



Network Infrastructure Design and Implementation for Infinx Electronics Company

Prepared By:
Zahid Saleh Muthanna Muhammed

Table of Contents

1.1 Abstract.....	4
2.1 Introduction.....	4
3.1 Objectives.....	4
4.1 Description.....	5
5.1 Requirements.....	6
6.1 Used Devices.....	7
7.1 Cables Used.....	7
7.2 Passwords.....	7
8.1 Infinx Electronics Network Topology	8
9.1 Subnetting and IP Addressing.....	9
10.1 Configuration of hostname, banner message, password, domain lookup, and password encryption on layer2 switches, layer3 switches, and layer3 routers.	10
11.1 VLANs.....	12
11.1.1 Overview of VLAN	12
11.1.2 Advantages of VLANs.....	12
11.1.3 Step#1 VLANs on Access Layer Switches.....	12
11.1.3 Step#2 VLANs and Inter-VLANS on Layer 3 Switches.....	13
12.1 IP Addresses Assignments.....	14
12.1.1 Layer 3 Switch Interfaces to Core routers Interfaces	14
12.1.2 Core Router's Interfaces to Layer 3 Switch Interfaces	15
12.1.3 Core Router's Interfaces to ISPs Router's Interfaces	16
12.1.4 ISPs Router's Interfaces to Core Router's Interfaces	17
13.1 Secure Shell (SSH) Configuration.....	18
13.1.1 Overview about SSH	18
13.1.2 Key points about SSH:.....	18
13.1.3 SSH Configuration on Multilayer Switch	18
13.1.4 SSH configuration on Layer 3 Routers.....	19
14.1 OSPF Configuration	20
14.1.1 Overview about OSPF	20
14.1.2 Key points about OSPF:.....	20
14.1.3 OSPF Configuration on Multilayer Switches	20
14.1.4 OSPF Configuration on Layer3 Routers	21
14.1.5 OSPF Configuration on ISPs Routers.....	21
14.1.6 OSPF, NAT Overload, RIP, ACL Show Results.....	22

15.1 Servers Infrastructure and configuration.....	23
15.1.1 DHCP Server.....	23
15.1.2 DNS Server.....	24
15.1.3 Web Server.....	25
15.1.4 FTP Server.....	26
15.1.5 Email Server.....	27
16.1 Wireless Configuration	28
17.1 IOT Door Configuration.....	30
19.1 Testing and verifying the connectivity	32
20.1 Conclusion.....	34

1.1 Abstract

This project involves designing a computer network topology using Cisco Packet Tracer for Infinix Electronics (a global mobile electronics company). Computer networks are crucial in today's business world due to the increasing number of electronic devices. Cisco Packet Tracer is a simulation software that allows for network design, configuration, and troubleshooting without affecting real networks. It provides a powerful tool for learning network design principles and allows visualization of network behavior and connectivity. With Packet Tracer, designers can create network topologies, test connectivity, and simulate the behavior of different network devices, such as routers, switches, servers, and access points.

2.1 Introduction

In today's world, computer networks have become an essential aspect of every business. With the increasing number of electronic devices, the demand for effective and efficient networks has increased as well. In this project, we are designing a network for a mobile electronics company called Infinix Electronics, which operates globally. The network will be designed using Cisco Packet Tracer, a simulation software that allows us to design, configure and troubleshoot networks.

Packet Tracer is a powerful tool that enables network designers to simulate network configurations and test them without disrupting real networks. It is an essential tool for learning about network design and allows network administrators to visualize network behavior and connectivity. With Packet Tracer, network designers can create network topologies, test connectivity, and simulate the behavior of network devices. The software includes a variety of networking devices, such as routers, switches, servers, and access points, which can be easily configured to simulate real-world scenarios.

The proposed network for Infinix Electronics will encompass four floors, and we will highlight the components and relevant connectivity in this report. We will implement VLANs and configure IP routing, DHCP, inter-VLAN-related configuration, etc. In this report, we will discuss the design and implementation of the network using Cisco Packet Tracer, focusing on the relevant configurations, the challenges faced, and their solutions.

3.1 Objectives

- Develop self-learning skills.
- Expand learner knowledge in the Network configuration in Cisco tracer.
- Apply the skills and knowledge learned in the course to design and configure institutional networks.
- Practicing Cisco tracer tool.

4.1 Description

Infinix Electronics, a global mobile electronics company, is currently expanding and in the process of relocating to a new building. However, the chosen building lacks an existing network infrastructure. Therefore, before the company can proceed with the move, it is essential to design and implement a new network system for the building. This network implementation project aims to provide seamless connectivity and communication throughout the building.

The new building will consist of four floors, each accommodating two departments:

- The first floor will house the Production and Customer Services departments.
- The second floor will accommodate Sales and Marketing.
- The third floor will be allocated to the Acc/Finance and HR departments.
- The fourth floor will serve as the IT department and house the room servers.
- Furthermore, it is recommended to establish a dedicated wireless network for the break area of the staff.

To ensure a smooth transition, Infinix Electronics hired us as a network engineers team. Our responsibility will be to design and implement a reliable, redundant network system that caters to the specific needs of each department and facilitates efficient communication and collaboration. By successfully completing this project, Infinix Electronics aims to establish a robust network infrastructure that supports its expanding operations and promotes seamless connectivity among its various departments in the new building.

5.1 Requirements

To ensure that the new network for Infenix Electronics meets the current business requirements and remains adaptable for future needs, we need to focus on logical design. The logical design encompasses the following requirements that will be implemented:

1. We'll implement a topology with a redundant connection at every layer for fault tolerance using two routers (*Core-R1, Core-R2*), two multilayer switches, and two routers' ISPs. *Note: (we'll use an AC Power Supply module in each multilayer switch to provide electrical power to the device)*
2. For each department, we'll allocate an access layer switch.
3. In order to achieve high performance, reliability, scalability, and simplified troubleshooting, we will use suitable cables for connecting routers, routers to multilayer switches, and switches to end systems.
4. In order for the routers to be connected with each other using DTE Serial connection, we will use **HWIC-2T** module inside each router (*The HWIC-2T is a Cisco 2-Port Serial High-Speed WAN Interface Card, providing 2 serial ports*).
5. We'll assign a wireless network to each department for users' connectivity using Wireless Access Point (WAP). *Note: (we will assign a password authentication for each WAP in each department to enhance security, control access, and better management for organization's wireless network)*.
6. We'll allocate different VLANs and subnets for each department with considering access ports and trunk ports on layer2 and layer3 switches.
7. For the access layer switches, multilayer switches, and core routers basic device settings, including hostnames, service password encryption, enable passwords, banner messages, and disable IP domain lookup should be implemented.
8. For efficient address allocation, network segmentation, efficient routing, and effective IP address management, we will perform a proper subnetting and IP addressing.
9. We will enable inter-VLAN routing on both layer3 switches to facilitate communication between departments.
10. For the devices on the room server, we will assign IP addresses statically. Also, we'll assign IP addresses to the multilayer switches for routing and switching functionality.
11. We'll use a dedicated DHCP server that is located in the server room to dynamically allocate IP addresses to devices in the network.
12. To provide a path for network traffic when there is no specific route available for a destination, we'll configure the default route in layer 3 switches and core routers.
13. We'll configure OSPF as the routing protocol to advertise routes on routers and layer3 switches and SSH on routers and layer3 switches for secure remote login,
14. For efficient address utilization, enabling multiple devices to share a single public IP address, providing internet access for private networks, we'll configure the NAT overload. In addition, we will configure standard and extended ACL.
15. After implementing the necessary technologies, we'll test the connectivity between departments to ensure proper configuration and functionality of all devices.

6.1 Used Devices

Device Name	Model	Quantity
<i>Router</i>	<i>2811</i>	<i>2</i>
<i>Router</i>	<i>2911</i>	<i>2</i>
<i>Multilayer Switch</i>	<i>3652-2PS</i>	<i>2</i>
<i>Access Layer Switch</i>	<i>2960-24TT</i>	<i>2</i>
<i>Access Point</i>	<i>AP-PT</i>	<i>8</i>
<i>PCs</i>	<i>-</i>	<i>43</i>
<i>Smartphone</i>	<i>-</i>	<i>11</i>
<i>Tablets</i>	<i>-</i>	<i>11</i>
<i>Printer</i>	<i>-</i>	<i>7</i>
<i>IoT Door</i>	<i>-</i>	<i>7</i>
<i>Server</i>	<i>-</i>	<i>5</i>

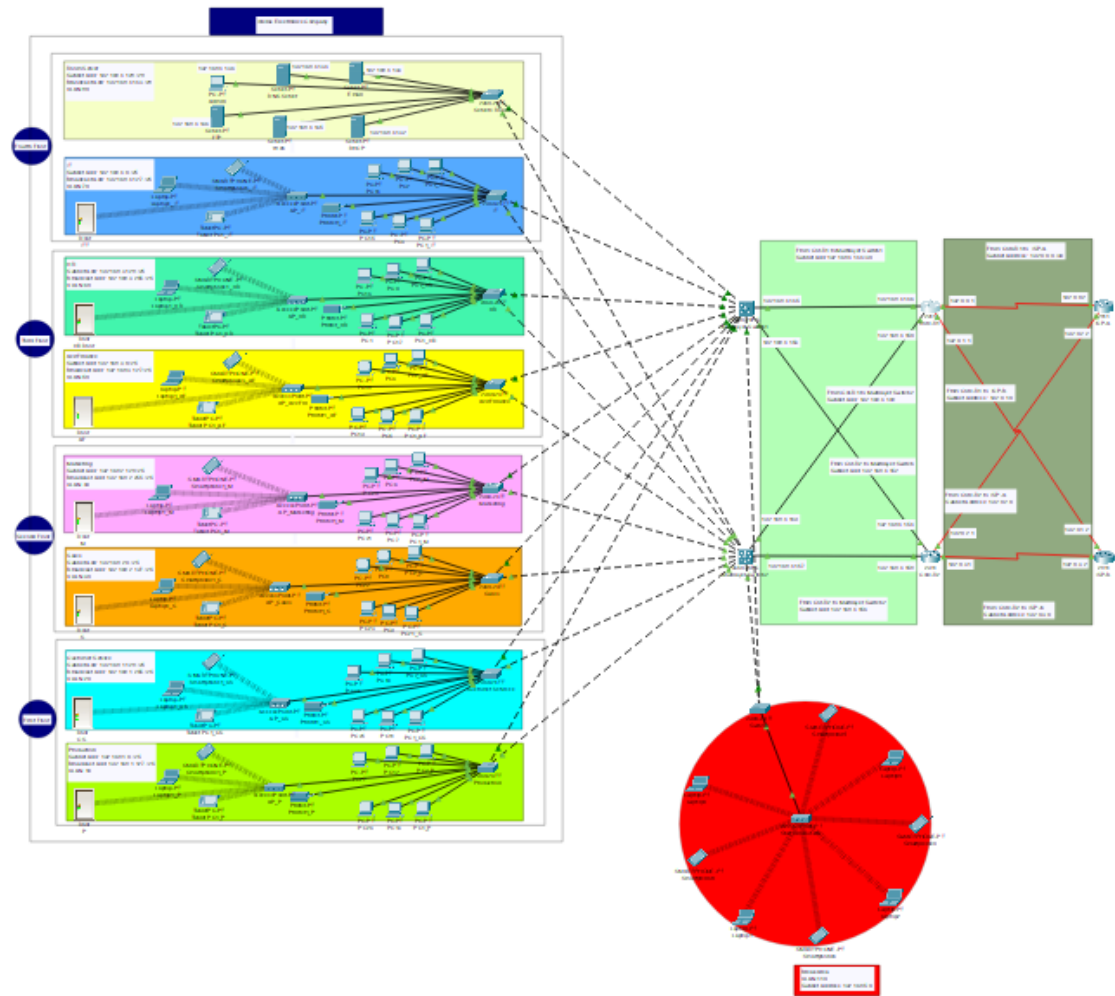
7.1 Cables Used

Location	Type
<i>Between Routers</i>	<i>Serial connection (DTE)</i>
<i>Routers' to multilayer switches</i>	<i>Straight-through connection</i>
<i>Multilayer switches to access layer switches</i>	<i>Cross-over connection</i>
<i>Access layer switches to end systems</i>	<i>Straight-through connection</i>

7.2 Passwords

Device	Password
<i>Production-Switch</i>	<i>production123</i>
<i>Customer Service Switch</i>	<i>cs1234</i>
<i>Acc/Finance Switch</i>	<i>acc/fin1234</i>
<i>Marketing-Switch</i>	<i>m1234</i>
<i>HR-Switch</i>	<i>hr1234</i>
<i>IT-Switch</i>	<i>it1234</i>
<i>Sales-Switch</i>	<i>s1234</i>
<i>Main Multilayer Switch1</i>	<i>mmls1</i>
<i>Main Multilayer Switch2</i>	<i>mmls2</i>
<i>Core-R1</i>	<i>corer1</i>
<i>Core-R2</i>	<i>corer2</i>

8.1 Infix Electronics Network Topology



9.1 Subnetting and IP Addressing

- *Four Floors (Each Two Departments)*

Subnet Name	Subnet Address	First Usable Host	Last Usable Host	Broadcast Address	Subnet Mask
Production	192.168.1.0	192.168.1.1	192.168.1.126	192.168.1.127	255.255.255.128/25
Customer Service	192.168.1.128	192.168.1.129	192.168.1.254	192.168.1.255	
Sales	192.168.2.0	192.168.2.1	192.168.2.126	192.168.2.127	
Marketing	192.168.2.128	192.168.2.129	192.168.2.254	192.168.2.255	
Acc/Finance	192.168.3.0	192.168.3.1	192.168.3.126	192.168.3.127	
HR	192.168.3.128	192.168.3.129	192.168.3.254	192.168.3.255	
IT	192.168.4.0	192.168.4.1	192.168.4.126	192.168.4.127	255.255.255.240/28
Server Room	192.168.4.128	192.168.4.129	192.168.4.142	192.168.4.143	

- *Between Core Routers' and Layer3 Switches*

Router and Multilayer Switch Names	Subnet Address	First Usable Host	Last Usable Host	Broadcast Address	Subnet Mask
Core-R1 to Multilayer Switch1	192.168.4.144	192.168.4.145	192.168.4.146	192.168.4.147	255.255.255.252/30
Core-R1 to Multilayer Switch1	192.168.4.148	192.168.4.149	192.168.4.150	192.168.4.151	
Core-R2 to Multilayer Switch1	192.168.4.152	192.168.4.153	192.168.4.154	192.168.4.155	
Core-R2 to Multilayer Switch1	192.168.4.156	192.168.4.157	192.168.4.158	192.168.4.159	

- *Between Core Routers and ISPs*

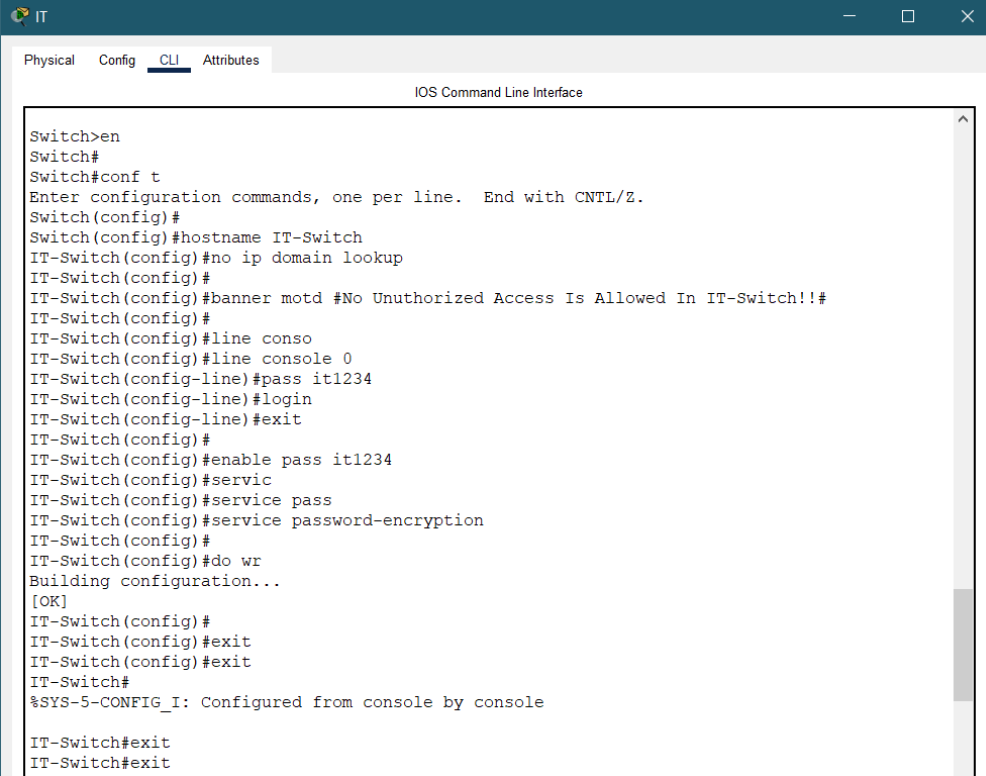
Router and ISP Names	Subnet Address	First Usable Host	Last Usable Host	Broadcast Address	Subnet Mask
Core-R1 to ISP1	192.0.0.0	192.0.0.1	192.0.0.2	192.0.0.3	255.255.255.252/30
Core-R1 to ISP2	192.0.1.0	192.0.1.1	192.0.1.2	192.0.1.3	
Core-R2 to ISP1	192.0.2.0	192.0.2.1	192.0.2.2	192.0.2.3	
Core-R2 to ISP2	192.0.3.0	192.0.3.1	192.0.3.2	192.0.3.3	

- *Wireless Subnet for Staff Break Area*

Subnet Name	Subnet Address	First Usable Host	Last Usable Host	Broadcast Address	Subnet Mask
Break Area	192.168.5.0	192.168.5.1	192.168.5.254	192.168.5.255	255.255.255.0/24

10.1 Configuration of hostname, banner message, password, domain lookup, and password encryption on layer2 switches, layer3 switches, and layer3 routers.

