

## Yoobee Server EC2 Linux instance at the Sydney region

*Machine 1*

Name: yoobee server

OS: Amazon Linux

Instance Type: t2. micro

Network: VPC-1

Subnet: Public subnet

Public IP: Enabled

A screenshot of a computer

AI-generated content may be incorrect.

Submission Image-: Yoobee Server\_2 sin EC2 Linux instance at the Singapore region

Name: yoobee Server 2

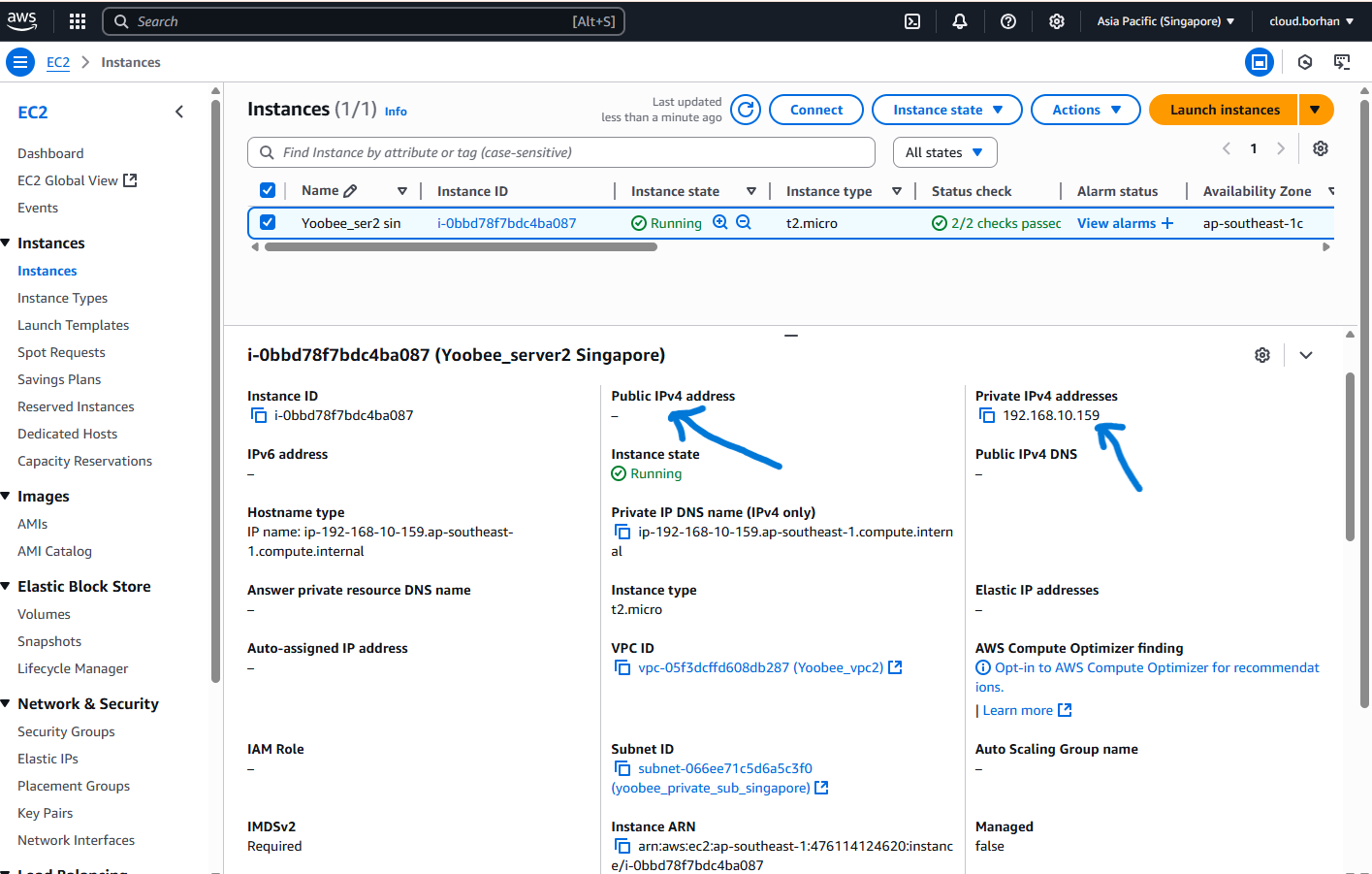
OS: Amazon Linux

Instance Type: t2. micro

Network: VPC-2

Subnet: Private subnet

Public IP: Disabled



## Submission Image: Yoobee Server EC2 Linux instance accessible using SSH port.

*Execution*

Try to SSH Into the yoobee server machine from our machine using the below command:

ssh -i "Key.pem" [ec2-user@3.107.245.86](mailto:ec2-user@3.107.245.869)

“Key.pem” is the key file assigned to yoobee server machine while launching the EC2 machine. We can see the exact ssh command from the connect option in the console.

