Computer Networks

Project

Course Instructor

Dr. Usman Habib

Group Members

Abdullah Daoud……(22I-2626)

Usman Ali……(22I-2725)

Faizan Rasheed……(22I-2734)

Section

SE-E

Date

Wednesday, April 14th, 2025

Spring 2025

Department of Software Engineering

FAST – National University of Computer & Emerging Sciences

Islamabad Campus

Table of Contents

NUCES SDN Project: Detailed Implementation Report in Cisco Packet Tracer

Abstract

This report provides a detailed account of the implementation of a Software-Defined Networking (SDN) project for NUCES using Cisco Packet Tracer. The project is structured into four parts: planning and design, network setup and configuration, SDN controller logic implementation, and traffic engineering with load balancing. Each part is documented with comprehensive details on the topology, configurations, testing procedures, challenges encountered, solutions applied, and deliverables. The project simulates SDN concepts within the constraints of Cisco Packet Tracer, achieving a functional network with inter-campus connectivity, traffic engineering, and load balancing.

Introduction

The objective of this project is to design and implement a network for NUCES using SDN principles in Cisco Packet Tracer. The network spans multiple campuses (Islamabad, Karachi, Lahore, Faisalabad, Peshawar, Multan) with a head office, incorporating SDN controller logic, traffic engineering, and load balancing. The project is executed in four parts as per the revised plan, using static routing, access control lists (ACLs), and load balancing techniques to simulate SDN functionality.

Part 1: Planning and Design

1.1. Objective

The goal of Part 1 is to design the network topology and IP addressing scheme for NUCES, ensuring scalability and support for SDN concepts such as centralized control, traffic engineering, and load balancing.

1.2. Network Topology Design

The topology is structured to represent a multi-campus network with a centralized head office:

- **Head Office**:
  - **Devices**: ‘HeadOffice-Controller’ (Cisco 2911 router), ‘HO-Switch1’ to ‘HO-Switch3’ (2950 switches), ‘HO-EdgeServer1’ and ‘HO-EdgeServer2’ (servers).
  - **Connections**: ‘HeadOffice-Controller’ connects to switches via ‘GigabitEthernet0/0‘. Switches connect to servers and other devices.
- **Campuses**: Six campuses (Islamabad, Karachi, Lahore, Faisalabad, Peshawar, Multan), each with:
  - **Devices**: A controller (e.g., ‘Islamabad-Controller‘), switches (e.g., ‘Isb-Lab1-Switch‘), 5 PCs per lab (e.g., ‘Isb-Lab1-PC1’ to ‘PC5’), and 1 edge server (e.g., ‘Isb-EdgeServer1‘).
  - **Connections**: PCs connect to switches, switches to the controller’s ‘GigabitEthernet0/0‘.
- **Backbone**:
  - **Primary Backbone**: Connects all controllers via ‘GigabitEthernet0/1’ interfaces.
  - **Backup Network**: A secondary network for redundancy, connected via ‘GigabitEthernet0/2’ interfaces through ‘Backup-Intermediary-Switch‘.

1.3. IP Addressing Scheme

A structured IP addressing scheme was designed to ensure clarity and scalability:

- **Head Office**: ‘10.0.0.0/24’
  - ‘HeadOffice-Controller‘ (G0/0): ‘10.0.0.1‘
  - ‘HO-Switch1‘: ‘10.0.0.2‘
  - ‘HO-Switch2‘: ‘10.0.0.3‘
  - ‘HO-Switch3‘: ‘10.0.0.4‘
  - ‘HO-EdgeServer1‘: ‘10.0.0.5‘

  - ‘HO-EdgeServer2‘: ‘10.0.0.6‘

- **Islamabad**: ‘10.0.1.0/24‘
  - ‘Islamabad-Controller‘ (G0/0): ‘10.0.1.1‘
  - ‘Isb-Lab1-PC1‘ to ‘PC5‘: ‘10.0.1.10‘ to ‘10.0.1.14‘
  - ‘Isb-Lab2-PC1‘ to ‘PC5‘: ‘10.0.1.15‘ to ‘10.0.1.19‘
  - ‘Isb-Lab3-PC1‘ to ‘PC5‘: ‘10.0.1.20‘ to ‘10.0.1.24‘
  - ‘Isb-Lab4-PC1‘ to ‘PC5‘: ‘10.0.1.25‘ to ‘10.0.1.29‘
  - ‘Isb-Lab5-PC1‘ to ‘PC5‘: ‘10.0.1.30‘ to ‘10.0.1.34‘
  - ‘Isb-EdgeServer1‘: ‘10.0.1.5‘
- **Karachi**: ‘10.0.2.0/24‘
  - ‘Karachi-Controller‘ (G0/0): ‘10.0.2.1‘
  - ‘Kar-Lab1-PC1‘ to ‘PC5‘: ‘10.0.2.10‘ to ‘10.0.2.14‘
  - ‘Kar-Lab2-PC1‘ to ‘PC5‘: ‘10.0.2.15‘ to ‘10.0.2.19‘
  - ‘Kar-Lab3-PC1‘ to ‘PC5‘: ‘10.0.2.20‘ to ‘10.0.2.24‘
  - ‘Kar-Lab4-PC1‘ to ‘PC5‘: ‘10.0.2.25‘ to ‘10.0.2.29‘
  - ‘Kar-Lab5-PC1‘ to ‘PC5‘: ‘10.0.2.30‘ to ‘10.0.2.34‘
  - ‘Kar-EdgeServer1‘: ‘10.0.2.5‘
- **Lahore**: ‘10.0.3.0/24‘
  - ‘Lahore-Controller‘ (G0/0): ‘10.0.3.1‘
  - ‘Lah-Lab1-PC1‘ to ‘PC5‘: ‘10.0.3.10‘ to ‘10.0.3.14‘
  - ‘Lah-Lab2-PC1‘ to ‘PC5‘: ‘10.0.3.15‘ to ‘10.0.3.19‘
  - ‘Lah-Lab3-PC1‘ to ‘PC5‘: ‘10.0.3.20‘ to ‘10.0.3.24‘
  - ‘Lah-Lab4-PC1‘ to ‘PC5‘: ‘10.0.3.25‘ to ‘10.0.3.29‘
  - ‘Lah-Lab5-PC1‘ to ‘PC5‘: ‘10.0.3.30‘ to ‘10.0.3.34‘
  - ‘Lah-EdgeServer1‘: ‘10.0.3.5‘
- **Faisalabad**: ‘10.0.4.0/24‘
  - ‘Faisalabad-Controller‘ (G0/0): ‘10.0.4.1‘

  - ‘Fsd-Lab1-PC1‘ to ‘PC5‘: ‘10.0.4.10‘ to ‘10.0.4.14‘
  - ‘Fsd-Lab2-PC1‘ to ‘PC5‘: ‘10.0.4.15‘ to ‘10.0.4.19‘
  - ‘Fsd-Lab3-PC1‘ to ‘PC5‘: ‘10.0.4.20‘ to ‘10.0.4.24‘
  - ‘Fsd-Lab4-PC1‘ to ‘PC5‘: ‘10.0.4.25‘ to ‘10.0.4.29‘
  - ‘Fsd-Lab5-PC1‘ to ‘PC5‘: ‘10.0.4.30‘ to ‘10.0.4.34‘
  - ‘Fsd-EdgeServer1‘: ‘10.0.4.5‘
- **Peshawar**: ‘10.0.5.0/24‘
  - ‘Peshawar-Controller‘ (G0/0): ‘10.0.5.1‘
  - ‘Pesh-Lab1-PC1‘ to ‘PC5‘: ‘10.0.5.10‘ to ‘10.0.5.14‘
  - ‘Pesh-Lab2-PC1‘ to ‘PC5‘: ‘10.0.5.15‘ to ‘10.0.5.19‘
  - ‘Pesh-Lab3-PC1‘ to ‘PC5‘: ‘10.0.5.20‘ to ‘10.0.5.24‘
  - ‘Pesh-Lab4-PC1‘ to ‘PC5‘: ‘10.0.5.25‘ to ‘10.0.5.29‘
  - ‘Pesh-Lab5-PC1‘ to ‘PC5‘: ‘10.0.5.30‘ to ‘10.0.5.34‘
  - ‘Pesh-EdgeServer1‘: ‘10.0.5.5‘
- **Multan**: ‘10.0.6.0/24‘
  - ‘Multan-Controller‘ (G0/0): ‘10.0.6.1‘
  - ‘Mul-Lab1-PC1‘ to ‘PC5‘: ‘10.0.6.10‘ to ‘10.0.6.14‘
  - ‘Mul-Lab2-PC1‘ to ‘PC5‘: ‘10.0.6.15‘ to ‘10.0.6.19‘
  - ‘Mul-Lab3-PC1‘ to ‘PC5‘: ‘10.0.6.20‘ to ‘10.0.6.24‘
  - ‘Mul-Lab4-PC1‘ to ‘PC5‘: ‘10.0.6.25‘ to ‘10.0.6.29‘
  - ‘Mul-Lab5-PC1‘ to ‘PC5‘: ‘10.0.6.30‘ to ‘10.0.6.34‘
  - ‘Mul-EdgeServer1‘: ‘10.0.6.5‘
- **Primary Backbone**: ‘10.0.10.0/24‘
  - ‘HeadOffice-Controller‘ (G0/1): ‘10.0.10.1‘
  - ‘Islamabad-Controller‘ (G0/1): ‘10.0.10.2‘
  - ‘Karachi-Controller‘ (G0/1): ‘10.0.10.3‘
  - ‘Lahore-Controller‘ (G0/1): ‘10.0.10.4‘
  - ‘Faisalabad-Controller‘ (G0/1): ‘10.0.10.5‘

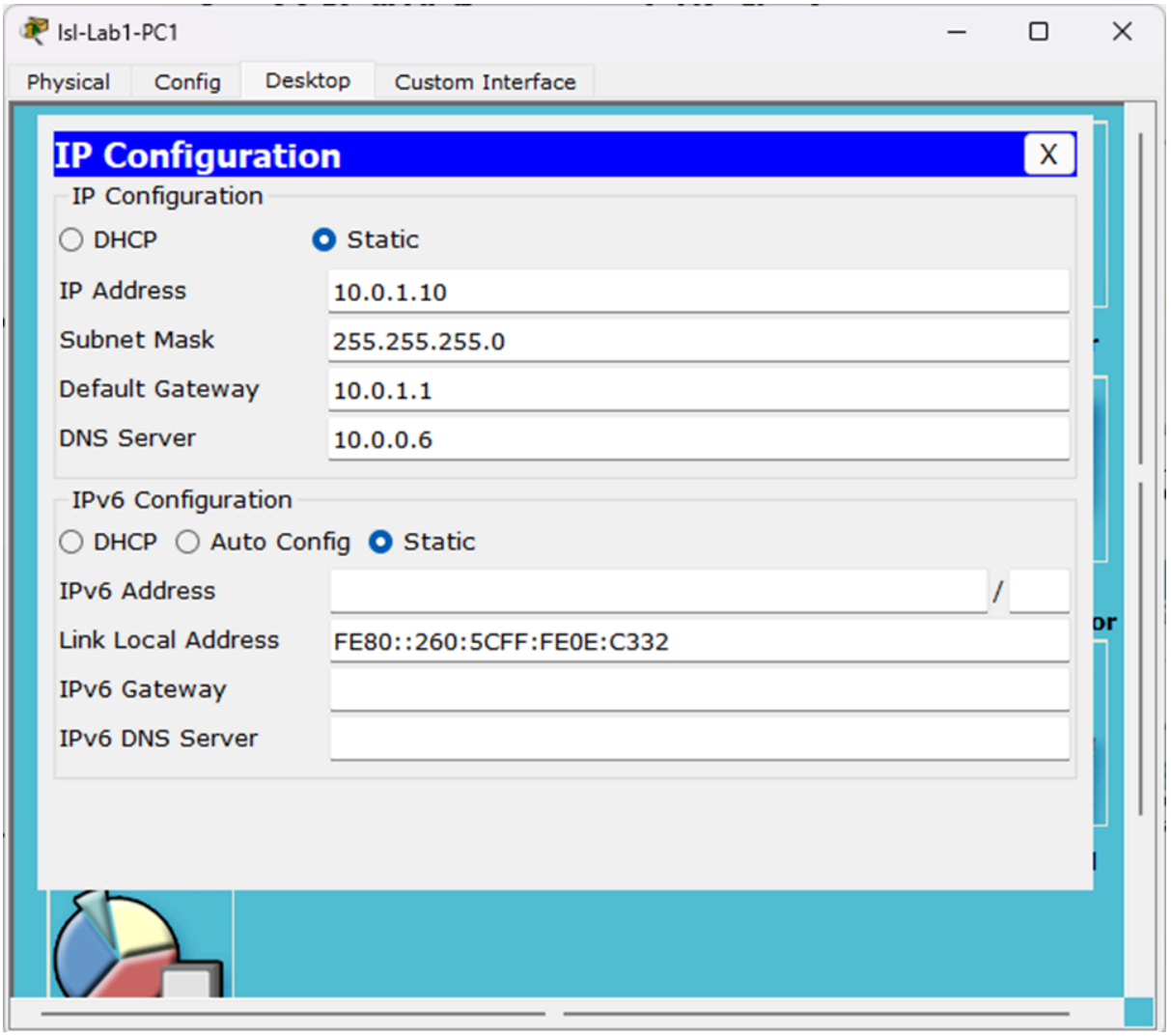
  - ‘Peshawar-Controller‘ (G0/1): ‘10.0.10.6‘
  - ‘Multan-Controller‘ (G0/1): ‘10.0.10.7‘
- **Backup Network**: ‘10.0.20.0/24‘
  - ‘Backup-Intermediary-Switch‘: ‘10.0.20.1‘
  - ‘HeadOffice-Controller‘ (G0/2): ‘10.0.20.2‘
  - ‘Islamabad-Controller‘ (G0/2): ‘10.0.20.3‘
  - ‘Karachi-Controller‘ (G0/2): ‘10.0.20.4‘
  - ‘Lahore-Controller‘ (G0/2): ‘10.0.20.5‘
  - ‘Faisalabad-Controller‘ (G0/2): ‘10.0.20.6‘
  - ‘Peshawar-Controller‘ (G0/2): ‘10.0.20.7‘
  - ‘Multan-Controller‘ (G0/2): ‘10.0.20.8‘

