



Basic Network For Trainee

Panthakit Totid



Outline

Day 1

- Chapter 1: Introduction to Networks
- Chapter 2: The OSI and TCP/IP Reference Model
- Chapter 3: Networking Devices
- Chapter 4: IPv4 Address and Subnetting (1)

Day 2

- Chapter 4: IPv4 Address and Subnetting (2)
- Chapter 5: Function of Routing
- Chapter 6: Static Routing

Day 3

- Chapter 7: STP Concepts and EtherChannel
- Chapter 8: VLANs and Trunks
- Chapter 9: Routing Between VLANs
- Chapter 10: Dynamic Routing (1)

Day 4

- Chapter 10: Dynamic Routing (2)
- Chapter 11: Layer 3 Redundancy
- Chapter 12: ACL Concepts

Panthakit Totid

Panthakit Totid



Basic Network For Trainee

Module 1

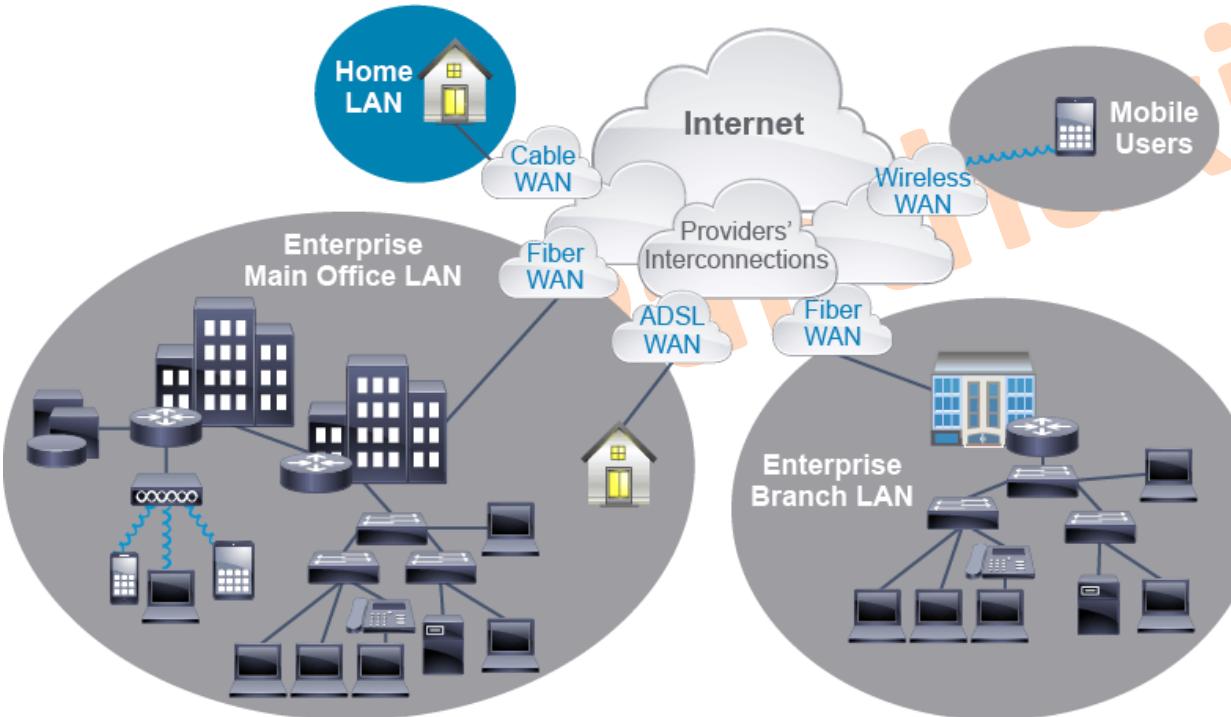
Introduction to Networks

Panthakit Totid



Introduction

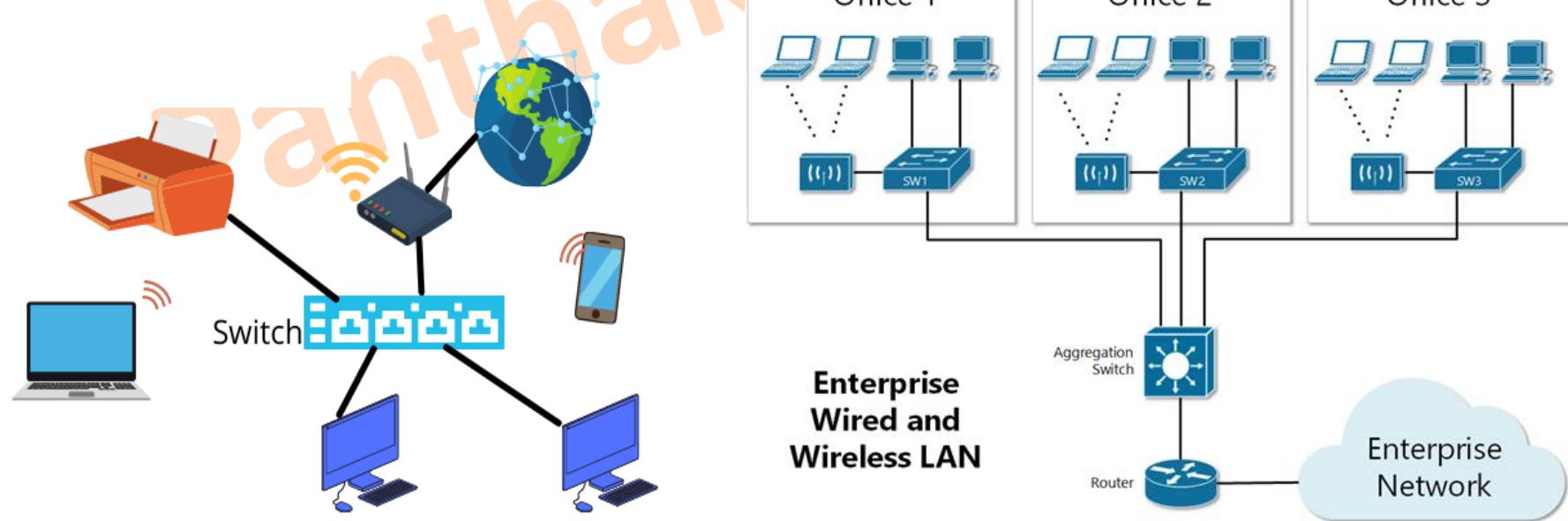
- At the most basic level, a “**network**” is defined as a group of systems interconnected to share resources. You can find examples of such systems and resources in a social network to share work experience or personal events or a computer network to share file storage, printer access, or internet connectivity.



