Islamic University of Technology

Computer Networks Lab

CSE 4512

Lab Manual

Author:
Md. Tanvir Hossain Saikat, Junior Lecturer
S. M. Sabit Bananee, Lecturer
Department of Computer Science & Engineering
Islamic University of Technology

Contents

Lab 00: Downloading and installing Cisco Packet Tracer **1**

Objectives: . . . 1

Introduction: . . . 1

Introduction to the Cisco Packet Tracer: . . . 1

Downloading and installing Cisco Packet Tracer: . . . 2

Lab 01: Introduction to Cisco Packet Tracer & Exploring a Basic Network Topology **3**

Objectives . . . 3

Lab Description . . . 3

Part A: Getting started with Packet Tracer UI . . . 3

Part B: Device catalogue . . . 4

Part C: Ports, modules and cable types . . . 4

Part D: Device utilities and software features . . . 4

Part E: Packet Tracer Basics . . . 5

Part F: Building the Topology . . . 6

Part G: IP Configuration . . . 6

Part H: Router-to-Router Connections (Ethernet or Serial) . . . 7

Part I: Adding More Ports to a Router Using Modules . . . 8

Part J: Establishing Routing (RIP) . . . 8

Part K: Simulation & Packet Capture . . . 9

Part L: Saving Configurations . . . 9

Lab 02: IP Subnetting and Variable Length Subnet Mask (VLSM) **11**

Objectives . . . 11

Lab Description . . . 11

Part A: Subnet (Subnetwork) . . . 11

Part B: IPv4 Addressing and Subnet Masks . . . 11

Part C: CIDR (Classless Inter-Domain Routing) Notation . . . 11

Part D: Variable Length Subnet Mask (VLSM) . . . 12

Part E: Basic IPv4 Subnetting . . . 12

Part F: Variable Length Subnet Mask (VLSM) Design . . . 12

Part G: RIP Configuration and Classless Routing . . . 13

Lab 03: DHCP and NAT Configuration **15**

Objectives . . . 15

Part A: Dynamic Host Configuration Protocol (DHCP) . . . 15

Part B: Configure DHCP (using server) . . . 15

Part C: Configure DHCP (using router) . . . 17

Part A: Standard Access Control Lists (ACL) . . . 18

Part B: NAT Overview . . . 18

Part C: Prerequisites: Defining Interfaces . . . 18

Part D: Configuration A: Static NAT . . . 19

Part E: Configuration B: Dynamic NAT . . . 19

Part F: Configuration C: PAT (NAT Overload) . . . 19

Part G: Configure ACL . . . 19

Part H: Configure Static NAT . . . 20

Part I: Configure Dynamic NAT . . . 21

Lab 04: Implementing Static and Dynamic Routing (RIP & OSPF) 24
Objectives: 24
Lab Description: 24
Part A: Static Routing 24
Part B: Dynamic Routing 25
Part C: Route Redistribution 26
Part D: Configure static routing 27
Part E: Configure RIP 29
Part F: Configure OSPF 30
Part G: Configure Route Redistribution 32

Lab 05: VLAN Configuration and Inter-VLAN Routing 34
Objectives: 34
Lab Description: 34
Part A: VLAN 34
Part B: Inter-VLAN Routing 37
Part C: Configure inter-VLAN routing using Router-on-a-Stick approach: 39

List of Figures

1 A sample network topology for the configuration of DHCP. 16
2 A sample network topology for the configuration of ACL. 20
3 A sample network topology for the configuration of NAT. 20
4 A sample network topology for the configuration of static routing. 27
5 A sample network topology for the configuration of RIP. 29
6 A sample network topology for the configuration of OSPF. 30
7 A sample network topology for Route Redistribution. 32
8 A Sample Network with VLANs 35
9 A logical representation of VLANs. 36
10 Access links and trunk links in a sample VLAN. 37
11 An example of a Router-on-a-Stick approach to inter-VLAN routing. 38
12 An example of an MLS approach to inter-VLAN routing. 38
13 A sample network topology for configuring inter-Vlan routing using the router-on-a-stick approach. 39

Lab 00: Downloading and installing Cisco Packet Tracer

Objectives:

- Get an introductory idea on the Cisco Packet Tracer.
- Download and install the Cisco Packet Tracer.

Introduction:

In this section (Section 1), you'll be working on simulating different real-life networks and understanding the networking concepts introduced in the theory course. Over the course of completing each lab, you will get to know how to build a network from scratch by fulfilling many constraints

We will focus on the Cisco Packet Tracer for this section. We have intentionally selected Cisco Packet Tracer due to its widespread industry adoption and relevance to your future career. Engaging with this network simulator will provide you with a thorough understanding of modern network components and help you develop practical skills essential for success in Network Professional certifications, such as CCNA, CCNP, and others that require proficiency in Cisco Packet Tracer.

Introduction to the Cisco Packet Tracer:

As the introduction mentions, the Cisco Packet Tracer is a widely used network simulator that simulates and visualizes different network topologies and components. It provides an easy-to-use drag-and-drop interface that enables users to create complex network topologies easily. Users can configure all types of proprietary Cisco devices using a simulated command line interface and simulate different application layers and basic routing protocols within those devices. In an educational setting, Cisco Packet Tracer is a good fit as it allows the designing of complex and large networks. Although it can't simulate production-level networks due to a limited feature set, the Cisco Packet Tracer is suitable for an introductory understanding of basic networking principles and their applicability in practical scenarios. You can consult Cisco's official webpage and Packet Tracer Wikipedia page for further reading.

Downloading and installing Cisco Packet Tracer:

1. To download the software, create an account on Cisco Networking Academy or Cisco Netacad: Go to https://www.netacad.com/courses/getting-started-cisco-packet-tracer and sign up.
2. Then go to https://www.netacad.com/resources/lab-downloads to download the most recent version (which is 9.0.0 at the time of writing) of packet tracer. Make sure you select the correct OS version for the software.
3. Launch the Packet Tracer installer and follow the on-screen instructions to complete the installation.
4. After installation, launch Cisco Packet Tracer by selecting its icon.
5. When launching Packet Tracer for the first time, you may be prompted to enable the multi-user setup. Select "Yes."
6. In the next window, enable "Keep me logged in (for 3 months)" and "Worldwide" as the server (just below the login button). Then click the Login button. This will take you to the authentication page in your browser.
7. When prompted, provide your credentials. Upon successful login, a confirmation message will appear; you can then close the browser window.
8. Cisco Packet Tracer will launch and you are ready to explore its features.
   If you need additional guidance, please refer to the Cisco Packet Tracer Download and Installation Instructions.
9. After installation, log in as a regular user using the credentials you created in Step i. Please avoid using the guest account, as it has a file limit of three. If you've followed all the steps correctly, you're ready to begin your work.
10. If you are using linux please follow this video from this link

Lab 01: Introduction to Cisco Packet Tracer & Exploring a Basic Network Topology

Duration: 1 hour 30 minutes | In-Lab Assessment
Tools: Cisco Packet Tracer

Objectives

By the end of this lab, students will be able to:

- Identify common Packet Tracer device types and explain their intended roles (end devices, network devices, IoT).
- Differentiate common physical and logical ports and modules on devices.
- Choose appropriate cable types for connections and attach cables to correct ports.
- Navigate Packet Tracer workspace: Logical vs Physical view, device box, connection toolbar, and device configuration panels.
- Use Real-Time and Simulation modes to observe basic frame traversal and PDU lifecycles.
- Open and use the built-in device utilities: Desktop apps, CLI/Console, and Programming editor (where available).
- Identify network "edge" and "core" devices.
- Build a LAN topology in Packet Tracer.
- Configure IP addressing and default gateways.
- Generate ICMP (ping) traffic and capture frames.

Lab Description

This lab introduces Cisco Packet Tracer's interface and device catalogue. You will place devices, inspect their physical ports and built-in utilities, connect devices with appropriate cables, and run simple simulations to observe how Packet Tracer models traffic and device behavior — all without changing IP or advanced routing settings. The focus is on understanding device types, components, ports and software features inside Packet Tracer.

After introduction to the cisco packet tracer software, this lab introduces to the students the basics of internet using a basic network topology, by showing different components.

Part 1: Introduction to Cisco Packet Tracer

Part A: Getting started with Packet Tracer UI

1. Start Cisco Packet Tracer and create a new file.
2. Locate and explore the main UI areas:
    - Device-Type Selection Box (bottom left): categories such as `End Devices`, `Switches`, `Routers`, `Wireless`, `Security`, `Connections`, `Generic`.

- Workspace panes: `Logical` and `Physical` tabs.
- Device/Connection toolbar (top): Inspect, Delete, Move, Place Note.
- Simulation controls (bottom right): Real-Time / Simulation toggle, Play, Capture/-Forward.

3. Save your file as `Lab1_PT_Intro.pkt`.

Part B: Device catalogue

1. From the `End Devices` category, drag the following and inspect each:
   - `PC-PT`: desktop PC — view `Physical` tab to see NIC ports, power LED, USB/serial approximations; open `Desktop` to see apps (PC GUI, Command Prompt, Web Browser, Programming).
   - `Laptop-PT`: laptop form factor — note built-in wireless NICs in some models and the `Physical` chassis.
   - `Server-PT`: server with virtual services — inspect the `Services` tab (HTTP, FTP, DNS, etc.) and `Config` pages (hostname, passwords).
   - `IP Phone / Generic IoT devices`: briefly inspect for IoT labs.
2. From the `Switches` & `Router` category:
   - `Switch-PT` (e.g. 2960): view physical ports (FastEthernet/Gigabit)
   - `Router-PT` (e.g. 2911): view physical ports (FastEthernet/Gigabit)
3. From the `Connections` palette, view cable types

Part C: Ports, modules and cable types

1. Click each placed device and open `Physical` (or `Config`) view to inspect available ports:
   - NIC ports labelled `FastEthernet0`, `GigabitEthernet0`, wireless NICs, USB/Serial console ports.
2. Cable types and when to use them:
   - **Copper Straight-through**: PC <-> Switch, Server <-> Switch, Access Point <-> Switch. (When the devices are of different types)
   - **Copper Crossover**: PC <-> PC (legacy) or Switch <-> Switch (older devices); Packet Tracer will often auto-select the correct cable. (When the devices are of same types)
   - **Console / RS-232 / Serial**: used for CLI/console access to routers/switches (learn how to connect console cable to device console port and to PC's `Terminal` app).

Part D: Device utilities and software features

1. For each end device (PC/Server), open the `Desktop` tab and explore built-in apps:
   - `Command Prompt` — use it to practice simple Packet Tracer commands that do not require IP (e.g., open the terminal to connect via console).

- `PC GUI / Web Browser / Email / Files` — view how Packet Tracer simulates apps on end devices.
- `Programming` (on supported devices) — open the editor to inspect example Python templates (do not need to run a networking script in this lab).
- `Terminal / CLI` — connect console cable from PC to device console and practice opening the console window to see the device prompt.