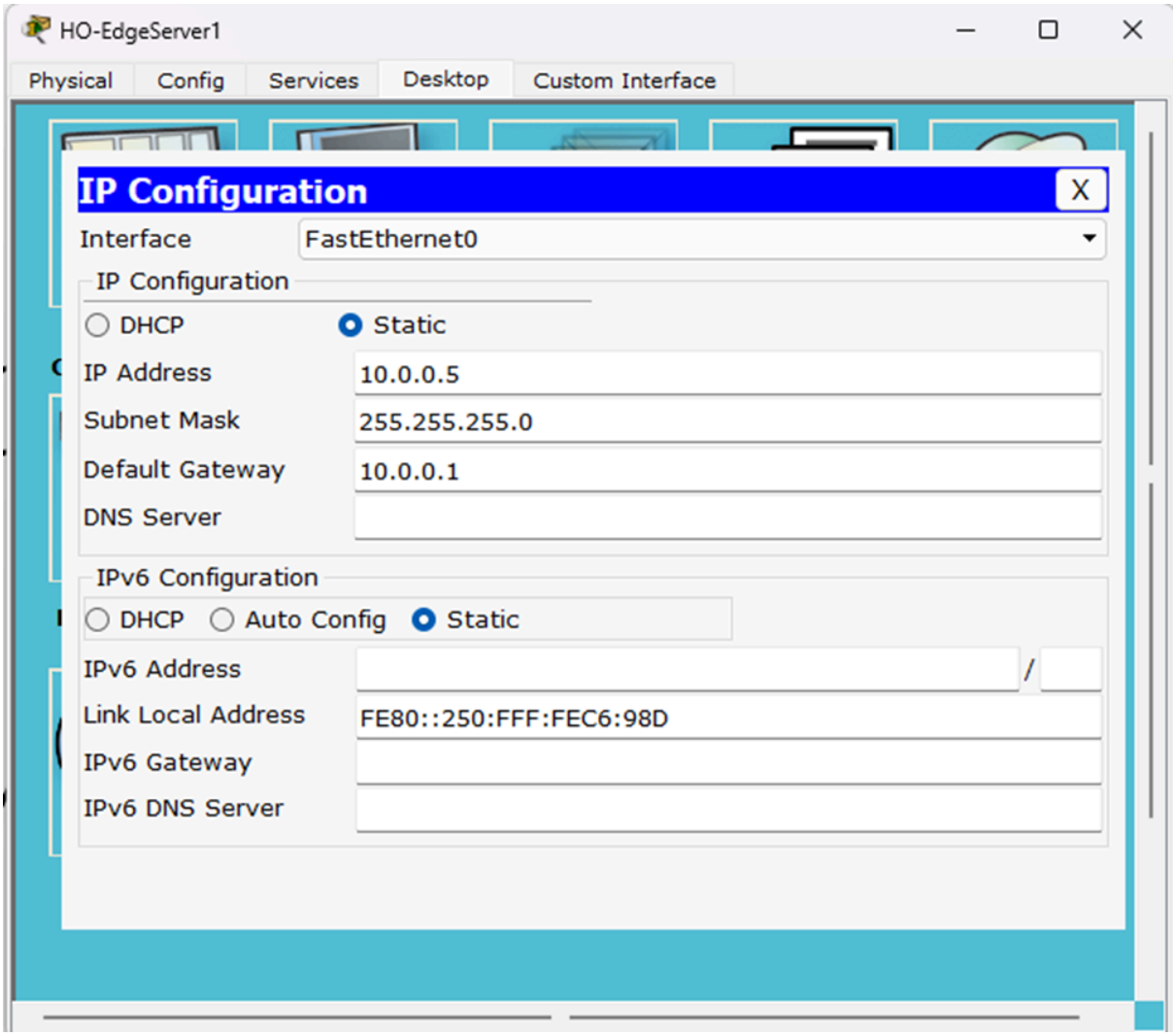
1.4. Planning for SDN Features

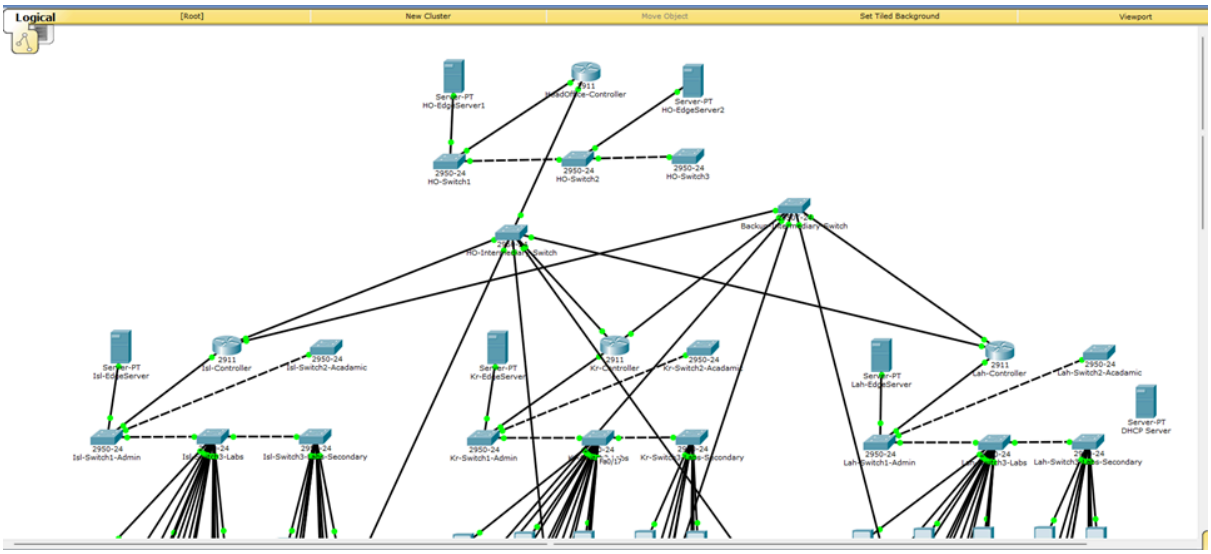
- **Centralized Control**: ‘HeadOffice-Controller’ acts as the central SDN controller, managing routing policies for all campuses.
- **Traffic Engineering**: Planned to use ACLs to prioritize server traffic and block unauthorized PC-to-PC traffic across campuses.
- **Load Balancing**: Planned to implement load balancing at the head office using a service IP, with traffic distributed across ‘HO-EdgeServer1’ and ‘HO-EdgeServer2’.

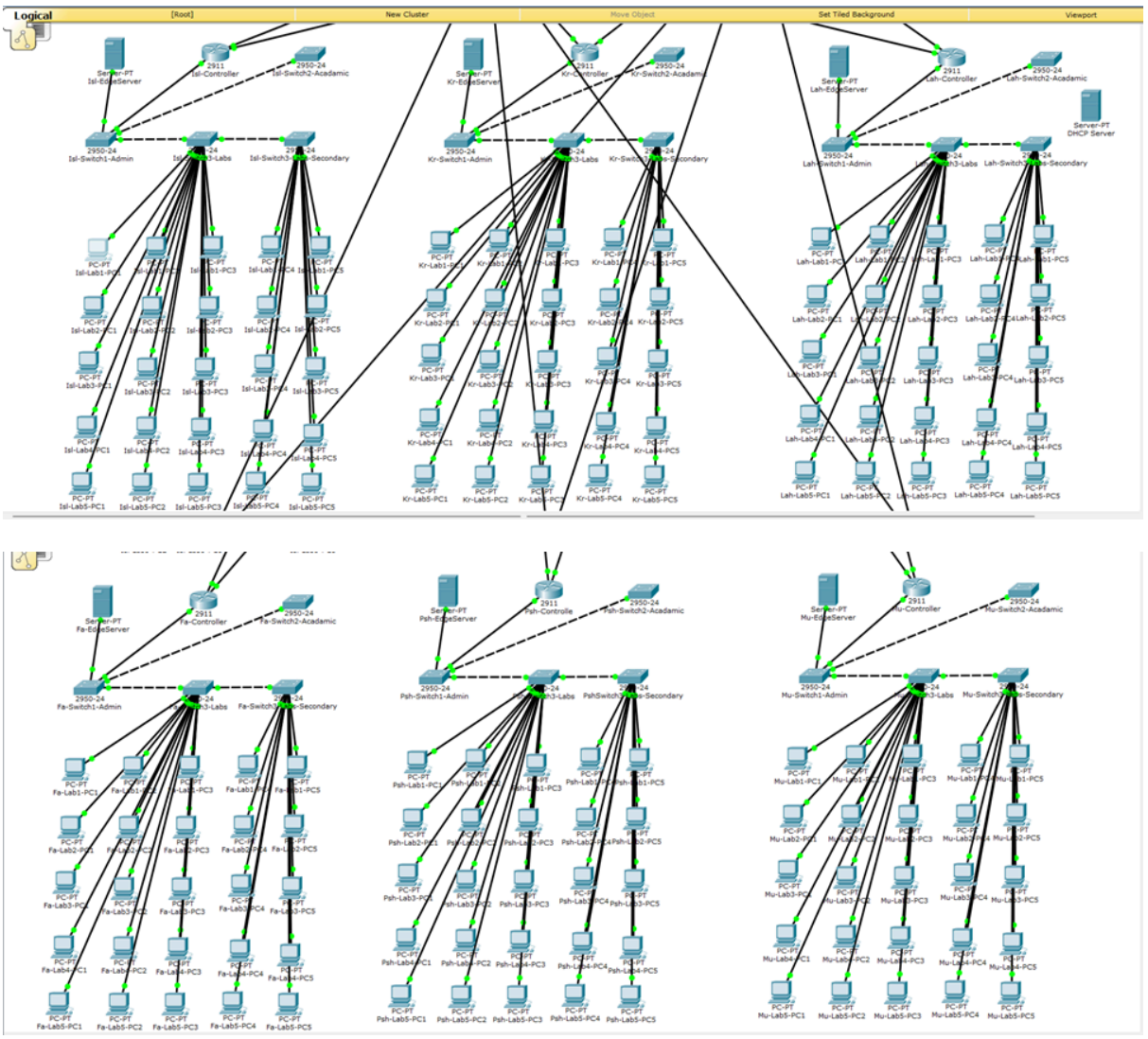
1.5. Challenges and Solutions

- **Challenge**: Ensuring the IP scheme supports future scalability.
- **Solution**: Used a /24 subnet for each campus, providing up to 254 addresses per campus, and reserved separate subnets for backbone and backup networks.


Isl-Lab1-PC1
Physical
Config
Desktop
Custom Interface
IP Configuration
X
IP Configuration
DHCP
Static
IP Address
10.0.1.10
Subnet Mask
255.255.255.0
Default Gateway
10.0.1.1
DNS Server
10.0.0.6
IPv6 Configuration
DHCP
Auto Config
Static
IPv6 Address
/
Link Local Address
FE80::260:5CFF:FE0E:C332
IPv6 Gateway
IPv6 DNS Server


HO-EdgeServer1
Physical
Config
Services
Desktop
Custom Interface
IP Configuration
X
Interface
FastEthernet0
IP Configuration
DHCP
Static
IP Address
10.0.0.5
Subnet Mask
255.255.255.0
Default Gateway
10.0.0.1
DNS Server
IPv6 Configuration
DHCP
Auto Config
Static
IPv6 Address
Link Local Address
FE80::250:FFF:FEC6:98D
IPv6 Gateway
IPv6 DNS Server


Logical
[Root]
New Cluster
Move Object
Set Tiled Background
Viewport



Part 2: Network Setup and Configuration

2.1. Objective

The goal of Part 2 is to set up the physical and logical network in Cisco Packet Tracer, configure devices with IP addresses, and establish basic connectivity within each campus.

2.2. Topology Setup

The topology was built in Cisco Packet Tracer as follows:

- **Head Office**:
  - Added ‘HeadOffice-Controller‘ (Cisco 2911 router).
  - Added ‘HO-Switch1‘, ‘HO-Switch2‘, ‘HO-Switch3‘ (2950 switches).

- Added ‘HO-EdgeServer1‘ and ‘HO-EdgeServer2‘.
- Connected ‘HeadOffice-Controller‘ (G0/0) to ‘HO-Switch1‘, which connects to ‘HO-Switch2‘ and ‘HO-Switch3‘.
- Connected servers to ‘HO-Switch2‘ and ‘HO-Switch3‘.

- **Campuses**:
  - For each campus (e.g., Islamabad):
  - Added ‘Islamabad-Controller‘ (Cisco 2911 router).
  - Added switches for each lab (e.g., ‘Isb-Lab1-Switch‘ to ‘Isb-Lab5-Switch‘).
  - Added 5 PCs per lab (e.g., ‘Isb-Lab1-PC1‘ to ‘PC5‘) and 1 edge server (‘Isb-EdgeServer1‘).
  - Connected PCs to their respective lab switches, switches to ‘Islamabad-Controller‘ (G0/0).
  - Repeated for Karachi, Lahore, Faisalabad, Peshawar, and Multan.

- **Backbone**:
  - Connected all controllers via their ‘GigabitEthernet0/1‘ interfaces to a central switch (‘BackboneSwitch‘) for the primary backbone (‘10.0.10.0/24‘).

2.3. Configuration

Devices were configured with IP addresses as per the Part 1 scheme:

- **HeadOffice-Controller**:

```
hostname HeadOffice-Controller
interface GigabitEthernet0/0
 ip address 10.0.0.1 255.255.255.0
 no shutdown
interface GigabitEthernet0/1
 ip address 10.0.10.1 255.255.255.0
 no shutdown
```

- **Islamabad-Controller**:

```
hostname Islamabad-Controller
interface GigabitEthernet0/0
 ip address 10.0.1.1 255.255.255.0
 no shutdown
interface GigabitEthernet0/1
 ip address 10.0.10.2 255.255.255.0
 no shutdown
```

- **Karachi-Controller**:

```
hostname Karachi-Controller
interface GigabitEthernet0/0
 ip address 10.0.2.1 255.255.255.0
 no shutdown
interface GigabitEthernet0/1
 ip address 10.0.10.3 255.255.255.0
 no shutdown
```

- **Lahore-Controller**:

```
hostname Lahore-Controller
interface GigabitEthernet0/0
 ip address 10.0.3.1 255.255.255.0
 no shutdown
interface GigabitEthernet0/1
 ip address 10.0.10.4 255.255.255.0
 no shutdown
```

- **Faisalabad-Controller**:

```
hostname Faisalabad-Controller
interface GigabitEthernet0/0
 ip address 10.0.4.1 255.255.255.0
 no shutdown
interface GigabitEthernet0/1
 ip address 10.0.10.5 255.255.255.0
 no shutdown
```

- **Peshawar-Controller**:

```
hostname Peshawar-Controller
interface GigabitEthernet0/0
 ip address 10.0.5.1 255.255.255.0
 no shutdown
interface GigabitEthernet0/1
 ip address 10.0.10.6 255.255.255.0
 no shutdown
```

- **Multan-Controller**:

```
hostname Multan-Controller
interface GigabitEthernet0/0
 ip address 10.0.6.1 255.255.255.0
 no shutdown
interface GigabitEthernet0/1
 ip address 10.0.10.7 255.255.255.0
 no shutdown
```

- **PCs and Servers**:
  - Configured via the Desktop tab in Packet Tracer.
  - Example for ‘Isb-Lab1-PC1‘:
    - IP: ‘10.0.1.10‘

∗ Subnet Mask: ‘255.255.255.0‘

∗ Default Gateway:

‘10.0.1.1‘ – Example for

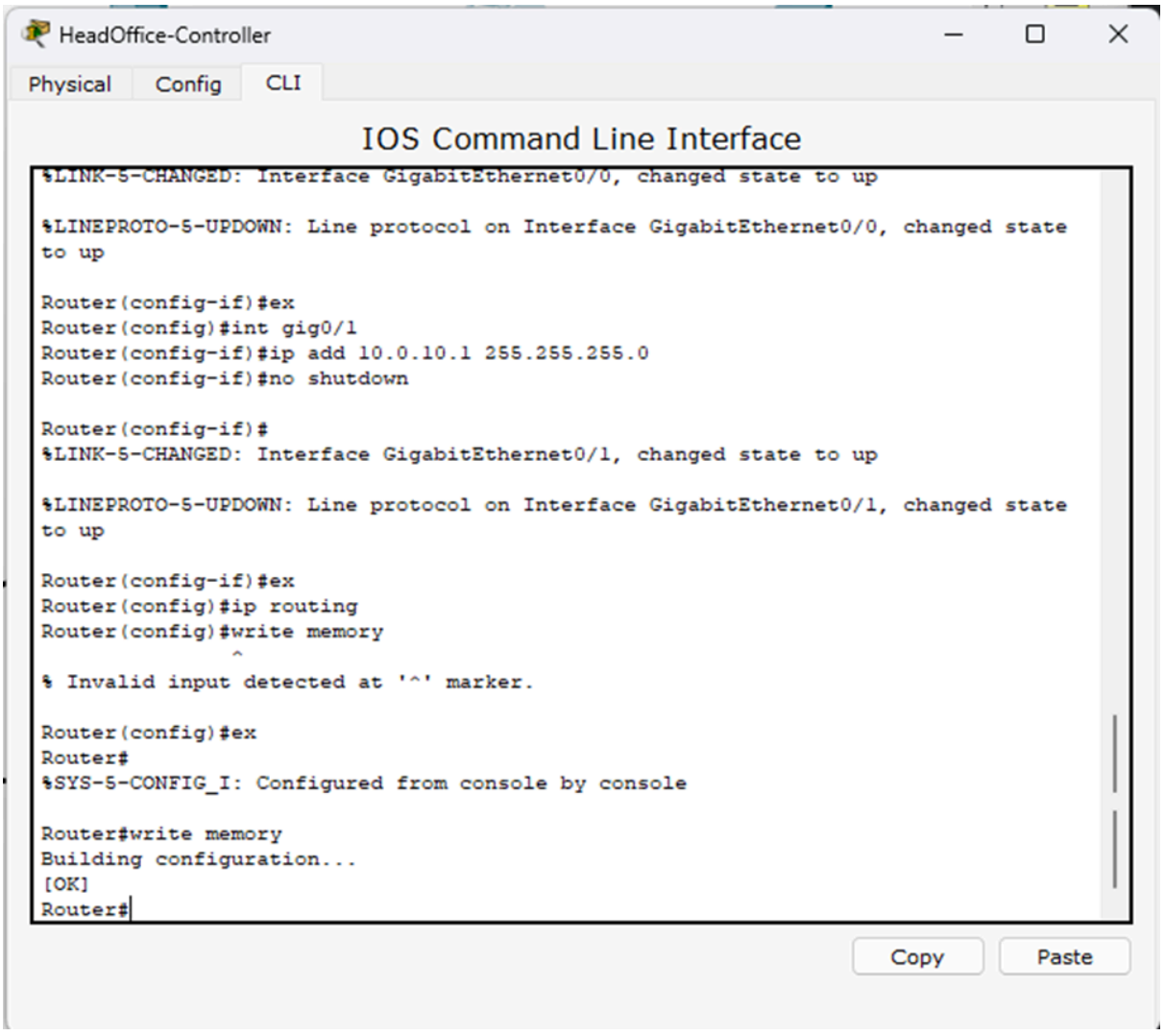
‘HO-EdgeServer1‘:

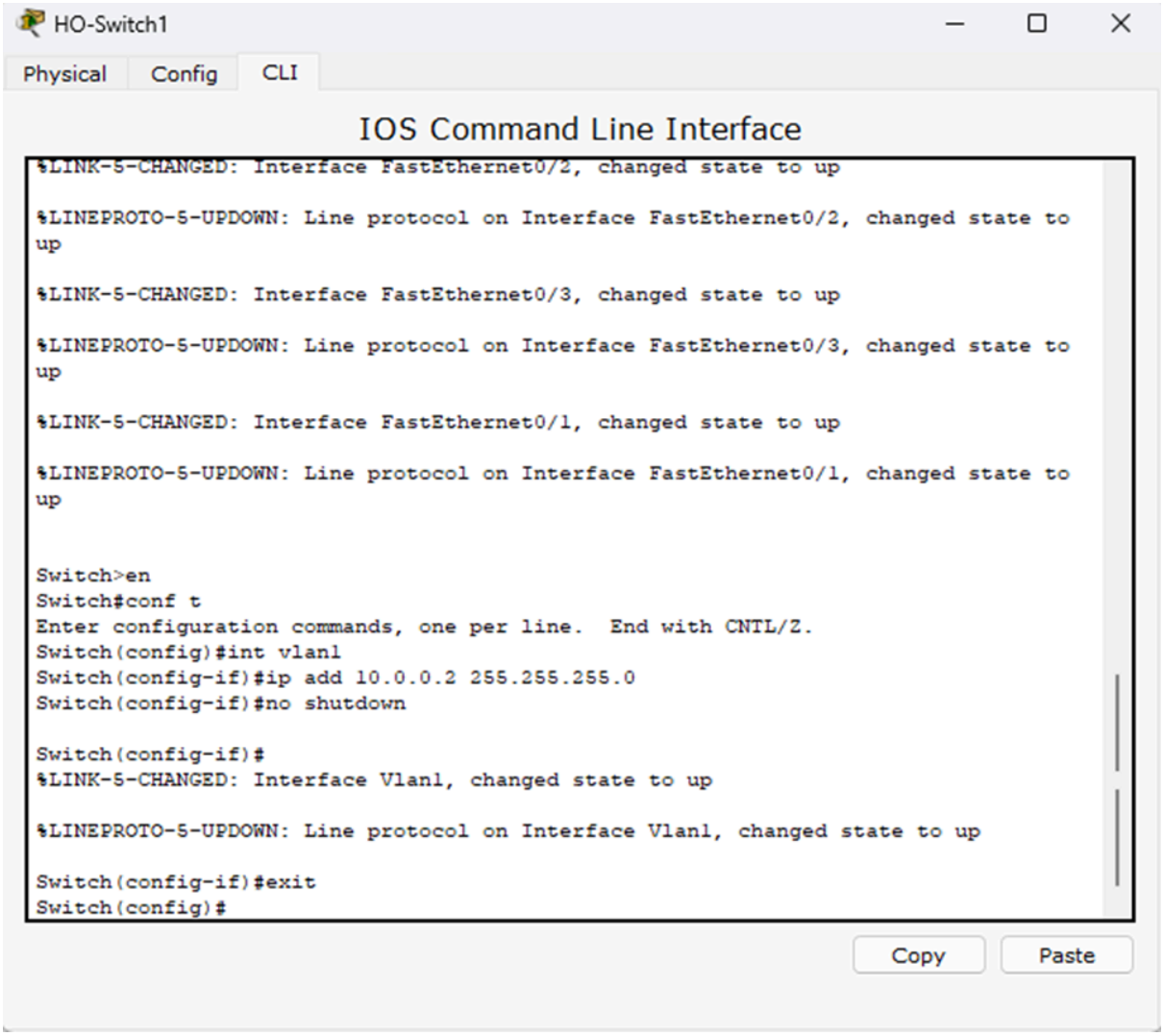
∗ IP: ‘10.0.0.5‘

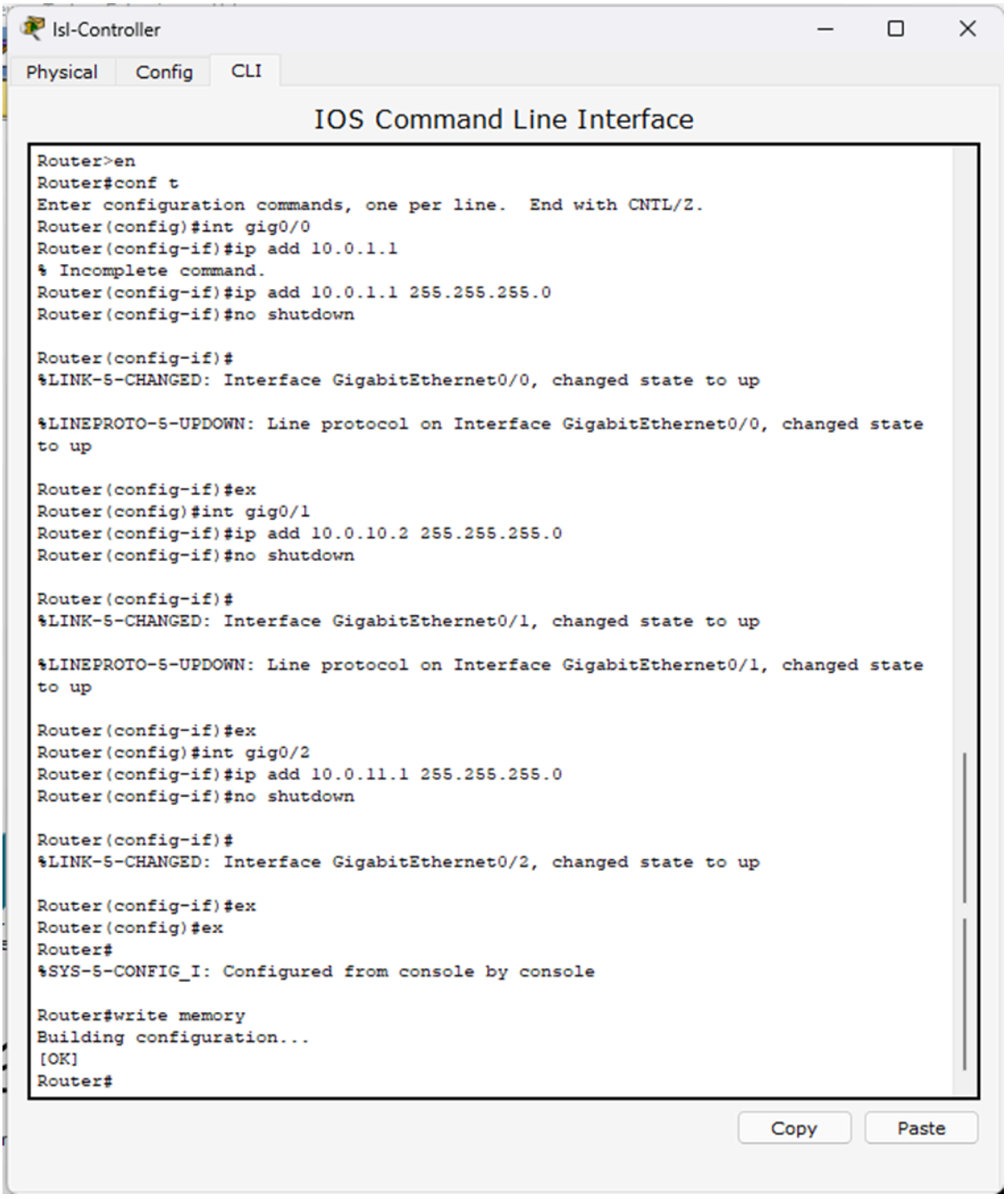
∗ Subnet Mask: ‘255.255.255.0‘

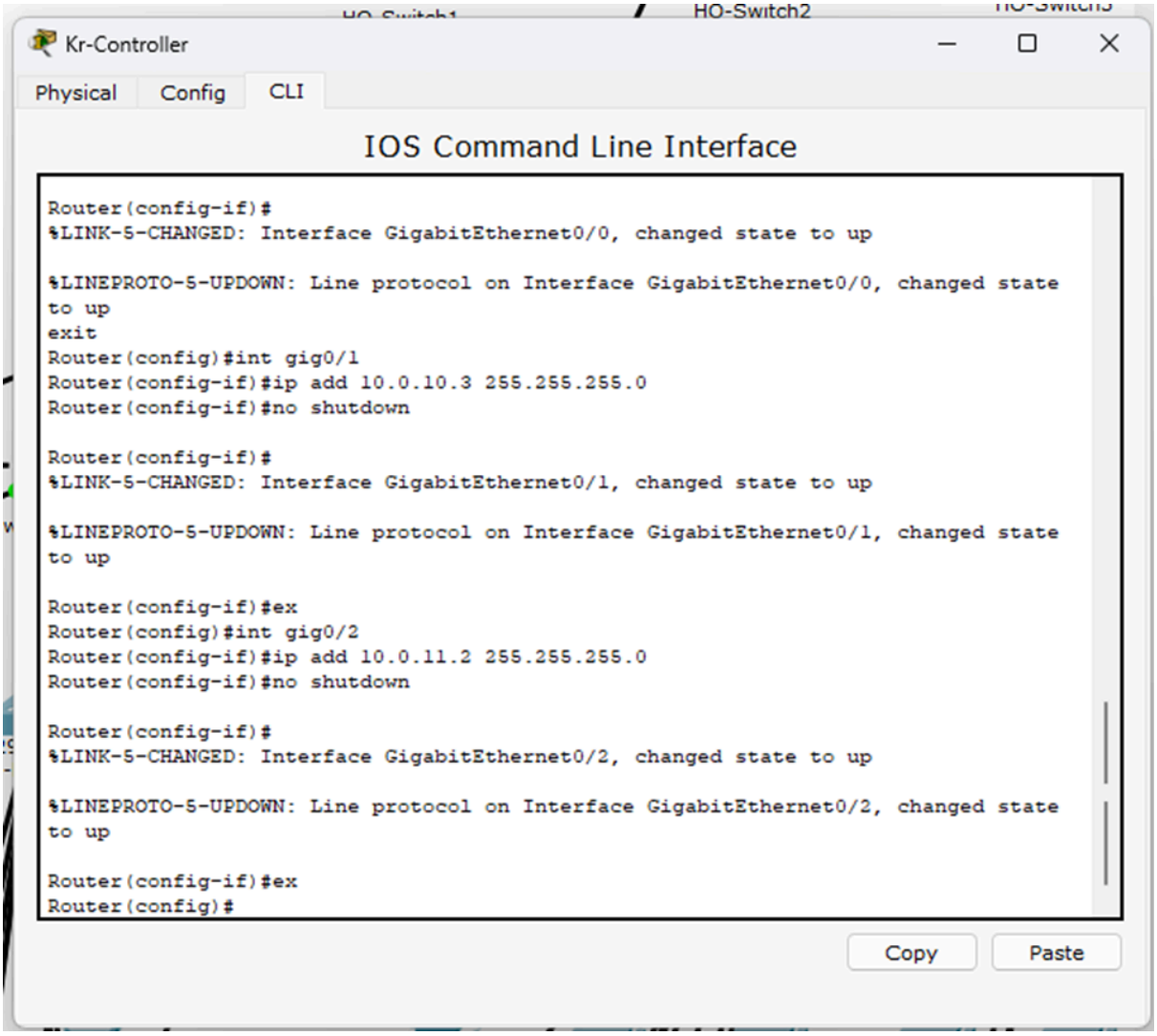
∗ Default Gateway: ‘10.0.0.1‘

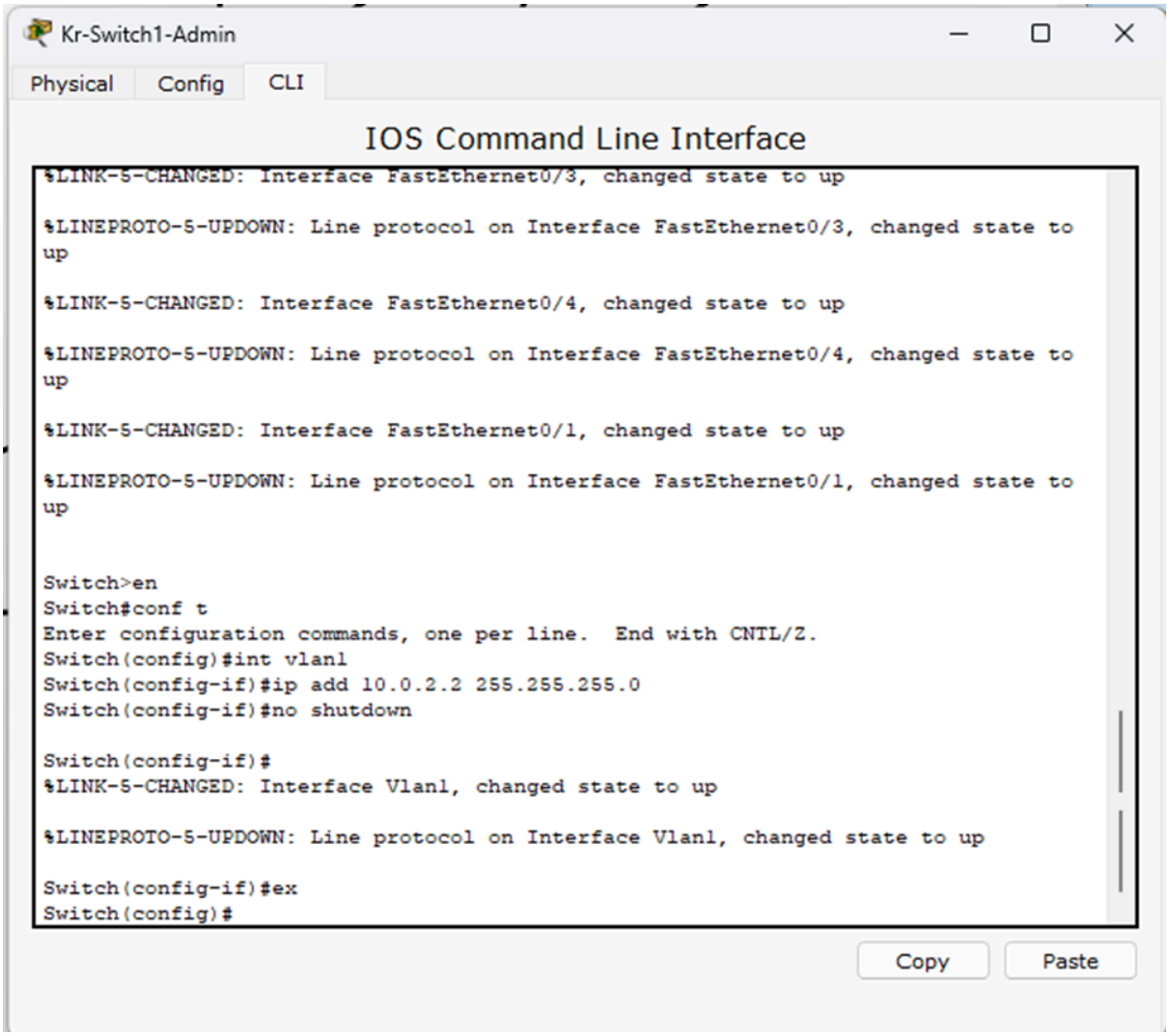
– Repeated for all PCs and servers across campuses.