- Network infrastructures vary greatly in terms of:
 - Size of the area covered
 - Number of users connected
 - Number and types of services available
 - Area of responsibility
- Two most common types of networks:
 - Local Area Network (LAN)**
 - Wide Area Network (WAN)**

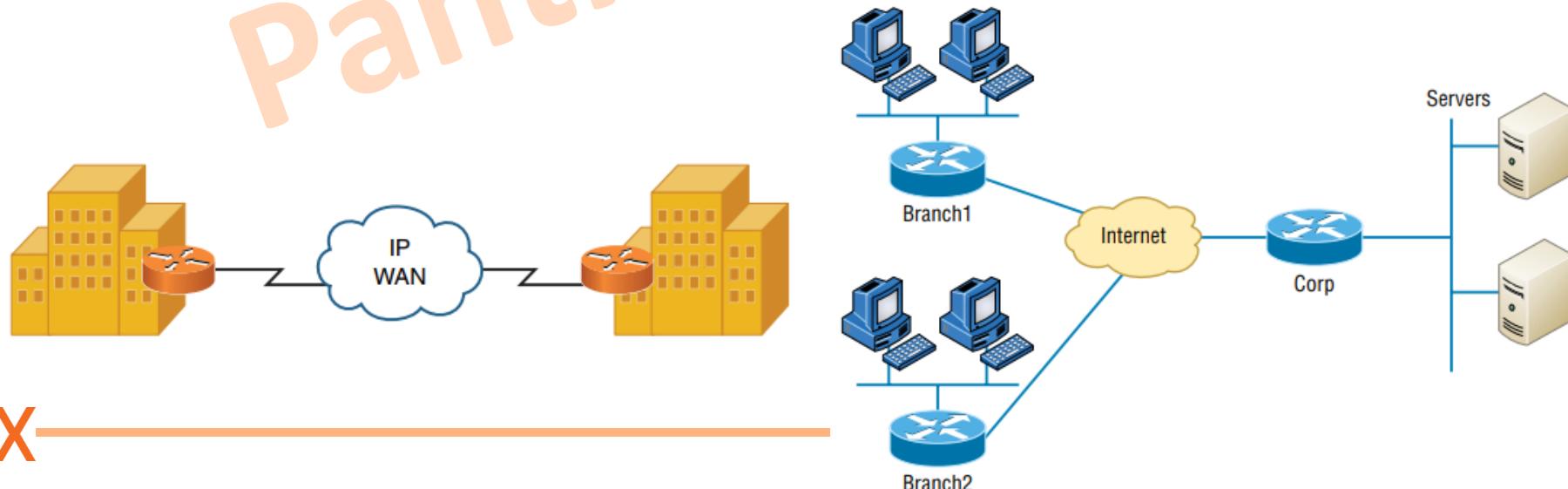
Introduction (Cont.)

- A **LAN** interconnects network components within a local area or **single location** (for example, within a building).
 - It can include both **residential networks** with a couple of computers and **enterprise networks** with hundreds of servers and thousands of workstations.
 - Typically, most of the **equipment** and **cabling** used on a LAN is owned and operated by the company or organization using the LAN.



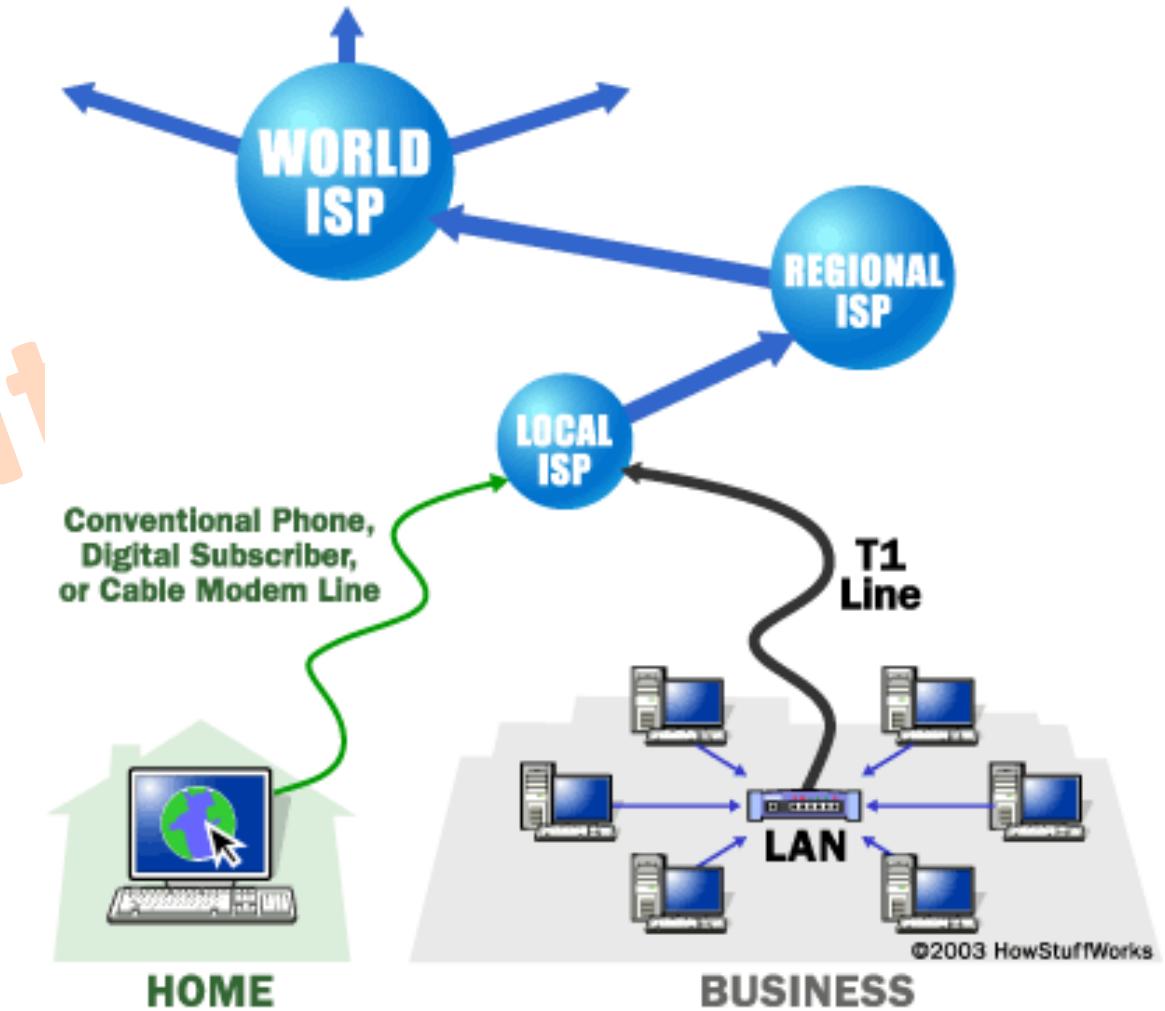
Introduction (Cont.)