2. For each network device (Switch/Router), open the `CLI` tab and explore the basic interface:
   - `Command-Line Interface (CLI)` — observe how devices boot up, watch the initial startup messages, and identify the default prompt (`Switch>` or `Router>`).
   - `User EXEC Mode` — practice entering basic commands such as `show ?`, `show version`, `show ip interface brief`.
   - `Privileged EXEC Mode` — enter privileged mode using `enable` and explore common diagnostic commands.
   - `Global Configuration Mode` — navigate using `configure terminal` and locate key configuration contexts such as interface configuration (`interface FastEthernet0/1`).
   - `Interface Exploration` — identify physical and logical interfaces, verify their default status, and understand how Packet Tracer models ports (e.g., shutdown state on routers).
   - `Basic Navigation` — practice moving between modes: User EXEC → Privileged → Global Config → Interface Config, then exit back to CLI.
3. Simulation vs Real-Time:
   - Switch to **Simulation Mode** and inspect a PDU as you create it (use the `Add Simple PDU` tool to simulate a generic frame between two devices). Observe PDU lifecycle, event list, and per-hop info.
   - Use Event List Filters to focus on specific protocol families (e.g., ARP, TCP) — you will not need to set IPs. However, you can still observe theoretical PDU processing in the simulated environment.
   - Use `Inspect` on a captured PDU to see fields and how Packet Tracer models headers.

Part 2: Exploring a Basic Network Topology

Part E: Packet Tracer Basics

1. **Components**
   - Devices Palette: End Devices → `PC-PT`.
   - Network Devices: Switches → `2960`; Routers → `2901`.
   - Connections: Click the lightning-bolt icon for cables (copper straight-through).
2. **Save Your File:** File → Save As → `Lab1_Topology.pkt`.

Part F: Building the Topology

1. Place Devices:
   - PC1: drag `PC-PT` (left).
   - Switch: drag `Switch-2960` (center).
   - Router: drag `Router-2901` (right of switch).
   - PC2: drag `PC-PT` (right).
2. Connect Devices:
   - PC1 (FastEthernet0) → Switch (FastEthernet0/1) using Copper Straight-Through.
   - Switch (FastEthernet0/2) → Router (GigabitEthernet0/0).
   - Router (GigabitEthernet0/1) → PC2 (FastEthernet0) using Copper Straight-Through.

Part G: IP Configuration

The IP is configured on the interface.

1. **PC1:**
   - Open PC1 → Desktop → IP Configuration:
     - IP: `192.168.1.10`
     - Subnet Mask: `255.255.255.0`
     - Default Gateway: `192.168.1.1`
2. **Router:**
   In Packet Tracer, click Router → **CLI** tab, then execute:

```
enable                                   ! Enter privileged EXEC mode
configure terminal                       ! Enter global configuration mode
  interface GigabitEthernet0/0            ! Select interface G0/0
    ip address 192.168.1.1 255.255.255.0  ! Assign IP
    no shutdown                          ! Bring up interface
  exit                                   ! Exit to global mode
  interface GigabitEthernet0/1            ! Select interface G0/1
    ip address 10.0.0.1 255.255.255.0     ! Assign IP
    no shutdown
  exit
end                                      ! Return to privileged EXEC
```

3. **PC2:**
   - Open PC2 → Desktop → IP Configuration:
     - IP: `10.0.0.10`
     - Subnet Mask: `255.255.255.0`
     - Default Gateway: `10.0.0.1`

Part H: Router-to-Router Connections (Ethernet or Serial)

Routers can be connected to each other using **Ethernet interfaces** or **Serial interfaces**. Serial interfaces are *not required* unless we are using traditional WAN technologies.

Option 1: Router-to-Router Using Ethernet (Recommended)

- No additional modules are needed.
- Use a Copper Straight-Through cable.
- Connect `GigabitEthernet0/1` of Router1 to `GigabitEthernet0/1` of Router2.

Example Configuration:

```
! Router1
enable
configure terminal
  interface GigabitEthernet0/1
    ip address 172.16.0.1 255.255.255.252
    no shutdown
end

! Router2
enable
configure terminal
  interface GigabitEthernet0/1
    ip address 172.16.0.2 255.255.255.252
    no shutdown
end
```

This is simpler and represents modern networks. Remember, both of the routers need to be in the same network

—

Option 2: Router-to-Router Using Serial Ports

To add serial interfaces:

- Click Router → **Physical** tab.
- Power off router.
- Insert `HWIC-2T` serial module.
- Power on router.
- Connect `Serial0/0/0` <-> `Serial0/0/0` using DCE/DTE cable.

Example Configuration:

```
! Router1
interface Serial0/0/0
  ip address 172.16.0.1 255.255.255.252
  no shutdown

! Router2
interface Serial0/0/0
```

```
  ip address 172.16.0.2 255.255.255.252
  no shutdown
```

Part I: Adding More Ports to a Router Using Modules

Routers such as the 2901 have limited built-in interfaces. Packet Tracer allows you to add additional ports (Ethernet, Serial, etc.) using hardware modules.

Steps to Add a Module:

1. Click the router → go to the **Physical** tab.
2. Power off the router using the power switch (top-left).
3. Drag a module into an empty expansion slot:
   - `HWIC-2T` — adds 2 serial interfaces.
   - `HWIC-4ESW` — adds 4 switch-like Ethernet ports.
   - `HWIC-1GE-SFP` — adds 1 GigabitEthernet port.
4. Power the router back on.
5. Go to the **CLI** tab and verify new interfaces using:

```
show ip interface brief
```

Example: Configuring a New Ethernet Port (G2/0)

```
enable
configure terminal
  interface GigabitEthernet2/0
    ip address 192.168.50.1 255.255.255.0
    no shutdown
  exit
end
```

After adding modules, you can connect more PCs, routers, or switches to these interfaces.

Part J: Establishing Routing (RIP)

Configure RIP on each router via the CLI:

```
enable
configure terminal
  router rip                         ! Enable RIP process
  version 2                          ! Use classless RIP v2
  network 192.168.1.0                ! Advertise LAN on G0/0
  network 10.0.0.0                   ! Advertise LAN on G0/1
  network 172.16.0.0                 ! Advertise serial link subnet
exit
end
```

Configure using GUI:

- Click the router and then click on config

- Then select RIP from the left side section
- There you can see an field to add network address and a button add
- Add all the addresses a router is connected to (including the router to router network).
- Do this in both the routers

Part K: Simulation & Packet Capture

- Switch Packet Tracer to **Simulation Mode**.
- On PC1 (Desktop → Command Prompt), run:

  ```
  ping 10.0.0.10  ! Send ICMP echo to PC2
  ```

- Observe the ICMP frames traversing the network

Part L: Saving Configurations

To save your changes:

```
enable                                ! Ensure privileged EXEC mode
copy running-config startup-config   ! Save running config to NVRAM
```

Student ID:

Lab 01 Task

In Islamic University of Technology (IUT), each computer lab has its own network. There are five labs, each with two PCs. The lab's network has a pattern of having 192.id for the first 2 portions of the network address. id is the last two digits of your student id

Your task is to design an interconnection such that any PC in Lab x can reach any PC in Lab y.

As IUT is in Board Bazar, there is a problem of frequent load shedding. For that, sometimes some of the network connecting devices might not work right. Keep this in mind when designing the network to solve this issue.

Lab 01 Grading Criteria

1. **Building the topology: 15 marks**
   - Lab's topology - 2 * 5 = 10 marks
   - Connection between the labs = 5 marks
2. **Configuring the PCs: 5 marks**
   - 10 PCs × .5 mark each
3. **Configuring the Routers: 10 marks**
4. **Verifying connectivity (ping): 10 marks**
   - Each lab's PCs local connectivity test - .5 marks each PC - 5 marks
   - Global connectivity test - .5 marks each PC - 5 marks
5. **Solving the load shedding problem: 10 marks**
6. **Total: 50 marks**

Lab 02: IP Subnetting and Variable Length Subnet Mask (VLSM)

Duration: 1 hour 30 minutes | In-Lab Assessment
Tools: Cisco Packet Tracer

Objectives

By the end of this lab, students will be able to:

- Understand the basics of IPv4 subnetting and CIDR notation.
- Subnet a fixed-length network to meet given host requirements.
- Apply Variable Length Subnet Mask (VLSM) to allocate address blocks efficiently.
- Design and implement a multi-LAN topology in Cisco Packet Tracer using VLSM.
- Configure classless routing using RIP version 2 and explain the purpose of disabling automatic summarization.

Lab Description

Part A: Subnet (Subnetwork)

A subnet is a logical, visible subdivision of an IP network. The practice of dividing a network into two or more smaller networks is called subnetting. This improves network efficiency, security, and manageability by:

1. Reducing Broadcast Traffic: Keeping local traffic within the subnet and confining broadcast domains.
2. Improving Security: Isolating different parts of the network from each other.
3. Efficient Address Allocation: Preventing large-scale waste of IP addresses.

Part B: IPv4 Addressing and Subnet Masks

An IPv4 address is a 32-bit number, usually represented in dotted-decimal notation (e.g., 192.168.1.1). It is divided into two parts: the Network ID (identifying the entire network/subnet) and the Host ID (identifying a specific device within that network).

The Subnet Mask is a 32-bit mask used to determine which part of the IP address is the Network ID and which part is the Host ID. The bits in the mask that are '1' correspond to the Network ID, and the bits that are '0' correspond to the Host ID.

Part C: CIDR (Classless Inter-Domain Routing) Notation

CIDR notation, also known as slash notation (e.g., /24), is a compact way to represent the subnet mask. The number after the slash indicates the number of bits in the IP address that are dedicated to the Network ID/Subnet ID. For example, a mask of /24 means the first 24 bits are for the network, leaving $32 - 24 = 8$ bits for the host addresses.

Part D: Variable Length Subnet Mask (VLSM)

VLSM is a subnetting technique that allows the use of different-sized subnets within the same network. Instead of creating a set of equal-sized subnets (Fixed-Length Subnet Mask or FLSM), VLSM allows you to size each subnet exactly according to its unique host requirements.

Benefit: VLSM dramatically increases IP address utilization efficiency, especially for point-to-point links which only require two usable host addresses.

Part E: Basic IPv4 Subnetting

1. **Theory Review:**
   - Recall that a subnet mask of /n means the first n bits are network / subnet and the remaining $32 - n$ bits host.
   - Compute # of subnets: 2^{Δ}, where Δ is the number of bits borrowed.
   - Compute hosts per subnet: $2^{\text{host bits}} - 2$, where the first address is the network address and the last address is the broadcast address.
2. **Exercise:** Given network `172.16.0.0/16`, create four equal-sized subnets.
   - Determine how many bits to borrow for 4 subnets.
   - Write each subnet's network address, subnet mask (in dotted decimal and CIDR), usable host range, and broadcast address.