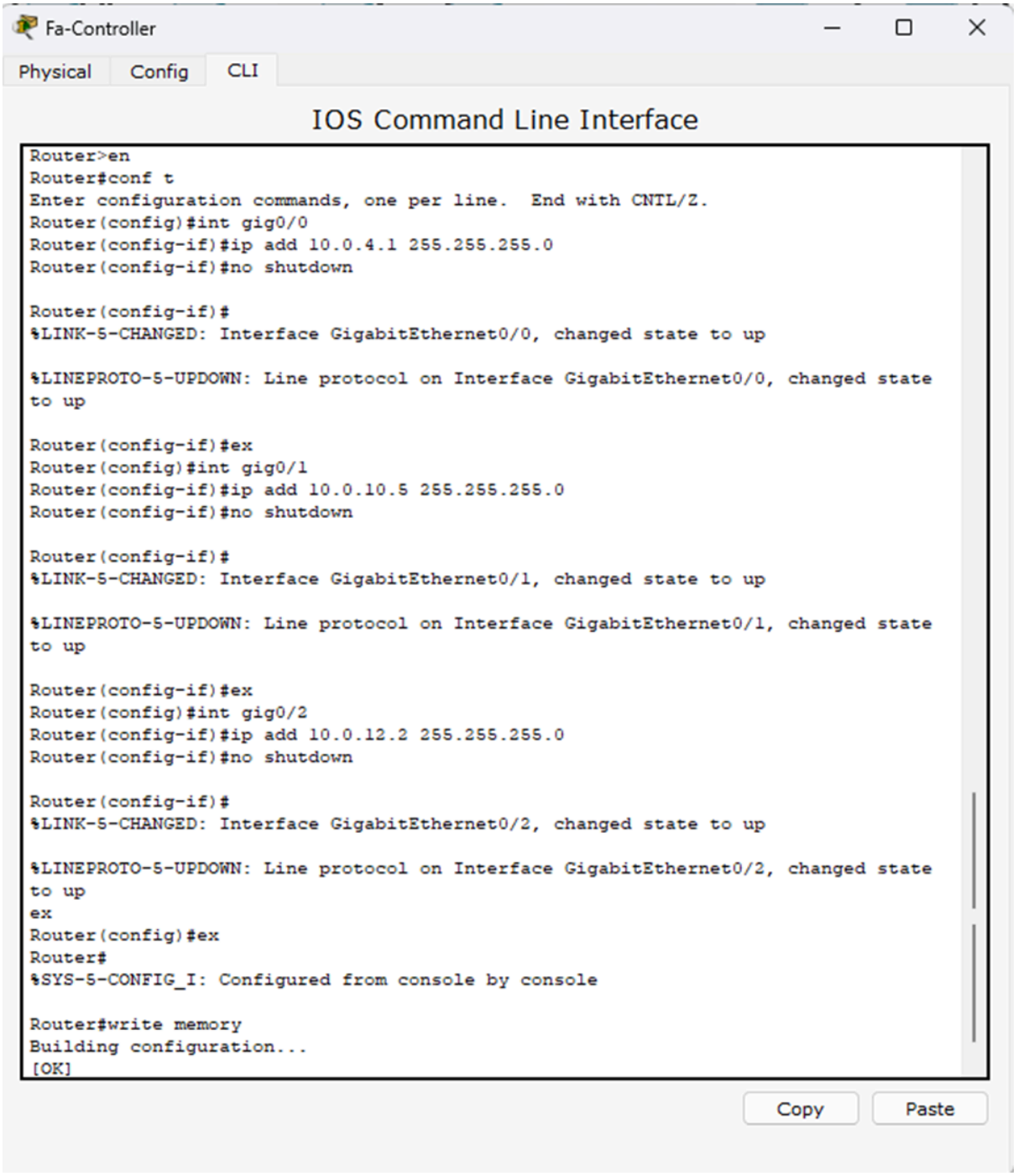

HO-Switch1
Physical
Config
CLI
IOS Command Line Interface
%LINK-5-CHANGED: Interface FastEthernet0/2, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/2, changed state to up
%LINK-5-CHANGED: Interface FastEthernet0/3, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/3, changed state to up
%LINK-5-CHANGED: Interface FastEthernet0/1, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/1, changed state to up
Switch>en
Switch#conf t
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)#int vlan1
Switch(config-if)#ip add 10.0.0.2 255.255.255.0
Switch(config-if)#no shutdown
Switch(config-if)#
%LINK-5-CHANGED: Interface Vlan1, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface Vlan1, changed state to up
Switch(config-if)#exit
Switch(config)#
Copy
Paste

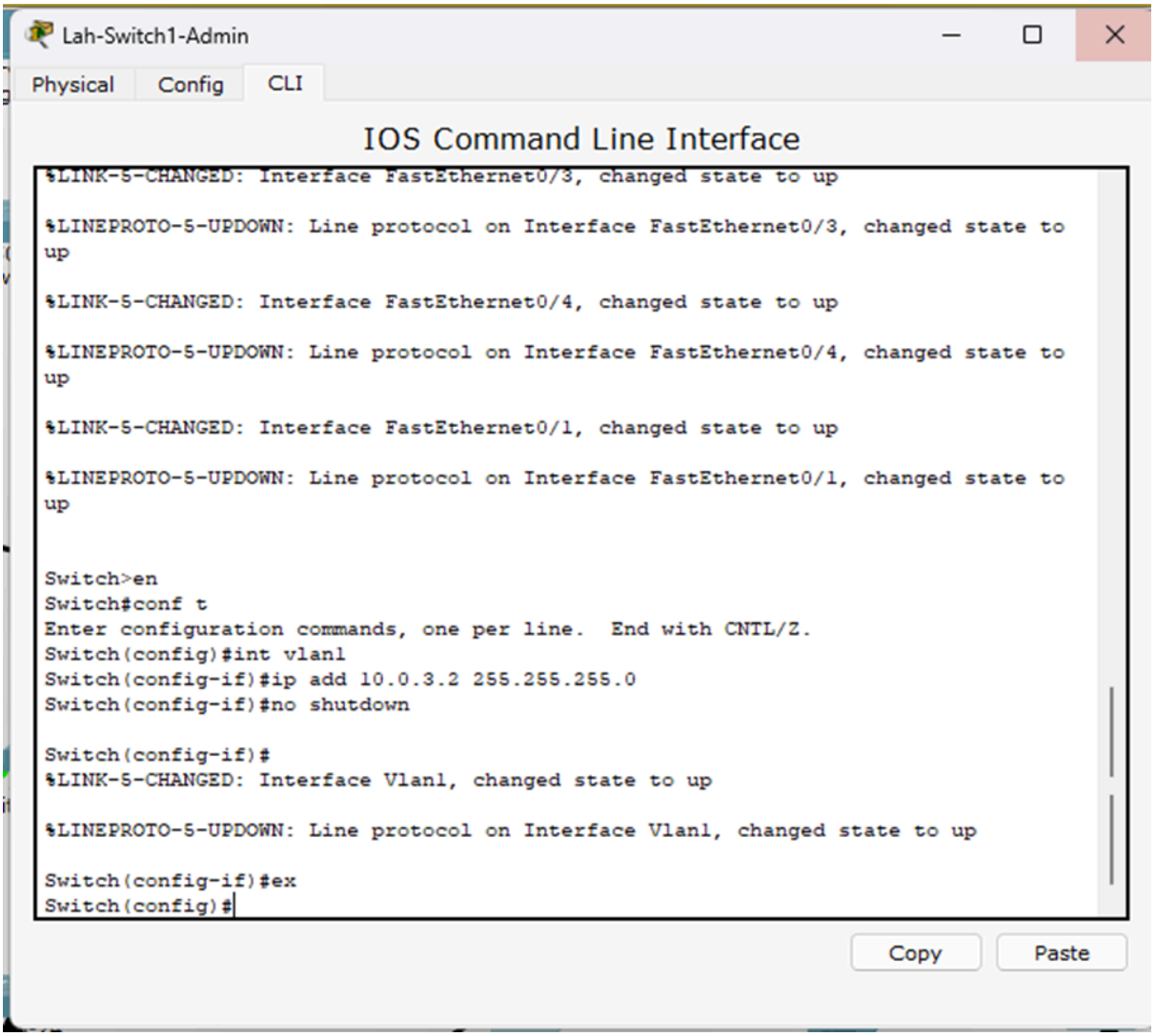

Isl-Controller
Physical
Config
CLI
IOS Command Line Interface
Router>en
Router#conf t
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#int gig0/0
Router(config-if)#ip add 10.0.1.1
% Incomplete command.
Router(config-if)#ip add 10.0.1.1 255.255.255.0
Router(config-if)#no shutdown
Router(config-if)#
%LINK-5-CHANGED: Interface GigabitEthernet0/0, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/0, changed state to up
Router(config-if)#ex
Router(config)#int gig0/1
Router(config-if)#ip add 10.0.10.2 255.255.255.0
Router(config-if)#no shutdown
Router(config-if)#
%LINK-5-CHANGED: Interface GigabitEthernet0/1, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/1, changed state to up
Router(config-if)#ex
Router(config)#int gig0/2
Router(config-if)#ip add 10.0.11.1 255.255.255.0
Router(config-if)#no shutdown
Router(config-if)#
%LINK-5-CHANGED: Interface GigabitEthernet0/2, changed state to up
Router(config-if)#ex
Router(config)#ex
Router#
%SYS-5-CONFIG_I: Configured from console by console
Router#write memory
Building configuration...
[OK]
Router#
Copy
Paste


HO-Switch2
Kr-Controller
Physical
Config
CLI
IOS Command Line Interface
Router(config-if)#
%LINK-5-CHANGED: Interface GigabitEthernet0/0, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/0, changed state to up
exit
Router(config)#int gig0/1
Router(config-if)#ip add 10.0.10.3 255.255.255.0
Router(config-if)#no shutdown
Router(config-if)#
%LINK-5-CHANGED: Interface GigabitEthernet0/1, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/1, changed state to up
Router(config-if)#ex
Router(config)#int gig0/2
Router(config-if)#ip add 10.0.11.2 255.255.255.0
Router(config-if)#no shutdown
Router(config-if)#
%LINK-5-CHANGED: Interface GigabitEthernet0/2, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/2, changed state to up
Router(config-if)#ex
Router(config)#
Copy
Paste


Kr-Switch1-Admin
Physical
Config
CLI
IOS Command Line Interface
%LINK-5-CHANGED: Interface FastEthernet0/3, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/3, changed state to up
%LINK-5-CHANGED: Interface FastEthernet0/4, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/4, changed state to up
%LINK-5-CHANGED: Interface FastEthernet0/1, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/1, changed state to up
Switch>en
Switch#conf t
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)#int vlan1
Switch(config-if)#ip add 10.0.2.2 255.255.255.0
Switch(config-if)#no shutdown
Switch(config-if)#
%LINK-5-CHANGED: Interface Vlan1, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface Vlan1, changed state to up
Switch(config-if)#ex
Switch(config)#
Copy
Paste


Lah-Controller
Physical
Config
CLI
IOS Command Line Interface
Router>en
Router#config t
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#int gig0/0
Router(config-if)#ip add 10.0.3.1 255.255.255.0
Router(config-if)#no shutdown
Router(config-if)#
%LINK-5-CHANGED: Interface GigabitEthernet0/0, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/0, changed state to up
Router(config-if)#ex
Router(config)#int gig0/1
Router(config-if)#ip add 10.0.10.4 255.255.255.0
Router(config-if)#no shutdown
Router(config-if)#
%LINK-5-CHANGED: Interface GigabitEthernet0/1, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/1, changed state to up
Router(config-if)#ex
Router(config)#int gig0/2
Router(config-if)#ip add 10.0.12.1 255.255.255.0
Router(config-if)#no shutdown
Router(config-if)#
%LINK-5-CHANGED: Interface GigabitEthernet0/2, changed state to up
Router(config-if)#ex
Router(config)#ex
Router#
%SYS-5-CONFIG_I: Configured from console by console
Router#write memory
Building configuration...
[OK]
Router#
Copy
Paste


Fa-Controller
Physical
Config
CLI
IOS Command Line Interface
Router>en
Router#conf t
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#int gig0/0
Router(config-if)#ip add 10.0.4.1 255.255.255.0
Router(config-if)#no shutdown
Router(config-if)#
%LINK-5-CHANGED: Interface GigabitEthernet0/0, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/0, changed state to up
Router(config-if)#ex
Router(config)#int gig0/1
Router(config-if)#ip add 10.0.10.5 255.255.255.0
Router(config-if)#no shutdown
Router(config-if)#
%LINK-5-CHANGED: Interface GigabitEthernet0/1, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/1, changed state to up
Router(config-if)#ex
Router(config)#int gig0/2
Router(config-if)#ip add 10.0.12.2 255.255.255.0
Router(config-if)#no shutdown
Router(config-if)#
%LINK-5-CHANGED: Interface GigabitEthernet0/2, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/2, changed state to up
ex
Router(config)#ex
Router#
%SYS-5-CONFIG_I: Configured from console by console
Router#write memory
Building configuration...
[OK]
Copy
Paste



2.4. Testing

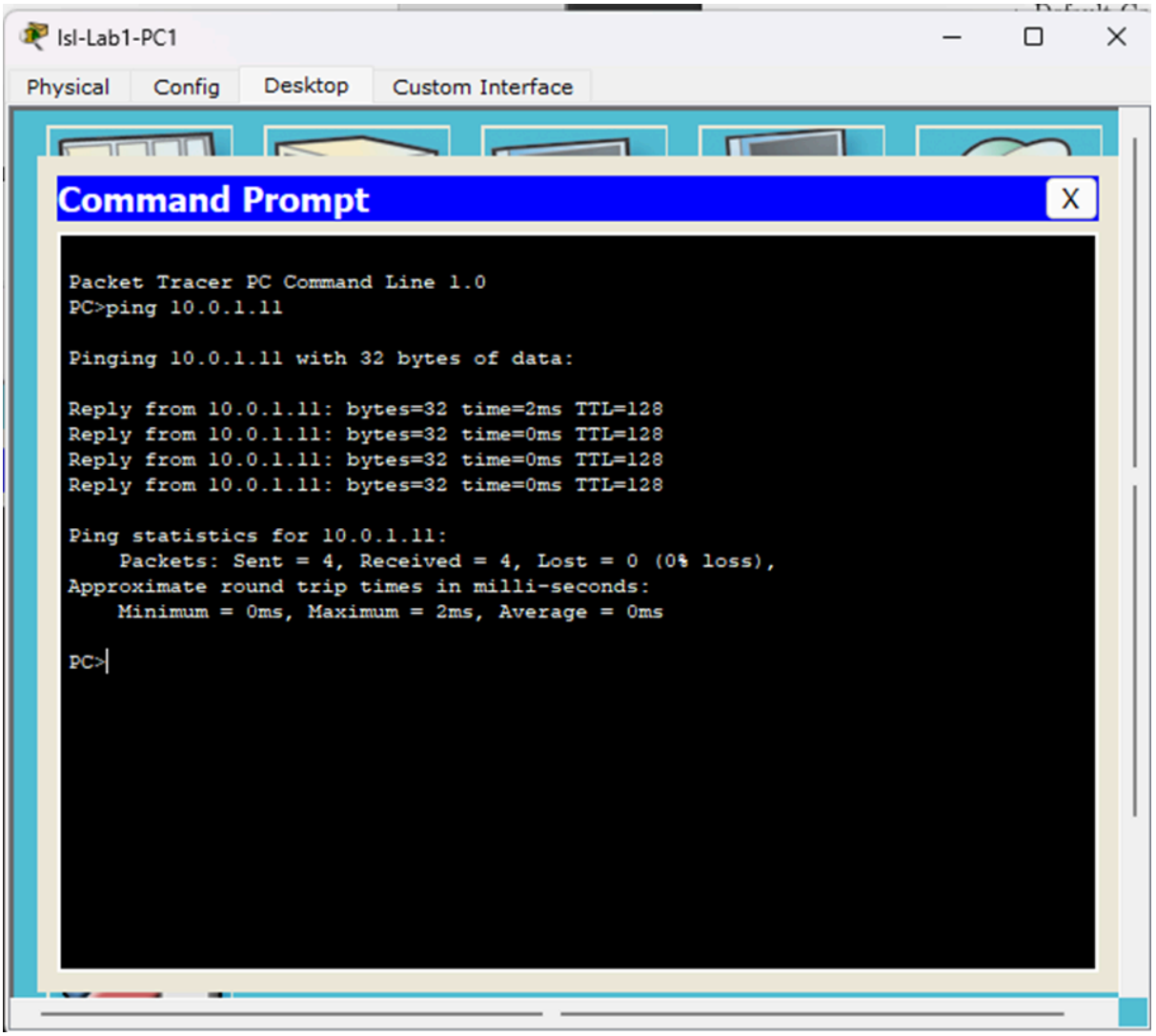
Basic connectivity tests were performed to verify the setup:

- **PC-to-PC within Campus**:
  - From ‘Isb-Lab1-PC1‘ (‘10.0.1.10‘) to ‘Isb-Lab1-PC2‘ (‘10.0.1.11‘):

```
Pinging 10.0.1.11 with 32 bytes of data:
Reply from 10.0.1.11: bytes=32 time=1ms TTL=128
Reply from 10.0.1.11: bytes=32 time=1ms TTL=128
Reply from 10.0.1.11: bytes=32 time=1ms TTL=128
Reply from 10.0.1.11: bytes=32 time=1ms TTL=128

Ping statistics for 10.0.1.11:
   Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
```

– Result: Success.