10.1.1 Layer2 Switch

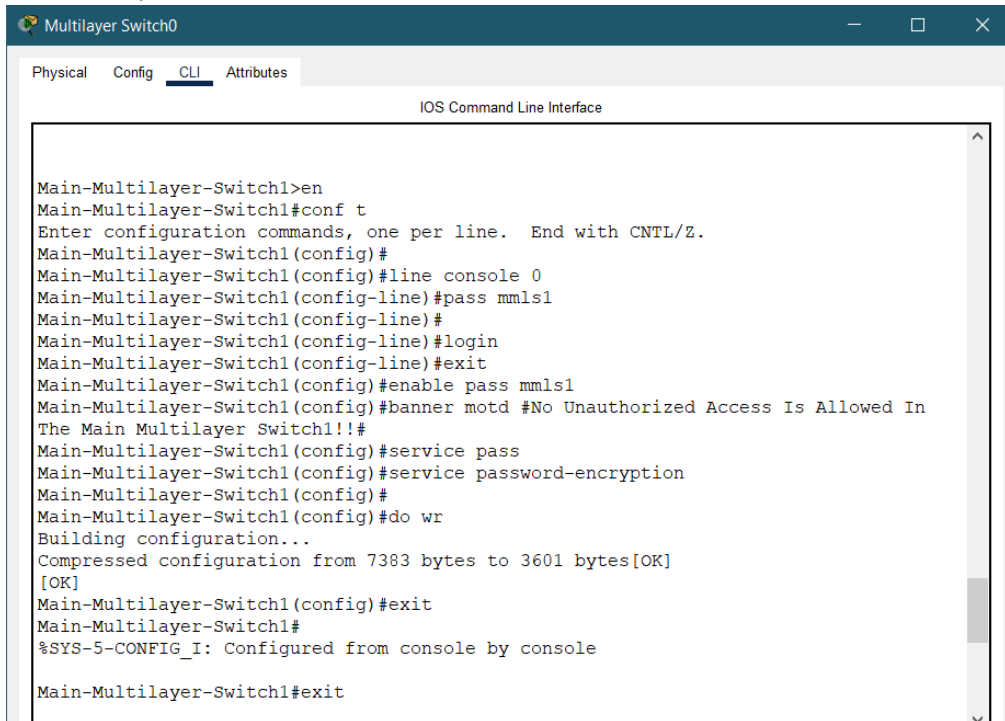


The screenshot shows a network configuration window titled "IT" with tabs for "Physical", "Config", "CLI", and "Attributes". The "CLI" tab is active, displaying the "IOS Command Line Interface". The interface shows a sequence of commands entered to configure a switch named "IT-Switch". The commands include setting the hostname, disabling domain lookup, configuring a banner message, setting console line parameters and password, enabling password encryption, and saving the configuration. The output shows the configuration being built and saved successfully.

```
Switch>en
Switch#
Switch#conf t
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)#
Switch(config)#hostname IT-Switch
IT-Switch(config)#no ip domain lookup
IT-Switch(config)#
IT-Switch(config)#banner motd #No Unauthorized Access Is Allowed In IT-Switch!!#
IT-Switch(config)#
IT-Switch(config)#line conso
IT-Switch(config)#line console 0
IT-Switch(config-line)#pass it1234
IT-Switch(config-line)#login
IT-Switch(config-line)#exit
IT-Switch(config)#
IT-Switch(config)#enable pass it1234
IT-Switch(config)#servic
IT-Switch(config)#service pass
IT-Switch(config)#service password-encryption
IT-Switch(config)#
IT-Switch(config)#do wr
Building configuration...
[OK]
IT-Switch(config)#
IT-Switch(config)#exit
IT-Switch(config)#exit
IT-Switch#
%SYS-5-CONFIG_I: Configured from console by console

IT-Switch#exit
IT-Switch#exit
```

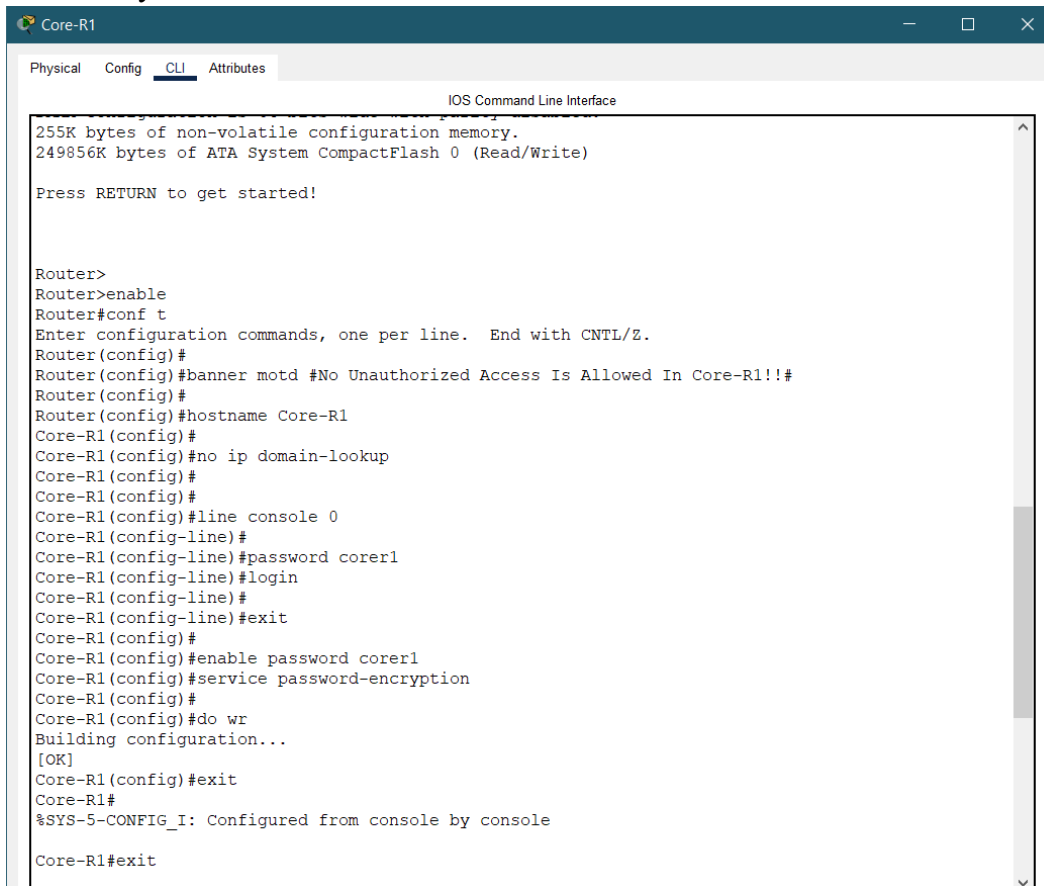
Layer3 switch



The screenshot shows a window titled "Multilayer Switch0" with tabs for Physical, Config, CLI, and Attributes. The CLI tab is active, displaying the IOS Command Line Interface. The user enters the following commands:

```
Main-Multilayer-Switch1>en
Main-Multilayer-Switch1#conf t
Enter configuration commands, one per line. End with CNTL/Z.
Main-Multilayer-Switch1(config)#
Main-Multilayer-Switch1(config)#line console 0
Main-Multilayer-Switch1(config-line)#pass mmls1
Main-Multilayer-Switch1(config-line)#
Main-Multilayer-Switch1(config-line)#login
Main-Multilayer-Switch1(config-line)#exit
Main-Multilayer-Switch1(config)#enable pass mmls1
Main-Multilayer-Switch1(config)#banner motd #No Unauthorized Access Is Allowed In
The Main Multilayer Switch1!!#
Main-Multilayer-Switch1(config)#service pass
Main-Multilayer-Switch1(config)#service password-encryption
Main-Multilayer-Switch1(config)#
Main-Multilayer-Switch1(config)#do wr
Building configuration...
Compressed configuration from 7383 bytes to 3601 bytes[OK]
[OK]
Main-Multilayer-Switch1(config)#exit
Main-Multilayer-Switch1#
%SYS-5-CONFIG_I: Configured from console by console
Main-Multilayer-Switch1#exit
```

10.1.3 Layer3 router



The screenshot shows a window titled "Core-R1" with tabs for Physical, Config, CLI, and Attributes. The CLI tab is active, displaying the IOS Command Line Interface. The user enters the following commands:

```
255K bytes of non-volatile configuration memory.
249856K bytes of ATA System CompactFlash 0 (Read/Write)

Press RETURN to get started!

Router>
Router>enable
Router#conf t
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#
Router(config)#banner motd #No Unauthorized Access Is Allowed In Core-R1!!#
Router(config)#
Router(config)#hostname Core-R1
Core-R1(config)#
Core-R1(config)#no ip domain-lookup
Core-R1(config)#
Core-R1(config)#line console 0
Core-R1(config-line)#
Core-R1(config-line)#password corer1
Core-R1(config-line)#login
Core-R1(config-line)#
Core-R1(config-line)#exit
Core-R1(config)#
Core-R1(config)#enable password corer1
Core-R1(config)#service password-encryption
Core-R1(config)#
Core-R1(config)#do wr
Building configuration...
[OK]
Core-R1(config)#exit
Core-R1#
%SYS-5-CONFIG_I: Configured from console by console
Core-R1#exit
```

11.1 VLANs

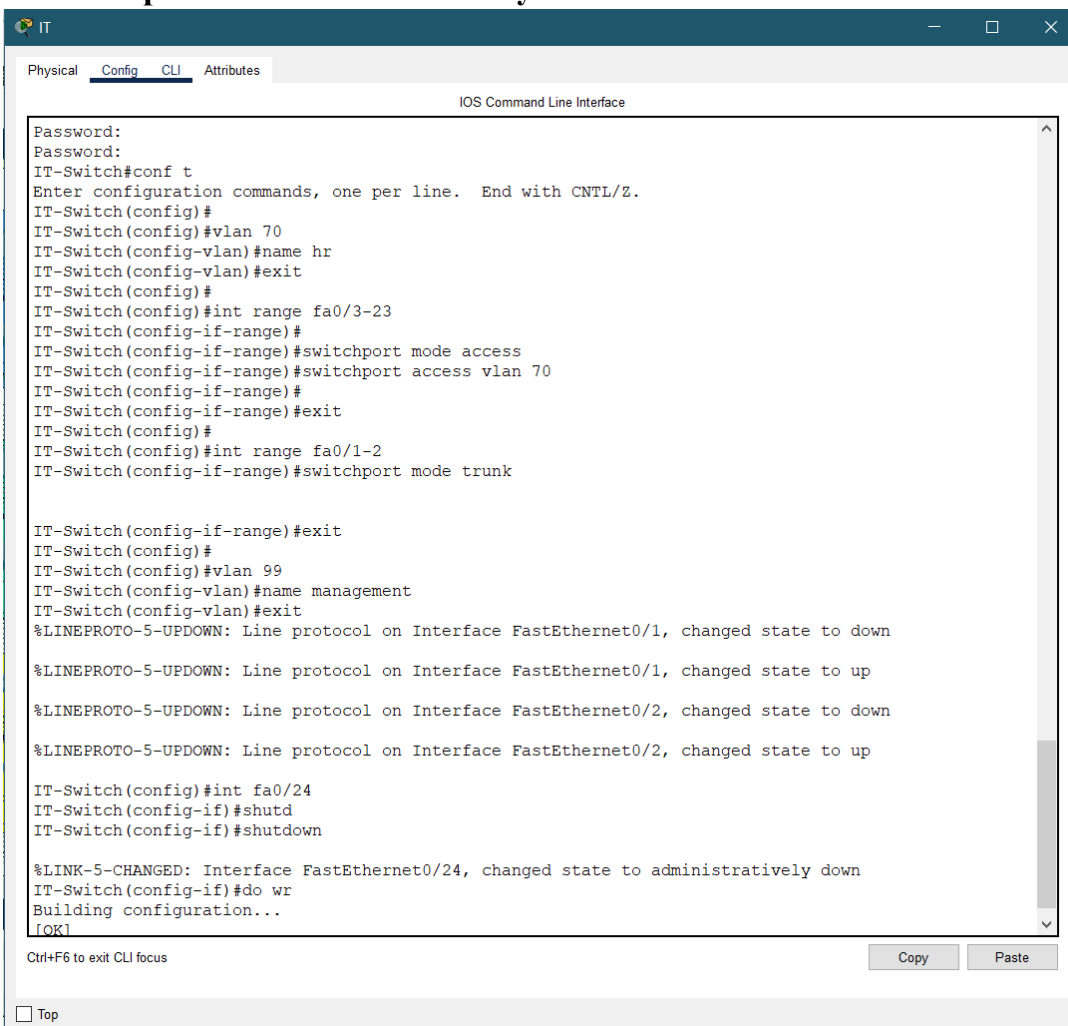
11.1.1 Overview of VLAN

A VLAN (Virtual Local Area Network) is a logical grouping of devices within a network, allowing them to communicate as if they were on the same physical network segment, regardless of their physical location.