Part F: Variable Length Subnet Mask (VLSM) Design

1. **Host Requirements Table:**

Subnet	Required Hosts
netA	14
netB	28
netC	2
netD	7
netE	28

2. **Design Steps:**
   (a) Order subnets by descending host requirement.
   (b) For each, choose the smallest CIDR block that accommodates the required hosts.
   (c) Allocate subnets sequentially from `204.15.5.0/24`, incrementing network address accordingly.
3. **Resulting VLSM Plan:**

Subnet	Network	Mask	Usable Hosts Range
netB	204.15.5.0	/27(255.255.255.224)	204.15.5.1 – 204.15.5.30
netE	204.15.5.32	/27	204.15.5.33 – 204.15.5.62
netA	204.15.5.64	/28(255.255.255.240)	204.15.5.65 – 204.15.5.78
netD	204.15.5.80	/28	204.15.5.81 – 204.15.5.94
netC	204.15.5.96	/30(255.255.255.252)	204.15.5.97 – 204.15.5.98

Note: The last address in each subnet is reserved as the broadcast address and is not usable for host assignment.

Part G: RIP Configuration and Classless Routing

To enable routing between the subnets defined above, configure RIP version 2 with the following guidelines:

1. On each router, enter configuration mode and enable RIP:

   ```
   router rip
     version 2
     no auto-summary
     network 204.15.5.0
   ```

2. **Why RIP Version 2?**
   RIP version 1 only supports classful routing and does not advertise subnet masks. Since this lab uses VLSM (variable-length subnet masks), it is essential to use RIP version 2, which supports classless routing and includes subnet mask information in its updates.

3. **Why `no auto-summary`?**
   By default, RIP summarizes routes to their classful boundaries (e.g., summarizing all subnets under 204.15.5.0/24). This behavior can cause incorrect routing in discontiguous networks. Disabling automatic summarization ensures that each subnet is advertised with its correct subnet mask, allowing precise routing between non-contiguous subnets.

4. Verify RIP routing with the following commands:

   - `show ip protocols` – to confirm RIP v2 is active.
   - `show ip route` – to verify RIP routes are installed.
   - `debug ip rip` – to monitor RIP updates in real-time (optional).

Student ID:

Lab 02 Task

Your company, Iutians Assemble, is expanding its headquarters and requires a new network design to support various departments. The IA headquarters consists of the following departments across multiple floors: Engineering - Floor 1, Marketing - Floor 2, Sales - Floor 3, Administration - Floor 4, Management - Floor 5

Suppose they all share the prefix **192.(i).(i+3 % 14)**. Below is the required devices for each of the departments:

$$\begin{aligned}
E(i) &= \big(7i+3\big) \bmod 41 \;+\; 10 && \text{(Engineering)}\\
M(i) &= \big(5i+11\big) \bmod 26 \;+\; 5 && \text{(Marketing)}\\
S(i) &= \big(11i+7\big) \bmod 36 \;+\; 5 && \text{(Sales)}\\
A(i) &= \big(3i+17\big) \bmod 18 \;+\; 3 && \text{(Administration)}\\
G(i) &= \big(13i+1\big) \bmod 9 \;+\; 2 && \text{(Management)}
\end{aligned}$$

where i is the last two digits your student ID
Now, by completing the following table with the most efficient design decision, design the topology with correct network flow, for each network attach at least 1 end devices.

Subnetting Worksheet:

Department	**Devices**	**Network**	**Mask**	**Usable Host Range**
Engineering	$E(i)$	____________	/______	__________________
Marketing	$M(i)$	____________	/______	__________________
Sales	$S(i)$	____________	/______	__________________
Administration	$A(i)$	____________	/______	__________________
Management	$G(i)$	____________	/______	__________________

Lab 02 Grading Criteria

1. **Subnet Calculation (30 pts)**
   - Each of the correct values in the table (2 pts)
2. **Topology & Configuration (20 pts)**
   - Correct topology & router, pc configuration (7 pts).
   - Correct routing (3 pts).
   - Send a packet from a single device in the network to all other devices (2 pts each = 10 pts)
3. **Total: 50 pts**

Lab 03: DHCP and NAT Configuration

Duration: 2 hours 30 minutes | In-Lab Assessment
Tools: Cisco Packet Tracer

Objectives

By the end of this lab, students will be able to:

- Set up DHCP both via a dedicated server and via a Cisco router.
- Inspect DHCP bindings and troubleshoot IP lease assignments.
- Configure Static, Dynamic, and Port-Address Translation (PAT) NAT on Cisco routers.
- Verify NAT translations and statistics.
- Understand and configure Access Control Lists (ACL).

Part 1: DHCP Theory and Configuration

Part A: Dynamic Host Configuration Protocol (DHCP)

DHCP is a network management protocol that automates the assignment of IP addresses and other parameters (such as Subnet Mask, Default Gateway, and DNS), allowing devices to communicate on a network without manual configuration.

- **Operation:** The DHCP server selects an IP from a predefined pool and assigns it to a client upon request.
- **Lease:** Addresses are granted for a limited time (a lease). Clients must renew this lease periodically to retain connectivity.
- **Relay Agents:** Since DHCP relies on broadcast messages, which do not cross routers, a *DHCP Relay Agent* is required if the Server and Client are on different nets/subnets. The agent forwards client requests to the remote server.

Part B: Configure DHCP (using server)

A router of device model 2811, two switches of device model 2960, two PCs, two laptops, and one server have been used in the sample topology shown in Figure 1 for the DHCP configuration.

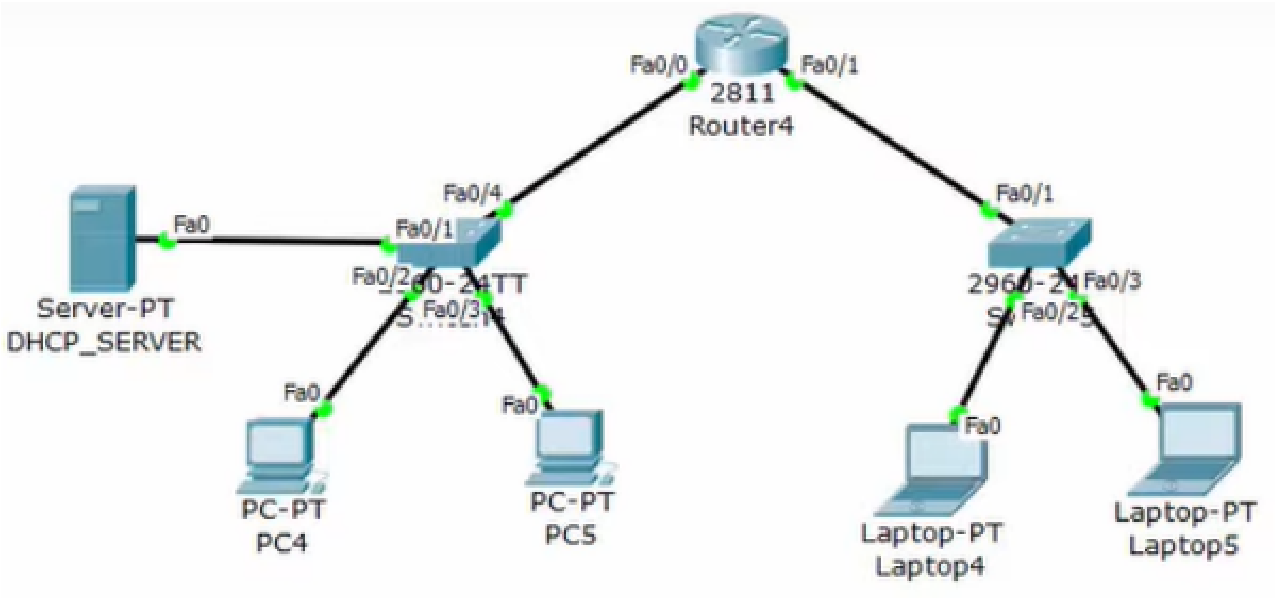

Figure 1: A sample network topology for the configuration of DHCP.

1. **Configure DHCP Server**

```
IP Address: 192.168.1.2
Default: 192.168.1.1
```

2. **Make DHCP Pools**

Go to Services and then DHCP, and change the following fields.

```
Pool Name: dotONEnetwork
Default: 192.168.1.1
Start IP: 192.168.1.3
Max Number: 20
```

Select **Add** to add the Address Pool to the server. Select **Save** to save the modification to the server.

To add another pool to the server, change the information as follows and then hit **Add** and **Save** buttons again.

```
Pool Name: dotTWOnetwork
Default: 192.168.2.1
Start IP: 192.168.2.2
Max Number: 20
```

Do not forget to turn on the DHCP server.

3. **Configure R1 Interfaces**

```
R1(config)#int fa0/0
R1(config-if)# ip address 192.168.1.1 255.255.255.0
R1(config-if)# ip helper-address 192.168.1.2
R1(config-if)#no shutdown
R1(config-if)#exit
R1(config)#int fa0/1
R1(config-if)# ip address 192.168.2.1 255.255.255.0
```

```
R1(config-if)# ip helper-address 192.168.1.2
R1(config-if)#no shutdown
R1(config-if)# end
R1# copy running-config startup-config
```

4. **Configure all the PCs**
   Just click DHCP and the server will do the rest.

5. **Verify**

   ```
   Ping PC1 from PC0
   ```

Part C: Configure DHCP (using router)

The exact network topology with the same device models, as shown in Figure 1, has been used for this configuration.

Configure the DHCP server in the router instead of a server.

```
R1(config)# ip dhcp pool dotONEnetwork
R1(dhcp-config)# default-router 192.168.1.1
R1(dhcp-config)# network 192.168.1.0 255.255.255.0
R1(dhcp-config)# exit

R1(config)# ip dhcp pool dotTWOnetwork
R1(dhcp-config)# default-router 192.168.2.1
R1(dhcp-config)# network 192.168.2.0 255.255.255.0
R1(dhcp-config)# exit
```

The rest of the configuration, i.e., Configure R1 Interfaces, Configure all the PCs, and Verify are same as the section .

Use **show ip dhcp binding** inside the router to see the status of the configured DHCP.

Part 2: NAT and ACL Theory and Configuration

Part A: Standard Access Control Lists (ACL)

An ACL is a sequential list of "permit" or "deny" statements applied to an interface. In this lab, we focus on **Standard IPv4 ACLs** (numbered 1–99), which filter traffic based solely on the **Source IP**.

Every ACL has an invisible "Deny All" rule at the very bottom. If a packet does not match any of your specific rules, it is **discarded**.
Solution: If you want to filter specific traffic but allow everything else, you must explicitly add `permit any` at the end of your list.

Configuration Steps

Step 1: Define the Rules

Use a number between 1 and 99. You must specify the source IP and a wildcard mask (inverse subnet mask).

```
! Format: access-list <1-99> {permit|deny} <source_ip> <wildcard>
Router(config)# access-list 10 deny 192.168.1.5 0.0.0.255
Router(config)# access-list 10 permit any
```

Step 2: Apply to an Interface

An ACL has no effect until applied to an interface's inbound or outbound direction.

```
! Format: ip access-group <number> {in|out}
Router(config)# interface g0/0
Router(config-if)# ip access-group 10 out
```

Step 3: Verify

```
Router# show access-lists
```

Part B: NAT Overview

NAT allows private IP networks to connect to the internet by translating private addresses to global public addresses.

1. **Static NAT:** One-to-One mapping (1 Private IP ↔ 1 Public IP). Used for servers.
2. **Dynamic NAT:** Many-to-Many mapping. Maps a private IP to the first available public IP from a pool.
3. **PAT (Overload):** Many-to-One mapping. Maps multiple private IPs to a single public IP using different source port numbers.