While doing it make sure the Security group rules allow for SSH traffic.

A screenshot of a computer screen

AI-generated content may be incorrect.

## Submission Image: Yoobee Server2 EC2 Linux instance not accessible using SSH port.

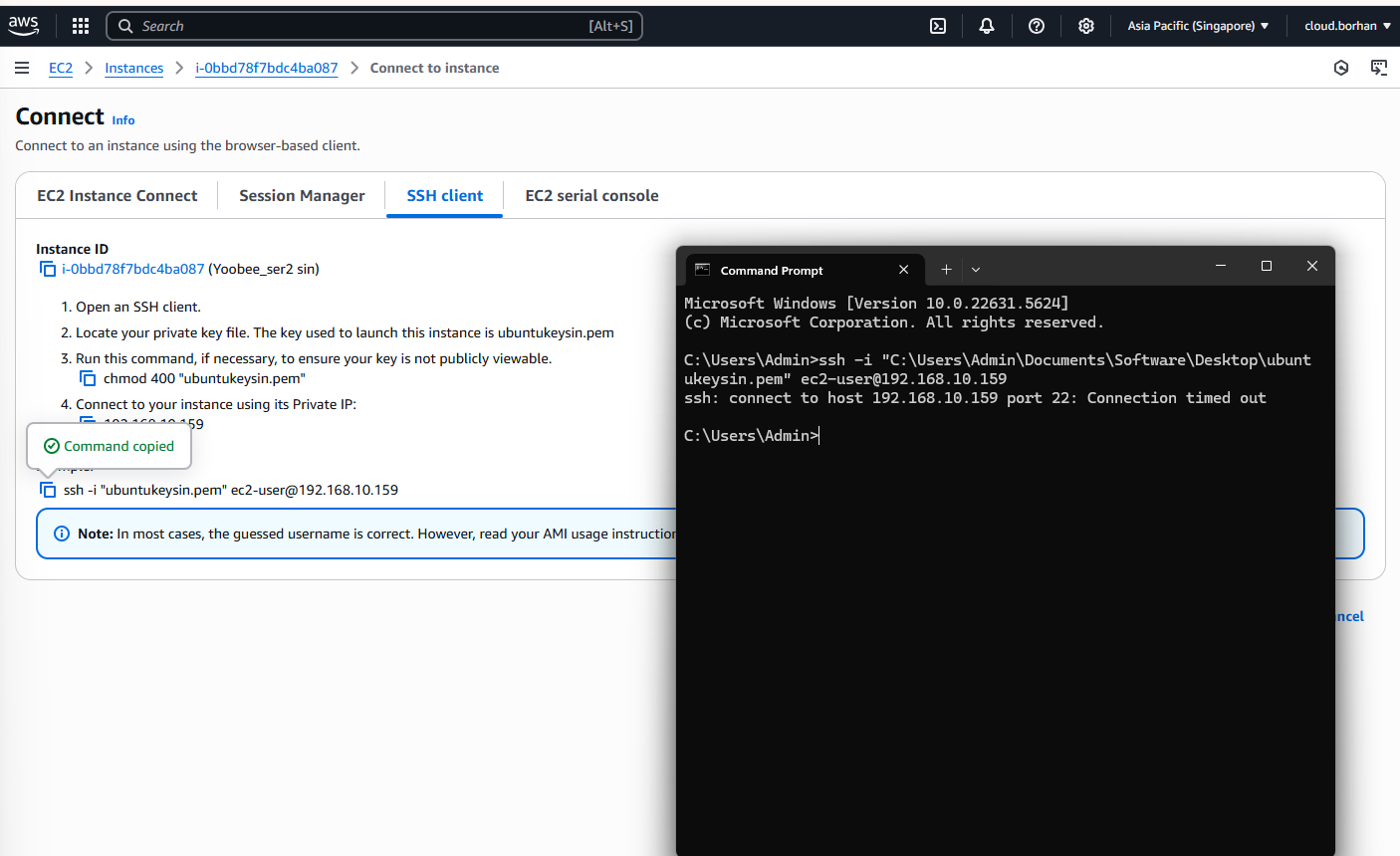
Try to SSH Into the Linux-2 machine from our machine using the command below:

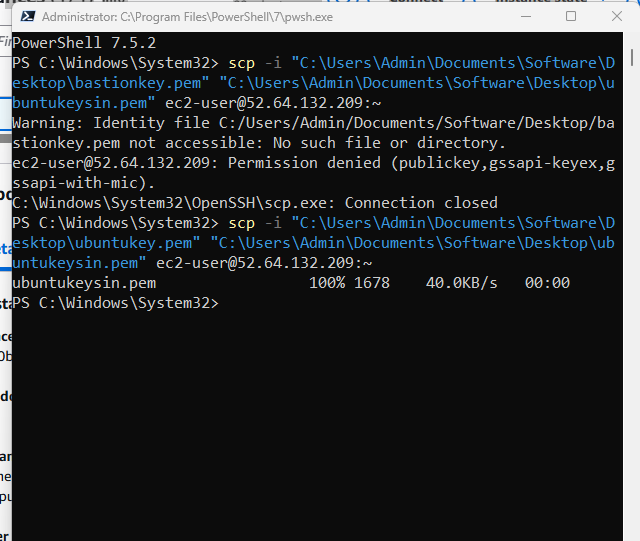
ssh -i "Key2.pem" ec2-user@192.168.10.159

“Key2.pem” is the key file assigned to server2 machine while launching the EC2 machine. We can see the exact ssh command from the connect option in the console.

While doing it make sure the Security group rules allow for SSH traffic.

Notice that the connection is getting timed out and not being established.





*Reason for Timeout*

If we look at the architecture diagram given above, while creating the Private subnet in VPC-2, we did not connect it with the internet using an internet gateway. We used an internet gateway to connect to VPC-1.

An internet gateway is a system that enables communication between a private network and the internet. Since VPC-2 is designed without any internet gateway, there is no way for any of the machines outside the network to reach server1 machine.

Hence, when we tried to SSH into the server2 machine using its private IP, we could not reach it.

In fact, for the server1 machine, if we are trying to SSH into it from outside the VPC-1 (in case of pinging from laptop) we will still not be able to reach it if we are connecting to the Private IP. That is why we enabled the public IP while creating the machine.

Stage 2

Setup

VPC Peering is a mechanism which enables two VPCs to connect directly between themselves using their private IPs without going through the internet. It is establishing a direct connection between the two networks and shields the network from the public internet.

For the second stage of the task, we will create VPC Peering between the Sydney region and Mumbai region using the acceptor requestor process. VPC-1 from Sydney will request a peering connection to VPC-2. In this case, VPC-1 becomes the Requestor.  VPC-2 must approve of the connection request to establish the VPC peering connection arising from VPC-1. Only if VPC-2 accepts, will the peering connection be active. Hence VPC-2 is the acceptor in this process.

After the Peering is created, we then must edit the route tables to route any requests to the other connected VPC via the VPC Peering. For e.g. any requests from VPC-1 to 192.168.0.0/24 CIDR (VPC-2) will be routed to the VPC Peering connection which will then route it to VPC-2 and vice versa.

Once the connection is established, we try to SSH into the EC2 machine in Sydney region. We also try to SSH into the EC2 machine in the Mumbai region. We then try to SSH into the EC2 machine in Mumbai region from the EC2 machine in Sydney region.

## Creating VPC Peering

Below is the Architecture of the Network after the VPC peering connection is established. Notice that the peering connection is established between the two VPCs directly without going through the internet.  A screen shot of a computer

AI-generated content may be incorrect.