- A **WAN** interconnects network components that are **geographically separated**. For example, a corporate headquarters might have multiple WAN connections to remote office sites.
 - A WAN is more likely to make use of a **service provider** network.
 - It is typically provided by different telecommunication providers using various technologies using different media such as fiber, copper, cable, ADSL, or wireless links.
 - In enterprise internetworks, WANs connect the main office, branches, SOHO, and mobile users.



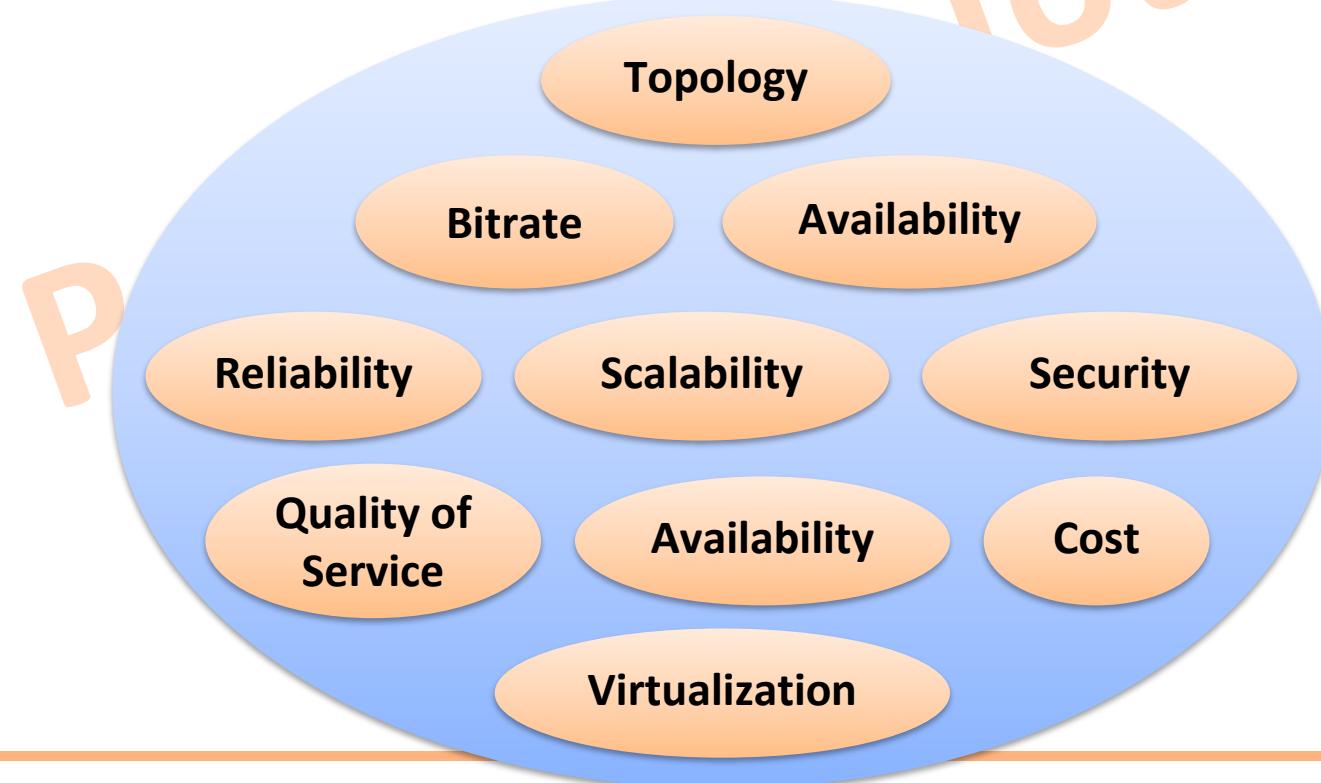
Introduction (Cont.)

- WAN



Characteristics of a Network

- When you purchase a mobile phone or a PC, the specifications list tells you the important characteristics of the device, just as specific characteristics of a network help describe its performance and structure.
- You can describe the qualities and features of a network by considering these characteristics:



Characteristics of a Network (Cont.)

- **Topology:** A network topology is the **arrangement** of its elements. Topologies give insight into physical connections and data flows among devices. In a carefully designed network, data flows are optimized, and the network performs as desired.
- **Bitrate:** Bitrate measures the data rate in **bits per second (bps)** of a given link in the network.
 - This measure is often referred to as **bandwidth** or **speed** in device configurations, which is sometimes thought of as speed.
 - However, it is not about how fast 1 bit is transmitted over a link—which is determined by the physical properties of the medium that propagates the signal—it is about the number of bits transmitted in a second.
 - Link bitrates commonly encountered today are 1 and 10 gigabits per second (1 or 10 billion bits per second).

Characteristics of a Network (Cont.)

- **Availability:** Availability indicates **how much time** a network is accessible and operational. Availability is expressed in terms of the percentage of time the network is operational.
 - This percentage is calculated as a ratio of the time in minutes that the network is available and the total number of minutes over an agreed period, multiplied by 100.

Device_X Information

- Reboots Every Hour
- Reboot Duration = 5 Minutes

$$\text{Availability} = \frac{\text{uptime}}{\text{uptime} + \text{downtime}}$$

$$\text{Reliability Mean Time Between Failures (MTBF)} = \frac{\text{time in service}}{\text{number of failures}}$$

$$\text{Device_X Availability} = \frac{\text{total uptime in a day in minutes}}{\text{total minutes in a day}} * 100 = \frac{24*55}{24*60} * 100 = \frac{1320}{1440} * 100 = 91.67\%$$

$$\text{Device_X Reliability MTBF} = \frac{\text{total time in service in a day}}{\text{number of failures in a day}} = \frac{24*55}{24} = \frac{1320}{24} = 55 \text{ minutes}$$

In other words, availability is the ratio of uptime and total time, expressed in percentage.