Part C: Prerequisites: Defining Interfaces

For any NAT configuration, you must define the boundaries of the network.

```
Router(config)# interface fa0/0        ! Interface facing LAN
Router(config-if)# ip nat inside
Router(config)# interface fa0/1        ! Interface facing Internet/ISP
Router(config-if)# ip nat outside
```

Part D: Configuration A: Static NAT

Goal: Map local server 10.0.0.5 permanently to public IP 50.0.0.5.

```
Router(config)# ip nat inside source static 10.0.0.5 50.0.0.5
```

Part E: Configuration B: Dynamic NAT

Goal: Allow internal subnet 192.168.10.0/24 to share a pool of public IPs (50.0.0.1 to 50.0.0.3).

```
! Step 1: Define traffic to be translated using an ACL
Router(config)# access-list 1 permit 192.168.10.0 0.0.0.255

! Step 2: Define the pool of Public IPs
Router(config)# ip nat pool MY_POOL 50.0.0.1 50.0.0.3 netmask 255.255.255.252

! Step 3: Link the ACL to the Pool
Router(config)# ip nat inside source list 1 pool MY_POOL
```

Part F: Configuration C: PAT (NAT Overload)

Goal: Allow hundreds of users to access the internet using a single Public IP (on the outgoing interface).

```
! Step 1: Define traffic (ACL)
Router(config)# access-list 1 permit 192.168.10.0 0.0.0.255

! Step 2: Enable Overload on the outside interface
Router(config)# ip nat inside source list 1 interface fa0/1 overload
```

Part G: Configure ACL

A router of device model 2911, three switches of device model 2960, and three PCs have been used in the sample topology shown in Figure 2 for the ACL configuration.

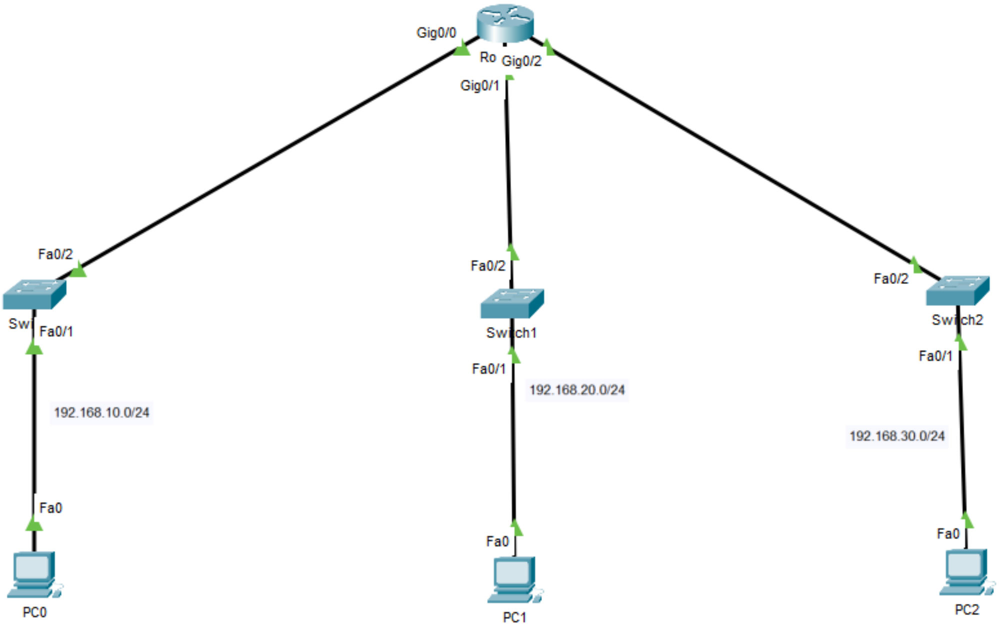

Figure 2: A sample network topology for the configuration of ACL.

1. **Define and Apply ACL**

```
Router(config)# access-list 1 deny 192.168.10.0 0.0.0.255
Router(config)# access-list 1 permit any
Router(config)# interface g0/2
Router(config-if)# ip access-group 1 out
```

Part H: Configure Static NAT

Two routers of device model 2811, one switch of device model 2960, two PCs, and one server have been used in the sample topology shown in Figure 3.

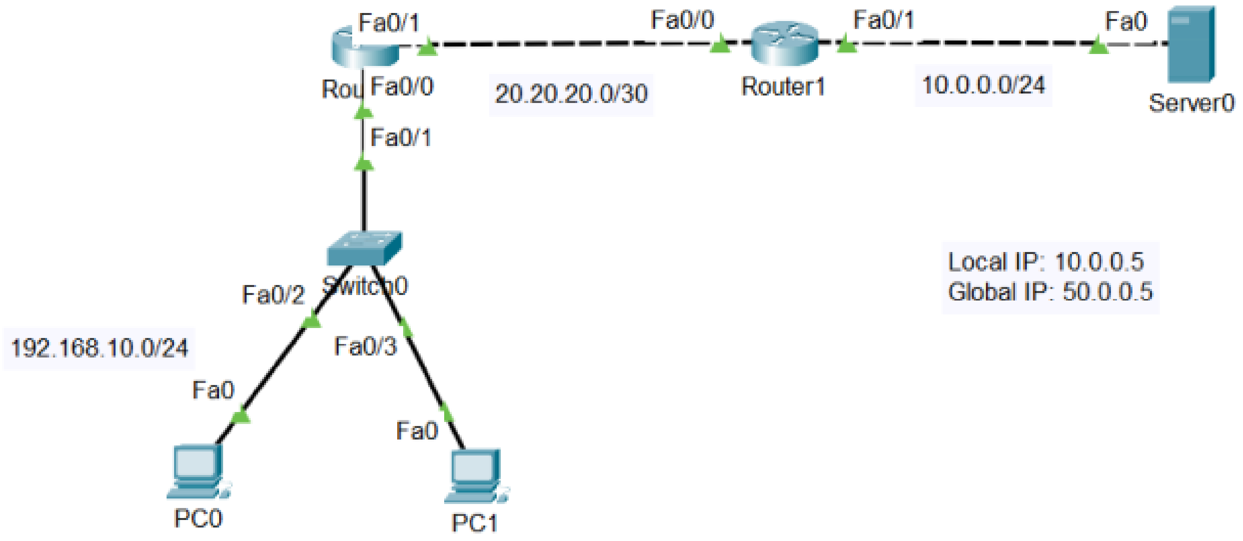

Figure 3: A sample network topology for the configuration of NAT.

1. **Enable static NAT inside Router1**

```
Router(config)# ip nat inside source static 10.0.0.5 50.0.0.5
Router(config)# int fa0/1
Router(config-if)# ip nat inside
Router(config)# int fa0/0
Router(config-if)# ip nat outside
```

2. **Verify NAT**

```
Router# show ip nat translations
```

Part I: Configure Dynamic NAT

Using the topology in Figure 3:

1. **Enable dynamic NAT inside Router1**

```
Router(config)# access-list 1 permit 192.168.10.0 0.0.0.255
Router(config)# ip nat pool NAT_Pool 50.0.0.1 50.0.0.3 netmask
    255.255.255.252
Router(config)#ip nat inside source list 1 pool NAT_Pool
Router(config)# int fa0/1
Router(config-if)# ip nat inside
Router(config)# int fa0/0
Router(config-if)# ip nat outside
```

Student ID:

Lab 03 Task: The MAGA Headquarters

Your company, Make Abdullahpur Great Again (MAGA), is expanding its headquarters and requires a new network design to support various departments. The MAGA headquarters consists of the following departments across multiple floors: **Engineering** (Floor 1), **Marketing** (Floor 2), **Sales** (Floor 3), **Administration** (Floor 4), **Management** (Floor 5).

Suppose they all share the prefix **192.(i).((i+3) % 10)**. Below are the required devices for each of the departments:

$$\begin{aligned}
E(n) &= 33 && \text{(Engineering)}\\
M(n) &= 7 && \text{(Marketing)}\\
S(n) &= 14 && \text{(Sales)}\\
A(n) &= 12 && \text{(Administration)}\\
G(n) &= 30 && \text{(Management)}
\end{aligned}$$

where, i is the last two digits of your id

Task Requirements

1. **Topology Implementation:** Implement and configure the topology. For each network, you need to add a minimum of 3 end devices.
2. **DHCP Configuration:**
   - Devices should get their IP automatically from the DHCP server.
   - **Engineering, Marketing, and Sales:** Devices should use a separate dedicated DHCP server for automatic IP assignment.
   - **Admin and Management:** Devices should use a Router as a DHCP server.
3. **NAT Configuration:**
   - Your company is connected to the internet via an "Internet Router".
   - You have purchased only **7 Global IPs**.
   - The Global IP address range starts from **110.i.(i+1).0/24**.
   - **Your own PC** must have a specific local IP that maps to its own exclusive Global IP
   - **All other devices** must share the remaining six Global IP addresses for accessing the internet

Lab 03 Grading Criteria

1. **Topology & Router Configuration (20 pts)**
   - Correct topology design (5 pts)
   - Correct router interfaces and basic setup (15 pts)
2. **DHCP Configuration (20 pts)**
   - Correct separate DHCP server configuration (10 pts)
   - Correct Router-as-DHCP configuration (10 pts)
3. **NAT Configuration (20 pts)**
   - Correct NAT Configuration (10 * 2 pts)
4. **Routing (10 pts)**
   - Correct Routing configuration to ensure connectivity (10 pts)
5. **Total: 70 pts**

Lab 04: Implementing Static and Dynamic Routing (RIP & OSPF)

Duration: 1 hours 30 minutes | In-Lab Assessment
Tools: Cisco Packet Tracer

Objectives:

- Define and describe the concept of routing
- Describe the concept of static routing and dynamic routing
- Configure static and dynamic routing in a given network topology

Lab Description:

Part A: Static Routing

Static routing requires the administrator to manually configure and update all routes in a router's table—changes in the network (like a link failure) must be addressed by hand, unlike dynamic routing, which adapts automatically.

Because it doesn't exchange update packets, static routing uses far fewer CPU cycles and less bandwidth, making it a lightweight backup solution in large networks rather than the primary method.

However, it demands complete knowledge of the network's topology and addresses, so it's impractical to use alone at scale. To handle packets whose destinations lack a specific entry, routers employ a default route, which directs such traffic out a predefined interface.

When working with CISCO devices, specifically for this lab, you will encounter two types of static default routes. They are **directly connected** static default route and **next-hop** static default route. The general format of the static routes is as follows:

```
ip route destination_network_prefix destination_prefix_mask (next-hop_address
| interface) [distance_metric]
```

To configure the directly connected static default routes, you will specify the **interface**. For the next-hop static default routes, you will specify the **next_hop address**. One special use case of the above command is configuring a **primary static default route** where both the `destination_network_prefix` and `Destination_prefix_mask` are 0.0.0.0. The IPv4 command format for specifying the primary static default route is given below:

```
ip route 0.0.0.0 0.0.0.0 (next-hop_address | interface) [distance_metric]
```

It means that **"packet from any IP address with any subnet mask gets sent to the specified next-hop address or interface."**

Another concept when configuring static routing is the **floating static route**. A floating route is nothing but the route that is used to forward a packet to a certain destination when main route is unavailable. The floating routes are defined by providing a higher **distance_metric** to a certain route. The default distance_metric, when it is not manually specified, is 1. The floating static routes are given higher numbers than 1. Routers always take the route with lower distance_metric when multiple routes to the same destination are available. That's why this floating static route will only be used when the main route (distance_metric is 1) is down or unavailable.

