Portfolio 02

Networking and Security Practice

BUILDING A VIRTUAL SANDBOXED NETWORK USING VIRTUALBOX

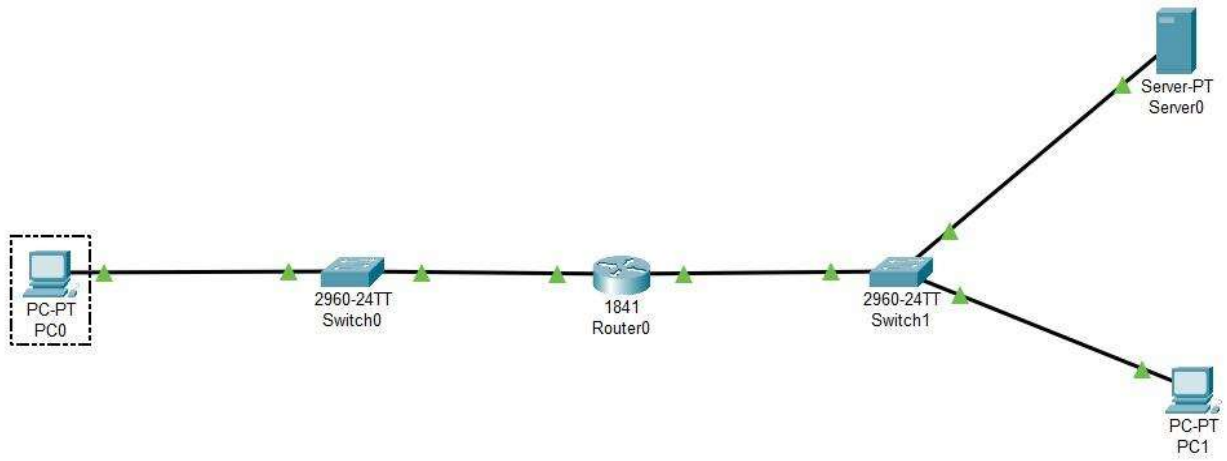
NAME: RITHUPARNA SANITHA DAYANAND

STUDENT ID: A000333680

Aims

This project's objective is to use VirtualBox to build a private, sandboxed virtual network. We can securely model real-world network situations in this isolated environment without affecting any operational systems. Several virtual machines (VMs) with private IP addresses will make up the network. We will get practical experience with networking topics with this project, such as IP subnetting, network interface configuration, and basic server setup. Important abilities in network design, planning, and organization will also be taught.

1: Network Diagram

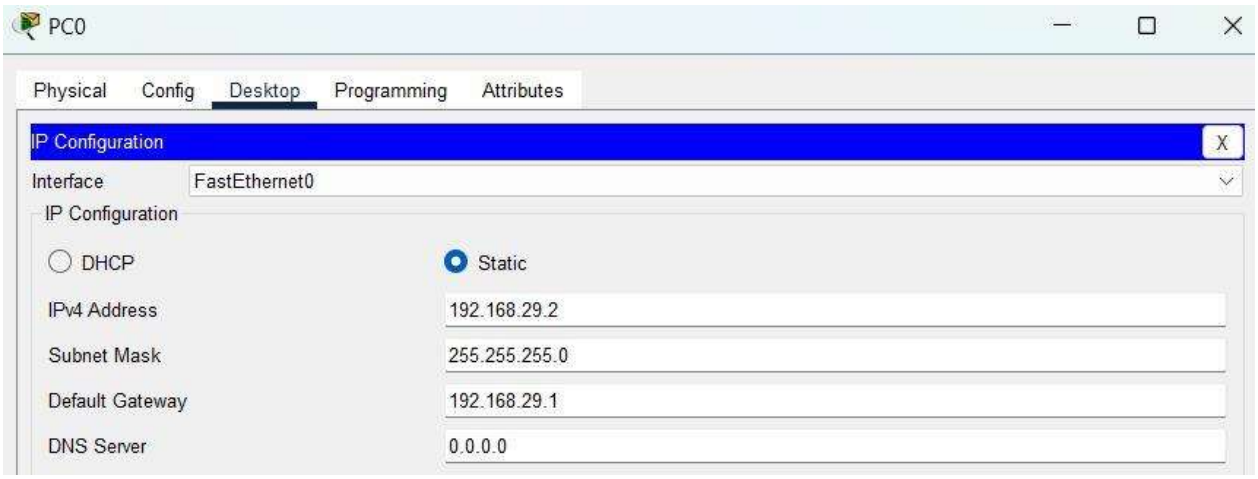


Gateway Router:

- Router1>enable
- Router1#configure terminal
- Router1[config]# interface fastethernet 0/0
- Router1(config-if)# ip address 192.168.29.1 255.0.0.0
- Router1(config-if)# no shutdown
- Router1(config-if)# exit
- Router1(config)# interface serial 0/1
- Router1(config-if)# ip address 192.168.129.1 255.0.0.0
- Router1(config-if)# no shutdown
- Router1(config-if)# exit

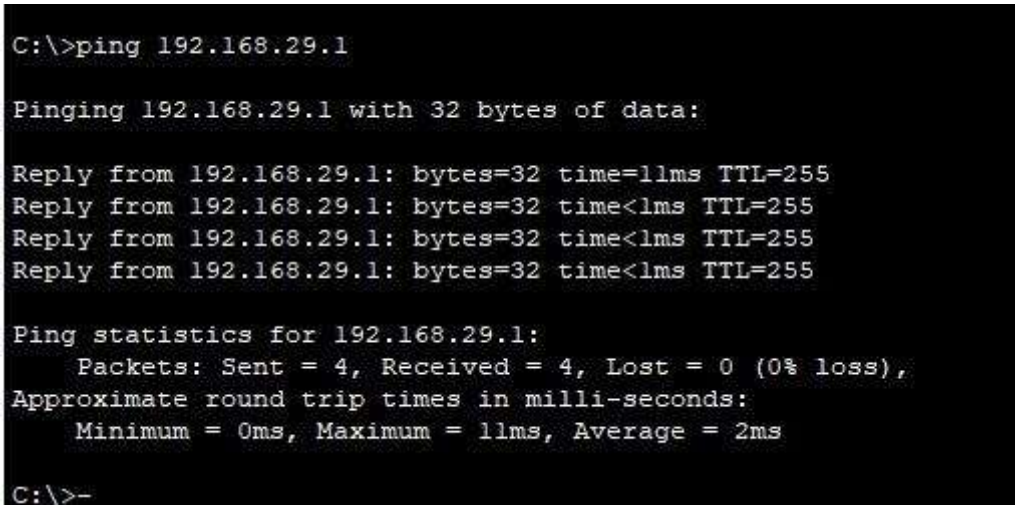
Desktop Setup

- Click and open the desktop ⏩ IP configuration ⏩ Change to static ⏩ Set IP address



- Ping 192.168.29.1

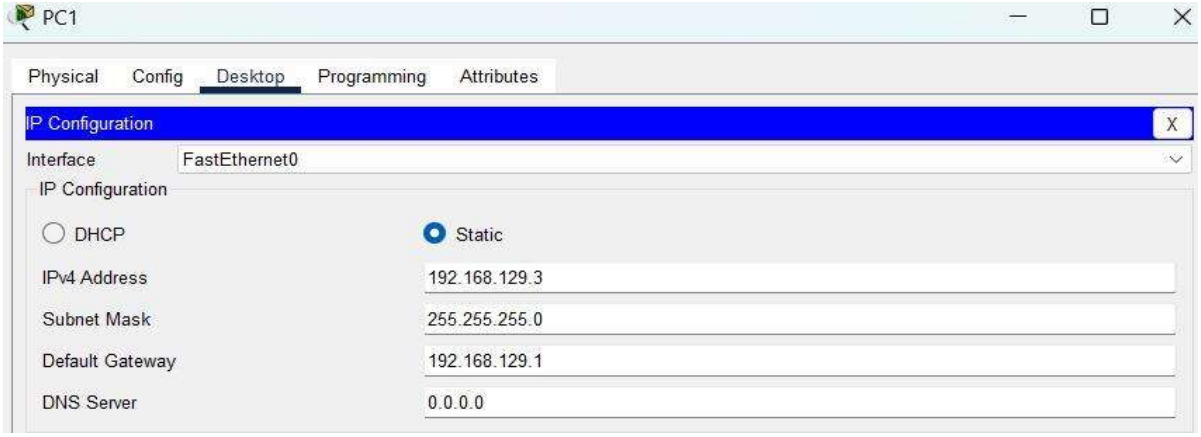
SCREENSHOT FOR PC TO GATEWAY PINGING



Application server:

- Open the application server icon by clicking it and go to desktop ⏩ IP configuration ⏩ change to static ⏩ Set IP address

SCREENSHOT OF SETING UP IP ADDRESS



- Click and open the Application Server and go to the service ⏩ Click HTTP ⏩ Turn ON HTTP and HTTPS ⏩ add HTML page in Application Server.
- Click the Application Server and it go to the desktop ⏩ Web Browser ⏩ Type the IP address 192.168.129.2

SCREENSHOT OF WEB SERVER SHOWS A WEB SITE IN WEB BROWSER

2: IP Address Table

DEVICE	ROLE	IP ADDRESS	SUBNET MASK	INTERFACE
Desktop VM	Management	192.168.29.2	255.255.255.0	enp0s3 [Internal]
Gateway Router VM (enp0s3)	Subnet 01 - Internal Network	10.0.0.16	255.255.255.0	enp0s3 [external]
Gateway Router VM (enp0s8)	Subnet 02 - Internal Network	192.168.29.1	255.255.255.0	enp0s8 [Internal]
Gateway Router VM (enp0s9)	Internet Access	192.168.129.3	255.255.255.0	enp0s9 [Internal]
Application Server VM	Server	192.168.129.2	255.255.255.0	enp0s3 [Internal]

3: Git Pages Lab Report

Configuration Step

We need three Virtual Machines for this

- Download and Install the Virtual Box (VM) in PC
- Download
  - Ubuntu Desktop (.iso format).
  - Ubuntu Server OS (.iso format).
  - Application Server in Bitnami WordPress (.ova format).
- Add this three to OS virtual machine

UBUNTU SERVER OS Configuration Steps:

Step 1: Create a New Virtual Machine (VM) for Ubuntu Server

- Open the VirtualBox.
- Click on “new” to create a new virtual machine.
- Name the VM (e.g., Ubuntu Server).
- Select the type as Linux and the version as Ubuntu (64-bit).
- Allocate memory (RAM) for the VM (e.g., 2048 MB or higher based on system’s capacity).
- Choose to create a virtual hard disk now and set a sufficient disk size (e.g., 10 GB).
- Click create

Step 2: Configuration of Network Interfaces

Here we need two network interfaces in the Ubuntu Server VM to act as a router between the two subnets.

- Go to Settings of your new VM.
- Select Network.
- Adapter 1: Set the network to "intnet" (internal network) for Subnet 1. Go to the Advanced tab and choose an adapter type like PCnet-FAST III or another supported option.

  Note: In the Ubuntu server terminal, this network adapter will typically be named enp0s8.

- Adapter 2: Enable the adapter and set it to a different internal network (for example, "intnet1") to be used for Subnet 2.
  Note: In the Ubuntu server terminal, this adapter will usually be named enp0s9.

- Adapter 3: Set the adapter to NAT. In the Ubuntu server terminal, this adapter is usually named **enp0s3**, and its IP address will be assigned automatically via DHCP. This adapter will be used to provide internet access through the host computer.

Step 3: Install Ubuntu Server

- Start VM and select the Ubuntu Server ISO as the boot disk.
- Go through the installation process:
- Set time zone, keyboard layout, etc.
- Create a user account and set a password.
- When asked, select the option to install OpenSSH Server (to enable remote access later if needed).
- Complete the installation, then reboot the VM.

Step 4: Configure Static IPs on the Network Interfaces

- After installing, we need to assign static IP addresses to both network interfaces (each in different subnets).
- Log to the Ubuntu Server VM.
- Edit network configuration file:
  - sudo nano /etc/netplan/00-installer-config.yaml
- Enter this code.

```
network:
  ethernets:
    enp0s3:
      addresses: [192.168.29.1/24]
      dhcp4: false
    enp0s8:
      addresses: [192.168.129.1/24]
      dhcp4: false
    enpes9:
      dhcp4: true
  version: 2
```

- Save it (CTRL + X) type (yes) and enter.
- Apply the network changes

- sudo netplan apply
- Ip a

Step 5: Enable IP Forwarding

To allow routing between the two subnets, you need to enable IP forwarding.