## Submission Image: vpc1 -vpc2 peering is

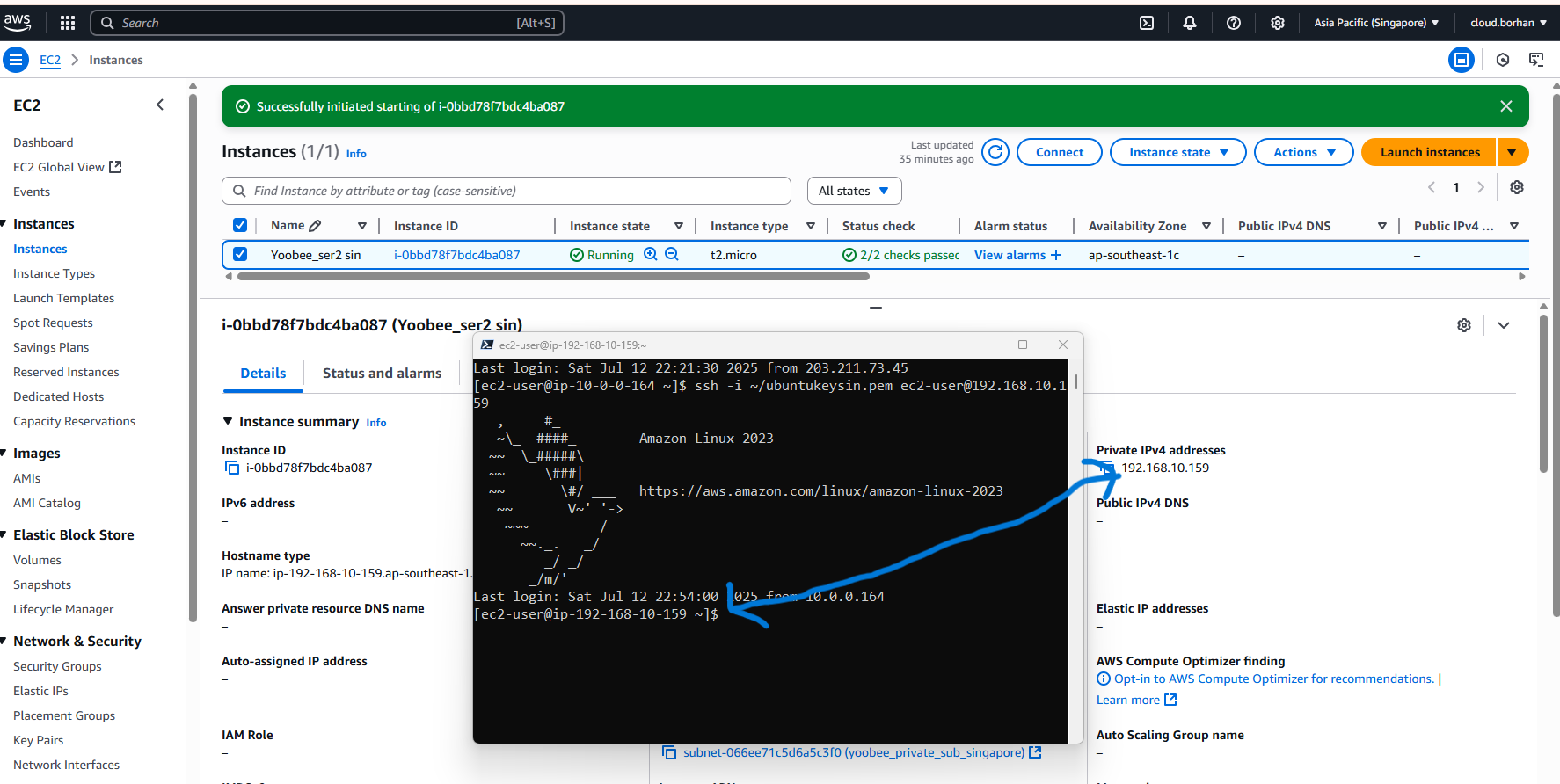
**Sydney vpc peering** A screenshot of a computer

AI-generated content may be incorrect.