– Repeated for all campuses (e.g., ‘Kar-Lab1-PC1‘ to ‘Kar-Lab1-PC2‘).

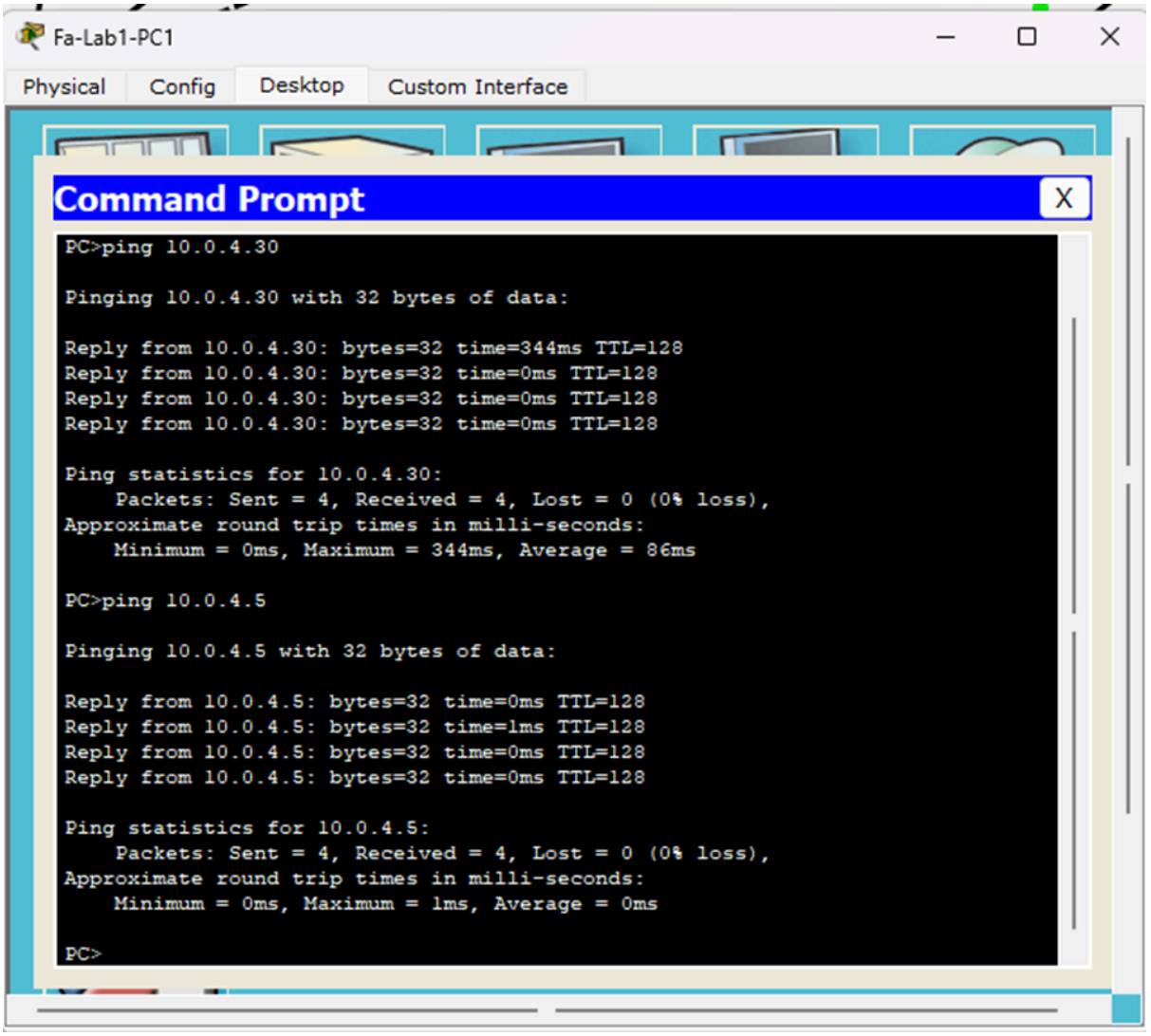


- **PC-to-PC across Campuses**:

  – From ‘Isb-Lab1-PC1‘ (‘10.0.1.10‘) to ‘Kar-Lab1-PC1‘ (‘10.0.2.10‘):

```
Pinging 10.0.2.10 with 32 bytes of data:
Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 10.0.2.10:
   Packets: Sent = 4, Received = 0, Lost = 4 (100%
   loss),
```

- Result: Failed (expected, as inter-subnet routing is not yet configured).



- **Simulation Mode**:
  - Used Simulation mode to observe packet flow within campuses.
  - Confirmed ARP requests and ICMP packets within the same subnet.

2.5. Challenges and Solutions

- **Challenge**: PCs initially failed to ping within the same subnet.
- **Solution**: Ensured correct default gateway settings on PCs (e.g., ‘10.0.1.1‘ for Islamabad PCs) and verified switch connectivity.

- **Challenge**: Interface status showed ‘down‘ on some controllers.
- **Solution**: Used ‘no shutdown‘ command on all interfaces to bring them up.

Part 3: Implementing SDN Controller Logic

3.1. Objective

The goal of Part 3 is to simulate SDN controller logic using network-layer policies (static routing) to enable inter-subnet connectivity across campuses and establish a backup network for redundancy.

3.2. Topology Updates

The topology was updated to include the backup network:

- **Backup Network** (‘10.0.20.0/24‘):
  - Added ‘Backup-Intermediary-Switch‘ (2950 switch).
  - Connected all controllers to ‘Backup-Intermediary-Switch‘ via their ‘GigabitEthernet0/2‘ interfaces.

3.3. Configuration

- **Enable IP Routing on All Controllers**:
  - Enabled routing to allow inter-subnet communication.
  - Example on ‘HeadOffice-Controller‘:

```
ip routing
```

  - Repeated on all Campus-Controllers.
- **Configure Backup Network Interfaces**:
  - On ‘HeadOffice-Controller‘:

```
interface GigabitEthernet0/2
 ip address 10.0.20.2 255.255.255.0
 no shutdown
```

  - On ‘Islamabad-Controller‘:

```
interface GigabitEthernet0/2
 ip address 10.0.20.3 255.255.255.0
 no shutdown
```

- On ‘Karachi-Controller‘:

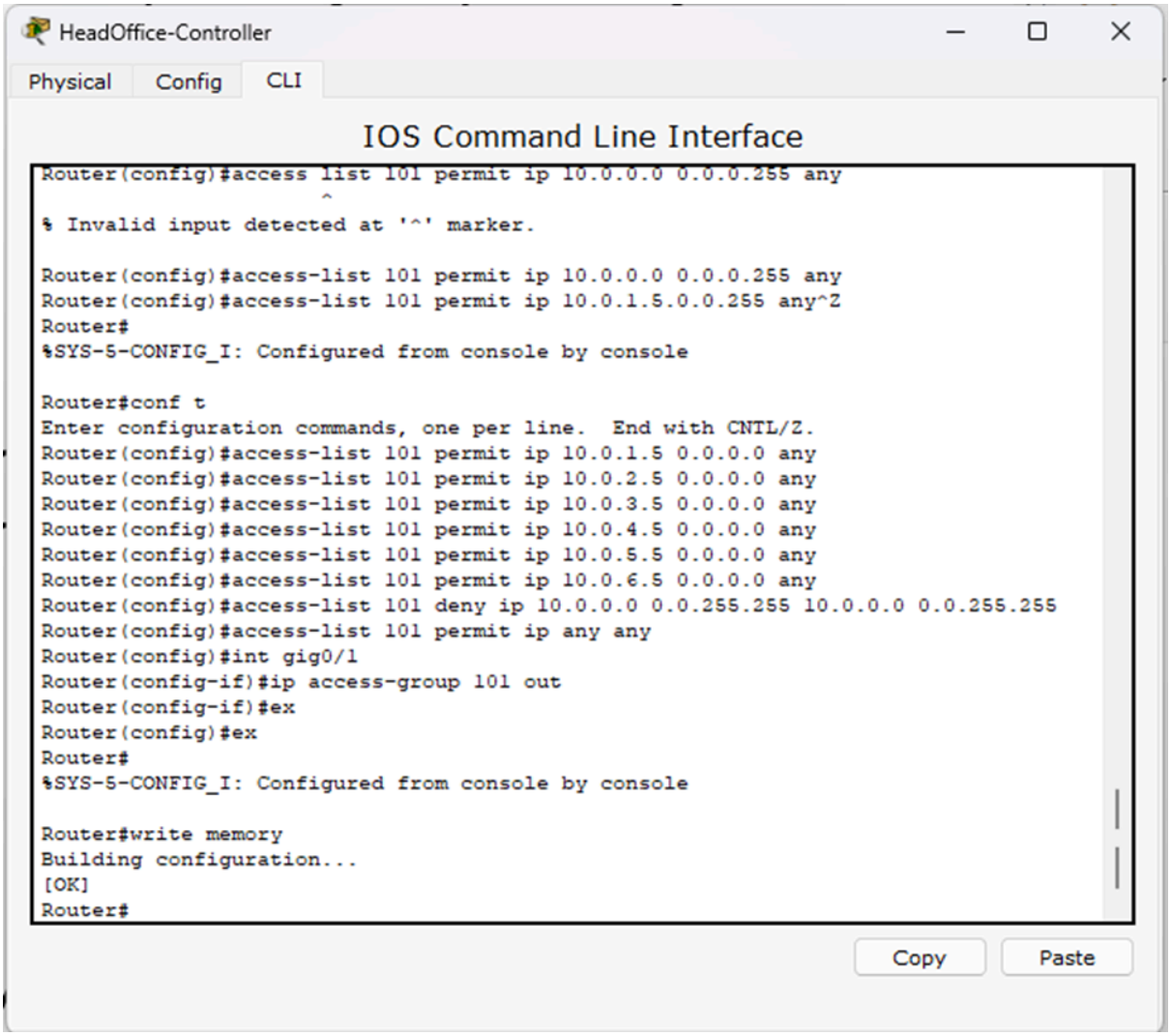
```
interface GigabitEthernet0/2
 ip address 10.0.20.4 255.255.255.0
 no shutdown
```

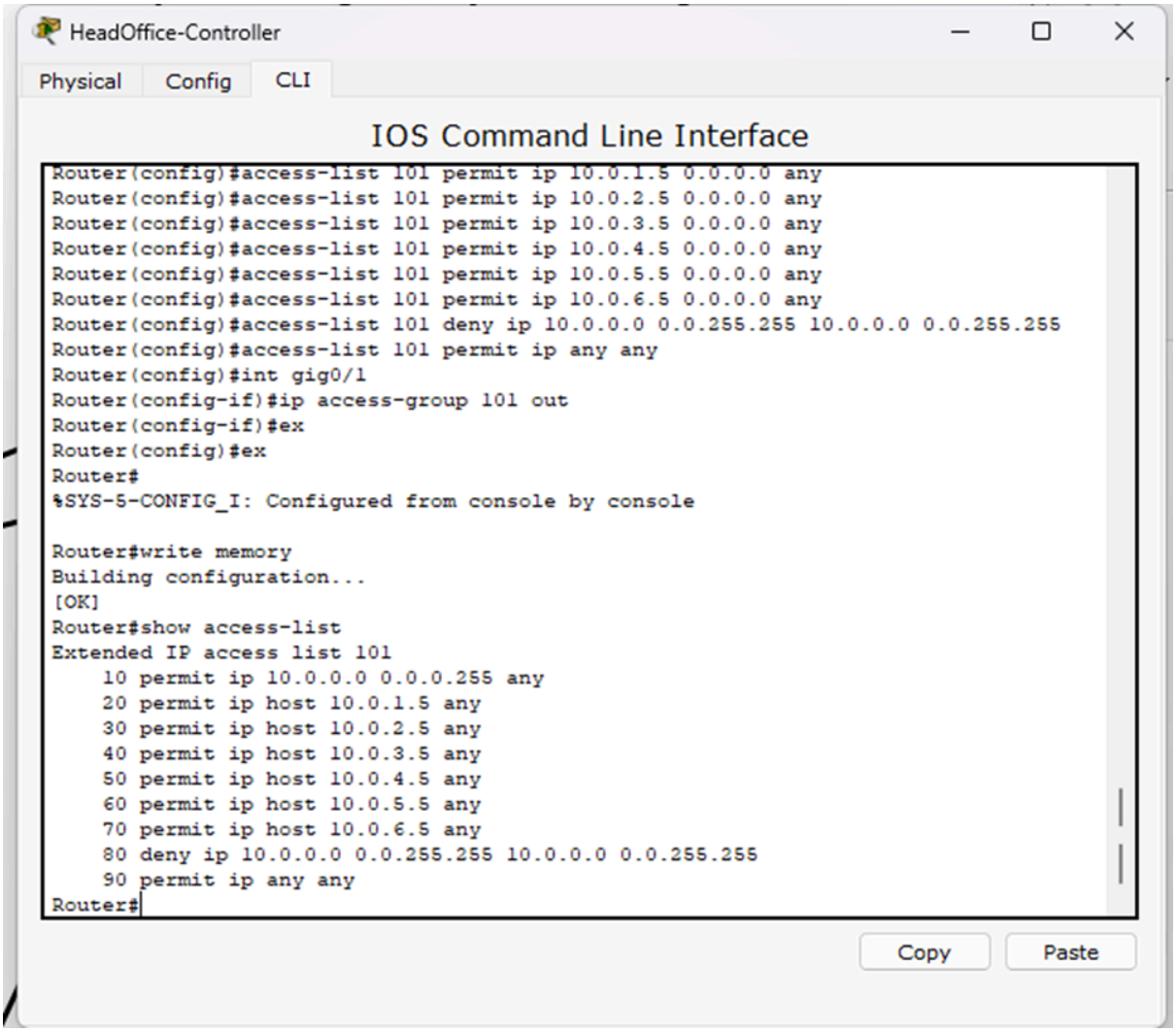
- On ‘Lahore-Controller‘:

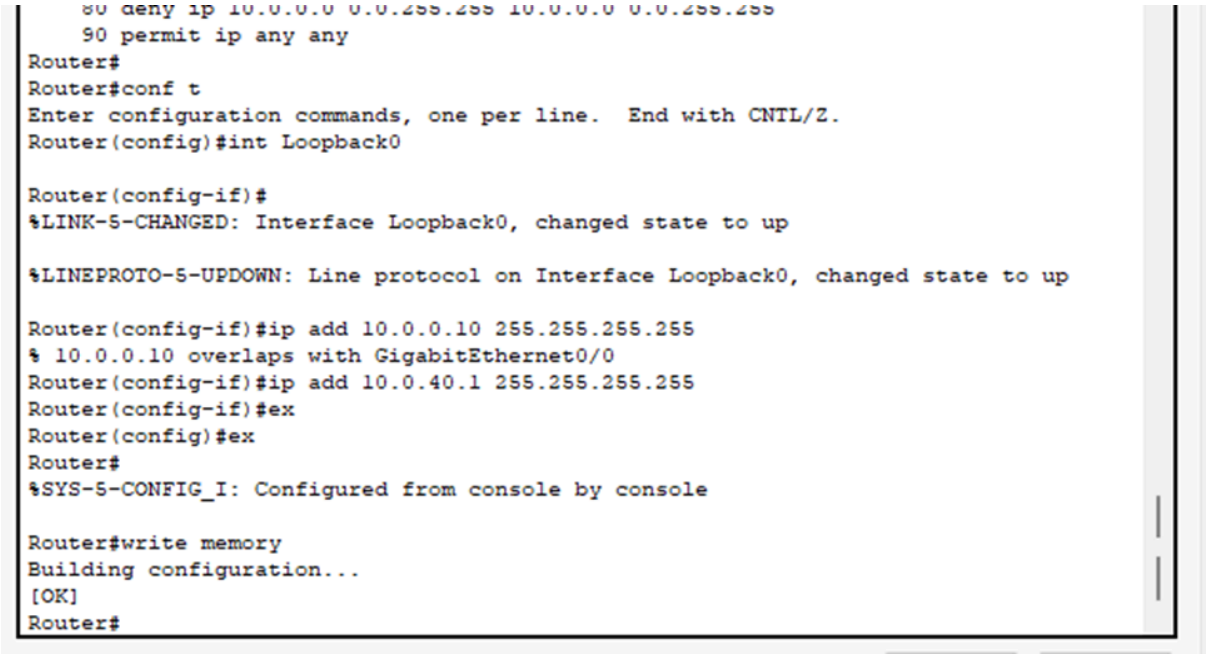
```
interface GigabitEthernet0/2
 ip address 10.0.20.5 255.255.255.0
 no shutdown
```

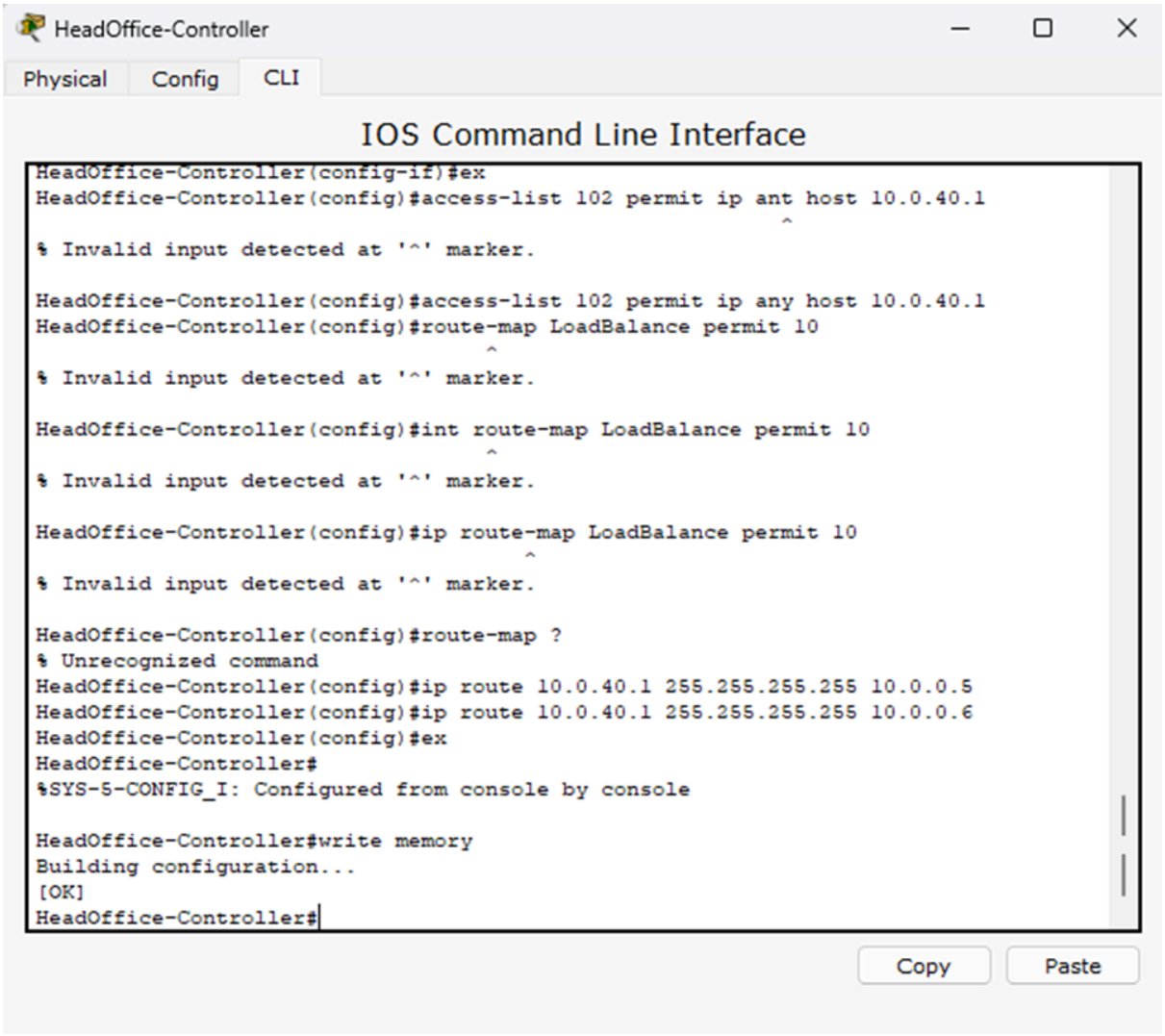
- On ‘Faisalabad-Controller‘:
- On ‘Peshawar-Controller‘:
- On ‘Multan-Controller‘:
- On ‘Backup-Intermediary-Switch‘:
  - Configured IP via Desktop tab: ‘10.0.20.1‘, Subnet Mask: ‘255.255.255.0‘.

- **Static Routes on HeadOffice-Controller**:
  - Configured routes to all campus subnets via the primary backbone.
- **Static Routes on Campus-Controllers**:
  - Example on ‘Karachi-Controller‘:
  - Repeated on all other Campus-Controllers, adjusting for their respective subnets.


HeadOffice-Controller
Physical
Config
CLI
IOS Command Line Interface
Router(config)#access list 101 permit ip 10.0.0.0 0.0.0.255 any
^
% Invalid input detected at '^' marker.
Router(config)#access-list 101 permit ip 10.0.0.0 0.0.0.255 any
Router(config)#access-list 101 permit ip 10.0.1.5.0.0.255 any^Z
Router#
%SYS-5-CONFIG_I: Configured from console by console
Router#conf t
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#access-list 101 permit ip 10.0.1.5 0.0.0.0 any
Router(config)#access-list 101 permit ip 10.0.2.5 0.0.0.0 any
Router(config)#access-list 101 permit ip 10.0.3.5 0.0.0.0 any
Router(config)#access-list 101 permit ip 10.0.4.5 0.0.0.0 any
Router(config)#access-list 101 permit ip 10.0.5.5 0.0.0.0 any
Router(config)#access-list 101 permit ip 10.0.6.5 0.0.0.0 any
Router(config)#access-list 101 deny ip 10.0.0.0 0.0.255.255 10.0.0.0 0.0.255.255
Router(config)#access-list 101 permit ip any any
Router(config)#int gig0/1
Router(config-if)#ip access-group 101 out
Router(config-if)#ex
Router(config)#ex
Router#
%SYS-5-CONFIG_I: Configured from console by console
Router#write memory
Building configuration...
[OK]
Router#
Copy
Paste


HeadOffice-Controller
Physical
Config
CLI
IOS Command Line Interface
Router(config)#access-list 101 permit ip 10.0.1.5 0.0.0.0 any
Router(config)#access-list 101 permit ip 10.0.2.5 0.0.0.0 any
Router(config)#access-list 101 permit ip 10.0.3.5 0.0.0.0 any
Router(config)#access-list 101 permit ip 10.0.4.5 0.0.0.0 any
Router(config)#access-list 101 permit ip 10.0.5.5 0.0.0.0 any
Router(config)#access-list 101 permit ip 10.0.6.5 0.0.0.0 any
Router(config)#access-list 101 deny ip 10.0.0.0 0.0.255.255 10.0.0.0 0.0.255.255
Router(config)#access-list 101 permit ip any any
Router(config)#int gig0/1
Router(config-if)#ip access-group 101 out
Router(config-if)#ex
Router(config)#ex
Router#
%SYS-5-CONFIG_I: Configured from console by console
Router#write memory
Building configuration...
[OK]
Router#show access-list
Extended IP access list 101
10 permit ip 10.0.0.0 0.0.0.255 any
20 permit ip host 10.0.1.5 any
30 permit ip host 10.0.2.5 any
40 permit ip host 10.0.3.5 any
50 permit ip host 10.0.4.5 any
60 permit ip host 10.0.5.5 any
70 permit ip host 10.0.6.5 any
80 deny ip 10.0.0.0 0.0.255.255 10.0.0.0 0.0.255.255
90 permit ip any any
Router#
Copy
Paste


90 permit ip any any
Router#
Router#conf t
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#int Loopback0
Router(config-if)#
%LINK-5-CHANGED: Interface Loopback0, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback0, changed state to up
Router(config-if)#ip add 10.0.0.10 255.255.255.255
% 10.0.0.10 overlaps with GigabitEthernet0/0
Router(config-if)#ip add 10.0.40.1 255.255.255.255
Router(config-if)#ex
Router(config)#ex
Router#
%SYS-5-CONFIG_I: Configured from console by console
Router#write memory
Building configuration...
[OK]
Router#



3.4. Testing

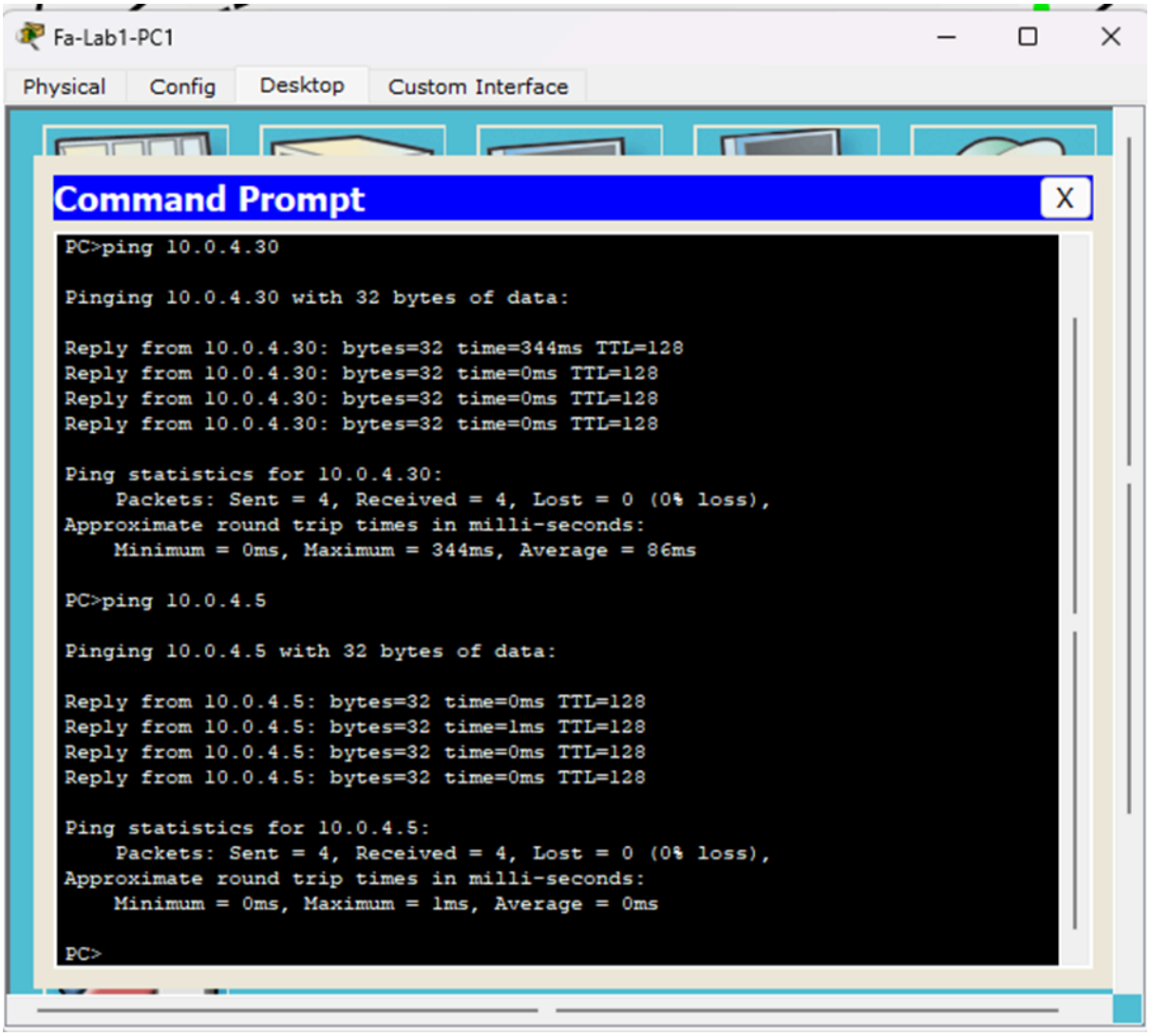
Comprehensive tests were performed to verify inter-subnet connectivity and backup network functionality:

- **PC-to-PC across Campuses**:
  - From ‘Isb-Lab1-PC1‘ (‘10.0.1.10‘) to ‘Fa-Lab5-PC1‘ (‘10.0.4.30‘):

```
Pinging 10.0.2.10 with 32 bytes of data:
Reply from 10.0.2.10: bytes=32 time=2ms TTL=126
Reply from 10.0.2.10: bytes=32 time=2ms TTL=126
Reply from 10.0.2.10: bytes=32 time=2ms TTL=126
Reply from 10.0.2.10: bytes=32 time=2ms TTL=126

Ping statistics for 10.0.2.10:
   Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
```

– Result: Success.

– Repeated for other campus pairs (e.g., ‘10.0.1.10‘ to ‘10.0.6.10‘).