11.1.2 Advantages of VLANs

1. Optimized Resource Utilization
2. Support for Virtualization
3. Simplified Network Management
4. Enhanced Network Security
5. Isolate traffic

11.1.3 Step#1 VLANs on Access Layer Switches



The screenshot shows a network configuration window titled "IT" with tabs for "Physical", "Config", "CLI", and "Attributes". The "CLI" tab is active, displaying the "IOS Command Line Interface". The interface shows a series of commands being entered into a terminal window. The commands configure two VLANs: VLAN 70 named "hr" and VLAN 99 named "management". For each VLAN, interfaces are configured as access ports. Additionally, interfaces fa0/1 and fa0/2 are configured as trunk ports. The configuration process concludes with a shutdown of interface fa0/24 and a confirmation message "Building configuration...".

```
IT
Physical Config CLI Attributes
IOS Command Line Interface

Password:
Password:
IT-Switch#conf t
Enter configuration commands, one per line. End with CNTL/Z.
IT-Switch(config)#
IT-Switch(config)#vlan 70
IT-Switch(config-vlan)#name hr
IT-Switch(config-vlan)#exit
IT-Switch(config)#
IT-Switch(config)#int range fa0/3-23
IT-Switch(config-if-range)#
IT-Switch(config-if-range)#switchport mode access
IT-Switch(config-if-range)#switchport access vlan 70
IT-Switch(config-if-range)#
IT-Switch(config-if-range)#exit
IT-Switch(config)#
IT-Switch(config)#int range fa0/1-2
IT-Switch(config-if-range)#switchport mode trunk

IT-Switch(config-if-range)#exit
IT-Switch(config)#
IT-Switch(config)#vlan 99
IT-Switch(config-vlan)#name management
IT-Switch(config-vlan)#exit
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/1, changed state to down
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/1, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/2, changed state to down
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/2, changed state to up

IT-Switch(config)#int fa0/24
IT-Switch(config-if)#shutd
IT-Switch(config-if)#shutdown

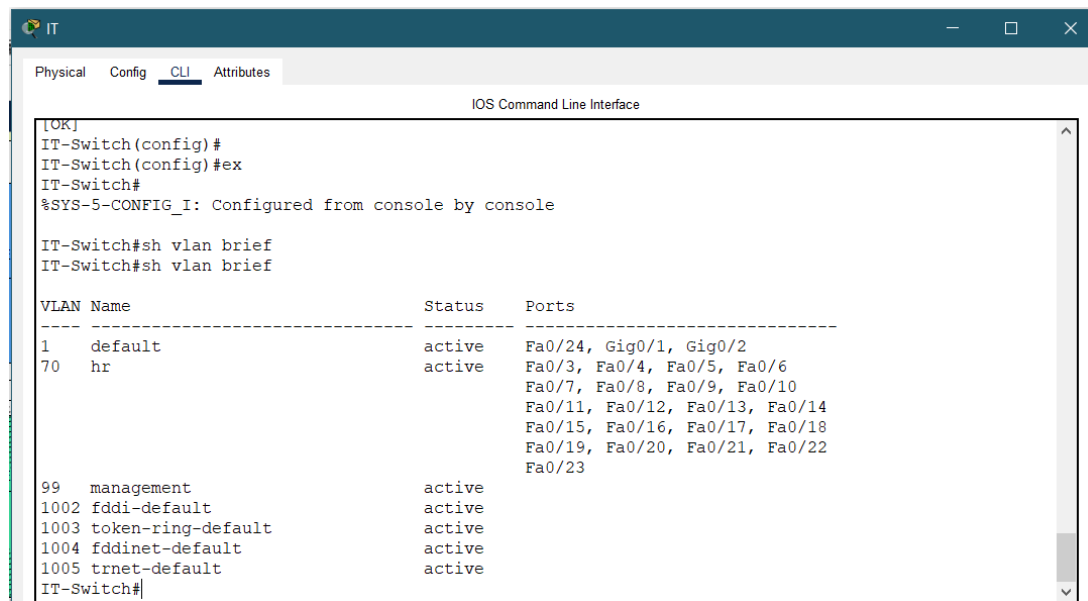
%LINK-5-CHANGED: Interface FastEthernet0/24, changed state to administratively down
IT-Switch(config-if)#do wr
Building configuration...
[OK]
```

Ctrl+F6 to exit CLI focus

Copy Paste

☐ Top

Result



The screenshot shows a network switch CLI window titled "IT" with tabs for Physical, Config, CLI, and Attributes. The CLI window displays the following commands and output:

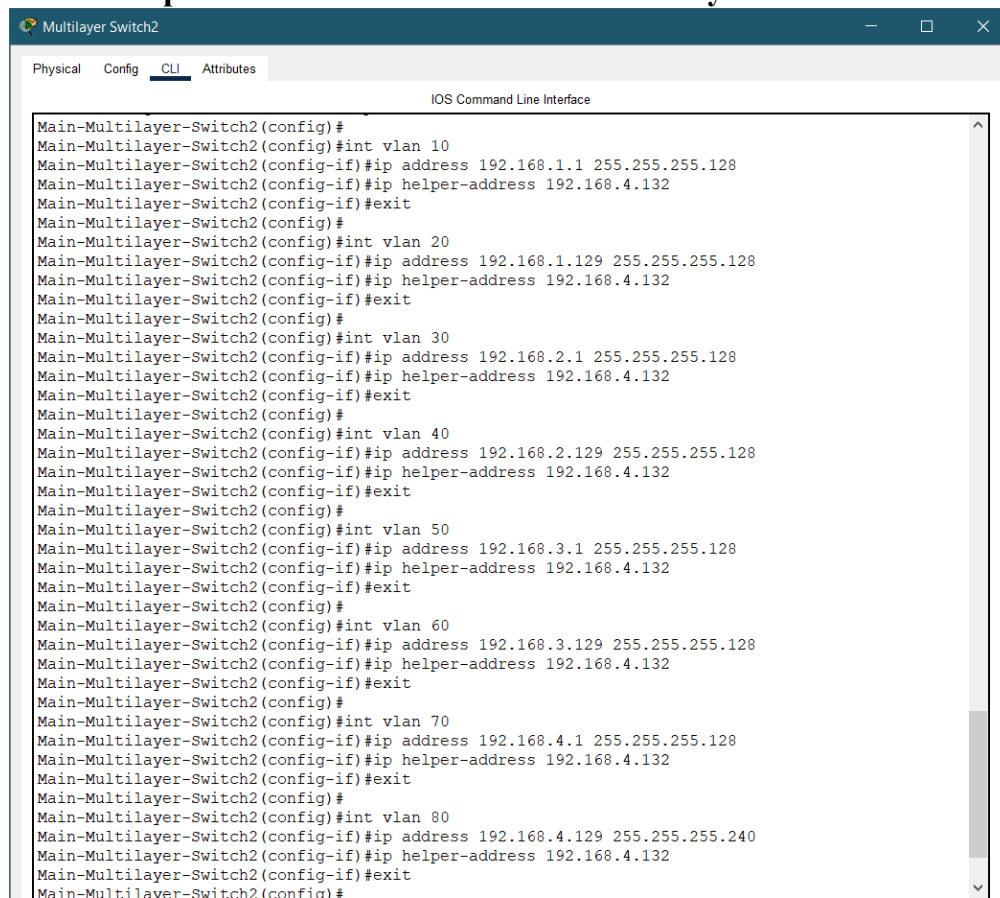
```
[OK]
IT-Switch(config)#
IT-Switch(config)#ex
IT-Switch#
%SYS-5-CONFIG_I: Configured from console by console

IT-Switch#sh vlan brief
IT-Switch#sh vlan brief
```

VLAN	Name	Status	Ports
1	default	active	Fa0/24, Gig0/1, Gig0/2
70	hr	active	Fa0/3, Fa0/4, Fa0/5, Fa0/6 Fa0/7, Fa0/8, Fa0/9, Fa0/10 Fa0/11, Fa0/12, Fa0/13, Fa0/14 Fa0/15, Fa0/16, Fa0/17, Fa0/18 Fa0/19, Fa0/20, Fa0/21, Fa0/22 Fa0/23
99	management	active	
1002	fddi-default	active	
1003	token-ring-default	active	
1004	fddinet-default	active	
1005	trnet-default	active	

```
IT-Switch#
```

11.1.3 Step#2 VLANs and Inter-VLANS on Layer 3 Switches



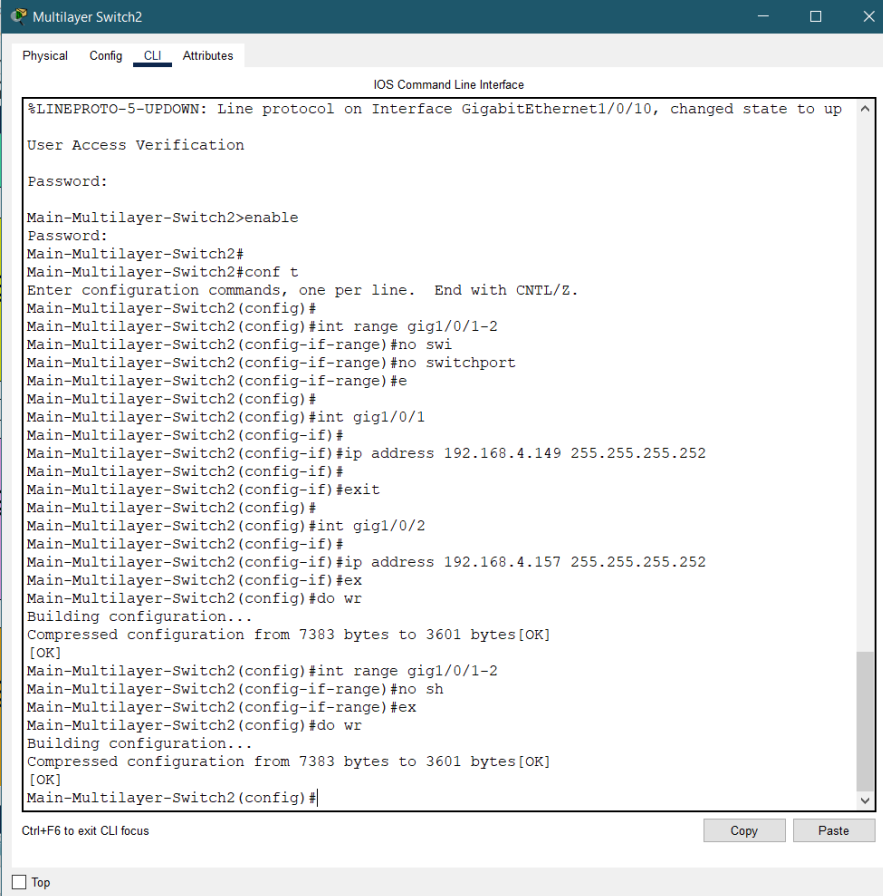
The screenshot shows a network switch CLI window titled "Multilayer Switch2" with tabs for Physical, Config, CLI, and Attributes. The CLI window displays the following commands and output:

```
Main-Multilayer-Switch2(config)#
Main-Multilayer-Switch2(config)#int vlan 10
Main-Multilayer-Switch2(config-if)#ip address 192.168.1.1 255.255.255.128
Main-Multilayer-Switch2(config-if)#ip helper-address 192.168.4.132
Main-Multilayer-Switch2(config-if)#exit
Main-Multilayer-Switch2(config)#
Main-Multilayer-Switch2(config)#int vlan 20
Main-Multilayer-Switch2(config-if)#ip address 192.168.1.129 255.255.255.128
Main-Multilayer-Switch2(config-if)#ip helper-address 192.168.4.132
Main-Multilayer-Switch2(config-if)#exit
Main-Multilayer-Switch2(config)#
Main-Multilayer-Switch2(config)#int vlan 30
Main-Multilayer-Switch2(config-if)#ip address 192.168.2.1 255.255.255.128
Main-Multilayer-Switch2(config-if)#ip helper-address 192.168.4.132
Main-Multilayer-Switch2(config-if)#exit
Main-Multilayer-Switch2(config)#
Main-Multilayer-Switch2(config)#int vlan 40
Main-Multilayer-Switch2(config-if)#ip address 192.168.2.129 255.255.255.128
Main-Multilayer-Switch2(config-if)#ip helper-address 192.168.4.132
Main-Multilayer-Switch2(config-if)#exit
Main-Multilayer-Switch2(config)#
Main-Multilayer-Switch2(config)#int vlan 50
Main-Multilayer-Switch2(config-if)#ip address 192.168.3.1 255.255.255.128
Main-Multilayer-Switch2(config-if)#ip helper-address 192.168.4.132
Main-Multilayer-Switch2(config-if)#exit
Main-Multilayer-Switch2(config)#
Main-Multilayer-Switch2(config)#int vlan 60
Main-Multilayer-Switch2(config-if)#ip address 192.168.3.129 255.255.255.128
Main-Multilayer-Switch2(config-if)#ip helper-address 192.168.4.132
Main-Multilayer-Switch2(config-if)#exit
Main-Multilayer-Switch2(config)#
Main-Multilayer-Switch2(config)#int vlan 70
Main-Multilayer-Switch2(config-if)#ip address 192.168.4.1 255.255.255.128
Main-Multilayer-Switch2(config-if)#ip helper-address 192.168.4.132
Main-Multilayer-Switch2(config-if)#exit
Main-Multilayer-Switch2(config)#
Main-Multilayer-Switch2(config)#int vlan 80
Main-Multilayer-Switch2(config-if)#ip address 192.168.4.129 255.255.255.240
Main-Multilayer-Switch2(config-if)#ip helper-address 192.168.4.132
Main-Multilayer-Switch2(config-if)#exit
Main-Multilayer-Switch2(config)#
```

Inter- vlans

12.1 IP Addresses Assignments

12.1.1 Layer 3 Switch Interfaces to Core routers Interfaces

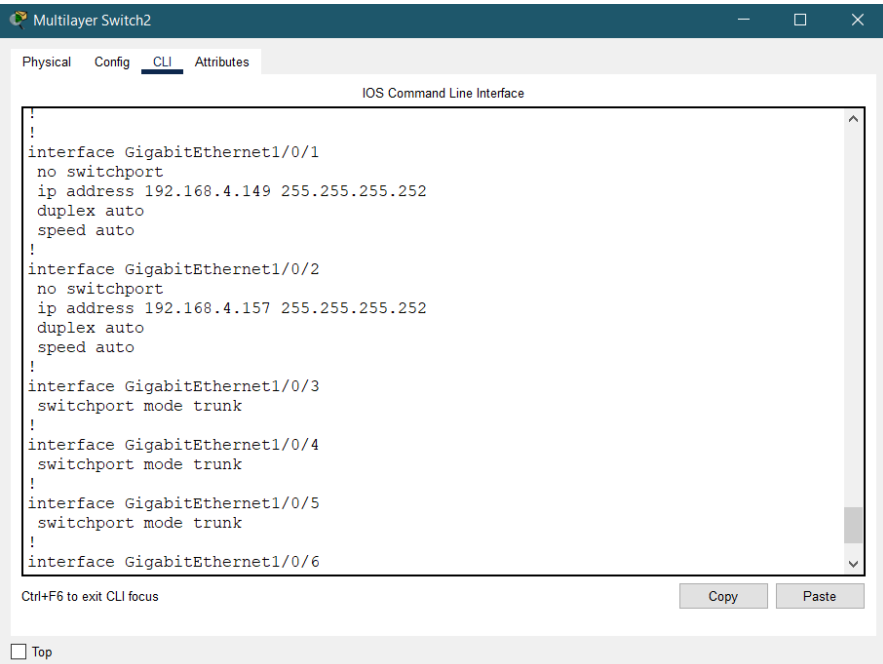


```
%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet1/0/10, changed state to up
User Access Verification

Password:

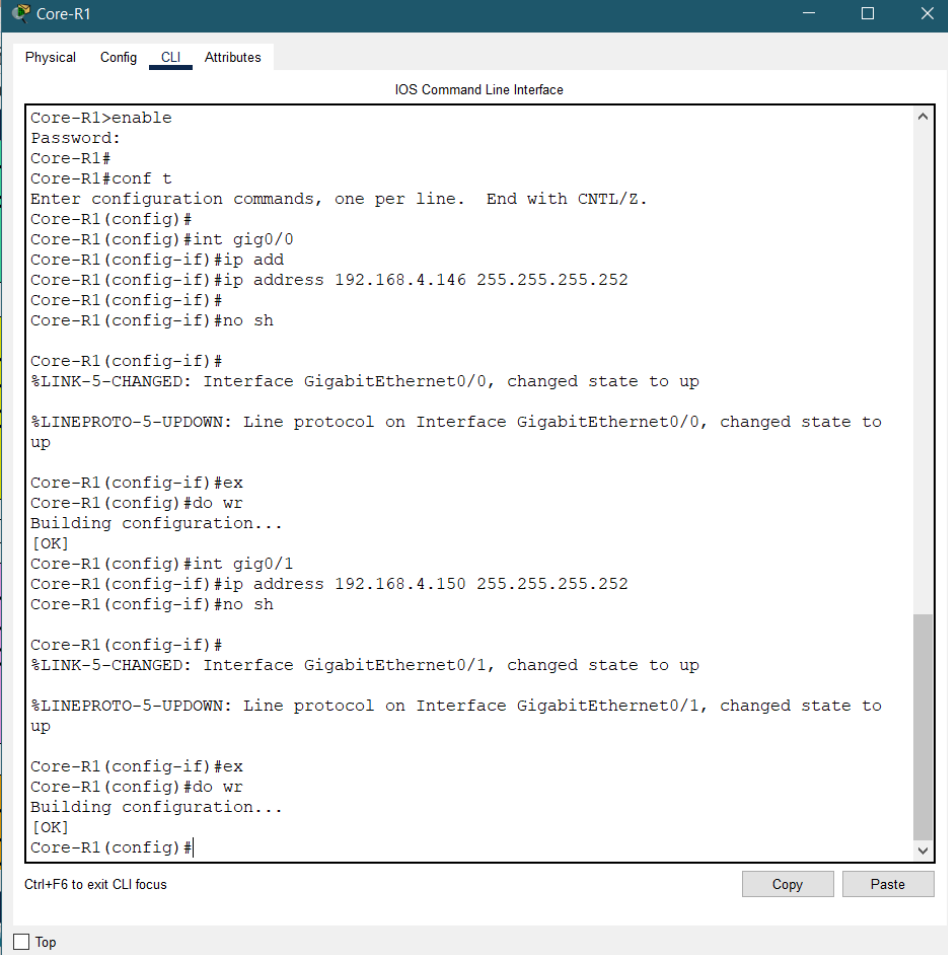
Main-Multilayer-Switch2>enable
Password:
Main-Multilayer-Switch2#
Main-Multilayer-Switch2#conf t
Enter configuration commands, one per line. End with CNTL/Z.
Main-Multilayer-Switch2(config)#
Main-Multilayer-Switch2(config)#int range gig1/0/1-2
Main-Multilayer-Switch2(config-if-range)#no swi
Main-Multilayer-Switch2(config-if-range)#no switchport
Main-Multilayer-Switch2(config-if-range)#e
Main-Multilayer-Switch2(config)#
Main-Multilayer-Switch2(config)#int gig1/0/1
Main-Multilayer-Switch2(config-if)#
Main-Multilayer-Switch2(config-if)#ip address 192.168.4.149 255.255.255.252
Main-Multilayer-Switch2(config-if)#
Main-Multilayer-Switch2(config-if)#exit
Main-Multilayer-Switch2(config)#
Main-Multilayer-Switch2(config)#int gig1/0/2
Main-Multilayer-Switch2(config-if)#
Main-Multilayer-Switch2(config-if)#ip address 192.168.4.157 255.255.255.252
Main-Multilayer-Switch2(config-if)#ex
Main-Multilayer-Switch2(config)#do wr
Building configuration...
Compressed configuration from 7383 bytes to 3601 bytes[OK]
[OK]
Main-Multilayer-Switch2(config)#int range gig1/0/1-2
Main-Multilayer-Switch2(config-if-range)#no sh
Main-Multilayer-Switch2(config-if-range)#ex
Main-Multilayer-Switch2(config)#do wr
Building configuration...
Compressed configuration from 7383 bytes to 3601 bytes[OK]
[OK]
Main-Multilayer-Switch2(config)#
```