Characteristics of a Network (Cont.)

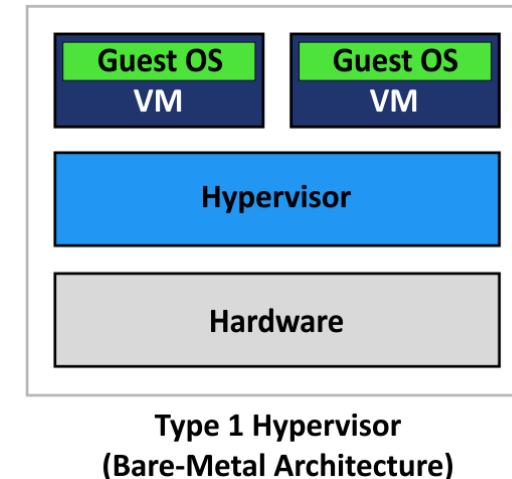
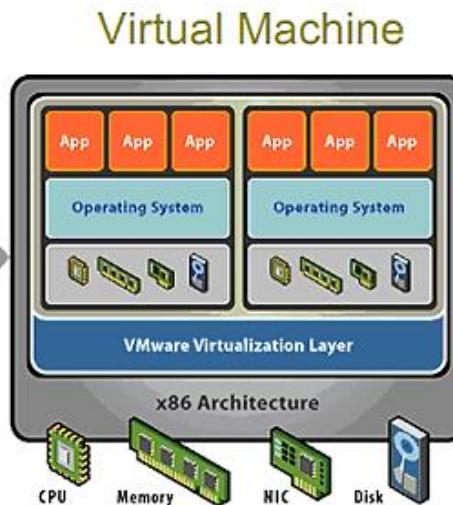
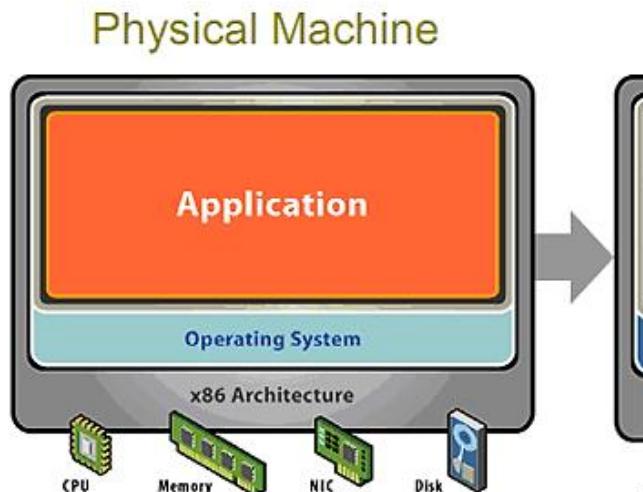
- **Availability:**

Availability %	Friendly Name	Downtime		
		per Year	per Month	per Week
90%	one nine	36.5 days	72 hours	16.8 hours
99%	two nines	3.65 days	7.2 hours	1.68 hours
99.5%	--	1.83 days	3.6 hours	50.4 minutes
99.9%	three nines	8.76 hours	43.8 minutes	10.1 minutes
99.95%	--	4.38 hours	21.56 minutes	5.04 minutes
99.99%	four nines	52.56 minutes	4.32 minutes	1.01 minutes
99.999%	five nines	5.26 minutes	25.9 seconds	6.05 seconds
99.9999%	six nines	31.5 seconds	2.59 seconds	0.605 seconds
99.99999%	seven nines	3.15 seconds	0.259 seconds	0.0605 seconds

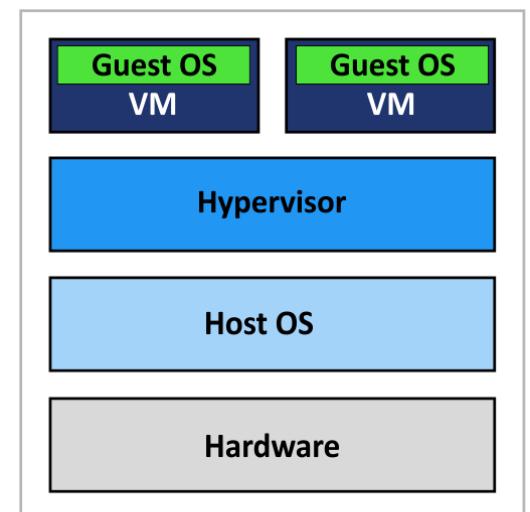
<https://manishsharma.blog/2020/02/04/design-for-availability-game-of-9s>

Characteristics of a Network (Cont.)

- **Virtualization:** Traditionally, network services and functions have only been provided via hardware. Network virtualization creates a software solution that **emulates network services and functions**.
- Virtualization solves many of the networking challenges in today's networks, helping organizations centrally automate and provision the network from a central management point.



Type 1 Hypervisor
(Bare-Metal Architecture)