**Singapore peering active** A screenshot of a computer

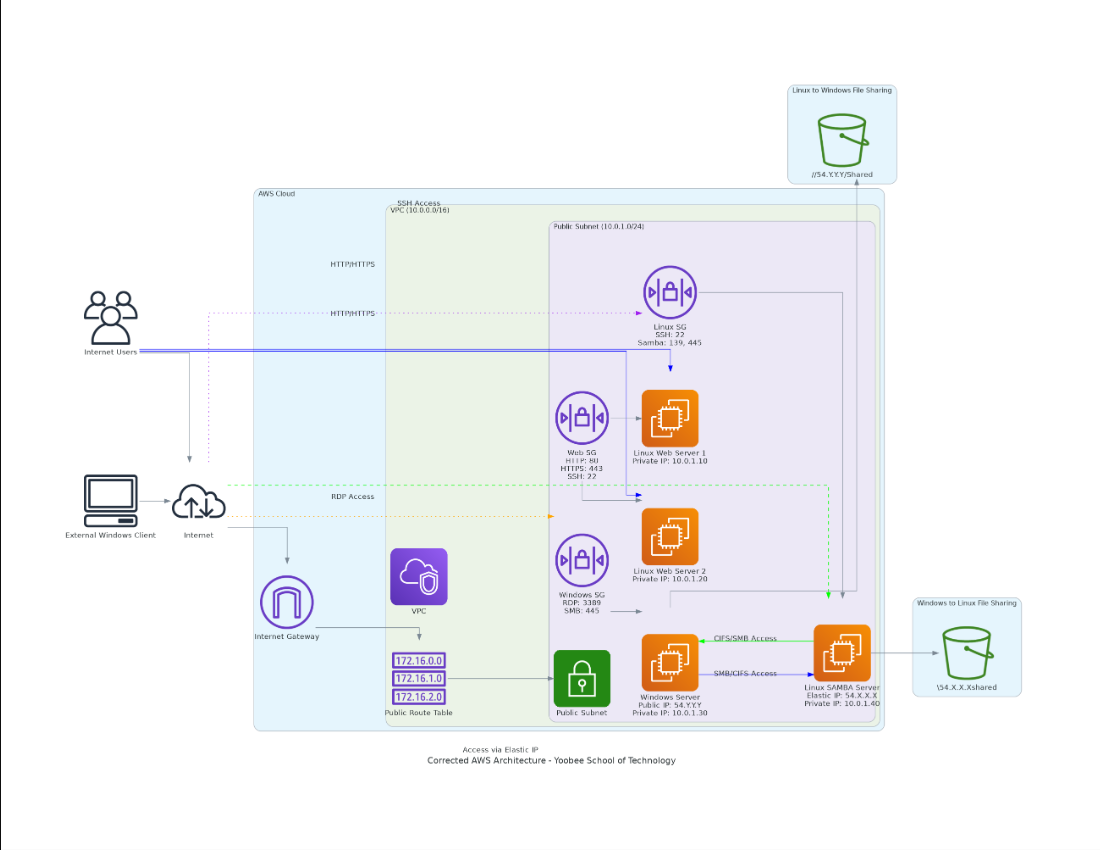
AI-generated content may be incorrect.