Result



```
!
!
interface GigabitEthernet1/0/1
no switchport
ip address 192.168.4.149 255.255.255.252
duplex auto
speed auto
!
interface GigabitEthernet1/0/2
no switchport
ip address 192.168.4.157 255.255.255.252
duplex auto
speed auto
!
interface GigabitEthernet1/0/3
switchport mode trunk
!
interface GigabitEthernet1/0/4
switchport mode trunk
!
interface GigabitEthernet1/0/5
switchport mode trunk
!
interface GigabitEthernet1/0/6
```

12.1.2 Core Router's Interfaces to Layer 3 Switch Interfaces



The screenshot shows the Core-R1 CLI window with the following commands and output:

```
Core-R1>enable
Password:
Core-R1#
Core-R1#conf t
Enter configuration commands, one per line. End with CNTL/Z.
Core-R1(config)#
Core-R1(config)#int gig0/0
Core-R1(config-if)#ip add
Core-R1(config-if)#ip address 192.168.4.146 255.255.255.252
Core-R1(config-if)#
Core-R1(config-if)#no sh

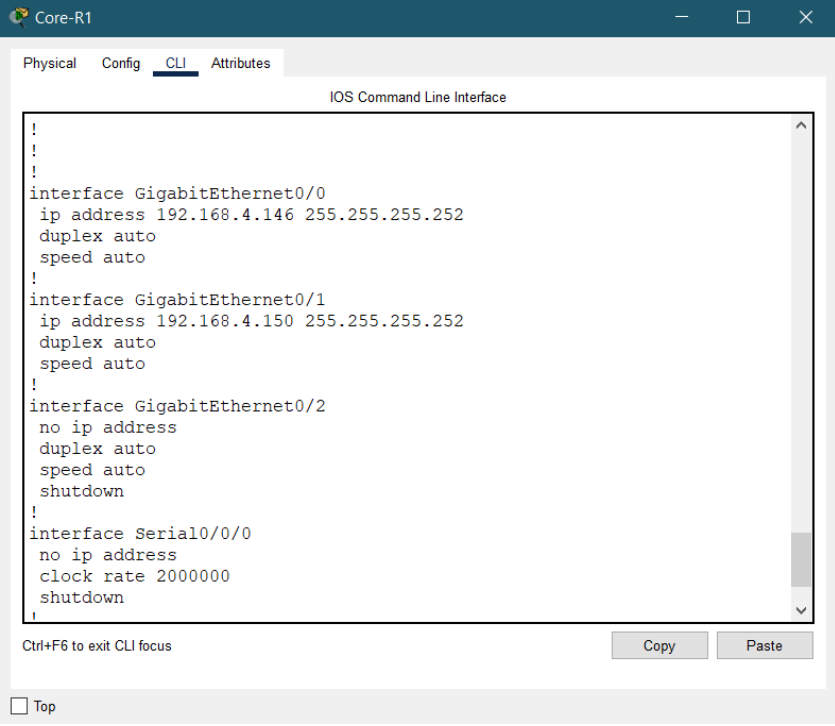
Core-R1(config-if)#
%LINK-5-CHANGED: Interface GigabitEthernet0/0, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/0, changed state to up

Core-R1(config-if)#ex
Core-R1(config)#do wr
Building configuration...
[OK]
Core-R1(config)#int gig0/1
Core-R1(config-if)#ip address 192.168.4.150 255.255.255.252
Core-R1(config-if)#no sh

Core-R1(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

Core-R1(config-if)#ex
Core-R1(config)#do wr
Building configuration...
[OK]
Core-R1(config)#
```

At the bottom of the window, there is a "Ctrl+F6 to exit CLI focus" message and "Copy" and "Paste" buttons. A "Top" button is also visible at the bottom left.



The screenshot shows the Core-R1 CLI window with the following configuration commands:

```
!
!
!
interface GigabitEthernet0/0
ip address 192.168.4.146 255.255.255.252
duplex auto
speed auto
!
interface GigabitEthernet0/1
ip address 192.168.4.150 255.255.255.252
duplex auto
speed auto
!
interface GigabitEthernet0/2
no ip address
duplex auto
speed auto
shutdown
!
interface Serial0/0/0
no ip address
clock rate 2000000
shutdown
!
```

At the bottom of the window, there is a "Ctrl+F6 to exit CLI focus" message and "Copy" and "Paste" buttons. A "Top" button is also visible at the bottom left.

12.1.3 Core Router's Interfaces to ISPs Router's Interfaces

The screenshot shows the Core-R2 IOS Command Line Interface with the CLI tab selected. The interface displays the following commands and output:

```
User Access Verification
Password:

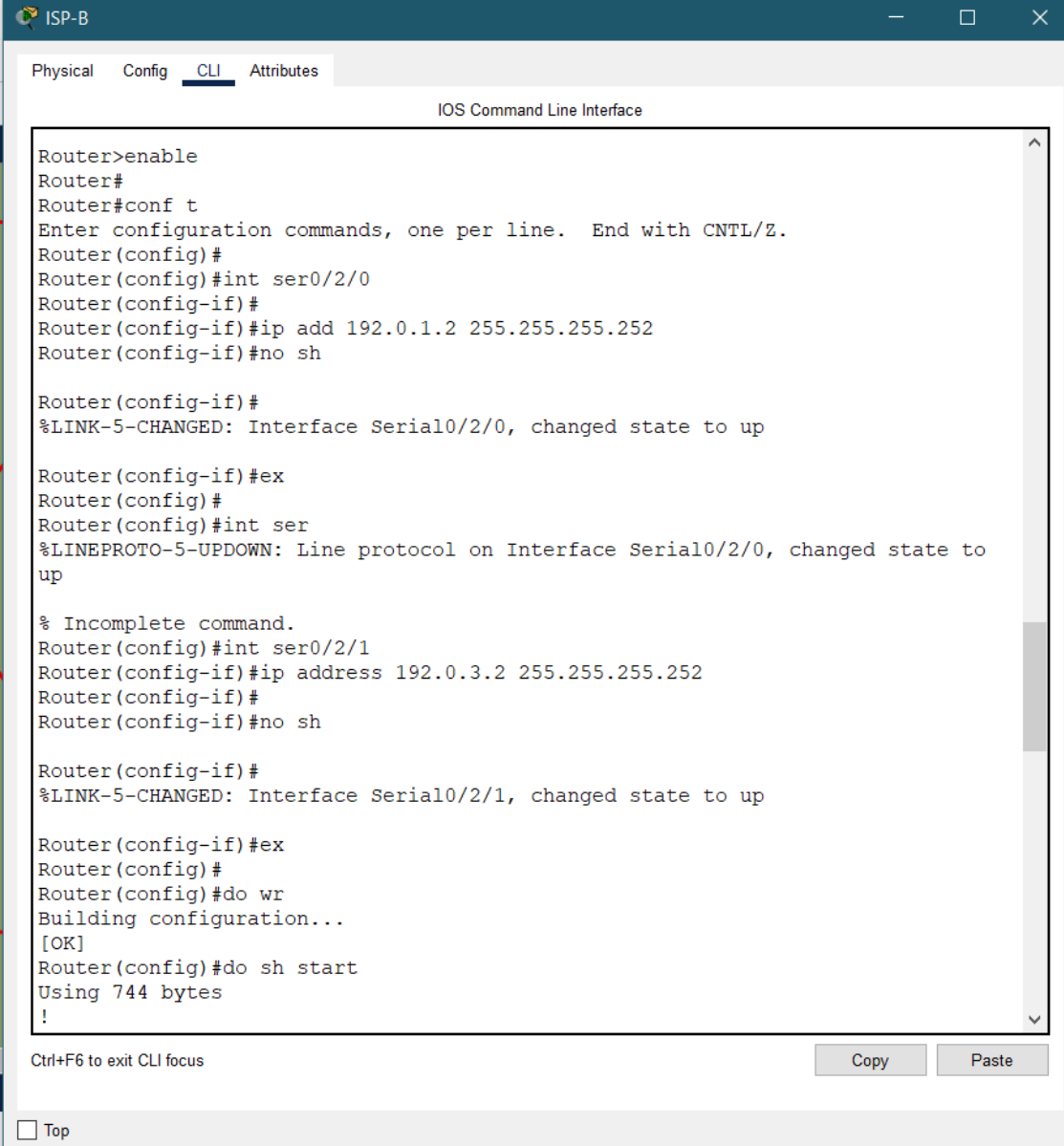
Core-R2>enable
Password:
Core-R2#
Core-R2#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Core-R2(config)#int range ser
Core-R2(config)#int range seria
Core-R2(config)#int ser
Core-R2(config)#int serial 0/0/0
Core-R2(config-if)#ip address 192.0.2.1 255.255.255.252
Core-R2(config-if)#no sh

%LINK-5-CHANGED: Interface Serial0/0/0, changed state to down
Core-R2(config-if)#ex
Core-R2(config)#
Core-R2(config)#int ser
Core-R2(config)#int serial 0/0/1
Core-R2(config-if)#
Core-R2(config-if)#ip add 192.0.3.1 255.255.255.252
Core-R2(config-if)#
Core-R2(config-if)#ex
Core-R2(config)#do wr
Building configuration...
[OK]
Core-R2(config)#int serial 0/0/1
Core-R2(config-if)#no sh

%LINK-5-CHANGED: Interface Serial0/0/1, changed state to down
Core-R2(config-if)#|
```

At the bottom of the CLI window, there is a status bar with the text "Ctrl+F6 to exit CLI focus" and two buttons: "Copy" and "Paste". Below the CLI window, there is a checkbox labeled "Top".

12.1.4 ISPs Router's Interfaces to Core Router's Interfaces



The screenshot shows a web-based interface for a router named 'ISP-B'. The 'CLI' tab is selected, displaying the 'IOS Command Line Interface'. The terminal window shows the following sequence of commands and outputs:

```
Router>enable
Router#
Router#conf t
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#
Router(config)#int ser0/2/0
Router(config-if)#
Router(config-if)#ip add 192.0.1.2 255.255.255.252
Router(config-if)#no sh

Router(config-if)#
%LINK-5-CHANGED: Interface Serial0/2/0, changed state to up

Router(config-if)#ex
Router(config)#
Router(config)#int ser
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial0/2/0, changed state to up

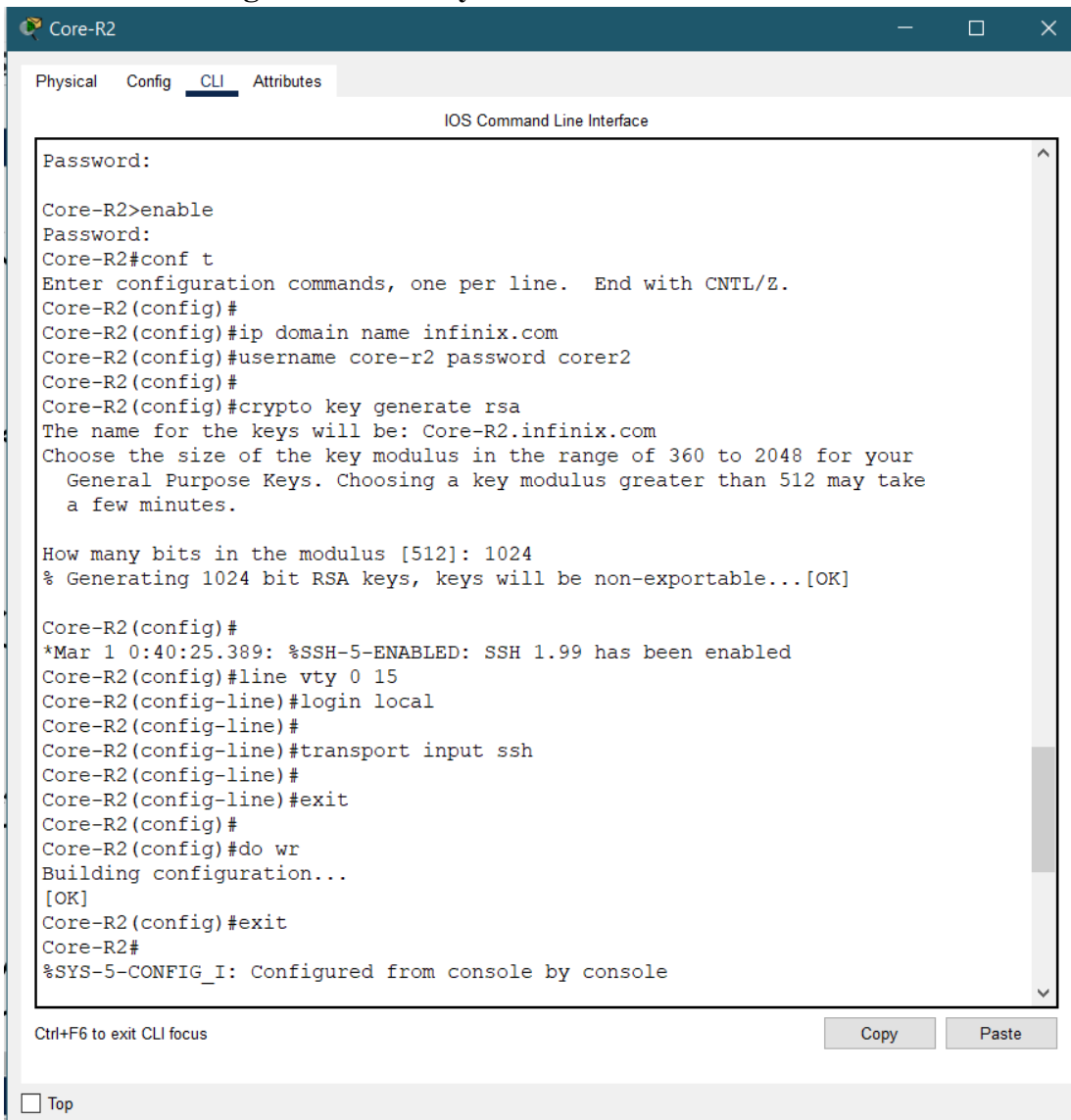
% Incomplete command.
Router(config)#int ser0/2/1
Router(config-if)#ip address 192.0.3.2 255.255.255.252
Router(config-if)#
Router(config-if)#no sh

Router(config-if)#
%LINK-5-CHANGED: Interface Serial0/2/1, changed state to up

Router(config-if)#ex
Router(config)#
Router(config)#do wr
Building configuration...
[OK]
Router(config)#do sh start
Using 744 bytes
!
```

At the bottom of the CLI window, there is a status bar with the text 'Ctrl+F6 to exit CLI focus' and two buttons: 'Copy' and 'Paste'. Below the CLI window, there is a checkbox labeled 'Top'.