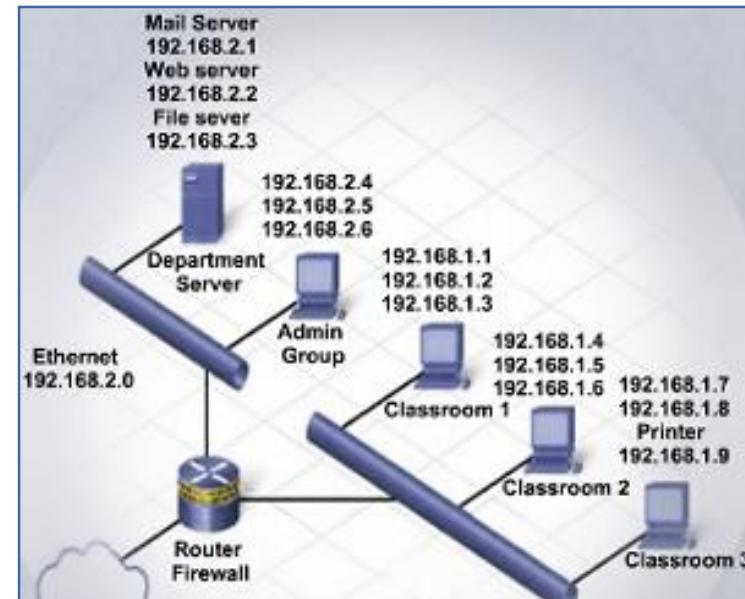
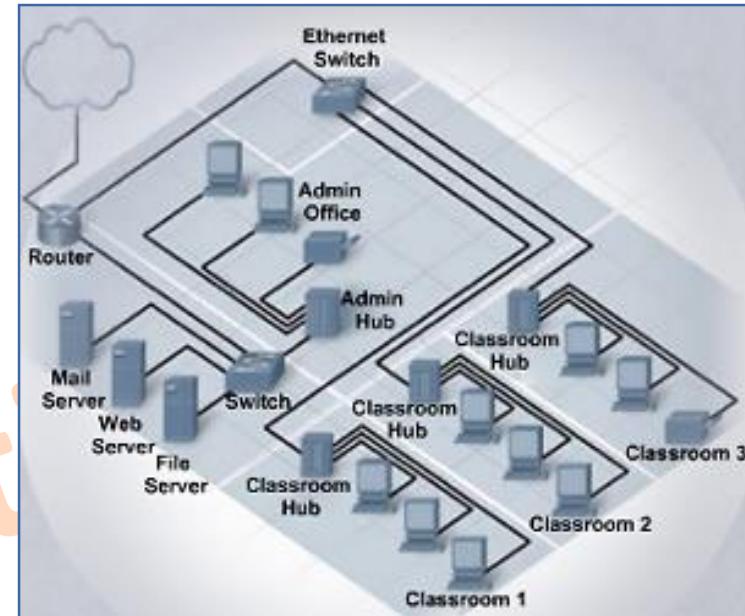


Type 2 Hypervisor
(Hosted Architecture)

Physical vs. Logical Topologies

Each network has both a physical and a logical topology.

- The **physical topology** of a network refers to the **physical layout** of the devices and cabling. The term node is commonly used when discussing topology diagrams. For networking topology diagrams, a node is a device.
- The **logical topology** is determined by the **intermediary devices and the protocols** chosen to implement the network.
 - How the hosts access the medium to communicate across the network.



Physical vs. Logical Topologies

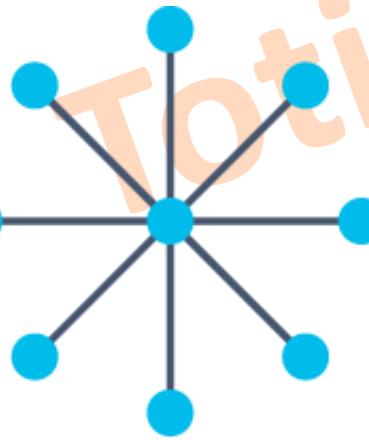
- The following figure represents some of the physical topologies that you may encounter.