Part B: Dynamic Routing

Dynamic routing lets routers select routes based on the real-time network condition. There will always be packets traveling around the networks to keep the routers up-to-date about the present network condition. Then, the routers will select an optimal route to a given destination based on some set of metrics. There are different types of dynamic routing protocols following different algorithms. The two most common ones are **Routing Information Protocol (RIP)** following distance-vector algorithm and **Open Shortest Path First (OSPF)** protocol following link state routing algorithm.

Routing Information Protocol (RIP)

RIP is somewhat obsolete due to its limitations and the advent of more modern and sophisticated protocols like OSPF, EIGRP, etc. Still, as this was one of the pioneering dynamic routing protocols that dominated the networking world for quite some time, you must understand this. This will help you understand the improvements made in newer protocols. There are two versions of RIP, namely RIPv1 and RIPv2. RIPv2 is the dominant one and is used in almost all cases where RIP is used. So, we'll focus on RIPv2 only for our lab. A major problem of RIPv1 was that it used to broadcast routing table updates every 30 seconds. You can imagine the flood of packets every 30 seconds that would take place where millions of networks are now interconnected, even if they are configured to send updates at random times. RIPv2 was designed to overcome this issue along with other improvements. You can learn more about these two versions along with RIPng (for IPv6) here.

Open Shortest Path First (OSPF) Routing Protocol

OSPF uses a Link-State algorithm: routers flood only their link states (not full tables) to neighbors, then each runs Dijkstra's Shortest-Path First to build routes. This makes OSPF more CPU-intensive than distance-vector protocols, but it converges faster and balances load better than RIP.

Key OSPF Concepts

1. **Area:** A logical grouping of routers sharing topology info. All non-backbone areas (any ID != 0) connect through the backbone (Area 0).
2. **Area Border Router (ABR):** Connects two areas and exchanges summaries between them.
3. **Autonomous System (AS):** A set of OSPF areas under one administrative domain.
4. **Designated Router (DR)/Backup DR:** On multi-access links (e.g. Ethernet), the DR/BDR reduce LSA traffic by acting as a hub for broadcasts.

5. **Router ID:** A unique 32-bit identifier (manually set or chosen as highest loopback/active IP). Requires process reset (clear ip ospf process).
6. **Cost:** Inverse of bandwidth: cost = reference-bandwidth / interface-bandwidth (default ref = 100 Mbps). Adjust with auto-cost reference-bandwidth and bandwidth commands.
7. **Wildcard Mask:** Used in network <IP> <wildcard> area <ID> to select interfaces—0 bits must match, 1 bits ignore (e.g. 0.0.0.255 for /24).
8. **Process ID:** Local identifier for an OSPF instance (router ospf <process_id>); multiple instances can run on one router.

Neighbor Maintenance

1. **Hello and Dead Intervals:** Hello packets sent every X seconds (default 10); if no hello is heard in 4 × Hello, the neighbor is declared down.
2. **Passive Interfaces:** Advertise networks but do not send hellos (common on LANs).

Part C: Route Redistribution

Redistributing routing information refers to the process of sharing routing information between different routing protocols within a network. The messages used to advertise or update routing information are protocol-specific; hence, information from one protocol cannot be directly used by another to update its routing information.

For instance, the router at the boundary of the OSPF and RIP networks can be set up to redistribute routes as follows:

Redistributing OSPF into RIP: When the OSPF router receives a route, it advertises it to the RIP domain, allowing branch offices using RIP to learn about routes to the internal OSPF network.

Redistributing RIP into OSPF: Conversely, the router can also redistribute routes learned from RIP into the OSPF network, ensuring that the internal OSPF network can reach the branch offices.

Part D: Configure static routing

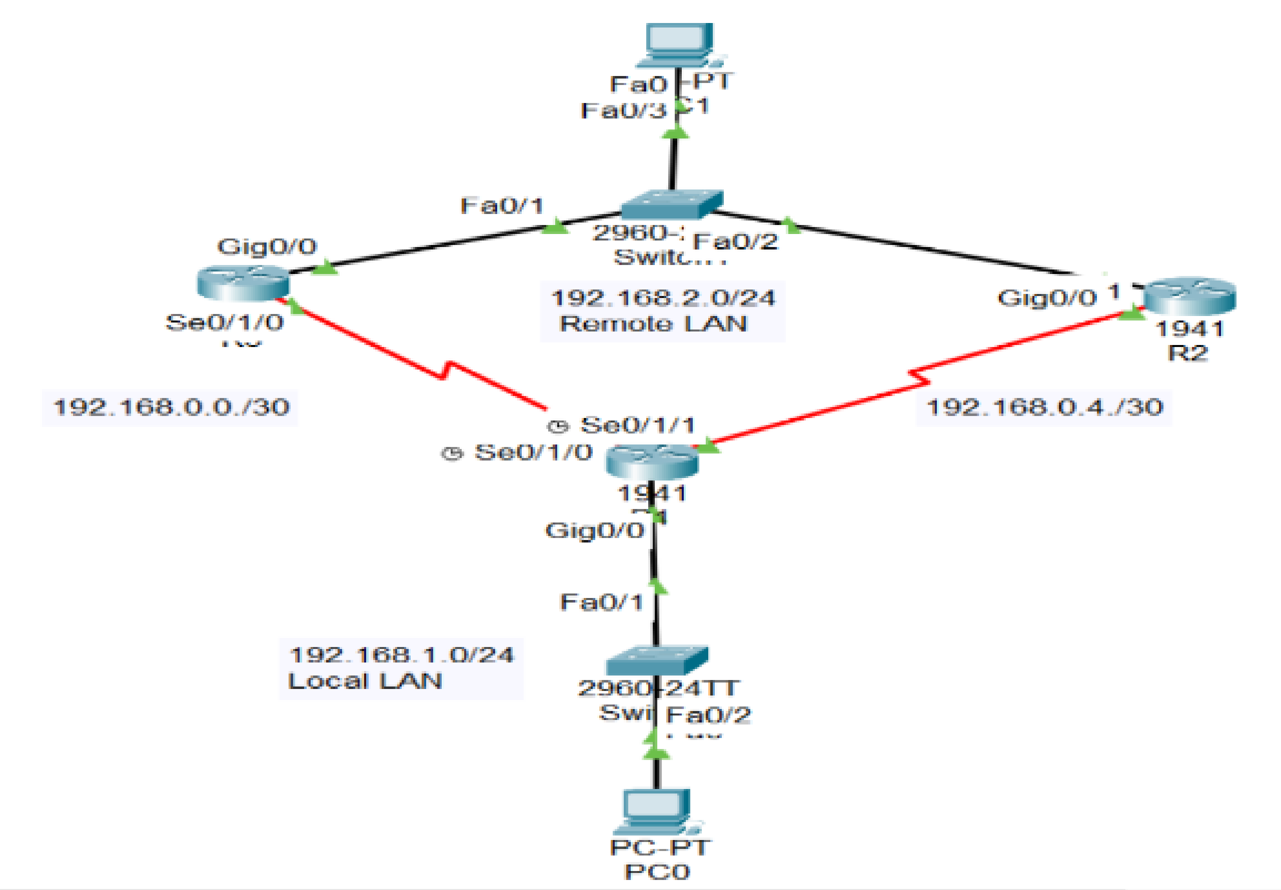

Figure 4: A sample network topology for the configuration of static routing.

1. **Configure R1 Interfaces**

```
R1(config)#int g0/0
R1(config-if)#ip add 192.168.1.1 255.255.255.0
R1(config-if)#desc connection-to-PC0
R1(config-if)#no shutdown
R1(config-if)#exit
R1(config)#int s0/1/0
R1(config-if)#ip add 192.168.0.2 255.255.255.252
R1(config-if)#desc connection-to-R3
R1(config-if)#clock rate 64000
R1(config-if)#no shutdown
R1(config-if)#exit
R1(config)#int s0/1/1
R1(config-if)#ip add 192.168.0.6 255.255.255.252
R1(config-if)#desc connection-to-R2
R1(config-if)#clock rate 64000
R1(config-if)#no shutdown
R1(config-if)#exit
```

2. **Configure R2 Interfaces**

```
R2(config)#int s0/1/1
R2(config-if)#ip add 192.168.0.5 255.255.255.252
R2(config-if)#desc connection-to-R1
R2(config-if)#no shutdown
R2(config-if)#exit
R2(config)#int g0/0
R2(config-if)#ip add 192.168.2.1 255.255.255.0
R2(config-if)#desc connection-to-RemoteLAN
R2(config-if)#no shutdown
R2(config-if)#exit
```

3. Configure R3 Interfaces

```
R3(config)#int s0/1/0
R3(config-if)#ip add 192.168.0.1 255.255.255.252
R3(config-if)#desc connection-to-R1
R3(config-if)#no shutdown
R3(config-if)#exit
R3(config)#int g0/0
R3(config-if)#ip add 192.168.2.2 255.255.255.0
R3(config-if)#desc connection-to-RemoteLAN
R3(config-if)#no shutdown
R3(config-if)#exit
```

4. Configure PC0

```
IP: 192.168.1.10
Mask: 255.255.255.0
Gateway: 192.168.1.1
```

5. Configure PC1

```
IP: 192.168.2.10
Mask: 255.255.255.0
Gateway: 192.168.2.1
```

6. Configure static routing to Remote LAN in R1

```
R1(config)#ip route 192.168.2.0 255.255.255.0 s0/1/1
```

It's a **directly connected** static default route.

```
R1(config)#ip route 192.168.2.0 255.255.255.0 192.168.0.1 5
```

It's a **next-hop floating** static default route.

7. Configure static routing to Local LAN in R2

```
R2(config)#ip route 192.168.1.0 255.255.255.0 s0/1/1
```

It's a **directly connected** static default route.

```
R2(config)#ip route 192.168.1.0 255.255.255.0 192.168.0.6 5
```

It's a **next-hop floating** static default route.

8. **Configure static routing to Local LAN in R3**

```
R2(config)#ip route 192.168.1.0 255.255.255.0 s0/1/0
```

It's a **directly connected** static default route.

```
R2(config)#ip route 192.168.1.0 255.255.255.0 192.168.0.2 5
```

It's a **next-hop floating** static default route.

9. **Verify**

```
Ping PC1 from PC0
```

Part E: Configure RIP

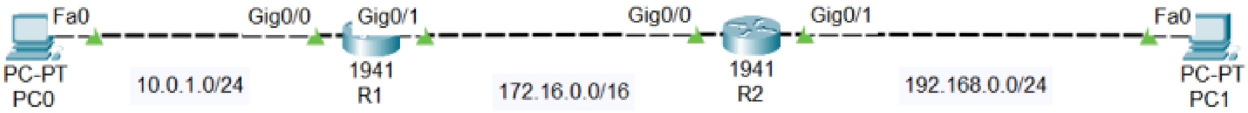

Figure 5: A sample network topology for the configuration of RIP.