13.1.4 SSH configuration on Layer 3 Routers



The screenshot shows a terminal window titled 'Core-R2' with tabs for Physical, Config, CLI, and Attributes. The CLI tab is active, displaying the 'IOS Command Line Interface'. The terminal output shows the following sequence of commands and responses:

```
Core-R2>enable
Password:
Core-R2#conf t
Enter configuration commands, one per line. End with CNTL/Z.
Core-R2(config)#
Core-R2(config)#ip domain name infinix.com
Core-R2(config)#username core-r2 password corer2
Core-R2(config)#
Core-R2(config)#crypto key generate rsa
The name for the keys will be: Core-R2.infinix.com
Choose the size of the key modulus in the range of 360 to 2048 for your
General Purpose Keys. Choosing a key modulus greater than 512 may take
a few minutes.

How many bits in the modulus [512]: 1024
% Generating 1024 bit RSA keys, keys will be non-exportable...[OK]

Core-R2(config)#
*Mar 1 0:40:25.389: %SSH-5-ENABLED: SSH 1.99 has been enabled
Core-R2(config)#line vty 0 15
Core-R2(config-line)#login local
Core-R2(config-line)#
Core-R2(config-line)#transport input ssh
Core-R2(config-line)#
Core-R2(config-line)#exit
Core-R2(config)#
Core-R2(config)#do wr
Building configuration...
[OK]
Core-R2(config)#exit
Core-R2#
%SYS-5-CONFIG_I: Configured from console by console
```

At the bottom of the terminal window, there is a status bar with the text 'Ctrl+F6 to exit CLI focus' on the left, and 'Copy' and 'Paste' buttons on the right. Below the terminal window, there is a 'Top' button with a checkbox icon.

Including RSA encryption with login to the local database

14.1 OSPF Configuration

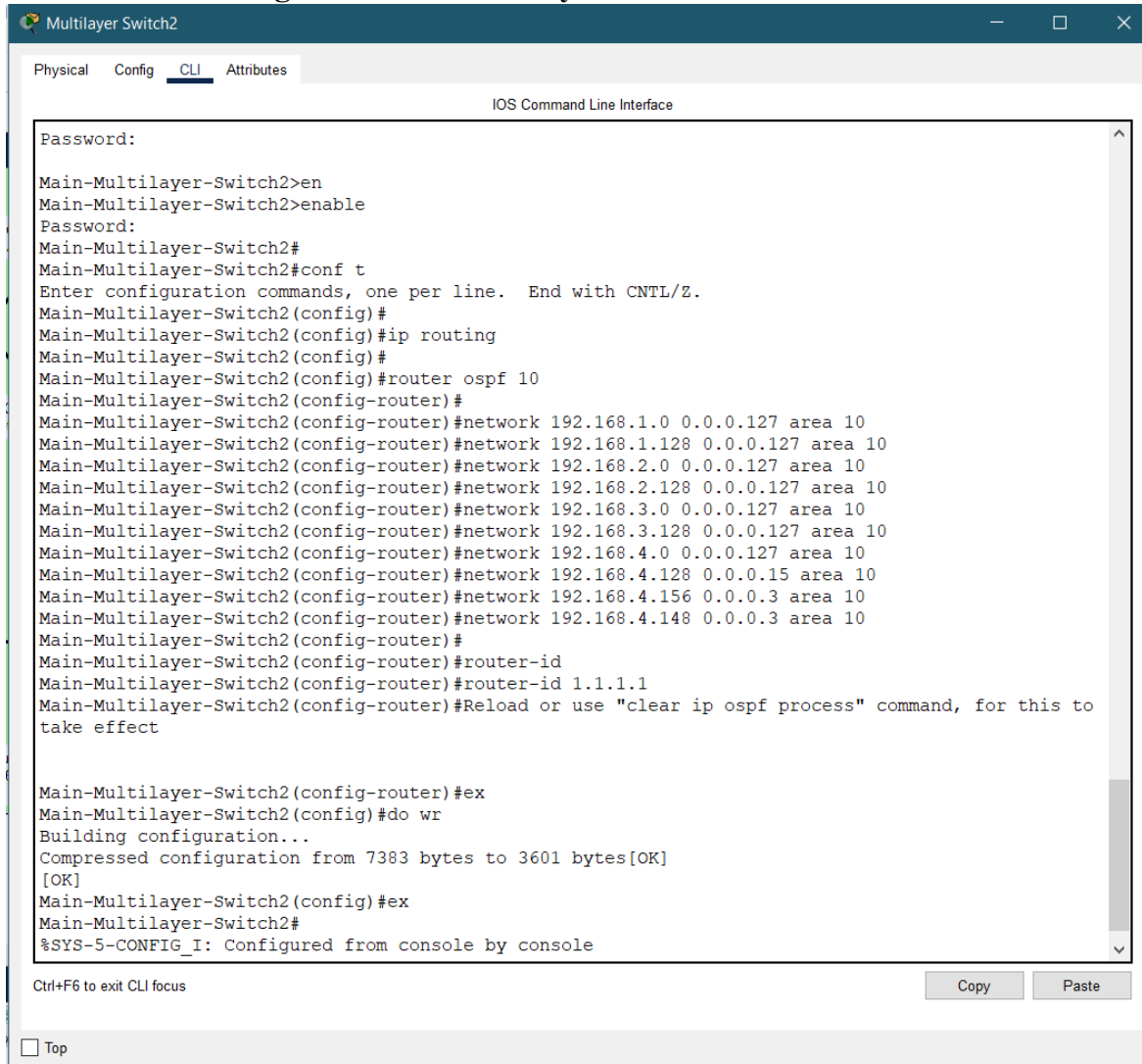
14.1.1 Overview about OSPF

OSPF stands for Open Shortest Path First, and it is a routing protocol used in computer networks to determine the best paths for forwarding network traffic. It is an interior gateway protocol (IGP) commonly used within an autonomous system (AS), such as an enterprise network or an internet service provider's network.

14.1.2 Key points about OSPF:

1. Link-State Protocol
2. Dynamic Routing Updates
3. Shortest Path Calculation
4. Load Balancing

14.1.3 OSPF Configuration on Multilayer Switches



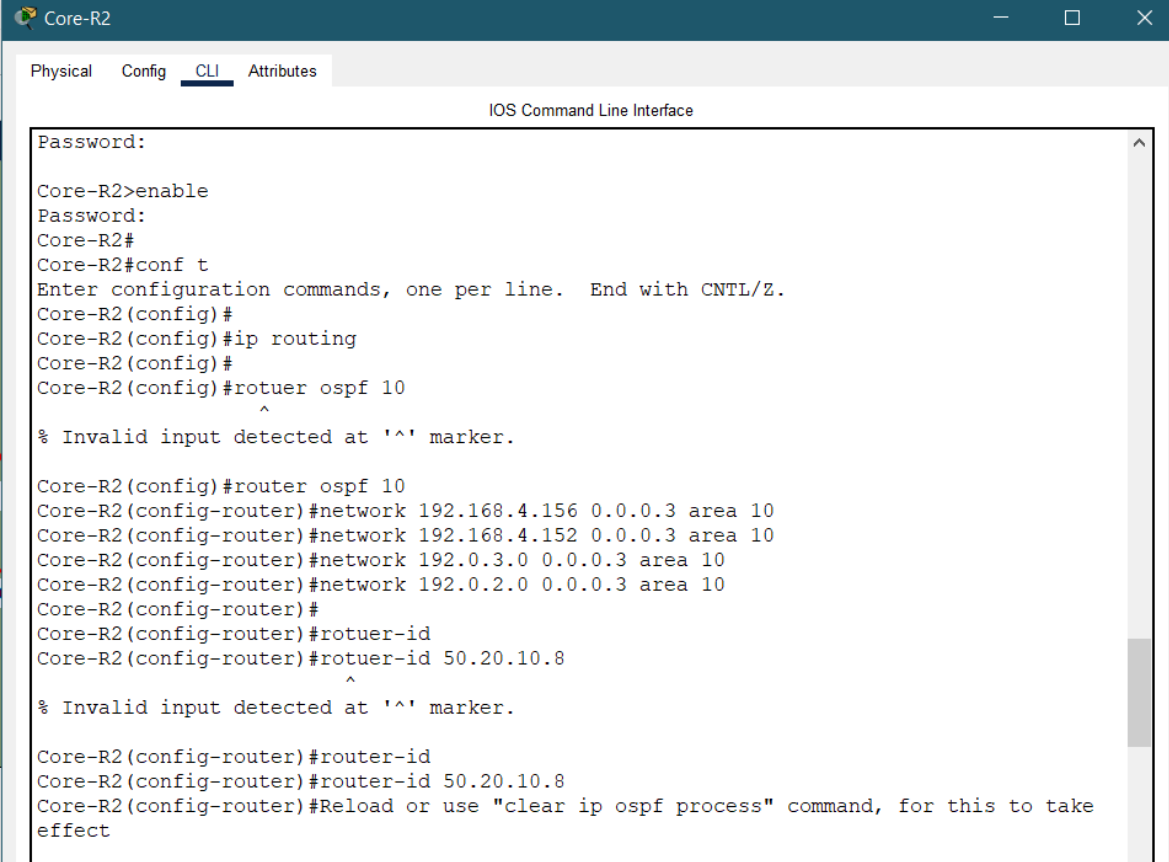
```
Multilayer Switch2
Physical Config CLI Attributes
IOS Command Line Interface

Password:
Main-Multilayer-Switch2>en
Main-Multilayer-Switch2>enable
Password:
Main-Multilayer-Switch2#
Main-Multilayer-Switch2#conf t
Enter configuration commands, one per line. End with CNTL/Z.
Main-Multilayer-Switch2(config)#
Main-Multilayer-Switch2(config)#ip routing
Main-Multilayer-Switch2(config)#
Main-Multilayer-Switch2(config)#router ospf 10
Main-Multilayer-Switch2(config-router)#
Main-Multilayer-Switch2(config-router)#network 192.168.1.0 0.0.0.127 area 10
Main-Multilayer-Switch2(config-router)#network 192.168.1.128 0.0.0.127 area 10
Main-Multilayer-Switch2(config-router)#network 192.168.2.0 0.0.0.127 area 10
Main-Multilayer-Switch2(config-router)#network 192.168.2.128 0.0.0.127 area 10
Main-Multilayer-Switch2(config-router)#network 192.168.3.0 0.0.0.127 area 10
Main-Multilayer-Switch2(config-router)#network 192.168.3.128 0.0.0.127 area 10
Main-Multilayer-Switch2(config-router)#network 192.168.4.0 0.0.0.127 area 10
Main-Multilayer-Switch2(config-router)#network 192.168.4.128 0.0.0.15 area 10
Main-Multilayer-Switch2(config-router)#network 192.168.4.156 0.0.0.3 area 10
Main-Multilayer-Switch2(config-router)#network 192.168.4.148 0.0.0.3 area 10
Main-Multilayer-Switch2(config-router)#
Main-Multilayer-Switch2(config-router)#router-id
Main-Multilayer-Switch2(config-router)#router-id 1.1.1.1
Main-Multilayer-Switch2(config-router)#Reload or use "clear ip ospf process" command, for this to
take effect

Main-Multilayer-Switch2(config-router)#ex
Main-Multilayer-Switch2(config)#do wr
Building configuration...
Compressed configuration from 7383 bytes to 3601 bytes[OK]
[OK]
Main-Multilayer-Switch2(config)#ex
Main-Multilayer-Switch2#
%SYS-5-CONFIG_I: Configured from console by console

Ctrl+F6 to exit CLI focus
Copy Paste
Top
```

14.1.4 OSPF Configuration on Layer3 Routers



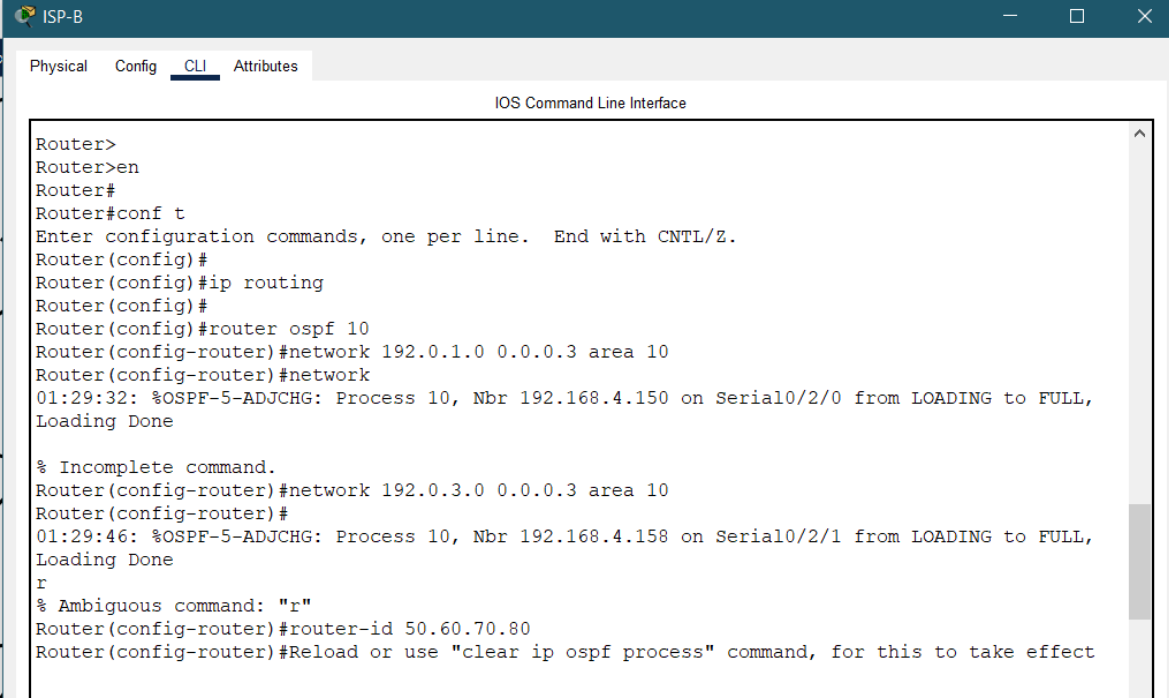
The screenshot shows the CLI of a router named Core-R2. The user enters 'enable' to get into privileged mode, then 'conf t' to enter configuration mode. They attempt to enter 'router ospf 10' but make a typo as 'rotuer ospf 10', which results in an 'Invalid input detected at '^' marker.' error. They then correct it to 'router ospf 10'. Next, they enter four 'network' commands to advertise the interfaces 192.168.4.156, 192.168.4.152, 192.0.3.0, and 192.0.2.0 into area 10. They then attempt to set the router ID with 'rotuer-id 50.20.10.8' (another typo for 'router-id'), which also results in an 'Invalid input detected at '^' marker.' error. Finally, they enter 'router-id 50.20.10.8' correctly. The output shows the OSPF process starting and loading the configuration.

```
Core-R2>enable
Password:
Core-R2#
Core-R2#conf t
Enter configuration commands, one per line. End with CNTL/Z.
Core-R2(config)#
Core-R2(config)#ip routing
Core-R2(config)#
Core-R2(config)#rotuer ospf 10
^
% Invalid input detected at '^' marker.

Core-R2(config)#router ospf 10
Core-R2(config-router)#network 192.168.4.156 0.0.0.3 area 10
Core-R2(config-router)#network 192.168.4.152 0.0.0.3 area 10
Core-R2(config-router)#network 192.0.3.0 0.0.0.3 area 10
Core-R2(config-router)#network 192.0.2.0 0.0.0.3 area 10
Core-R2(config-router)#
Core-R2(config-router)#rotuer-id
Core-R2(config-router)#rotuer-id 50.20.10.8
^
% Invalid input detected at '^' marker.

Core-R2(config-router)#router-id
Core-R2(config-router)#router-id 50.20.10.8
Core-R2(config-router)#Reload or use "clear ip ospf process" command, for this to take effect
```