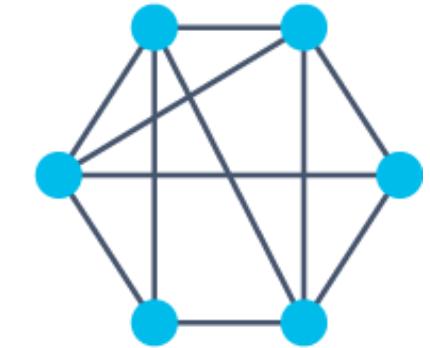
Bus Topology



Ring Topology



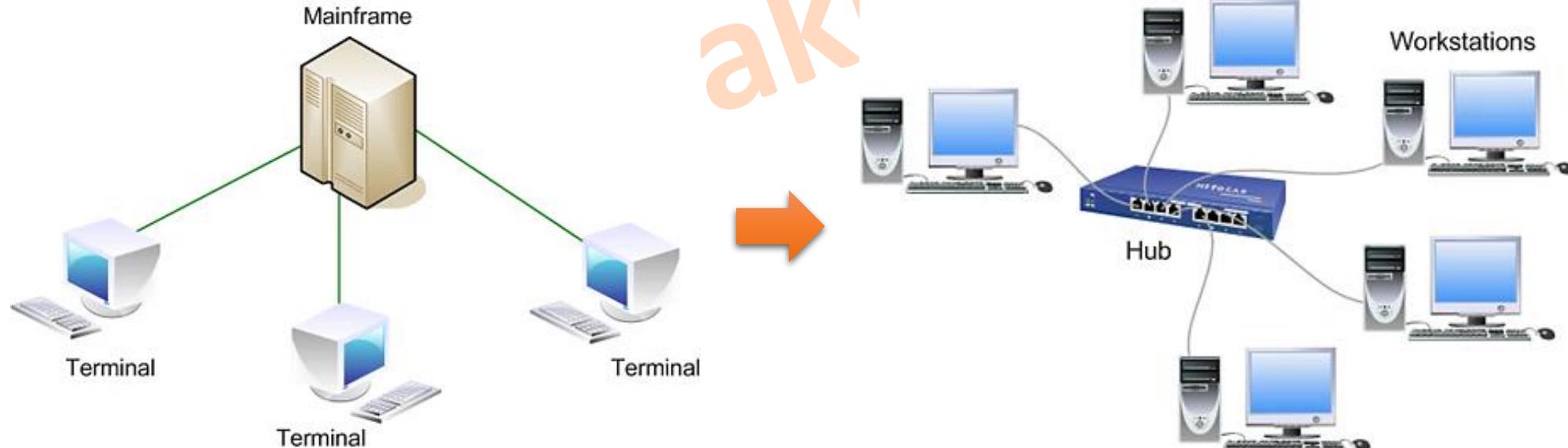
Star Topology



Mesh Topology

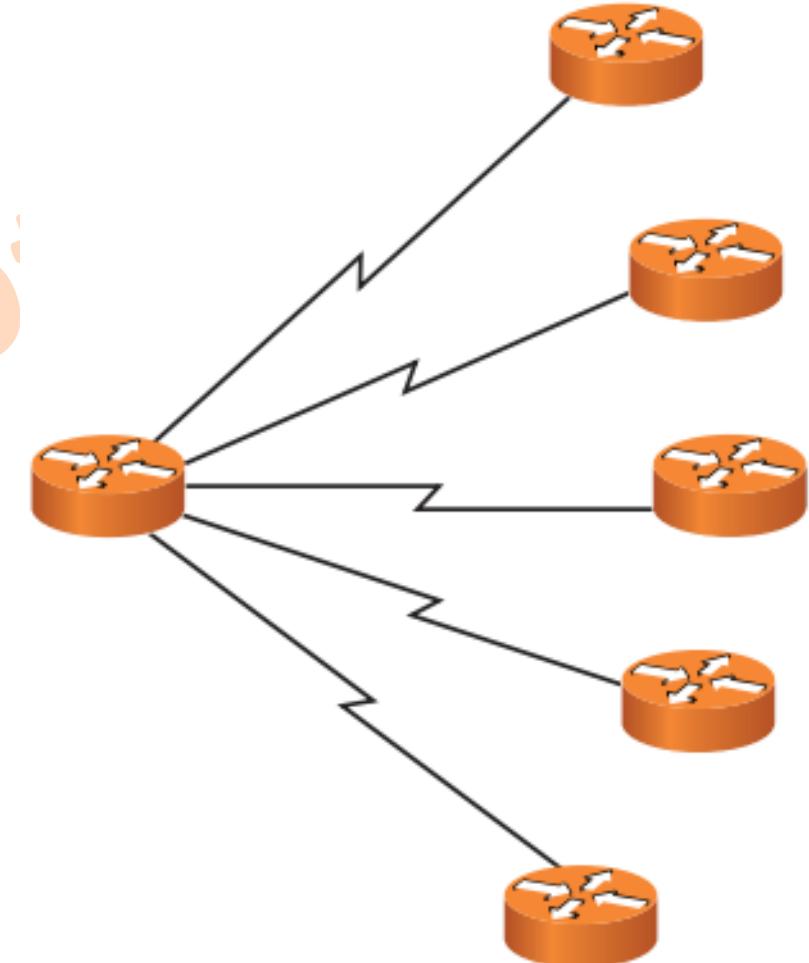
Star Topology

- In star topology network, there is a **central switch (or Hub)** through which all network nodes or devices transmit or receive data.
- A single node failure does not bring the network down (only that single node is affected).
 - However, the failure if the central switch brings down the entire network.



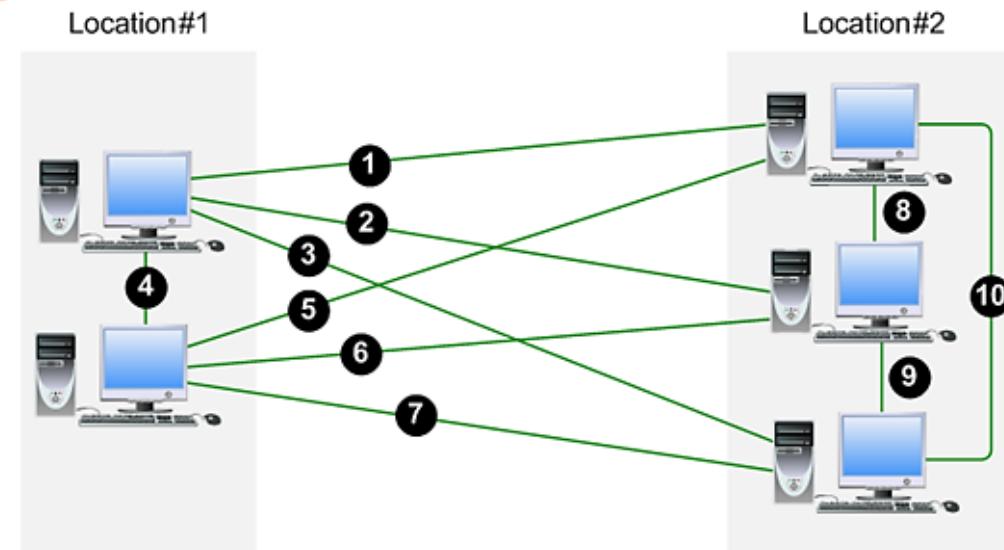
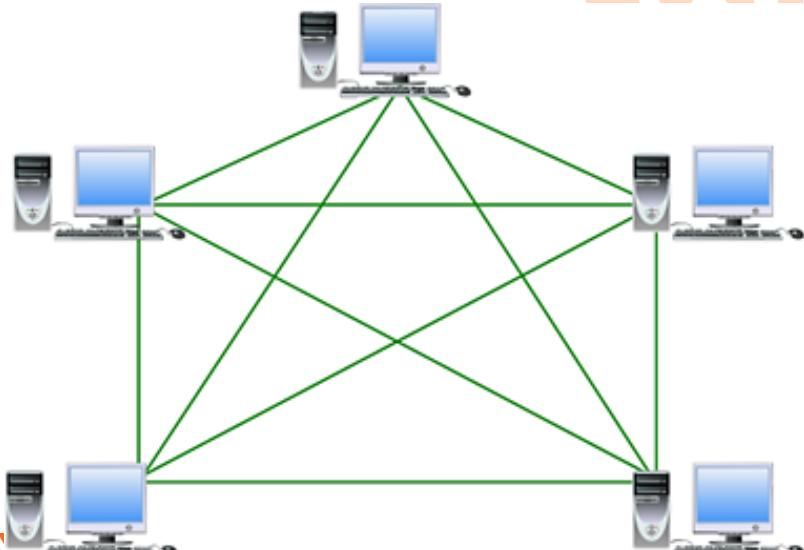
Star Topology (Cont.)

- This topology also called the **hub and spoke topology**.
 - When interconnecting **multiple sites** (for example, multiple corporate locations) via WAN links, a **hub-and-spoke topology** may be used, with a WAN link from each remote site (**spoke site**) to the main site (**hub site**).
 - This topology helps **minimize WAN expenses** by not directly connecting any two spoke locations.
 - If two spoke locations need to communicate with each other, their communication is sent via the hub location.