1. **Configure R1 Interfaces**

```
R1(config)# int g0/0
R1(config-if)# ip address 10.0.1.1 255.255.255.0
R1(config-if)# no shutdown
R1(config-if)# exit
R1(config)# int g0/1
R1(config-if)# ip address 172.16.0.1 255.255.0.0
R1(config-if)# no shutdown
R1(config-if)# exit
R1# copy running-config startup-config
```

2. **Configure R2 Interfaces**

```
R2(config)# int g0/1
R2(config-if)# ip address 192.168.0.1 255.255.255.0
R2(config-if)# no shutdown
R2(config-if)# exit
R2(config)# int g0/0
R2(config-if)# ip address 172.16.0.2 255.255.0.0
R2(config-if)# no shutdown
R2(config-if)# exit
R2# copy running-config startup-config
```

3. **Configure PC0**

```
IP: 10.0.1.10
Mask: 255.255.255.0
Gateway: 10.0.1.1
```

4. **Configure PC1**

```
IP: 192.168.0.10
Mask: 255.255.255.0
Gateway: 192.168.0.1
```

5. **Configure RIP in R1**

```
R1(config)#router rip
R1(config-router)#version 2
R1(config-router)#network 10.0.0.0
R1(config-router)#network 172.16.0.0
```

6. **Configure RIP in R2**

```
R2(config)#router rip
R2(config-router)#version 2
R2(config-router)#network 192.168.0.0
R2(config-router)#network 172.16.0.0
```

7. **Verify**

```
R1# show ip route rip

Ping PC1 from PC0
```

Part F: Configure OSPF

The OSPF concepts and the detailed implementation of the OSPF routing protocol can be found at this link. You are suggested to read the article and practice the commands from the article for implementation. Below is a simple configuration of OSPF for the given network topology:

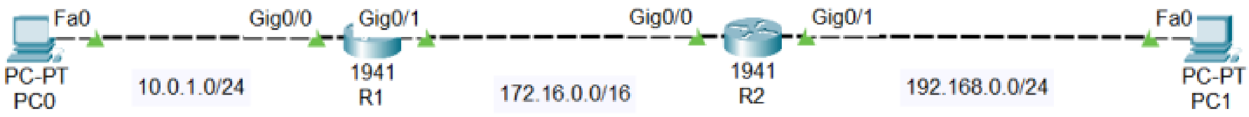

Figure 6: A sample network topology for the configuration of OSPF.

1. **Configure R1 Interfaces**

```
R1(config)# int g0/0
R1(config-if)# ip address 10.0.1.1 255.255.255.0
R1(config-if)# no shutdown
R1(config-if)# exit
```

```
R1(config)# int g0/1
R1(config-if)# ip address 172.16.0.1 255.255.0.0
R1(config-if)# no shutdown
R1(config-if)# exit
R1# copy running-config startup-config
```

2. Configure R2 Interfaces

```
R2(config)# int g0/1
R2(config-if)# ip address 192.168.0.1 255.255.255.0
R2(config-if)# no shutdown
R2(config-if)# exit
R2(config)# int g0/0
R2(config-if)# ip address 172.16.0.2 255.255.0.0
R2(config-if)# no shutdown
R2(config-if)# exit
R2# copy running-config startup-config
```

3. Configure PC0

```
IP: 10.0.1.10
Mask: 255.255.255.0
Gateway: 10.0.1.1
```

4. Configure PC1

```
IP: 192.168.0.10
Mask: 255.255.255.0
Gateway: 192.168.0.1
```

5. Configure OSPF in R1

```
R1(config)# router ospf 1
R1(config-router)# network 10.0.1.0 0.0.0.255 area 0
R1(config-router)# network 172.16.0.0 0.0.255.255 area 0
```

6. Configure OSPF in R2

```
R2(config)# router ospf 1
R2(config-router)# network 192.168.0.0 0.0.0.255 area 0
R2(config-router)# network 172.16.0.0 0.0.255.255 area 0
```

7. Verify

```
R1# show ip ospf neighbor
R2# show ip ospf neighbor

Ping PC1 from PC0
```

Part G: Configure Route Redistribution

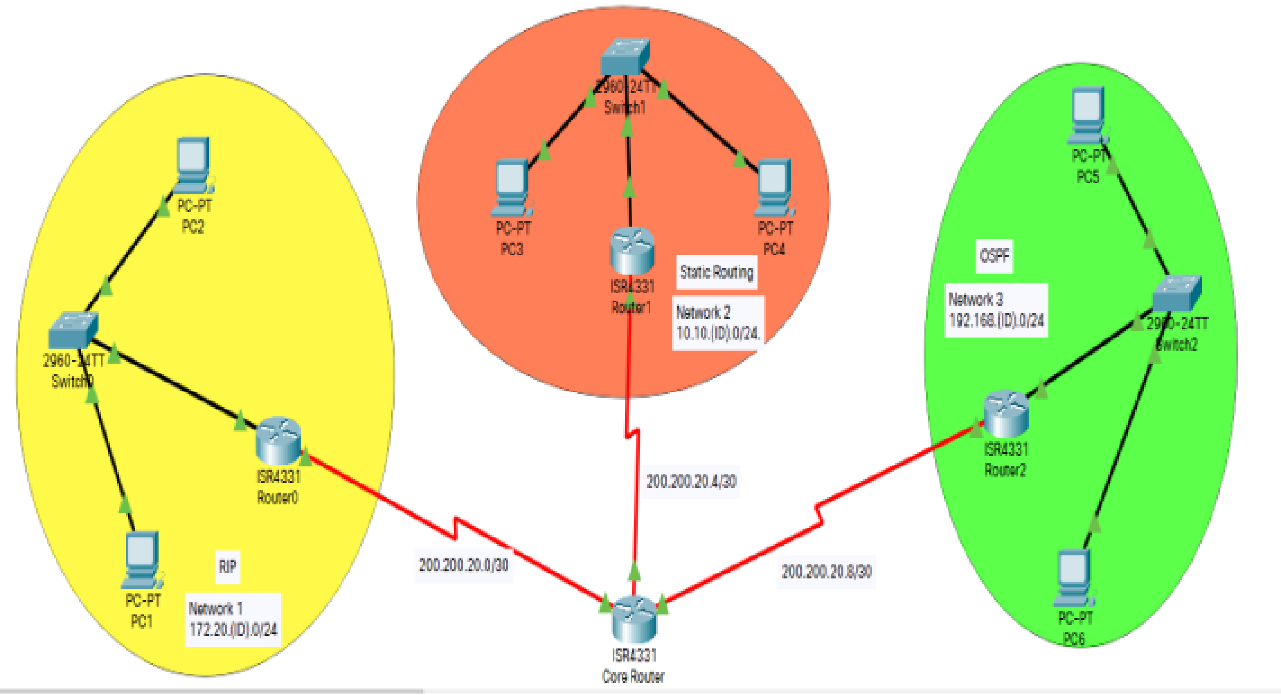

Figure 7: A sample network topology for Route Redistribution.

The core router is running three different routing protocols for the connected subnetworks. Route redistribution is necessary for this network design to enable communication among the subnetworks. It allows routes from one routing protocol to be advertised in another.

1. **Redistribute RIP and Static into OSPF**

```
CoreRouter(config)#router ospf 1
CoreRouter(config-router)#redistribute rip metric 1 subnets
CoreRouter(config-router)#redistribute static metric 1 subnets
```

2. **Redistribute OSPF and Static into RIP**

```
CoreRouter(config)#router rip
CoreRouter(config-router)#redistribute ospf 1 metric 1
CoreRouter(config-router)#redistribute static metric 1
```

3. **Verify**
   To verify the route redistribution, use the `ping` messages from **PC1** to **PC5** after successfully assigning the ip addresses to the PCs and routers' interfaces and successfully configuring routing protocols as shown in Figure 7.

Student ID:

Lab 04 Task

There are two labs for IUT CSE 2022 batch - Lab Alpha (Net Add- 192.168.id.0/24 and 192.168.id+1.0/24), Lab Beta (Net Add- 192.168.id+2.0/24, 192.168.id+3.0/24). Lab Alpha has 14 PCs connected to it and uses RIP. Lab Beta has 16 PCs connected to it and uses OSPF for routing. Now, there is a core ICT router which can re-route all the routing information from these 2 labs to the ICT dept network(192.168.id+4.0/24). All the ip addresses will be assigned automatically using DHCP. Implement this scenario. id is the last two digits of your student id.

Lab 04 Grading Criteria

1. **Correct Topology & Router Configuration (25 pts)**
   - Correct topology (5 pts)
   - Correct router configuration (20 pts)
2. **DHCP (10 pts)**
3. **Routing (5 pts)**
4. **Total: 40 pts**

Lab 05: VLAN Configuration and Inter-VLAN Routing

Duration: 1 hours 30 minutes | In-Lab Assessment
Tools: Cisco Packet Tracer

Objectives:

- Define and describe the concept of VLAN
- Describe the advantages of VLAN
- Design and implement VLAN and inter-VLAN routing

Lab Description:

As with other labs, this lab will also build up on the concepts and techniques of previous labs. So, make sure you have properly understood the previous lab contents.

Part A: VLAN

VLAN or Virtual LAN (Local Area Network) is a logical grouping of networking devices. When we create VLAN, we actually break a large broadcast domain into smaller broadcast domains. Consider VLAN as a subnet. Just as two different subnets cannot communicate with each other without a router, different VLANs also require a router to communicate.

Advantages of VLAN

VLAN provides the following advantages:

- Solve the broadcast problem.
- Reduce the size of broadcast domains.
- Allow us to add an additional layer of security.
- Make device management easier.
- Allow us to implement the logical grouping of devices by function instead of location.

VLAN Example

To understand VLAN more clearly, let's take an example.