14.1.5 OSPF Configuration on ISPs Routers

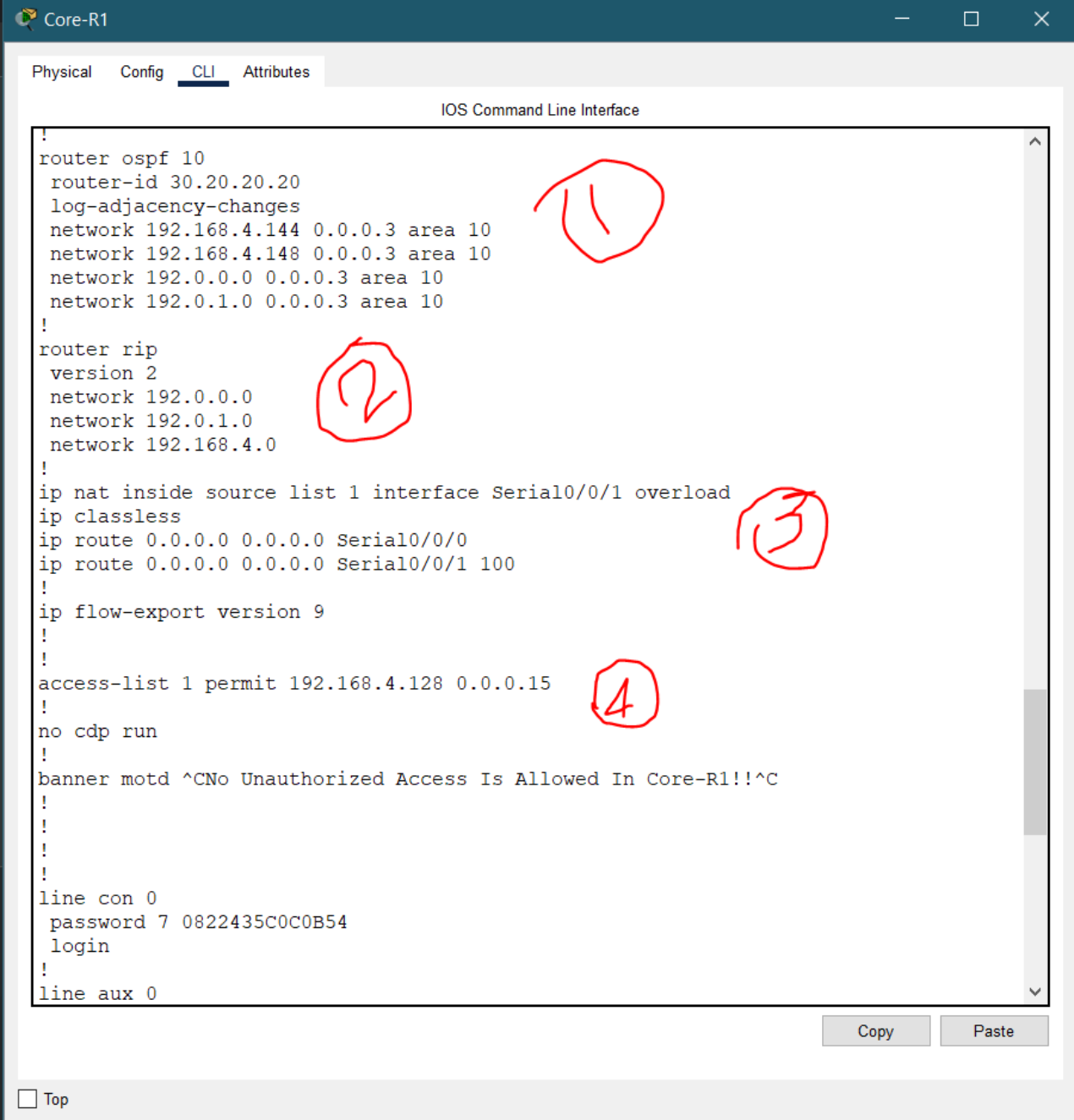


The screenshot shows the CLI of a router named ISP-B. The user enters 'en' to get into privileged mode, then 'conf t' to enter configuration mode. They enter 'ip routing' and then 'router ospf 10'. They enter two 'network' commands for 192.0.1.0 and 192.0.3.0 in area 10. The output shows the OSPF process starting and loading the configuration. They then attempt to enter 'r' for 'router-id', which results in an 'Ambiguous command: "r"' error. Finally, they enter 'router-id 50.60.70.80' correctly. The output shows the OSPF process starting and loading the configuration.

```
Router>
Router>en
Router#
Router#conf t
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#
Router(config)#ip routing
Router(config)#
Router(config)#router ospf 10
Router(config-router)#network 192.0.1.0 0.0.0.3 area 10
Router(config-router)#network
01:29:32: %OSPF-5-ADJCHG: Process 10, Nbr 192.168.4.150 on Serial0/2/0 from LOADING to FULL,
Loading Done

% Incomplete command.
Router(config-router)#network 192.0.3.0 0.0.0.3 area 10
Router(config-router)#
01:29:46: %OSPF-5-ADJCHG: Process 10, Nbr 192.168.4.158 on Serial0/2/1 from LOADING to FULL,
Loading Done
r
% Ambiguous command: "r"
Router(config-router)#router-id 50.60.70.80
Router(config-router)#Reload or use "clear ip ospf process" command, for this to take effect
```

14.1.6 OSPF, NAT Overload, RIP, ACL Show Results



Core-R1

Physical Config CLI Attributes

IOS Command Line Interface

```
!
router ospf 10
router-id 30.20.20.20
log-adjacency-changes
network 192.168.4.144 0.0.0.3 area 10
network 192.168.4.148 0.0.0.3 area 10
network 192.0.0.0 0.0.0.3 area 10
network 192.0.1.0 0.0.0.3 area 10
!
router rip
version 2
network 192.0.0.0
network 192.0.1.0
network 192.168.4.0
!
ip nat inside source list 1 interface Serial0/0/1 overload
ip classless
ip route 0.0.0.0 0.0.0.0 Serial0/0/0
ip route 0.0.0.0 0.0.0.0 Serial0/0/1 100
!
ip flow-export version 9
!
!
access-list 1 permit 192.168.4.128 0.0.0.15
!
no cdp run
!
banner motd ^CNo Unauthorized Access Is Allowed In Core-R1!!^C
!
!
!
line con 0
password 7 0822435C0C0B54
login
!
line aux 0
```

Copy Paste

☐ Top

Handwritten annotations: 1 (next to OSPF network commands), 2 (next to RIP network commands), 3 (next to NAT configuration), 4 (next to ACL configuration).

15.1 Servers Infrastructure and configuration

15.1.1 DHCP Server: is a network server that automatically assigns IP addresses and network configuration parameters to devices on a network. It dynamically manages the allocation of IP addresses and other network settings to clients, simplifying the process of network configuration and management.

DHCP

PhysicalConfigServicesDesktopProgrammingAttributes

SERVICES

HTTP

DHCP

DHCPv6

TFTP

DNS

SYSLOG

AAA

NTP

EMAIL

FTP

IoT

VM Management

Radius EAP

DHCP

InterfaceFastEthernet0ServiceOnOff

Pool NameserverPool

Default Gateway0.0.0.0

DNS Server0.0.0.0

Start IP Address :1921684128

Subnet Mask:255255255240

Maximum Number of Users :512

TFTP Server:0.0.0.0

WLC Address:0.0.0.0

AddSaveRemove

Pool Name	Default Gateway	DNS Server	Start IP Address	Subnet Mask	Max User	TFTP Server	WLC Address
Break Area	192.168.5.1	192.168.4.134	192.168.5.2	255.255.255.0	254	0.0.0.0	0.0.0.0
IT	192.168.4.1	192.168.4.134	192.168.4.2	255.255.255.128	126	0.0.0.0	0.0.0.0
HR	192.168.3.129	192.168.4.134	192.168.3.130	255.255.255.128	126	0.0.0.0	0.0.0.0
Acc/Finance	192.168.3.1	192.168.4.134	192.168.3.2	255.255.255.128	126	0.0.0.0	0.0.0.0
Marketing	192.168.2.129	192.168.4.134	192.168.2.130	255.255.255.128	126	0.0.0.0	0.0.0.0
Sales	192.168.2.1	192.168.4.134	192.168.2.2	255.255.255.128	126	0.0.0.0	0.0.0.0
Customer Service	192.168.1.129	192.168.4.134	192.168.1.130	255.255.255.128	126	0.0.0.0	0.0.0.0
Production	192.168.1.1	192.168.4.134	192.168.1.2	255.255.255.128	126	0.0.0.0	0.0.0.0
serverPool	0.0.0.0	0.0.0.0	192.168.4.128	255.255.255.240	512	0.0.0.0	0.0.0.0

Top

In the DHCP server, we accessed the DHCP server configuration and identified the subnets we wanted to create pools for, ensuring each subnet had a unique network address and subnet mask. We created a DHCP pool for each subnet, specifying the IP address range default gateway, DNS server, and other relevant options. We repeated this process for each subnet. We saved and applied the changes, and then tested the DHCP pools by connecting devices within the subnets to ensure proper IP address assignment. We referred to the documentation or resources specific to our DHCP server software for detailed instructions. For the IP addresses of the DHCP server, we assigned it statically.

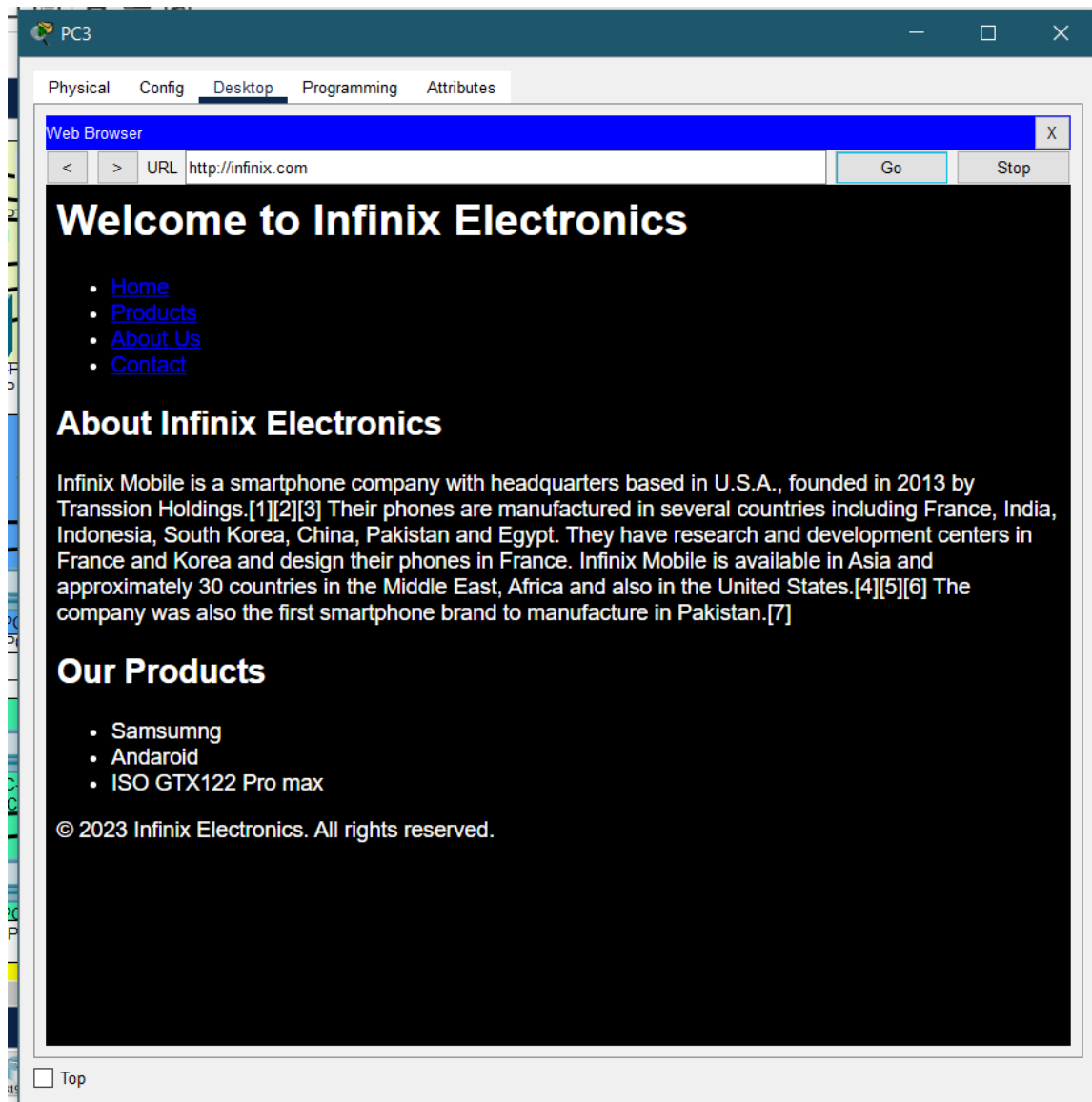
15.1.2 DNS Server: is a network server that translates domain names into their corresponding IP addresses. It acts as a directory service that allows users to access websites, send emails, or use other internet services by using easy-to-remember domain names instead of numeric IP addresses.

The screenshot shows the 'DNS Server' configuration window. The 'Services' tab is selected, and the 'DNS' service is turned 'On'. Under 'Resource Records', a record for 'infinix.com' is listed with an 'A Record' type and IP address '192.168.4.135'. The 'DNS Cache' button is visible at the bottom.

No.	Name	Type	Detail
0	infinix.com	A Record	192.168.4.135

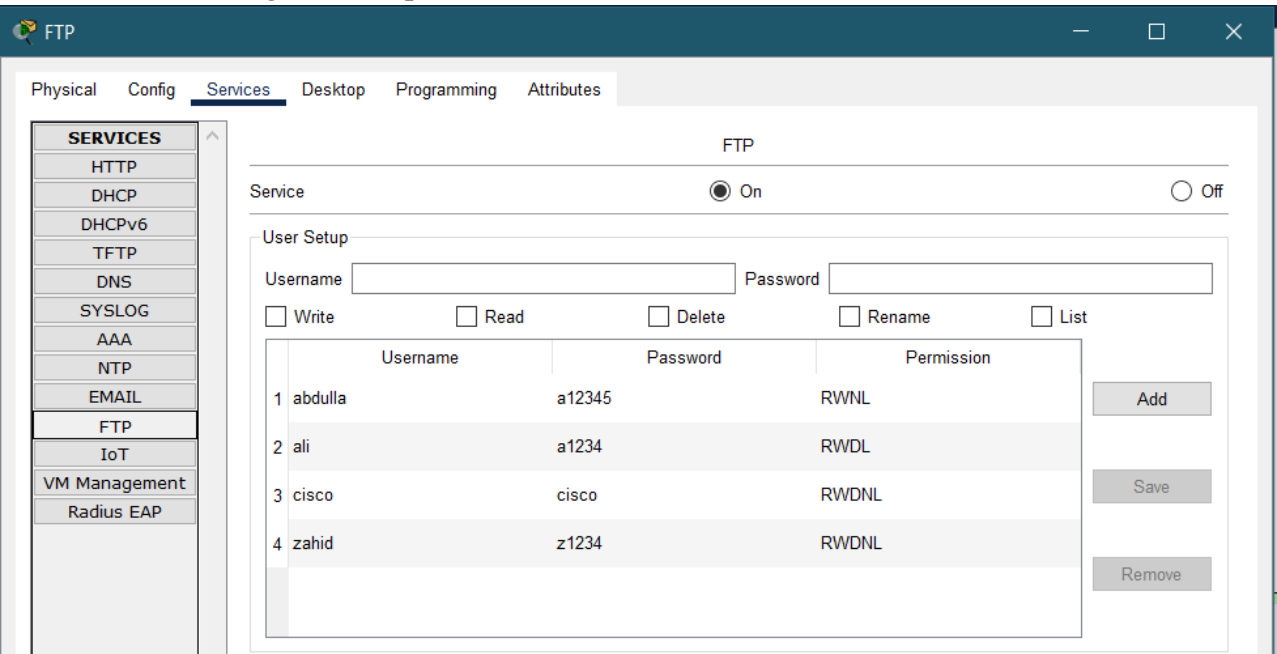
In the DNS server, we added the IP address of the web server of Infinix company. We did this to enable DNS resolution and allow clients to access the website hosted on the Infinix web server using its domain name. By associating the IP address with the domain name in the DNS server, we ensured that when users entered the Infinix website's domain name in their web browsers, the DNS server would provide the corresponding IP address, allowing the browser to establish a connection to the correct web server and retrieve the website's content.