- Open the sysctl configuration file:
  -sudo nano /etc/sysctl.conf
- Uncommand the line (or add it if not present):
  - net.ipv4.ip_forward=1
- Apply the changes:
  - sudo sysctl –p

configured iptables to allow forwarding using:

- Allow forwarding between enp0s8 and enp0s9

sudo iptables -A FORWARD -i enp0s8 -o enp0s9 -j ACCEPT

sudo iptables -A FORWARD -i enp0s9 -o enp0s8 -j ACCEPT

- Allow forwarding between enp0s3 and the internal interfaces

sudo iptables -A FORWARD -i enp0s3 -o enp0s8 -j ACCEPT

sudo iptables -A FORWARD -i enp0s8 -o enp0s3 -j ACCEPT

sudo iptables -A FORWARD -i enp0s3 -o enp0s9 -j ACCEPT

sudo iptables -A FORWARD -i enp0s9 -o enp0s3 -j ACCEPT

- Enable NAT on enp0s3 for internet access

sudo iptables -t nat -A POSTROUTING -o enp0s3 -j MASQUERADE

- made the changes permanent using:

sudo apt install iptables-persistent

sudo netfilter-persistent save

sudo netfilter-persistent reload

Ubuntu OS Configuration Steps:

Step: Installing Ubuntu Desktop in VM

To creare a virtual machine and install a GUI-based OS from an ISO file

- Launch VirtualBox
- Click “New” to create a new machine
- Fill in the details as requested so that the new machine is:
  - Named “Ubuntu Desktop”
  - The ISO named “ubuntu-24.04.1-desktop-amd64.iso” is selected
  -The checkbox for “Skip Unattended Installation” is checked
- Click “Next”. Now check the machine has:
  - 2048mb of Base Memory
  - Two processors
- Click “Next”. Create a Virtual Hard Disk for the machine. It should be a minimum of 25GB.
- Click “Next”. Verify settings are as above and click “Finish”.
- Click “Start” to boot the virtual machine
- Select “Try or Install Ubuntu”
- Click “Install Ubuntu”
- You need to create an Ubuntu account and click Restart Now.

Step 2: Configure the Network Interfaces

We need one network interface on the Ubuntu Desktop VM.

- Go to Settings of your new VM.
- Select Network.
- Adapter 1: Set to Internal Network (name it differently, e.g., intnet).
- Open Ubuntu desktop in VM and login
- Go to the settings ⏩ Network ⏩ enp0s3 setting ⏩ IPV4 ⏩ IPV4 Method change to “MANNUAL” ⏩ Add address =192.168.29.2, Add netmask =255.255.255.0 or 24, Add gateway =192.168.29.1
- Apply
- Disconnect the Network and connect.

Bitnami Web Application Configuration Step:

To create Bitnami virtual machine using VirtualBox

Step 1: Install Bitnami Application in VM.

- Click “File” then “Import appliance”.
- Navigate to the file named “bitnami-wordpress-6.3.1-r0-debian-11-amd64.ova”.
- Click “Open”.
- Click “Next”.
- Click “Finish”.
- Click “Start” to launch the machine.
- The first-time log in details is displayed on-screen. You will be required to reset the password on your first log in.
- Close the Bitnami.