## Submission Image: Yoobee\_Web\_Server\_2 EC2 Linux instance accessible using SSH port from EC2 Linux Yoobee Server located in Sydney VPC.

****

# Task 2: Discuss and develop strategies to mitigate security risks of open source and cloud-based components. Practical Demonstration-2

## Design topology:

****

## Submission Image: EC2 Linux instances in common security group

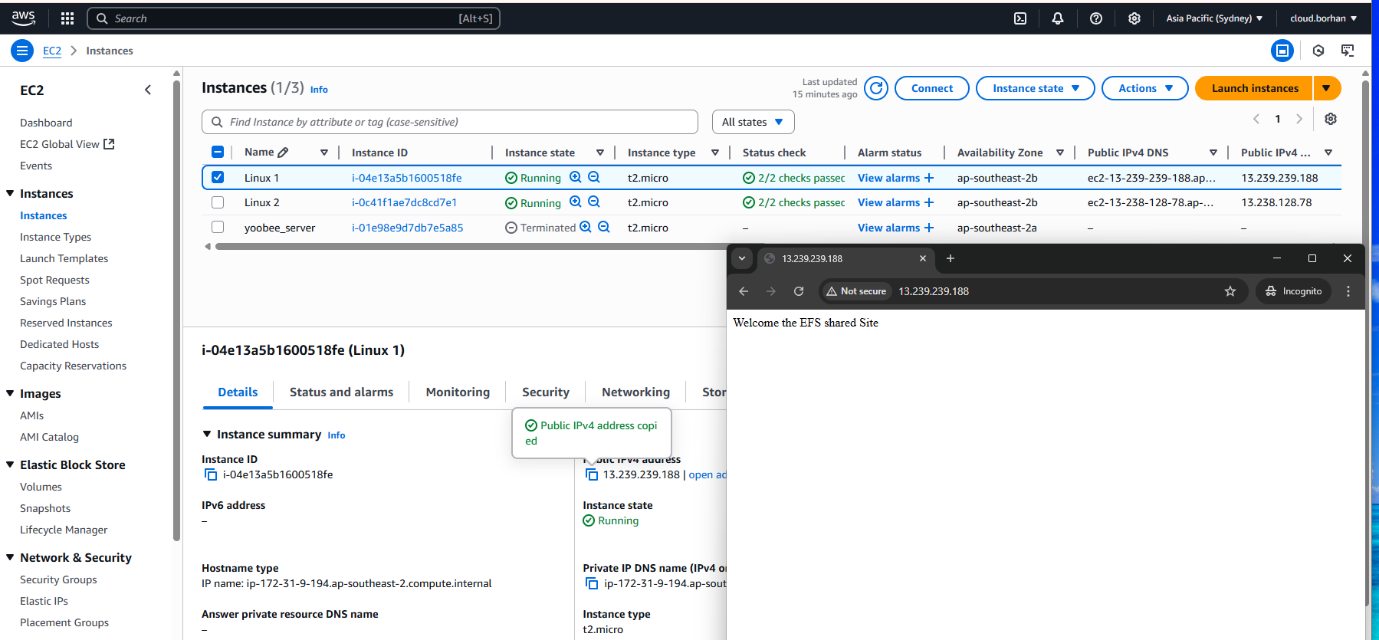
A screenshot of a computer

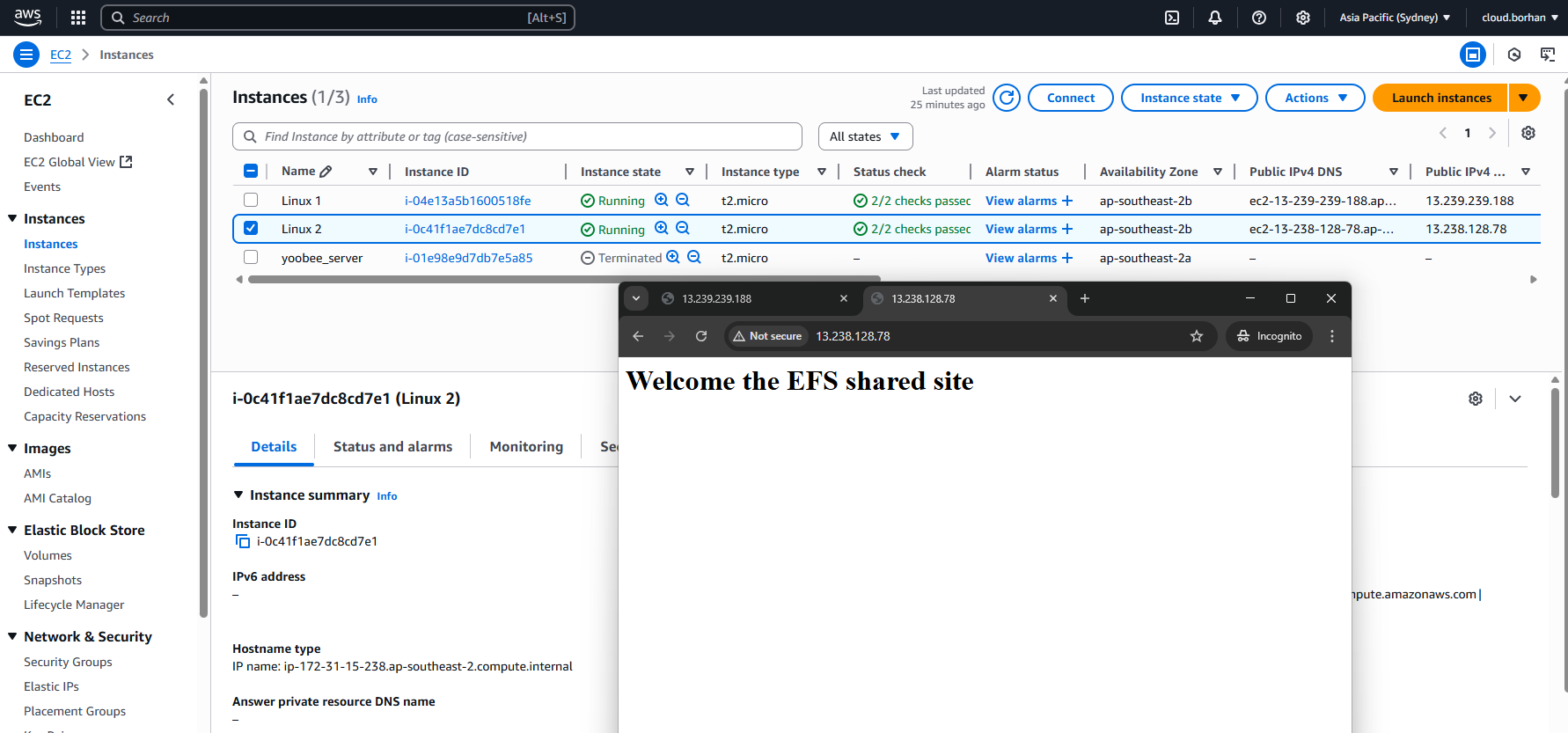
AI-generated content may be incorrect.