15.1.3 Web Server is a software application that runs on a computer and serves web pages and other web content to clients upon request. It is a fundamental component of the World Wide Web, allowing websites and web applications to be accessible to users over the Internet.

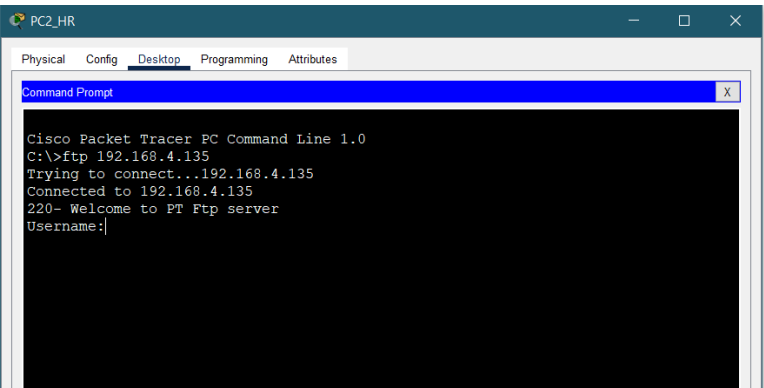


Here is the output of the web server.

15.1.4 FTP Server: is a type of server software that facilitates the transfer of files between computers over a network. It enables users to upload, download, and manage files on a remote server using the FTP protocol.



We added users to the FTP server and assigned specific permissions to them for security and access control purposes. By adding users and configuring permissions, we ensured that only authorized individuals or entities could access and interact with the files and directories stored on the FTP server.



This figure shows that the FTP server is accessible

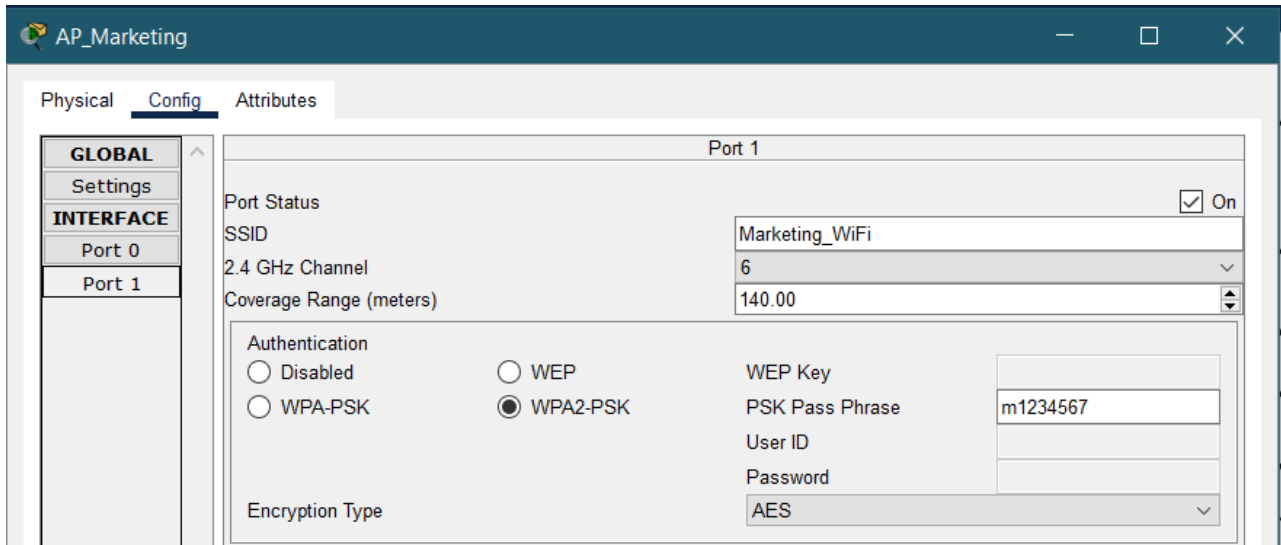
15.1.5 Email Server: is a server computer or software application that handles the sending, receiving, storage, and routing of email messages. It provides the infrastructure and protocols necessary for email communication within a network or across the internet.

The screenshot shows the 'Email' configuration page in a network device's web interface. The page has a dark blue header with the title 'Email' and standard window controls. Below the header is a navigation bar with tabs: 'Physical', 'Config', 'Services' (selected), 'Desktop', 'Programming', and 'Attributes'. On the left side, there is a 'SERVICES' menu with various options: HTTP, DHCP, DHCPv6, TFTP, DNS, SYSLOG, AAA, NTP, EMAIL (selected), FTP, IoT, VM Management, and Radius EAP. The main content area is titled 'EMAIL' and contains two sections: 'SMTP Service' and 'POP3 Service'. Both services have radio buttons for 'ON' (selected) and 'OFF'. Below these, there is a 'Domain Name' field with the value 'infinix.com' and a 'Set' button. A 'User Setup' section follows, with 'User' and 'Password' input fields. Below the input fields, a list of users is displayed: 'zahid', 'abdulla', and 'ali'.

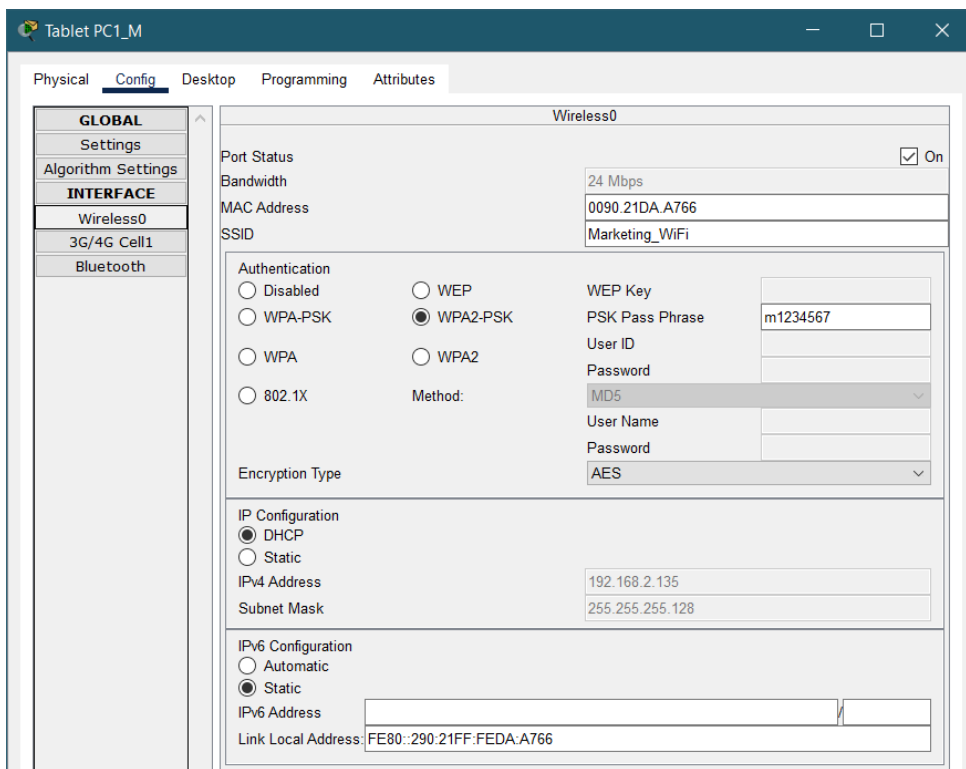
The screenshot shows the 'Desktop' configuration page in a network device's web interface. The page has a dark blue header with the title 'Desktop' and standard window controls. Below the header is a navigation bar with tabs: 'Physical', 'Config', 'Services', 'Desktop' (selected), 'Programming', and 'Attributes'. On the left side, there is a 'SERVICES' menu with various options: HTTP, DHCP, DHCPv6, TFTP, DNS, SYSLOG, AAA, NTP, EMAIL, FTP, IoT, VM Management, and Radius EAP. The main content area is titled 'IP Configuration' and contains two sections: 'IP Configuration' and 'IPv6 Configuration'. The 'IP Configuration' section has radio buttons for 'DHCP' and 'Static' (selected). Below these, there are input fields for 'IPv4 Address' (192.168.4.133), 'Subnet Mask' (255.255.255.240), 'Default Gateway' (192.168.4.129), and 'DNS Server' (192.168.4.134). The 'IPv6 Configuration' section has radio buttons for 'Automatic' and 'Static' (selected). Below these, there are input fields for 'IPv6 Address' (empty), 'Link Local Address' (FE80::200:CFF:FE82:396D), 'Default Gateway' (empty), and 'DNS Server' (empty). Below the IP configuration sections, there is a section for '802.1X' configuration. It has a checkbox for 'Use 802.1X Security' (unchecked). Below this, there is a dropdown menu for 'Authentication' (MD5) and input fields for 'Username' and 'Password'.

16.1 Wireless Configuration

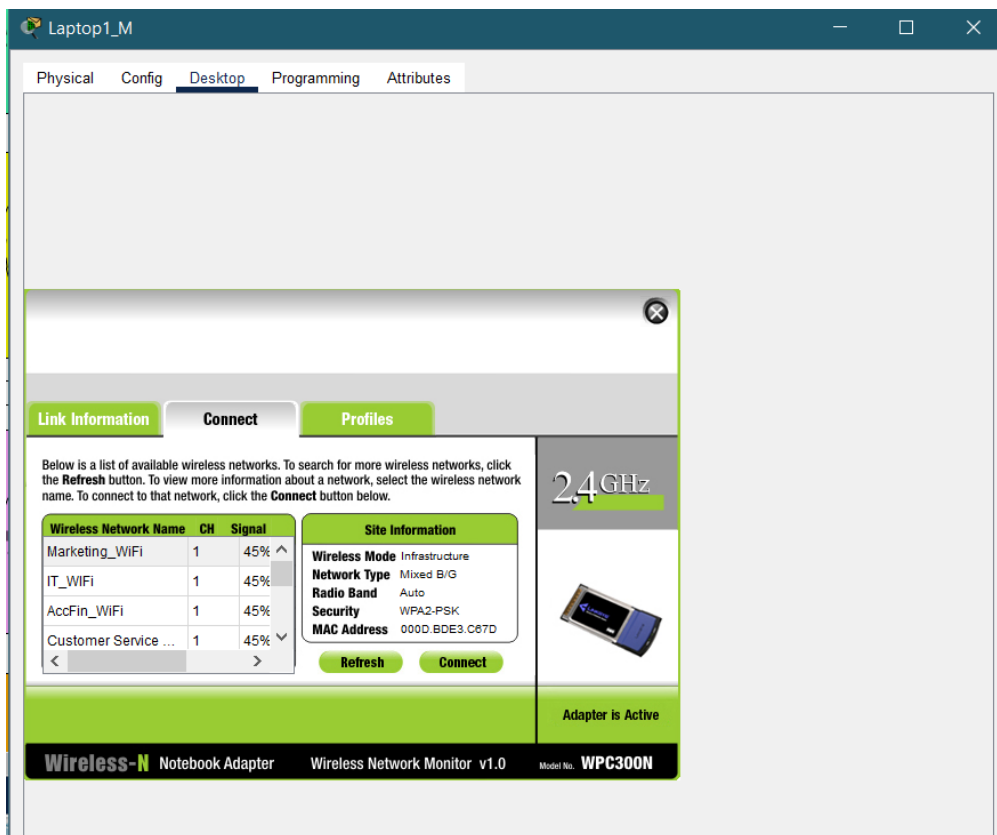
AP is a network device that allows wireless devices, such as laptops, smartphones, and tablets, to connect to a wired network and access network resources and services without requiring a physical connection.



For each AP in every subnet, we added an access point and configure its SSID (name) and setup password for authentication.



Here is a wireless host that is connected to the access point.



Here is a laptop that shows the wireless networks that are available.

17.1 IOT Door Configuration

S

Specifications Physical **Config** Attributes

GLOBAL

Settings

Algorithm Settings

Files

INTERFACE

Wireless0

Bluetooth

Global Settings

Display Name S

Serial Number PTT0810UBLD-

Interfaces Wireless0

Gateway/DNS IPv4

☒ DHCP

☐ Static

Default Gateway 192.168.2.1

DNS Server 192.168.4.134

Gateway/DNS IPv6

☐ Automatic

☒ Static

Default Gateway

DNS Server

IoT Server

☒ None

☐ Home Gateway

☐ Remote Server

Server Address


User Name

Password


Connect

ITT

SpecificationsPhysicalConfigAttributes



Zoom: 90%



Door

Open / Close / Unlock / Lock

Features:

- Registration Server Compatible
- Ability to vent Carbon Dioxide and Carbon Monoxide

Usage:

- Connect to the Door from SBC/MCU/Thing with IoT Custom Cable
- Use customWrite function to control the door and lock

Direct Control:

- ALT-click on keyhole to lock/unlock
- ALT-click on door to open/close

Local Control:

- Connect device to SBC/MCU/Thing. Use the "customWrite" API per Data Specifications

Remote Control:

- Connect device to Registration Server using Config Tab

NOTE: opening and closing the door is not remote controllable

Data Specifications:

Message Format: [door].[lock]

door: 0 = closed, 1 = open, -1 = don't care

lock: 0 = unlock, 1 = lock, -1 = don't care

Example:

Connect a SBC to the door, send a customWrite and open the door to vent the Carbon Dioxide and Carbon Monoxide level