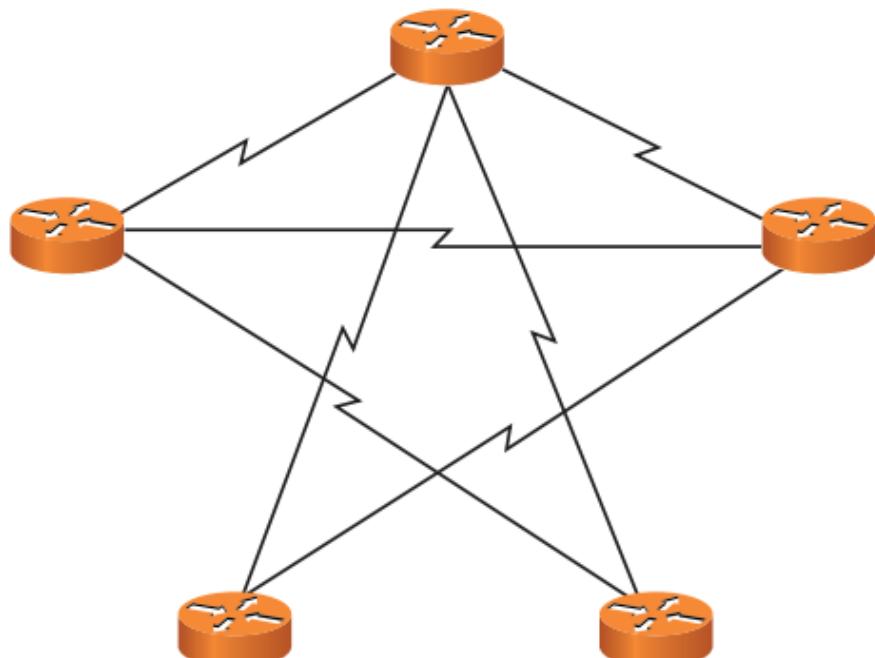
Mesh Topology

- In mesh topology network, each node **connects to all other nodes directly**. To find the number of physical links in a fully connected mesh network with n nodes. We need $n(n - 1)/2$ physical links.
- The advantage is that there is more than one path between any two nodes. In case of link failures, a **redundant path** can be found.
 - A full-mesh topology is **highly fault tolerant**.



Mesh Topology (Cont.)

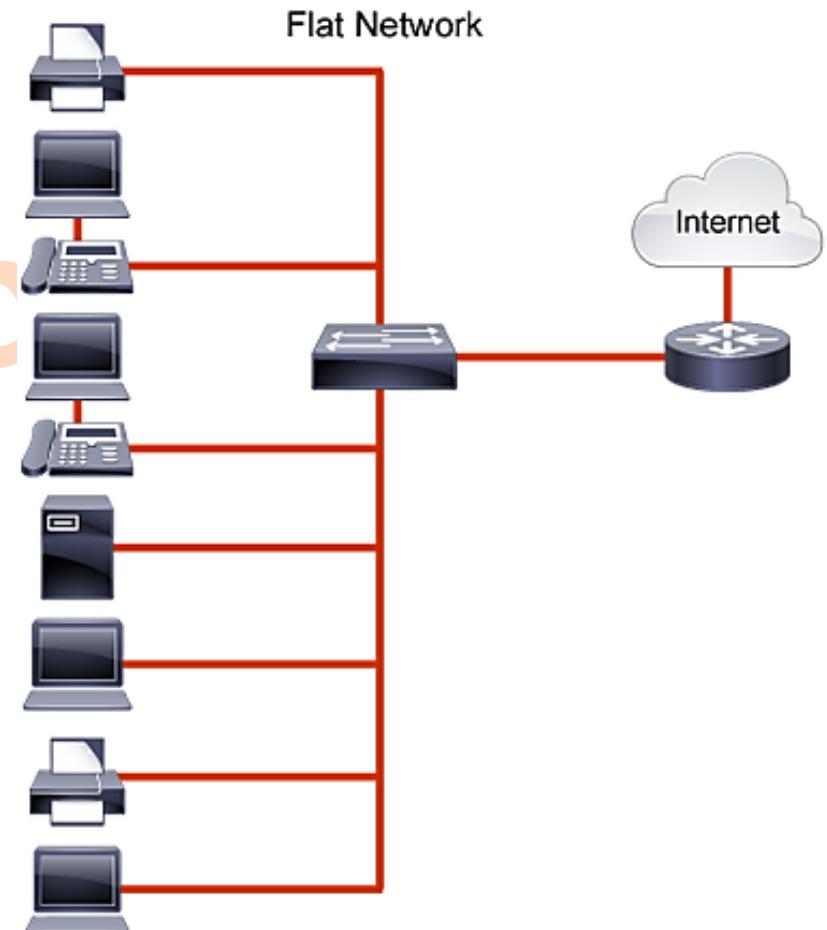
- A **partial-mesh** WAN topology is a **hybrid** of hub-and-spoke topology and full-mesh topology.
- Specifically, a partial-mesh topology can be designed to offer an optimal route between selected sites while avoiding the expense of interconnecting every site to every other site.



A network designer must consider network traffic patterns and strategically add links interconnecting sites that have higher volumes of traffic between themselves.