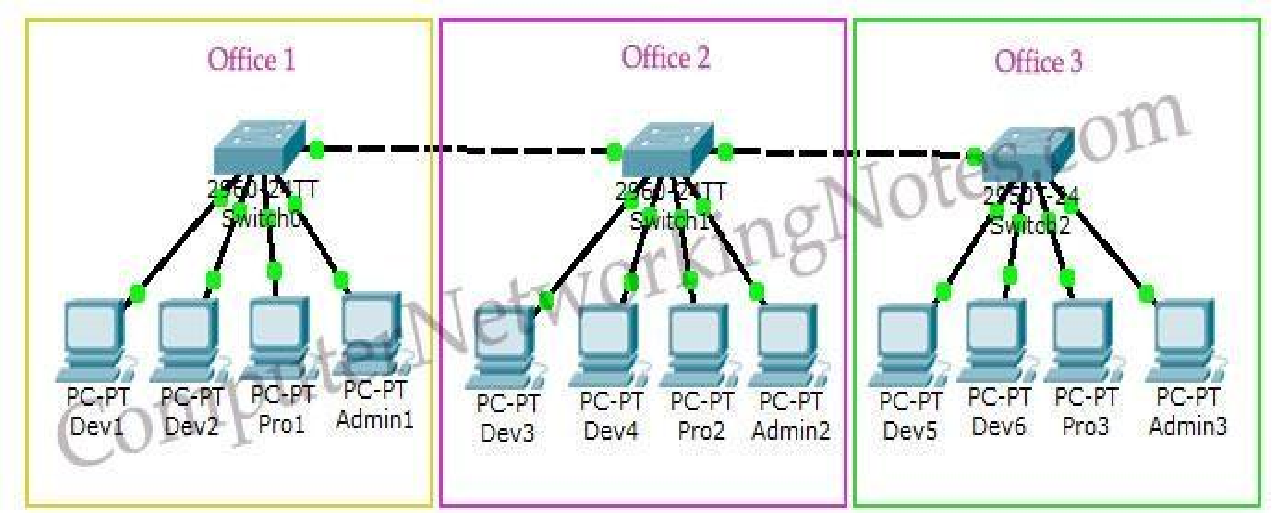

Figure 8: A Sample Network with VLANs

- Our company has three offices.
- All offices are connected with back-links (links connecting switches).
- The company has three departments: Development, Production, and Administration.
- The Development department has six computers (PCs).
- The Production and the Administration department has three PCs separately.
- Each office has two PCs from the Development department and one from both the Production and the Administration departments.
- The Production and the Administration departments have sensitive information that must be separated from the Development department.

With the default configuration, all computers connected to the same switch share a single broadcast domain. The Development department can access the administration or the production department resources.
With VLAN, we can create logical boundaries over the physical network. Assume we created three VLANs for our network and assigned them to the related computers.

- VLAN **Admin** for the Administration department.
- VLAN **Dev** for the Development department.
- VLAN **Pro** for the Production department.

Physically, we changed nothing, but logically, we grouped devices according to their function. These groups [VLANs] need routers of a layer-3 switch to communicate with each other. Logically, our network looks like the one shown in Figure 9.

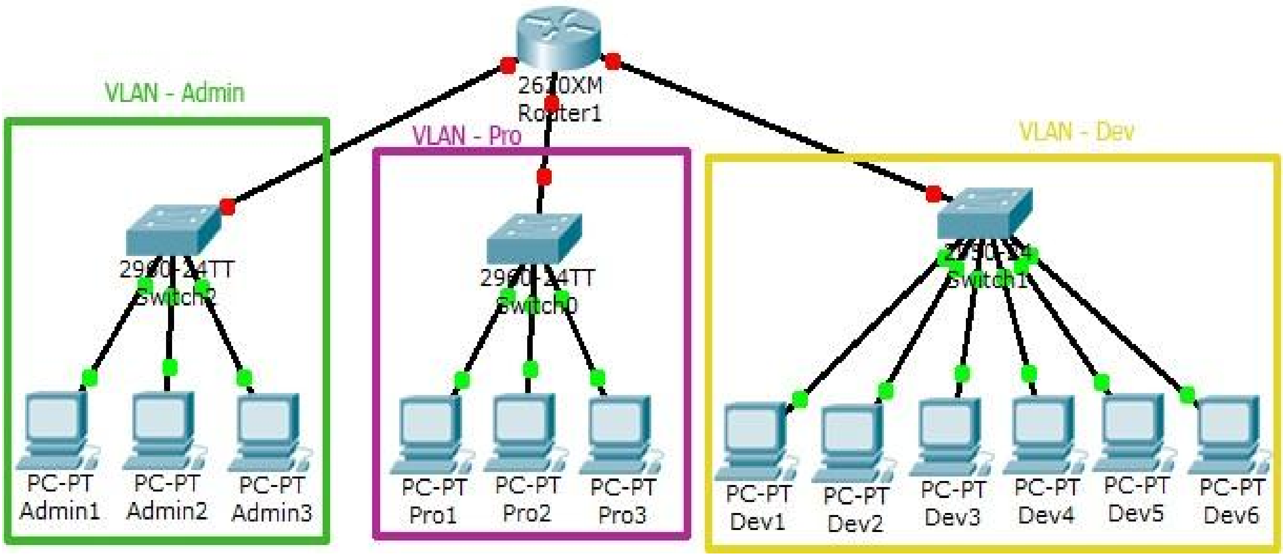

Figure 9: A logical representation of VLANs.

With the help of VLAN, we have separated our single network into three small networks (sub-networks). These networks do not share their broadcast domains with each other, which improves network performance and enhances security. Now, the Development department cannot access the Administration and the Production departments directly.

VLAN Connections

During the configuration of VLAN on ports, we need to know what type of connection it has. Switch supports two types of VLAN connection:

1. Access link
2. Trunk link

Access link

An access link is a connection where a switch port is connected to a device that has a standardized Ethernet NIC. Standard NIC only understands IEEE 802.3 or Ethernet II frames. Access link connection can only be assigned with a single VLAN. That means all devices connected to this port will be in the same broadcast domain.

Trunk link

A Trunk link is a connection where a switch port is connected to a device that is capable of understanding multiple VLANs. Usually, a trunk link connection is used to connect two switches or switches to a router. Remember when we said that VLAN could span anywhere in the network? That is basically due to the trunk link connection. Trunking allows us to send or receive VLAN information across the network. To support trunking, the original Ethernet frame is modified to carry VLAN information.
Figure 10 demonstrates access links and trunk links in a VLAN.

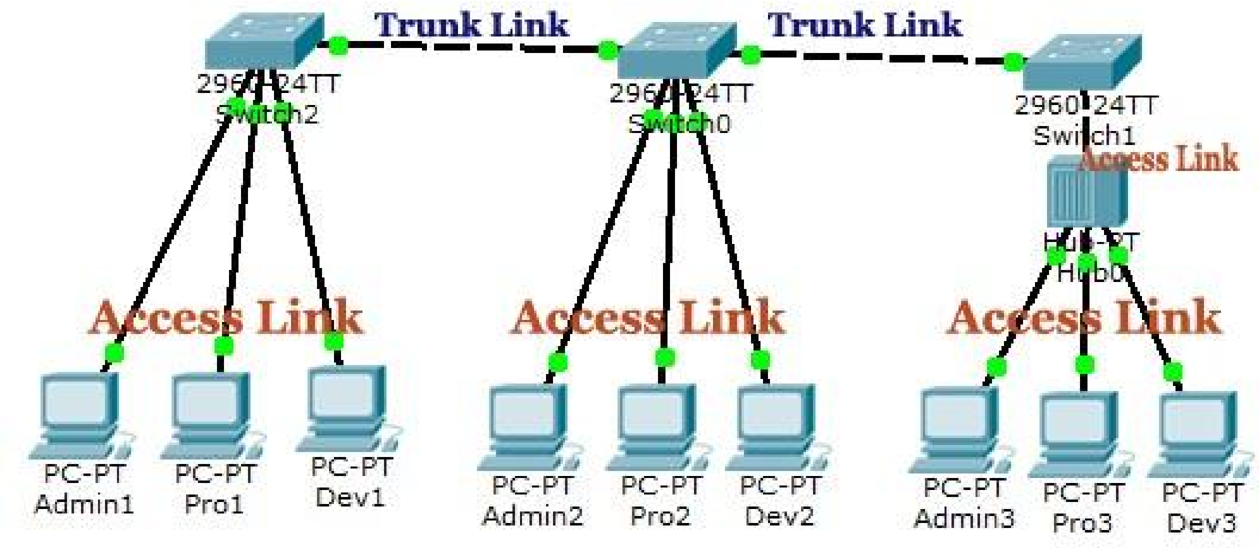

Figure 10: Access links and trunk links in a sample VLAN.

Part B: Inter-VLAN Routing

Inter-VLAN routing is a process for forwarding network traffic from one VLAN to another using a layer-3 device. Two common approaches to inter-VLAN routing are the router-on-a-stick approach and the layer-3 switch, which uses switch virtual interfaces (SVIs).

In the router-on-a-stick approach, one of the router's physical interfaces is configured as an 802.1Q trunk port so it can understand VLAN tags. Note that VLAN tags are used to identify packets belonging to different VLANs so that they can be routed to the appropriate VLAN members. Separate logical subinterfaces are created for each VLAN on that trunk port. Each subinterface is configured with an IP address from the VLAN it represents. The configured subinterfaces are software-based virtual interfaces. VLAN members (hosts) are configured to use the subinterface address as a default gateway. When VLAN-tagged traffic enters the router interface, it is forwarded to the VLAN subinterface. After a routing decision is made based on the destination IP network address, the router determines the exit interface for the traffic and sends out the packet through that interface. The router-on-a-stick method of inter-VLAN routing does not scale beyond 50 VLANs. For this reason, a layer-3 switch using SVIs is used for a scalable solution. Figure 11 is an example of a router-on-a-stick approach to inter-Vlan routing.

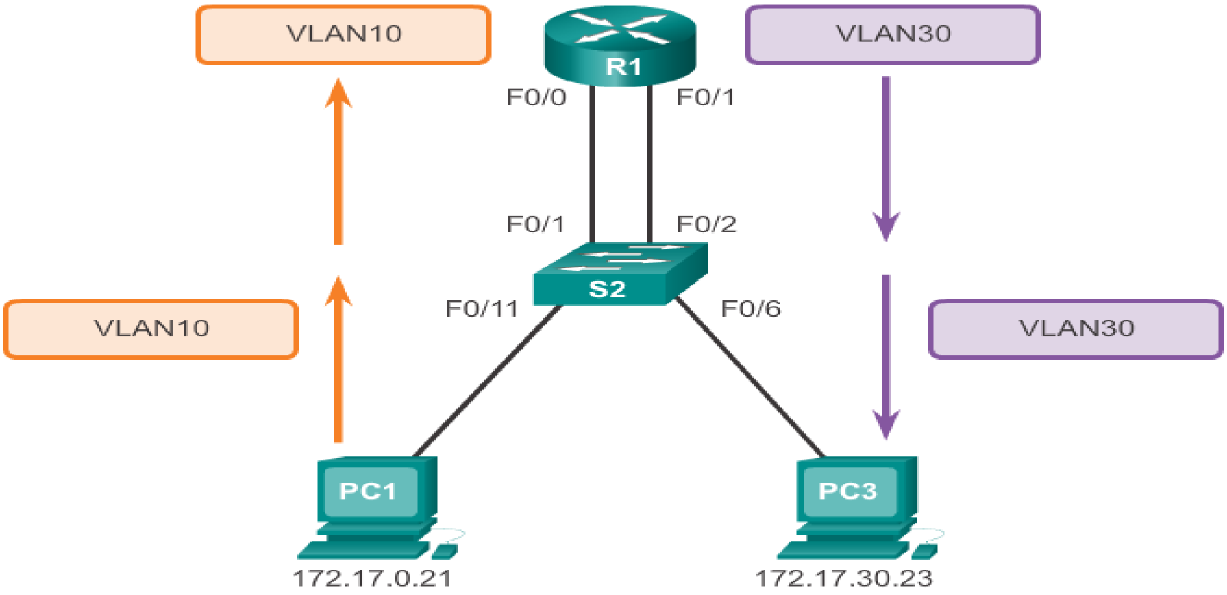

Figure 11: An example of a Router-on-a-Stick approach to inter-VLAN routing.