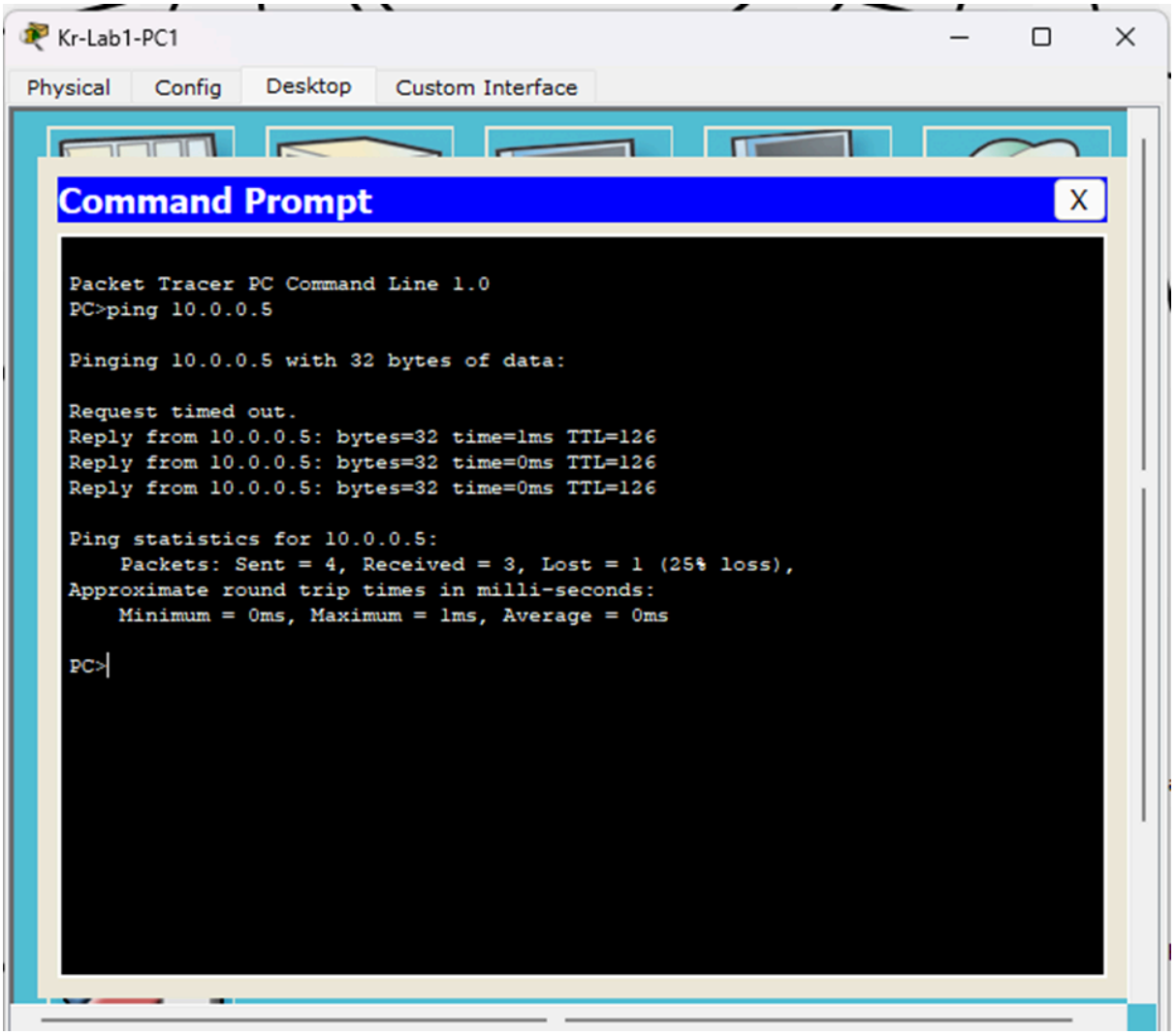


- **PC-to-Server across Campuses**:

  – From ‘Kar-Lab1-PC1‘ (‘10.0.2.10‘) to ‘HO-EdgeServer1‘ (‘10.0.0.5‘):

```
Pinging 10.0.0.5 with 32 bytes of data:
Reply from 10.0.0.5: bytes=32 time=2ms TTL=126
Reply from 10.0.0.5: bytes=32 time=2ms TTL=126
Reply from 10.0.0.5: bytes=32 time=2ms TTL=126
Reply from 10.0.0.5: bytes=32 time=2ms TTL=126

Ping statistics for 10.0.0.5:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
```

– Result: Success.

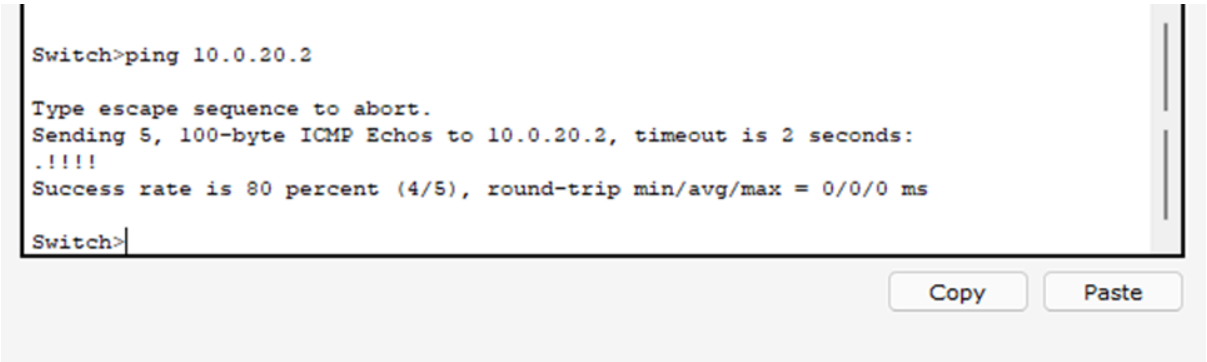


- **Backup Network Test**:
  – From ‘Backup-Intermediary-Switch‘ (‘10.0.20.1‘) to ‘HeadOffice-Controller‘ (G0/2, ‘10.0.20.2‘):

```
Pinging 10.0.20.2 with 32 bytes of data:
Reply from 10.0.20.2: bytes=32 time=1ms TTL=128
Reply from 10.0.20.2: bytes=32 time=1ms TTL=128
Reply from 10.0.20.2: bytes=32 time=1ms TTL=128
Reply from 10.0.20.2: bytes=32 time=1ms TTL=128
```

```
Ping statistics for 10.0.20.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
```

- Result: Success.



- Repeated for other controllers (e.g., ‘10.0.20.1‘ to ‘10.0.20.3‘).

- **Simulation Mode**:
  - Filtered for ICMP packets in Simulation mode.
  - Observed packet flow from ‘10.0.1.10‘ to ‘10.0.2.10‘, confirming the path: ‘Islamabad-Controller‘ → ‘HeadOffice-Controller‘ → ‘Karachi-Controller‘.
  - Verified backup network connectivity by simulating link failure on the primary backbone (manually disabling ‘GigabitEthernet0/1‘ interfaces) and testing via ‘10.0.20.0/24‘.

3.5. Challenges and Solutions

- **Challenge**: Initial inter-subnet pings failed due to missing routes.
- **Solution**: Added static routes on all controllers, ensuring each controller knows how to reach other subnets via ‘HeadOffice-Controller‘.
- **Challenge**: Backup network connectivity issues due to interface status.
- **Solution**: Ensured all ‘GigabitEthernet0/2‘ interfaces were up using ‘no shutdown‘ and verified IP assignments.

Part 4: Traffic Engineering and Load Balancing

4.1. Objective

The goal of Part 4 is to implement traffic engineering using ACLs to manage traffic flows and simulate load balancing across servers at the head office, ensuring efficient resource utilization.

4.2. Configuration

- **Traffic Engineering on HeadOffice-Controller**:
  - Configured an ACL to prioritize server traffic and block unauthorized PC-to-PC traffic across campuses.

```
access-list 101 permit ip 10.0.0.0 0.0.0.255 any
access-list 101 permit ip 10.0.1.5 0.0.0.0 any
access-list 101 permit ip 10.0.2.5 0.0.0.0 any
access-list 101 permit ip 10.0.3.5 0.0.0.0 any
access-list 101 permit ip 10.0.4.5 0.0.0.0 any
access-list 101 permit ip 10.0.5.5 0.0.0.0 any
access-list 101 permit ip 10.0.6.5 0.0.0.0 any
access-list 101 deny ip 10.0.0.0 0.0.255.255 10.0.0.0 0.0.255.255
access-list 101 permit ip any any
interface GigabitEthernet0/1
 ip access-group 101 out
```

  - **Explanation**:
    * Permits traffic from head office subnet (‘10.0.0.0/24‘) to any destination.
    * Permits traffic from each campus edge server (e.g., ‘10.0.1.5‘) to any destination.
    * Denies PC-to-PC traffic between campuses (e.g., ‘10.0.1.0/24‘ to ‘10.0.2.0/24‘).
    * Permits all other traffic.
    * Applied on ‘GigabitEthernet0/1‘ (backbone interface) in the outbound direction.

- **Load Balancing on HeadOffice-Controller**:
  - Configured a loopback interface as the service IP and used static routes for load balancing.

```
interface Loopback0
 ip address 10.0.40.1 255.255.255.255 ip
route 10.0.40.1 255.255.255.255 10.0.0.5 ip
route 10.0.40.1 255.255.255.255 10.0.0.6
```

- **Explanation**:
  - ‘10.0.40.1‘ acts as a virtual service IP for clients to access head office servers.
  - Two equal-cost static routes distribute traffic to ‘HO-EdgeServer1‘ (‘10.0.0.5‘) and ‘HOEdgeServer2‘ (‘10.0.0.6‘).

- **Routes on Campus-Controllers**:
  - Added routes to ensure all campuses can reach the service IP ‘10.0.40.1‘.
  - Example on ‘Karachi-Controller‘:

```
ip route 10.0.40.1 255.255.255.255 10.0.10.1
```

  - On ‘Islamabad-Controller‘:

```
ip route 10.0.40.1 255.255.255.255 10.0.10.1
```

  - On ‘Lahore-Controller‘:

```
ip route 10.0.40.1 255.255.255.255 10.0.10.1
```

  - On ‘Faisalabad-Controller‘:

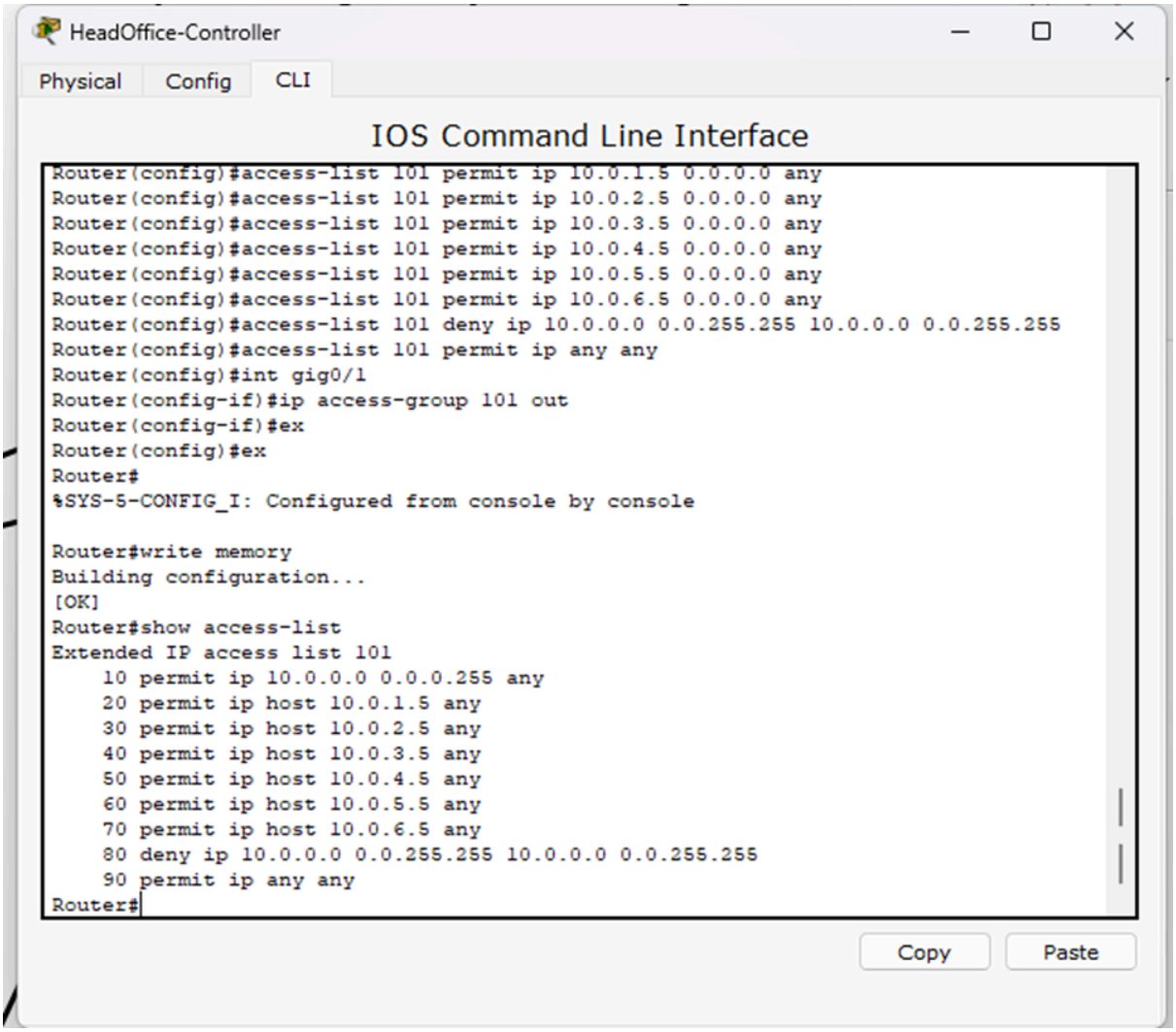
```
ip route 10.0.40.1 255.255.255.255 10.0.10.1
```

  - On ‘Peshawar-Controller‘:

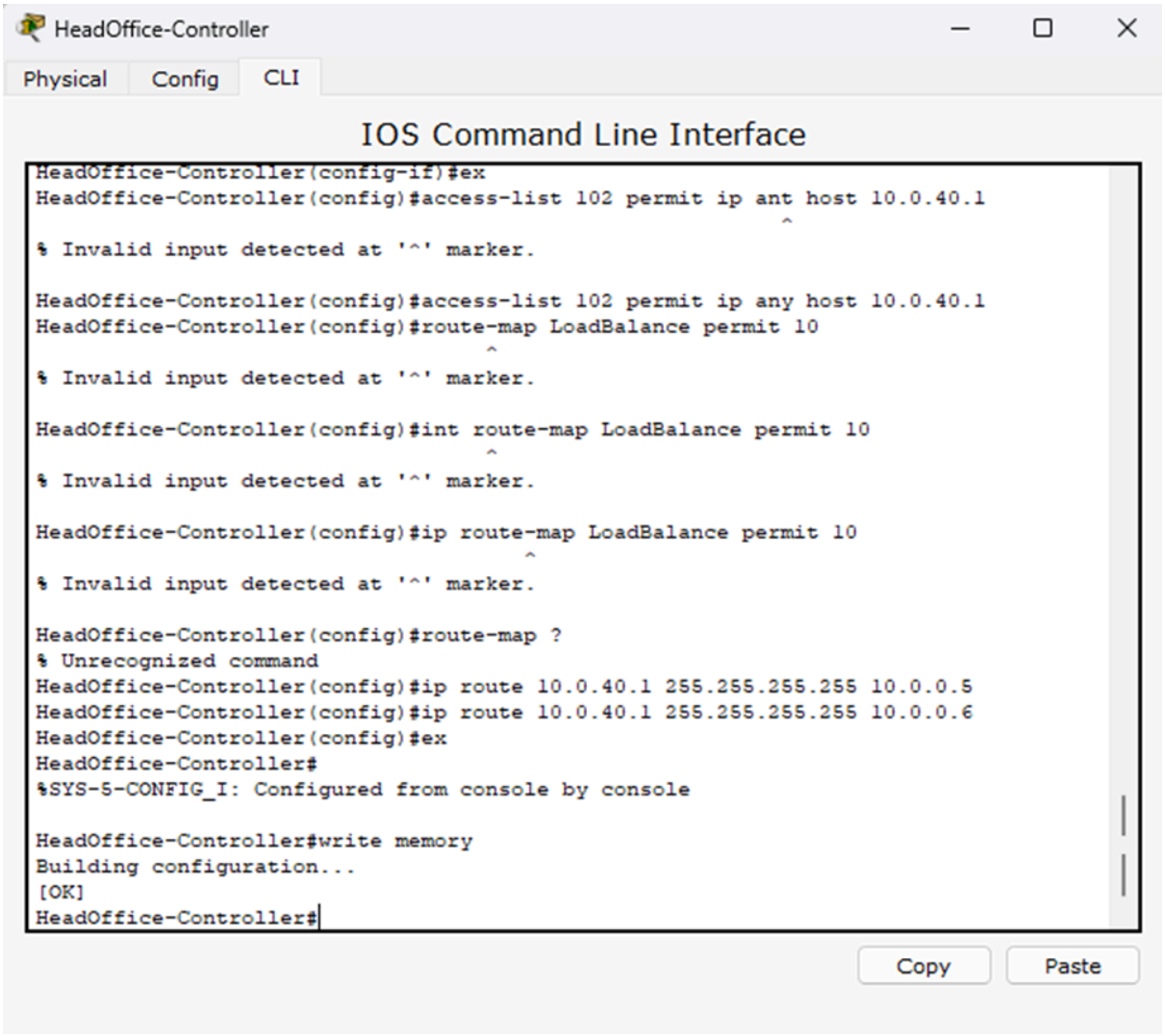
```
ip route 10.0.40.1 255.255.255.255 10.0.10.1
```