☐ Top

Edit

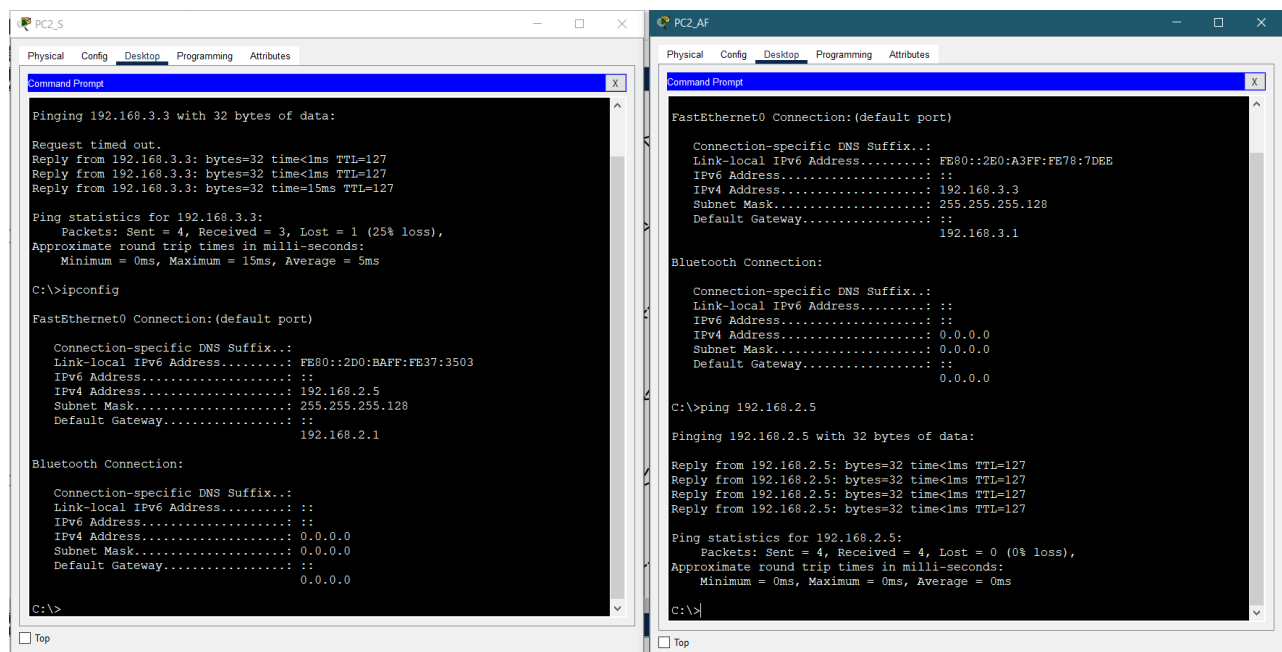
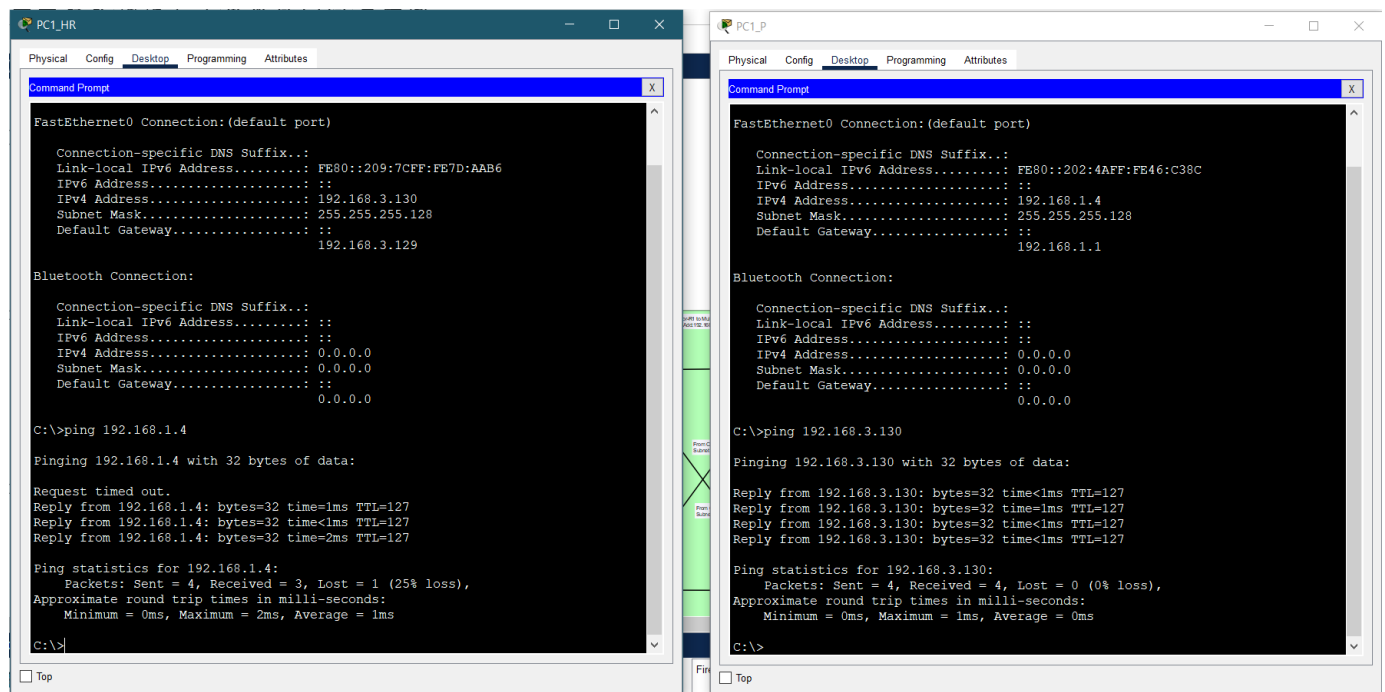
Advanced

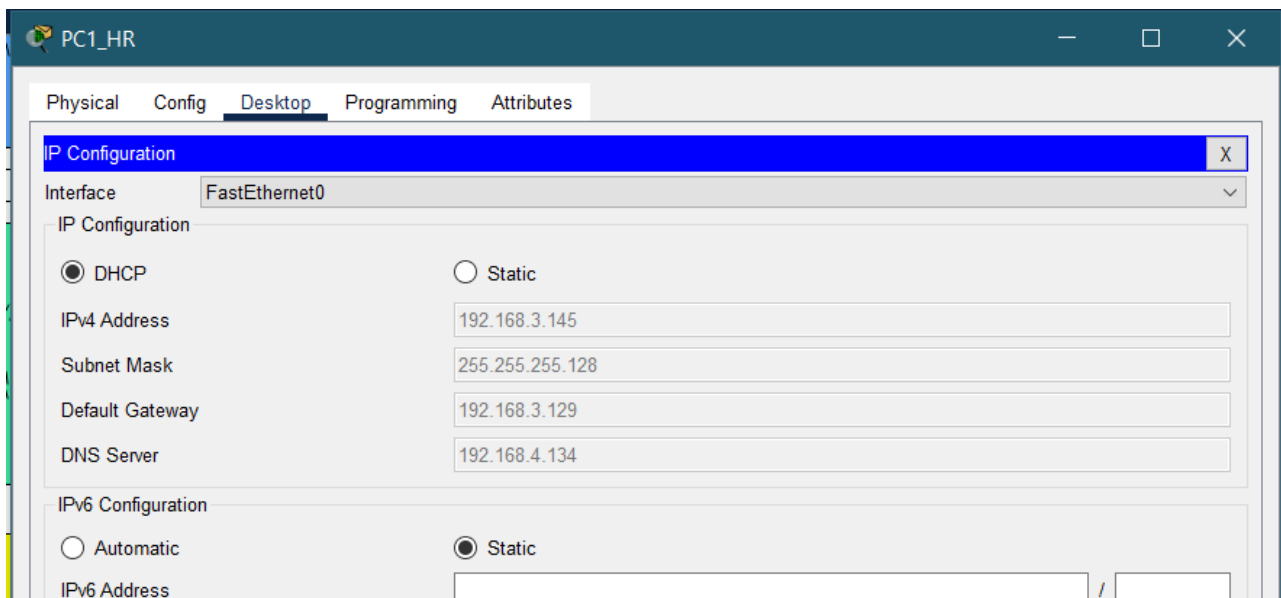
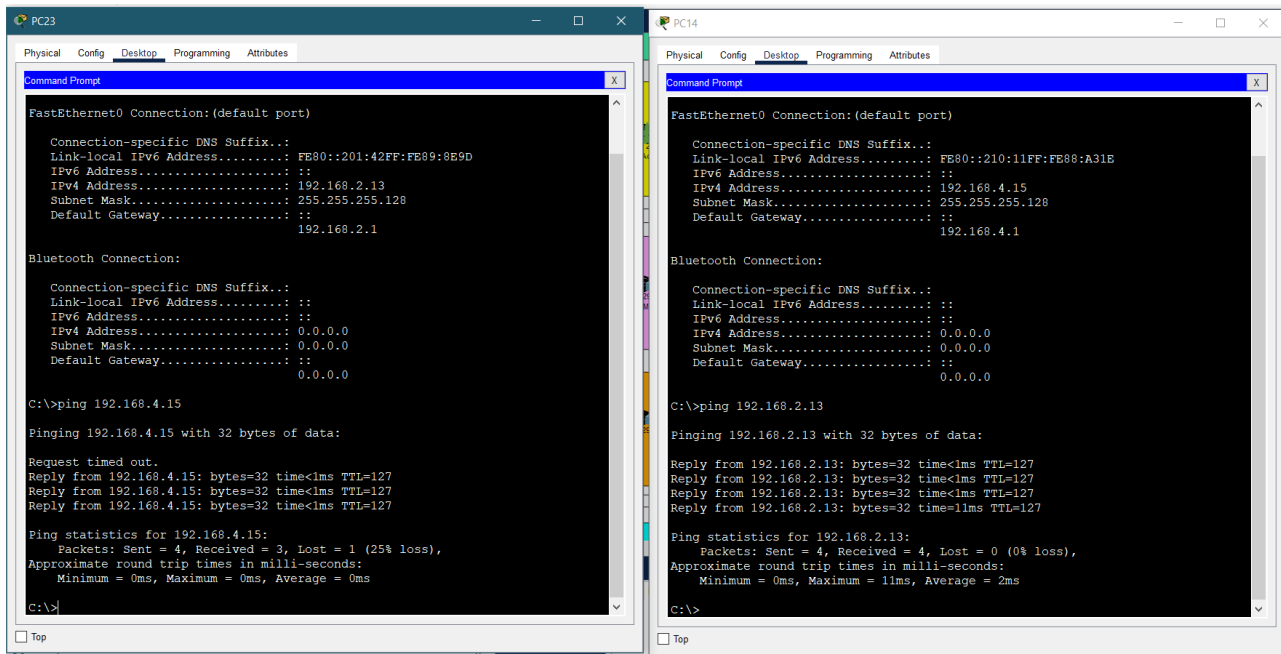
18.1 Routing Tables for the core routers, ISP routers, and multilayer switches

Routing Table for Multilayer Switch2					Routing Table for Core-R1					Routing Table for ISP-A				
Type	Network	Port	Next Hop IP	Metric	Type	Network	Port	Next Hop IP	Metric	Type	Network	Port	Next Hop IP	Metric
S	0.0.0.0/0	GigabitEthernet1/0/1	---	1/0	S	0.0.0.0/0	Serial0/0/0	---	1/0	C	192.0.0.0/30	Serial0/2/0	---	0/0
O	192.0.0.0/30	GigabitEthernet1/0/1	192.168.4.150	110/65	C	192.0.0.0/30	Serial0/0/0	---	0/0	L	192.0.0.2/32	Serial0/2/0	---	0/0
O	192.0.1.0/30	GigabitEthernet1/0/1	192.168.4.150	110/65	L	192.0.0.1/32	Serial0/0/0	---	0/0	O	192.0.1.0/30	Serial0/2/0	192.0.0.1	110/128
O	192.0.2.0/30	GigabitEthernet1/0/2	192.168.4.150	110/65	C	192.0.1.0/30	Serial0/0/1	---	0/0	C	192.0.2.0/30	Serial0/2/1	---	0/0
O	192.0.3.0/30	GigabitEthernet1/0/2	192.168.4.150	110/65	L	192.0.1.1/32	Serial0/0/1	---	0/0	L	192.0.2.2/32	Serial0/2/1	---	0/0
C	192.168.1.0/25	Vlan10	---	0/0	O	192.0.2.0/30	GigabitEthernet0/0	192.168.4.145	110/66	O	192.0.3.0/30	Serial0/2/1	192.0.2.1	110/128
C	192.168.1.128/25	Vlan20	---	0/0	O	192.0.2.0/30	GigabitEthernet0/1	192.168.4.149	110/66	O	192.168.1.0/25	Serial0/2/1	192.0.2.1	110/66
C	192.168.2.0/25	Vlan30	---	0/0	O	192.0.3.0/30	GigabitEthernet0/0	192.168.4.145	110/66	O	192.168.1.0/25	Serial0/2/0	192.0.0.1	110/66
C	192.168.2.128/25	Vlan40	---	0/0	O	192.0.3.0/30	GigabitEthernet0/1	192.168.4.149	110/66	O	192.168.1.128/25	Serial0/2/1	192.0.2.1	110/66
C	192.168.3.0/25	Vlan50	---	0/0	O	192.168.1.0/25	GigabitEthernet0/0	192.168.4.145	110/2	O	192.168.1.128/25	Serial0/2/0	192.0.0.1	110/66
C	192.168.3.128/25	Vlan60	---	0/0	O	192.168.1.0/25	GigabitEthernet0/1	192.168.4.149	110/2	O	192.168.2.0/25	Serial0/2/1	192.0.2.1	110/66
C	192.168.4.0/25	Vlan70	---	0/0	O	192.168.1.128/25	GigabitEthernet0/0	192.168.4.145	110/2	O	192.168.2.0/25	Serial0/2/0	192.0.0.1	110/66
C	192.168.4.128/28	Vlan80	---	0/0	O	192.168.1.128/25	GigabitEthernet0/1	192.168.4.149	110/2	O	192.168.2.128/25	Serial0/2/1	192.0.2.1	110/66
O	192.168.4.144/30	GigabitEthernet1/0/1	192.168.4.150	110/2	O	192.168.2.0/25	GigabitEthernet0/0	192.168.4.145	110/2	O	192.168.2.128/25	Serial0/2/0	192.0.0.1	110/66
C	192.168.4.148/30	GigabitEthernet1/0/1	---	0/0	O	192.168.2.0/25	GigabitEthernet0/1	192.168.4.149	110/2	O	192.168.3.0/25	Serial0/2/1	192.0.2.1	110/66
O	192.168.4.152/30	GigabitEthernet1/0/2	192.168.4.150	110/2	O	192.168.2.128/25	GigabitEthernet0/0	192.168.4.145	110/2	O	192.168.3.0/25	Serial0/2/0	192.0.0.1	110/66
C	192.168.4.156/30	GigabitEthernet1/0/2	---	0/0	O	192.168.2.128/25	GigabitEthernet0/1	192.168.4.149	110/2	O	192.168.3.128/25	Serial0/2/1	192.0.2.1	110/66
C	192.168.5.0/24	Vlan110	---	0/0	O	192.168.3.0/25	GigabitEthernet0/0	192.168.4.145	110/2	O	192.168.3.128/25	Serial0/2/0	192.0.0.1	110/66
					O	192.168.3.0/25	GigabitEthernet0/1	192.168.4.149	110/2	O	192.168.4.0/25	Serial0/2/1	192.0.2.1	110/66
					O	192.168.3.128/25	GigabitEthernet0/0	192.168.4.145	110/2	O	192.168.4.0/25	Serial0/2/0	192.0.0.1	110/66
					O	192.168.3.128/25	GigabitEthernet0/1	192.168.4.149	110/2	O	192.168.4.128/28	Serial0/2/1	192.0.2.1	110/66
					O	192.168.4.0/25	GigabitEthernet0/0	192.168.4.145	110/2	O	192.168.4.128/28	Serial0/2/0	192.0.0.1	110/66
					O	192.168.4.0/25	GigabitEthernet0/1	192.168.4.149	110/2	O	192.168.4.144/30	Serial0/2/0	192.0.0.1	110/65
					O	192.168.4.128/28	GigabitEthernet0/0	192.168.4.145	110/2	O	192.168.4.148/30	Serial0/2/0	192.0.0.1	110/65
					O	192.168.4.128/28	GigabitEthernet0/1	192.168.4.149	110/2	O	192.168.4.152/30	Serial0/2/1	192.0.2.1	110/65

Letter “o” specify the OSPF routing.

19.1 Testing and verifying the connectivity





As shown in this figure, the DHCP server is **enabled**.

20.1 Conclusion

In conclusion, we have designed a comprehensive network for Infinix Electronics, a global mobile electronics company, using Cisco Packet Tracer simulation software. The network encompasses four floors and incorporates various components such as routers, switches, servers, and wireless access points. Our goal was to create an efficient and secure network that meets the company's requirements for high performance, reliability, scalability, and simplified troubleshooting.

Throughout the project, we implemented VLANs, IP routing, DHCP, inter-VLAN configuration, and NAT overload for efficient address utilization. We also configured OSPF as the routing protocol for route advertisement and SSH for secure remote login. By assigning dedicated DHCP servers and implementing proper subnetting and IP addressing, we ensured effective address allocation and management.

To enhance security, we assigned password authentication for each wireless access point, enabling better control and management of the organization's wireless network. Additionally, we implemented ACLs (Access Control Lists) to control network traffic and protect against unauthorized access.

We paid particular attention to fault tolerance by incorporating redundant connections at every layer, ensuring network stability and continuity. Proper cabling and device settings were implemented for optimal performance and connectivity.

Overall, our network design and configuration in Cisco Packet Tracer met the project objectives, providing Infinix Electronics with a robust and secure network infrastructure that supports seamless communication and data transfer between departments.