A layer-3 switch is also known as a Multi-Layer Switch (MLS) as it operates both in layer-2 and layer-3. A switch virtual interface (SVI) is created for each VLAN i.e. one SVI is for one VLAN. The function of a SVI is the same as the router interface in case of the router-on-a-stick approach. It processes the incoming and outgoing packets of the VLANs and routes them accordingly. As the packets do not leave the switch to be routed to a different network, the latency is very low compared to router-on-a-stick approach. This MLS approach is employed in most modern enterprise systems due to its scalability and faster routing. Figure 12 is an example of an MLS approach to inter-VLAN routing.

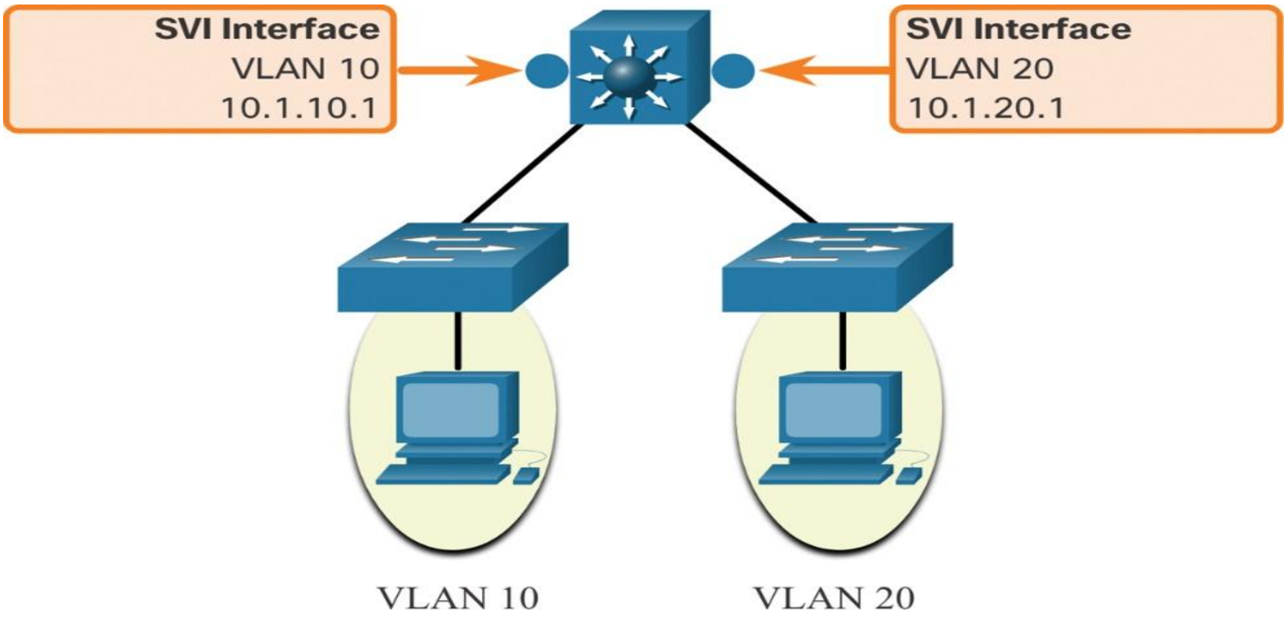

Figure 12: An example of an MLS approach to inter-VLAN routing.

Part C: Configure inter-VLAN routing using Router-on-a-Stick approach:

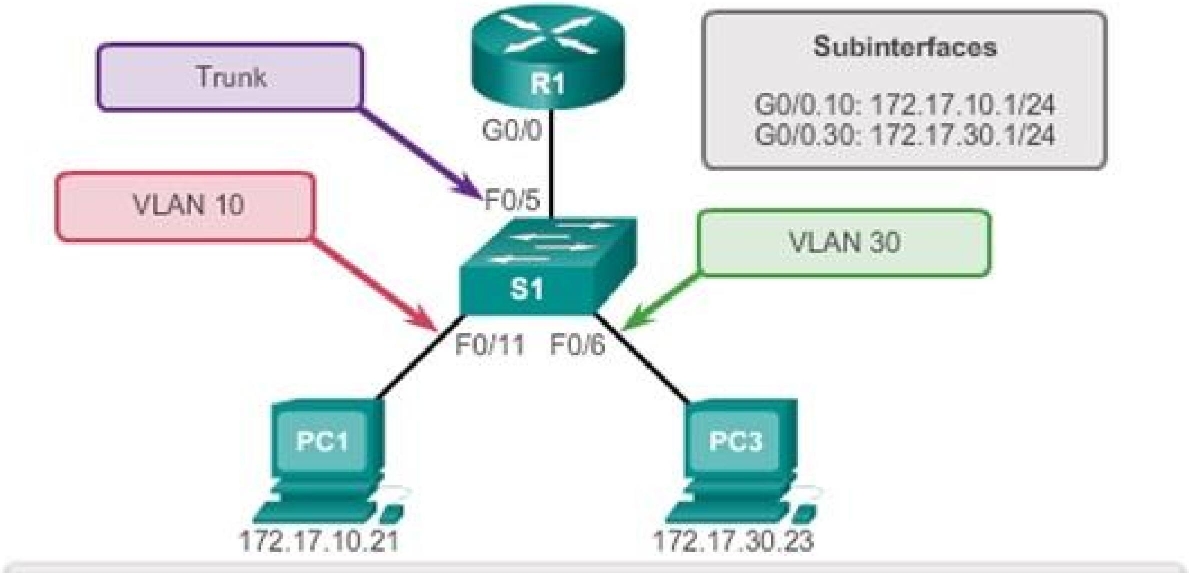

Figure 13: A sample network topology for configuring inter-Vlan routing using the router-on-a-stick approach.

In this section, we will configure the network topology in Figure 13 consisting of two VLANs using the router-on-a-stick approach.

1. At first, configure two (2) Vlans with VLAN ID 10 and 30 inside the switch.

```
S1(config)# vlan 10
S1(config-vlan)# exit
S1(config)# vlan 30
S1(config-vlan)# exit
S1(config)# exit
S1# show vlan

VLAN Name                             Status    Ports
---- -------------------------------- ---------
   -------------------------------
1    default                          active    Fa0/1, Fa0/2, Fa0
   /3, Fa0/4
                                                Fa0/7, Fa0/8, Fa0
                                                   /9, Fa0/10
                                                Fa0/12, Fa0/13,
                                                   Fa0/14, Fa0/15
                                                Fa0/16, Fa0/17,
                                                   Fa0/18, Fa0/19
                                                Fa0/20, Fa0/21,
                                                   Fa0/22, Fa0/23
                                                Fa0/24, Gig0/1,
                                                   Gig0/2
10   VLAN0010                         active    Fa0/11
30   VLAN0030                         active    Fa0/6
```

```
1002 fddi-default                     active
1003 token-ring-default               active
1004 fddinet-default                  active
1005 trnet-default                    active

VLAN Type  SAID       MTU   Parent RingNo BridgeNo Stp  BrdgMode
   Trans1 Trans2
---- ----- ---------- ----- ------ ------ -------- ---- --------
   ------ ------
1    enet  100001     1500  -      -      -        -    -        0
         0
10   enet  100010     1500  -      -      -        -    -        0
         0
30   enet  100030     1500  -      -      -        -    -        0
         0
1002 fddi  101002     1500  -      -      -        -    -        0
         0
1003 tr    101003     1500  -      -      -        -    -        0
         0
1004 fdnet 101004     1500  -      -      -        ieee -        0
         0
1005 trnet 101005     1500  -      -      -        ibm  -        0
         0

VLAN Type  SAID       MTU   Parent RingNo BridgeNo Stp  BrdgMode
   Trans1 Trans2
---- ----- ---------- ----- ------ ------ -------- ---- --------
   ------ ------

Remote SPAN VLANs
------------------------------------------------------------------------------


Primary Secondary Type              Ports
------- --------- -----------------
   ------------------------------------------
```

2. Now, configure the interfaces of the switch belonging to each VLAN.
   The interfaces that connect PCs will be the access link.

```
S1(config)# interface Fast-Ethernet 0/11
S1(config-if)# switchport mode access
```

This command configures the interface as an access link (see the theory section to understand an access link).

```
S1(config-if)# switchport access vlan 10
```

This command assigns VLAN 10 to access ports.

```
S1(config-if)# no shutdown

S1(config)# interface Fast-Ethernet 0/6
S1(config-if)# switchport mode access
```

```
S1(config-if)# switchport access vlan 30
S1(config-if)# no shutdown
```

The interface connected to the router will be the trunk link.

```
S1(config)# interface Fast-Ethernet 0/5
S1(config-if)# switchport mode trunk
```

This command configures the interface as a trunk link (see the theory section to understand a trunk link).

```
S1(config-if)# switchport trunk allowed vlan all
```

This command specifies the list of VLANs on the trunk link. In this case, we have allowed all the VLANs.

```
S1(config-if)# no shutdown
```

3. Finally, configure the router subinterfaces.

```
R1(config)# interface g0/0.10
R1(config-subif)# encapsulation dot1q 10
R1(config-subif)# ip address 172.17.10.1 255.255.255.0
R1(config-subif)# exit
R1(config)# interface g0/0.30
R1(config-subif)# encapsulation dot1q 30
R1(config-subif)# ip address 172.17.30.1 255.255.255.0
R1(config-subif)# exit
R1(config)# interface g0/0
R1(config-if)# no shutdown
```

The command `encapsulation dot1q ##` enables IEEE 802.1Q encapsulation of network traffic on the specified subinterface. Also remember to specify the VLAN id after the interface identifier e.g., `interface g0/0.10`

4. Now, verify the inter-Vlan routing configuration and subinterfaces by issuing the command `show ip route`.

```
R1# show ip route
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route

Gateway of last resort is not set

```

```
     172.17.0.0/16 is variably subnetted, 4 subnets, 2 masks
C       172.17.10.0/24 is directly connected, GigabitEthernet0/0.10
L       172.17.10.1/32 is directly connected, GigabitEthernet0/0.10
C       172.17.30.0/24 is directly connected, GigabitEthernet0/0.30
L       172.17.30.1/32 is directly connected, GigabitEthernet0/0.30
```

Student ID:

Lab 5 Task

There are in total three departments (EEE, ME, AEE) in University X. They all reside in Academic Building 1. You are the network administrator and have been given the Network - 192.id.id+1.0/24 to be used for the connection. EEE has 25 devices, ME has 30 devices, and AEE has 15 devices. You have divided the network to waste minimum possible IP addresses. But, they are all connected with a single LAN instead of three separate LAN. Now, divide them logically so that they don't share the same broadcast domain. (Attach only two devices for each departments but interfaces should be configured for all the number of devices), use minimum two switches.

Lab 5 Grading Criteria

1. **Correct Implementation (40 pts)**
   - Correct subnetting (10 pts)
   - Correct VLAN configuration in switches and router (30 pts)
   - Successful completion of sending packets (10 pts)
2. **Total: 50 pts**