Step 2: Configure the Network Interfaces

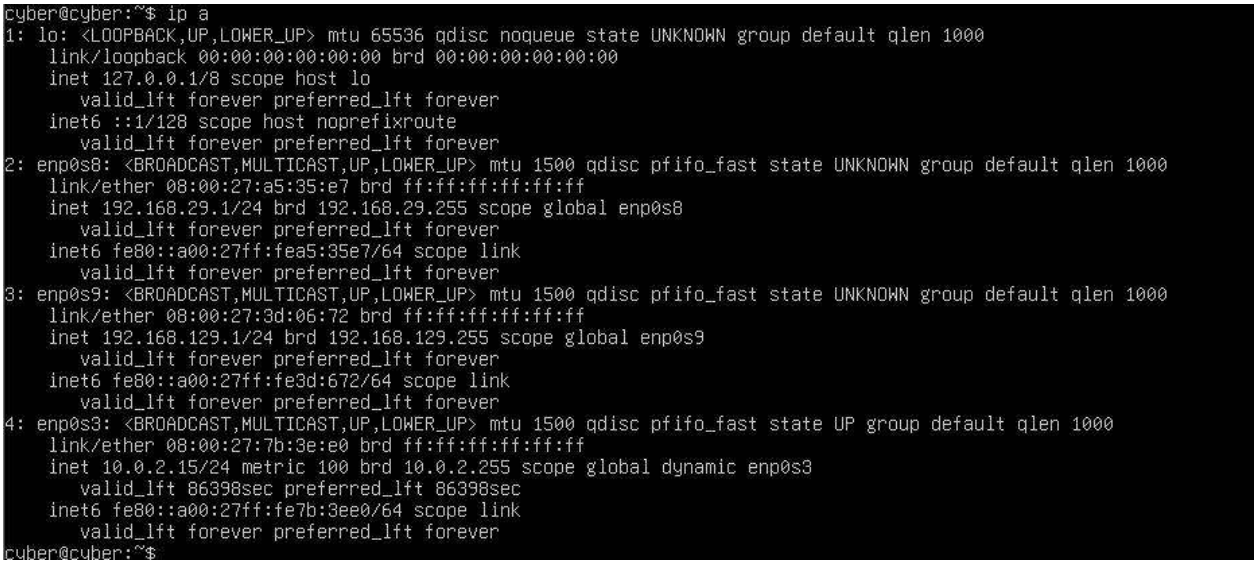
- Go to the Settings of new VM.
- Select the Network.
- Adapter 1: Set to Internal Network [[name it differently, e.g., intnet1].
- Open Bitnami Application in VM and login.
- Type the code
  - sudo nano /etc/network/interfaces
  - suto enp0s3 iface enp0s3 inet static address 192.168.129.2 netmask 255.255.255.0 gateway 192.168.129.1
  - Save it (CTRL + X) type (yes) and enter.
  - sudo ifdown enp0s3 && sudo ifup enp0s3
  - sudo systemctl restart networking
  - ip a

4: Functional Test Results

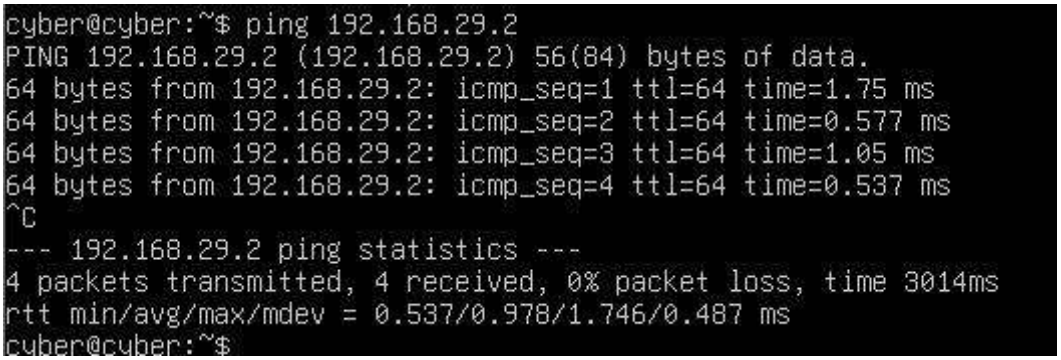
SCREENSHOTS FOR ALL THE FUCTIONAL TEST RESULT

UBUNTU SERVER OS:

- **Network IP configuration for ubuntu server**

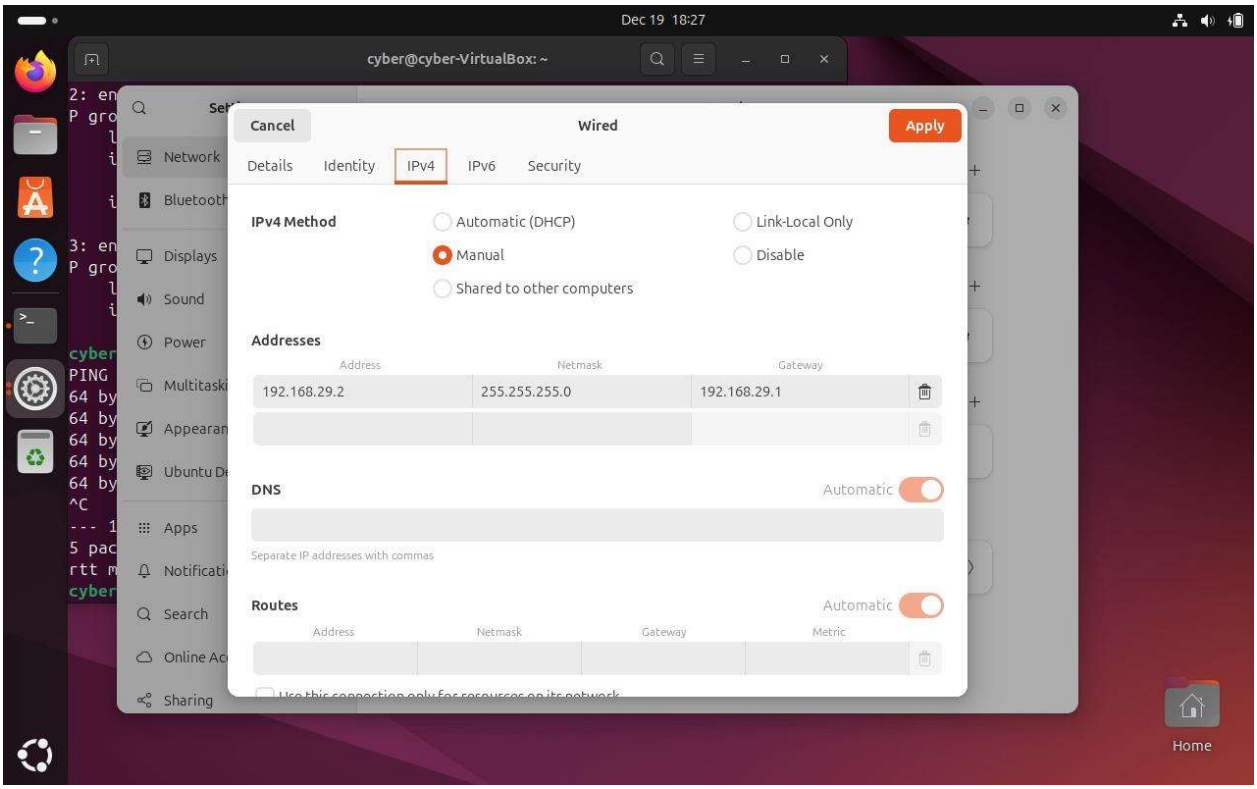


- **PING ubuntu Server to Ubuntu desktop**



UBUNTU DESKTOP OS:

- **IP configuration for desktop**



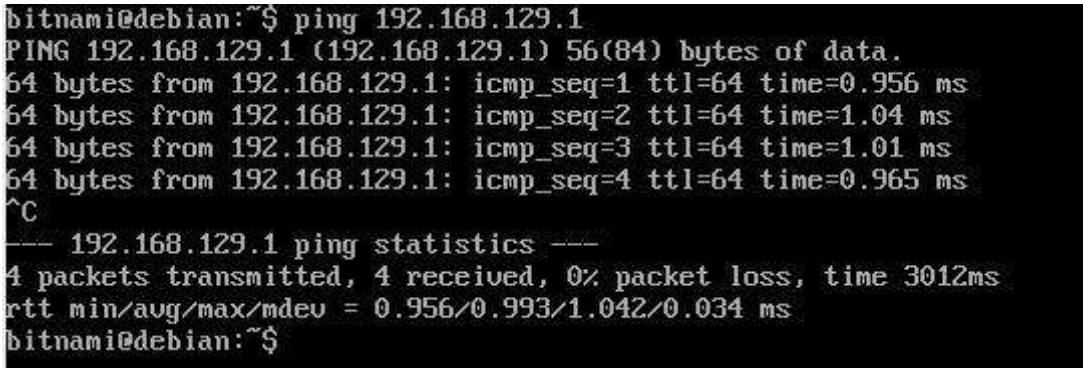
- **Ping Ubuntu desktop to ubuntu server**

```
cyber@cyber-VirtualBox: $ ping 192.168.29.1
PING 192.168.29.1 (192.168.29.1) 56(84) bytes of data.
64 bytes from 192.168.29.1: icmp_seq=1 ttl=64 time=0.882 ms
64 bytes from 192.168.29.1: icmp_seq=2 ttl=64 time=0.833 ms
64 bytes from 192.168.29.1: icmp_seq=3 ttl=64 time=0.887 ms
64 bytes from 192.168.29.1: icmp_seq=4 ttl=64 time=0.959 ms
^C
--- 192.168.29.1 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3017ms
rtt min/avg/max/mdev = 0.833/0.890/0.959/0.044 ms
```

BITNAMI APPLICATION SERVER

- **IP configuration for Bitnami**

- **Ping Bitnami application server to ubuntu server**



SCREEN RECORDING LINK OF MY YOUTUBE VEDIO PERFORMING FUNCTIONAL TEST:

https://youtu.be/uK88oVgs-zc