## Submission Image: EC2 Linux instances in common security group

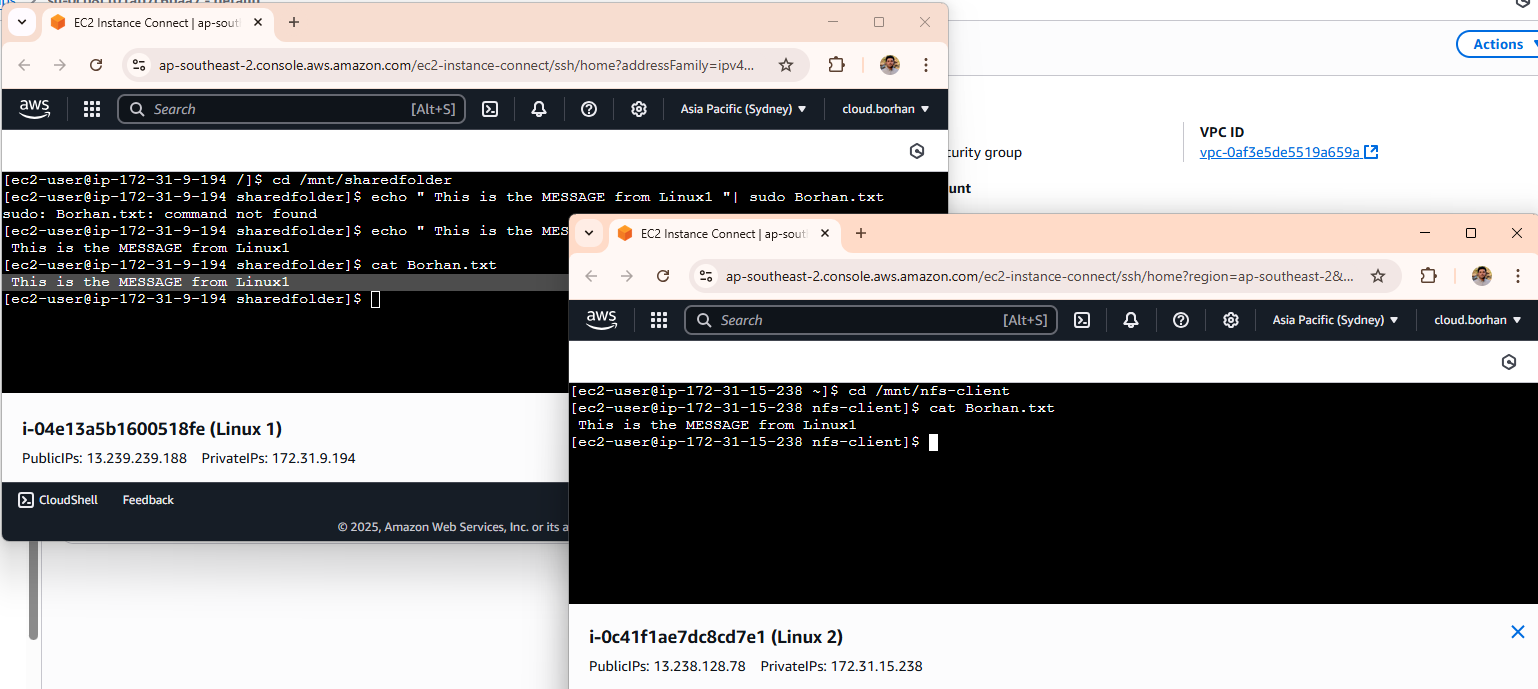
**A screenshot of a computer

AI-generated content may be incorrect.**

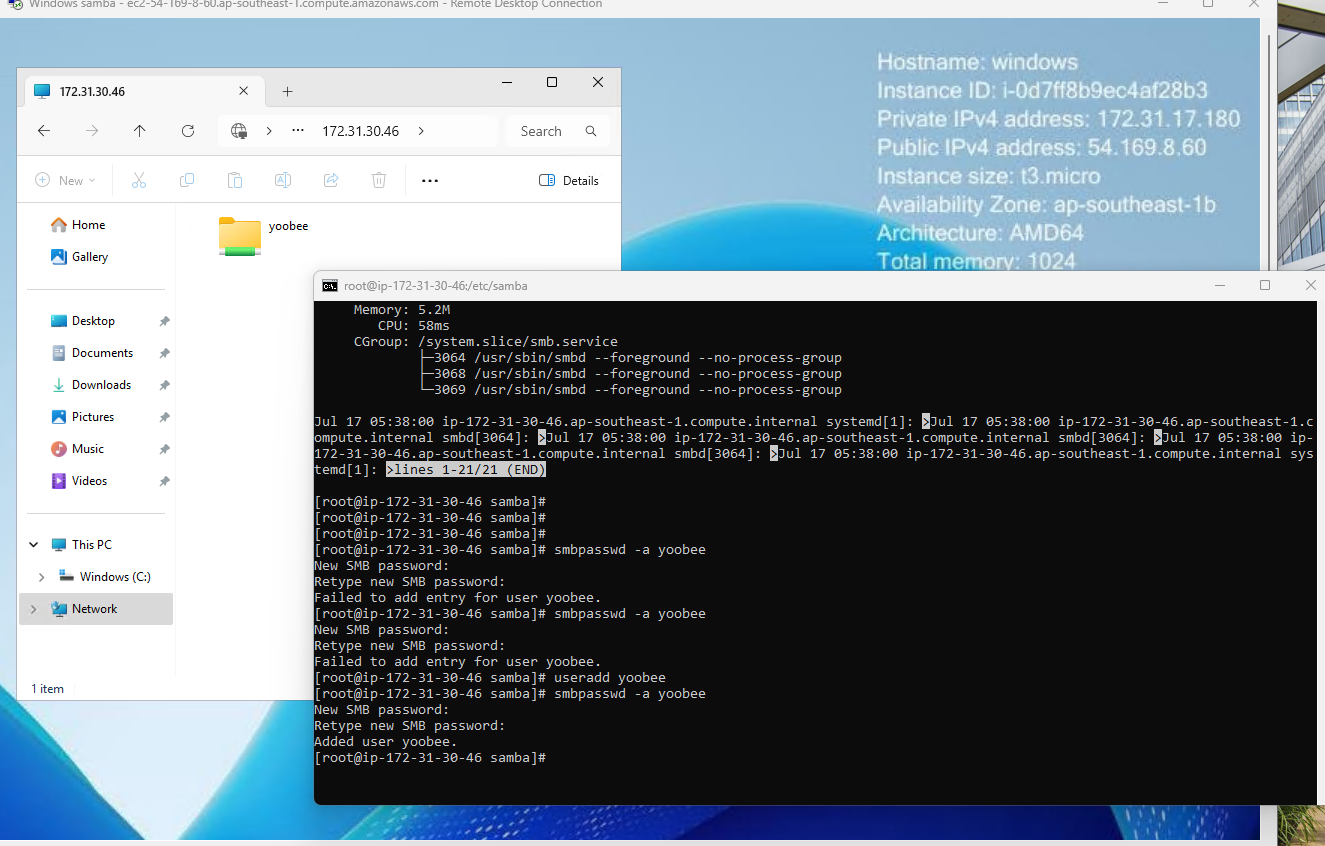
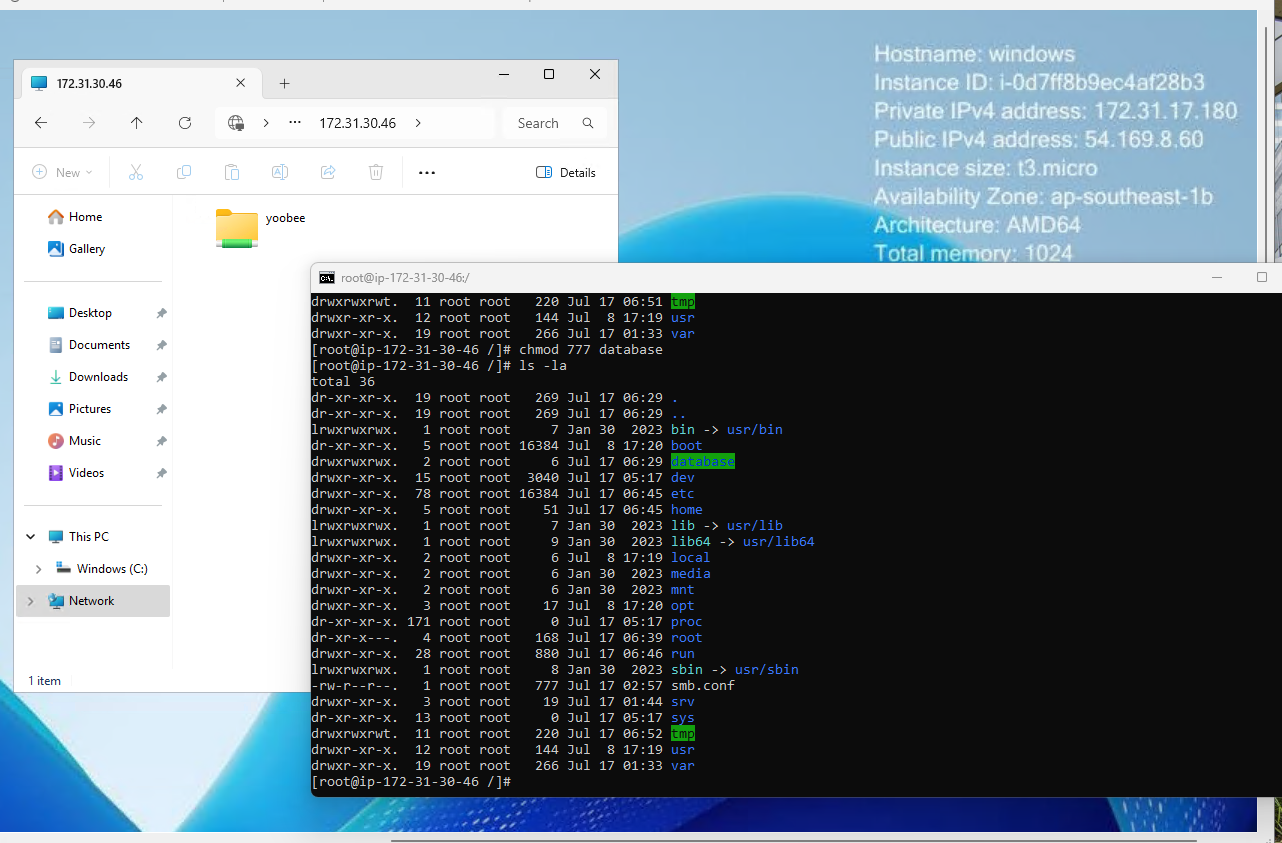
Server1 Is accessible:

Server2 is Also Accessible:

## Now sharing file with server 2 is accessible as you can see the message “This is the message from linux1 which accessible in Linux 2”



## Submission Image: Sample file access with SAMBA Server

From Linux to windows 

## Submission Image: Sample file access in Windows with samba client