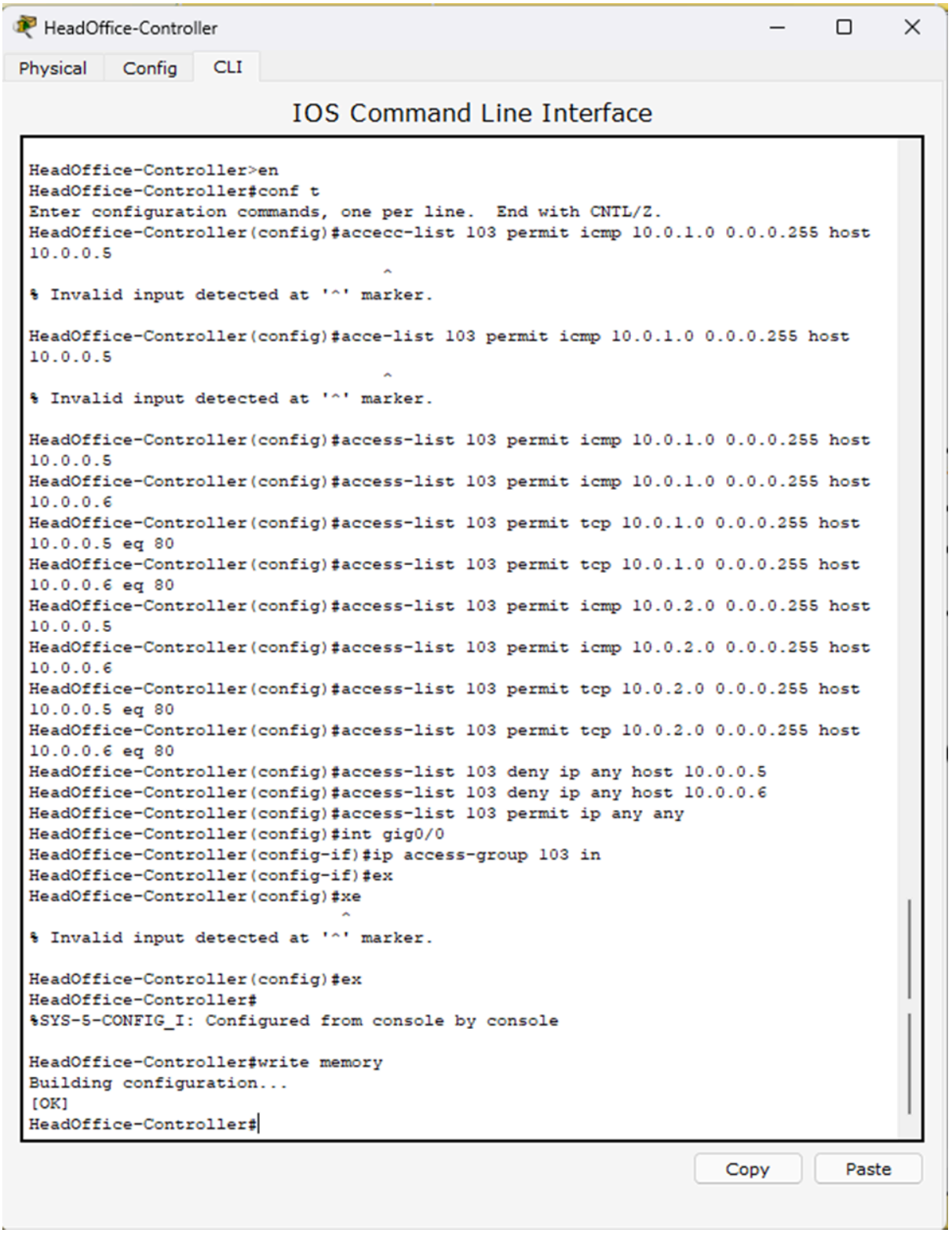
  - On ‘Multan-Controller‘:

```
ip route 10.0.40.1 255.255.255.255 10.0.10.1
```




```
    90 permit ip any any
Router#
Router#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#int Loopback0

Router(config-if)#
%LINK-5-CHANGED: Interface Loopback0, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback0, changed state to up

Router(config-if)#ip add 10.0.0.10 255.255.255.255
% 10.0.0.10 overlaps with GigabitEthernet0/0
Router(config-if)#ip add 10.0.40.1 255.255.255.255
Router(config-if)#ex
Router(config)#ex
Router#
%SYS-5-CONFIG_I: Configured from console by console

Router#write memory
Building configuration...
[OK]
Router#
```


HeadOffice-Controller
Physical
Config
CLI
IOS Command Line Interface
HeadOffice-Controller(config-if)#ex
HeadOffice-Controller(config)#access-list 102 permit ip ant host 10.0.40.1
% Invalid input detected at '^' marker.
HeadOffice-Controller(config)#access-list 102 permit ip any host 10.0.40.1
HeadOffice-Controller(config)#route-map LoadBalance permit 10
% Invalid input detected at '^' marker.
HeadOffice-Controller(config)#int route-map LoadBalance permit 10
% Invalid input detected at '^' marker.
HeadOffice-Controller(config)#ip route-map LoadBalance permit 10
% Invalid input detected at '^' marker.
HeadOffice-Controller(config)#route-map ?
% Unrecognized command
HeadOffice-Controller(config)#ip route 10.0.40.1 255.255.255.255 10.0.0.5
HeadOffice-Controller(config)#ip route 10.0.40.1 255.255.255.255 10.0.0.6
HeadOffice-Controller(config)#ex
HeadOffice-Controller#
%SYS-5-CONFIG_I: Configured from console by console
HeadOffice-Controller#write memory
Building configuration...
[OK]
HeadOffice-Controller#
Copy
Paste



4.3. Testing

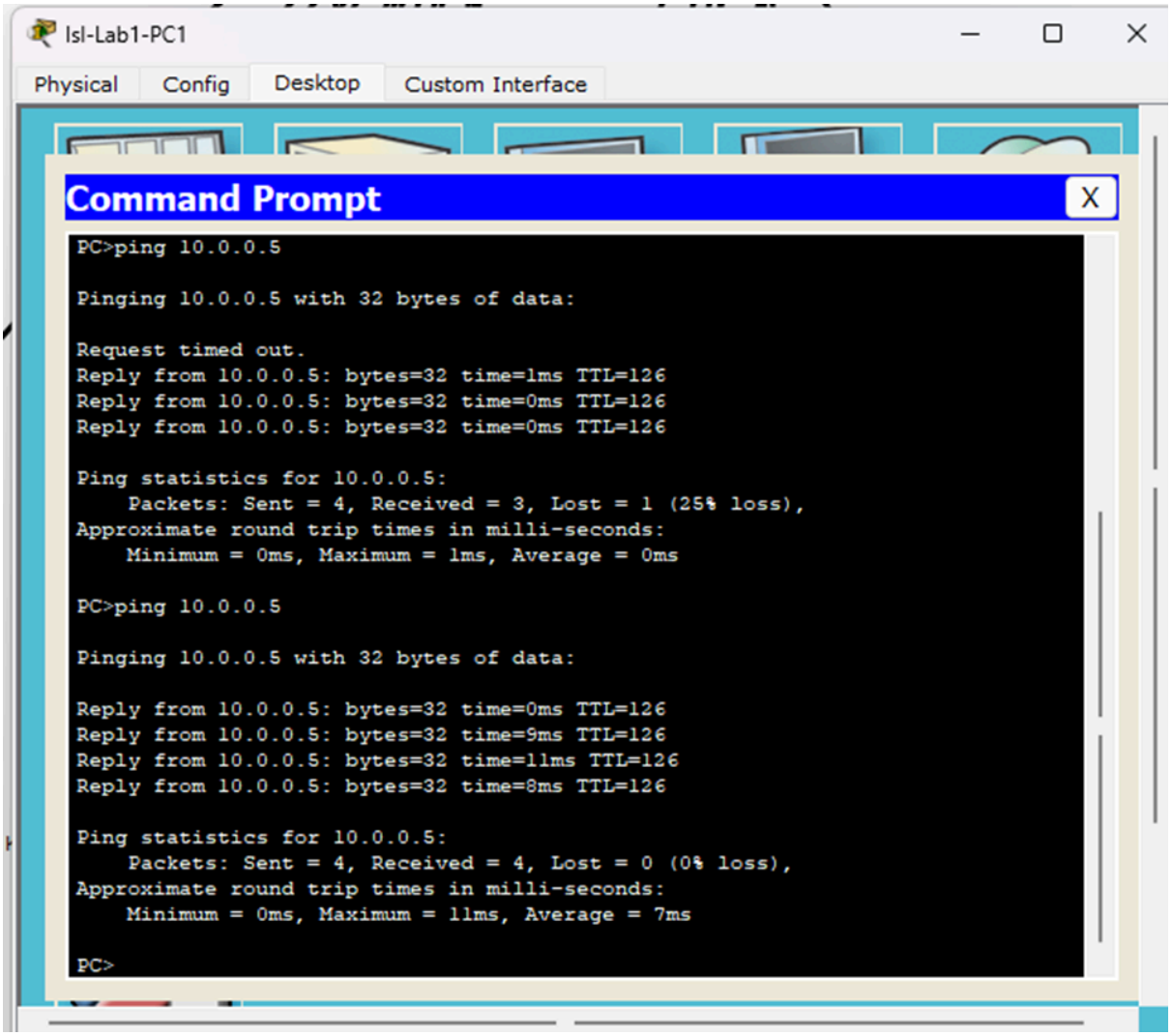
Detailed tests were conducted to verify traffic engineering and load balancing:

- **Traffic Engineering Tests**:
  - **Server Traffic (Allowed)**:

∗ From ‘Isb-Lab1-PC1‘ (‘10.0.1.10‘) to ‘HO-EdgeServer1‘ (‘10.0.0.5‘):

```
Pinging 10.0.0.5 with 32 bytes of data:
Reply from 10.0.0.5: bytes=32 time=2ms TTL=126
Reply from 10.0.0.5: bytes=32 time=2ms TTL=126
Reply from 10.0.0.5: bytes=32 time=2ms TTL=126
Reply from 10.0.0.5: bytes=32 time=2ms TTL=126

Ping statistics for 10.0.0.5:
    Packets: Sent = 4, Received = 4, Lost = 0 (0%
    loss),
```

∗ Result: Success (server traffic allowed).



– **PC-to-PC across Campuses (Blocked)**:

∗ From ‘Isb-Lab1-PC1‘ (‘10.0.1.10‘) to ‘Mul-Lab1-PC1‘ (‘10.0.6.10‘):

```
Pinging 10.0.6.10 with 32 bytes of data:
Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 10.0.6.10:
Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
```

∗ Result: Failed (expected, as ACL 101 blocks PC-to-PC traffic across campuses).

– **Backup Network (Unaffected)**:

∗ From ‘Backup-Intermediary-Switch‘ (‘10.0.20.1‘) to ‘Karachi-Controller‘ (G0/2, ‘10.0.20.4‘):

```
Pinging 10.0.20.4 with 32 bytes of data:
Reply from 10.0.20.4: bytes=32 time=1ms TTL=128
Reply from 10.0.20.4: bytes=32 time=1ms TTL=128
Reply from 10.0.20.4: bytes=32 time=1ms TTL=128
Reply from 10.0.20.4: bytes=32 time=1ms TTL=128

Ping statistics for 10.0.20.4:
   Packets: Sent = 4, Received = 4, Lost = 0 (0%
   loss),
```

∗ Result: Success (backup network unaffected by ACL).

- **Load Balancing Tests**:

– **Reachability of Service IP**:

∗ From ‘Kar-Lab1-PC1‘ (‘10.0.2.10‘) to ‘10.0.40.1‘:

```
Pinging 10.0.40.1 with 32 bytes of data:
Reply from 10.0.40.1: bytes=32 time=2ms TTL=126
Reply from 10.0.40.1: bytes=32 time=2ms TTL=126
Reply from 10.0.40.1: bytes=32 time=2ms TTL=126
Reply from 10.0.40.1: bytes=32 time=2ms TTL=126
```

```
Ping statistics for 10.0.40.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0%
    loss),
```

* Result: Success.

– **Load Balancing Verification**:

* Used Simulation mode to observe traffic distribution.

* Sent multiple pings from ‘Kar-Lab1-PC1‘ (‘10.0.2.10‘) and ‘Kar-Lab1-PC2‘ (‘10.0.2.11‘) to ‘10.0.40.1‘.

* Observed packets alternating between ‘HO-EdgeServer1‘ (‘10.0.0.5‘) and ‘HO-EdgeServer2‘ (‘10.0.0.6‘), confirming equal-cost load balancing.

• **Verification Commands**:

– On ‘HeadOffice-Controller‘:

* show access-lists:

```
Extended IP access list 101
    10 permit ip 10.0.0.0 0.0.0.255 any
    20 permit ip host 10.0.1.5 any
    30 permit ip host 10.0.2.5 any
    40 permit ip host 10.0.3.5 any
    50 permit ip host 10.0.4.5 any
    60 permit ip host 10.0.5.5 any
    70 permit ip host 10.0.6.5 any
    80 deny ip 10.0.0.0 0.0.255.255 10.0.0.0
    0.0.255.255
    90 permit ip any any
```

* show ip route:

```
10.0.0.0/8 is variably subnetted, ...
  10.0.40.1/32 is directly connected, Loopback0
            10.0.40.1/32
                      [1/0
                      ]
                      via
```

```
10.0
.0.5
[1/0
]
via
10.0
.0.6
```

– On ‘Karachi-Controller‘:

```
* show ip route:
  10.0.0.0/8 is variably subnetted, ...
    10.0.40.1/32 [1/0] via 10.0.10.1
```

4.4. Challenges and Solutions

- **Challenge**: Policy-Based Routing (PBR) using ‘route-map‘ for load balancing failed with errors

(‘

- **Solution**: Due to Packet Tracer’s limitation (lack of PBR support on the Cisco 2911 router), used equal-cost static routes (‘ip route 10.0.40.1 255.255.255.255 10.0.0.5‘ and ‘ip route 10.0.40.1 255.255.255.255 10.0.0.6‘) to achieve load balancing.

- **Challenge**: Initial ping to ‘10.0.40.1‘ failed with “Destination host unreachable” from ‘Karachi-

Controller‘.

- **Solution**: Identified missing route to ‘10.0.40.1‘ on Campus-Controllers. Added ‘ip route 10.0.40.1 255.255.255.255 10.0.10.1‘ on all Campus-Controllers, resolving the issue.

- **Challenge**: ACL initially blocked more traffic than intended.

- **Solution**: Adjusted ACL sequence to ensure server traffic (‘10.0.0.0/24‘ and campus edge servers) is permitted before the deny rule for PC-to-PC traffic.

Conclusion

This project successfully implemented an SDN-based network for NUCES in Cisco Packet Tracer across four parts:

- **Part 1**: Designed a detailed topology and IP addressing scheme for a multi-campus network, ensuring scalability and support for SDN features.
- **Part 2**: Set up the network in Cisco Packet Tracer, configured IP addresses, and established basic connectivity within campuses.
- **Part 3**: Simulated SDN controller logic using static routing, enabling inter-subnet connectivity and implementing a backup network for redundancy.
- **Part 4**: Implemented traffic engineering with ACLs to manage traffic flows and load balancing using static routes to distribute server traffic at the head office.

Despite challenges such as Cisco Packet Tracer's limitations (e.g., lack of PBR support), alternative methods like static routes were used to achieve the project objectives. The final network supports intercampus communication, prioritizes server traffic, blocks unauthorized PC-to-PC traffic, and balances server load at the head office, meeting all requirements of the revised plan.