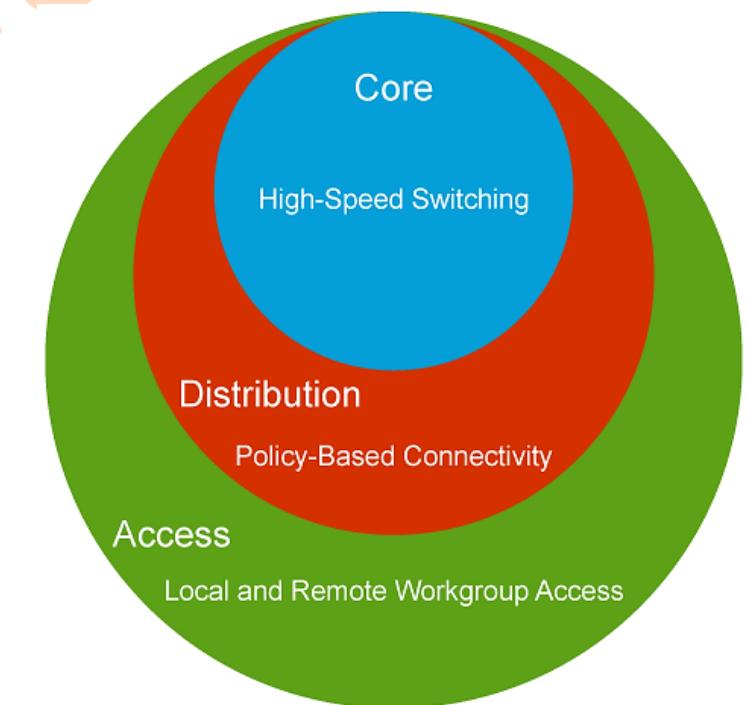
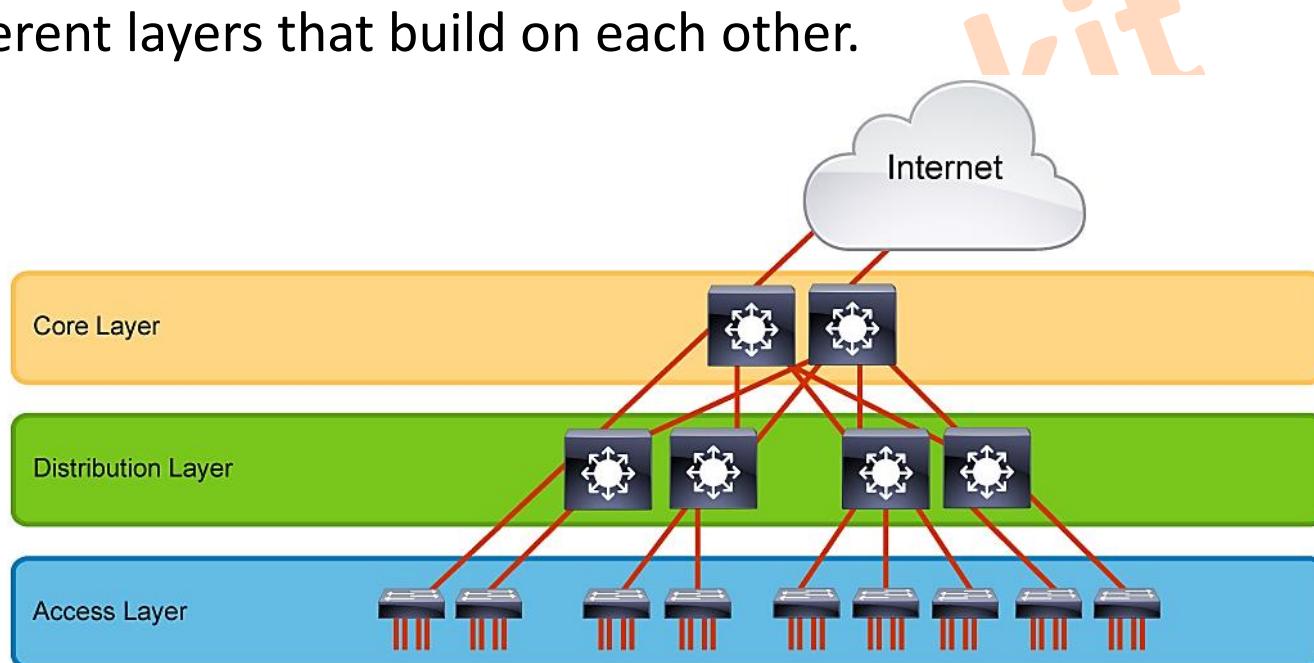
Hierarchical Network Design

- A **flat enterprise** campus network is where all PCs, servers, and printers are connected to each other using Layer 2 switches.
 - All devices on this subnet will share available bandwidth and all are members of the ***same broadcast domain***.
 - Wasting bandwidth and computational resources.
- Flat networks do not scale to meet the needs of most enterprise networks or of many small and medium-size businesses.
- You can use a **Layer 3 device** such as a router or a Layer 3 switch to segment a network.
 - Broadcasts that originate within a subnet will not propagate beyond the edge of the LAN segment.



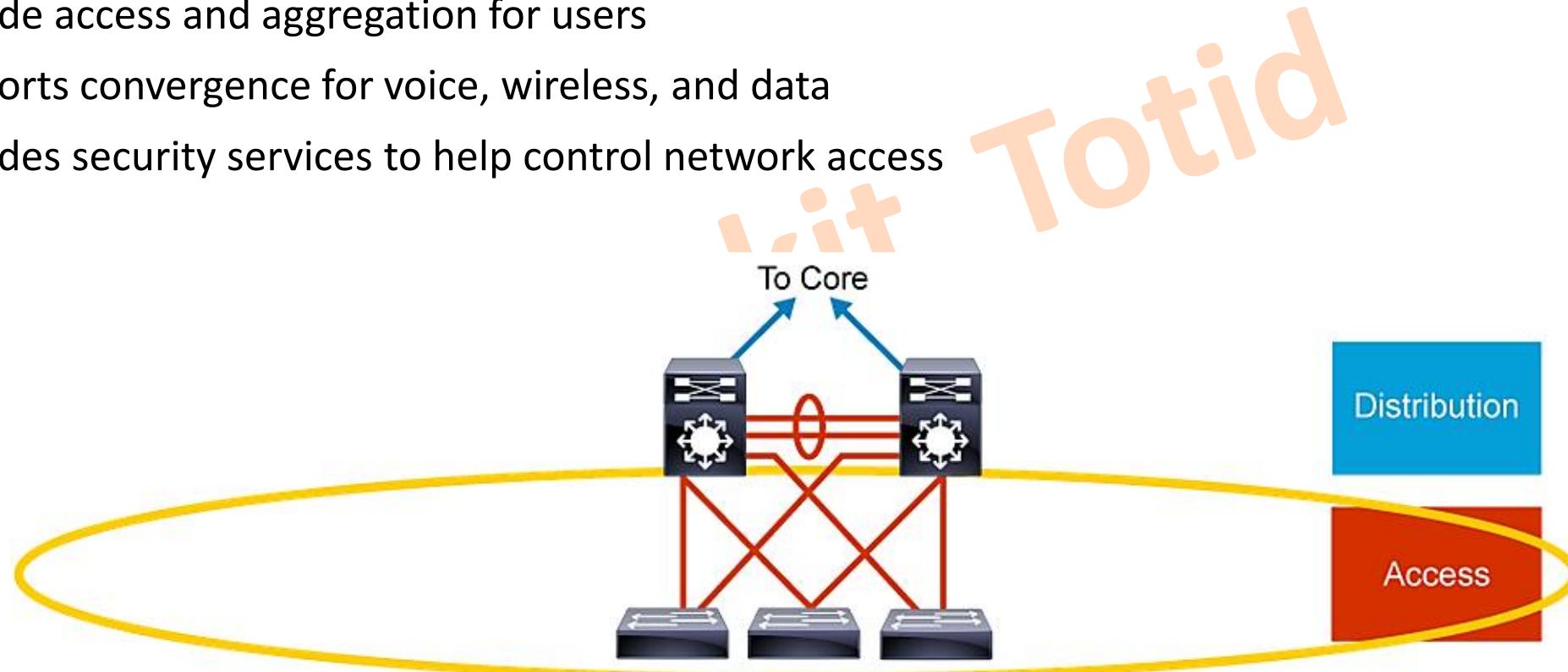
Campus Network Design

- Hierarchical models for internetwork design allow you to design internetworks in layers.
- The **Cisco Enterprise Campus Architecture models** also uses layers to simplify the task that is required for internetworking.
- Leveraging the hierarchical model also simplifies campus network design by allowing focus at different layers that build on each other.



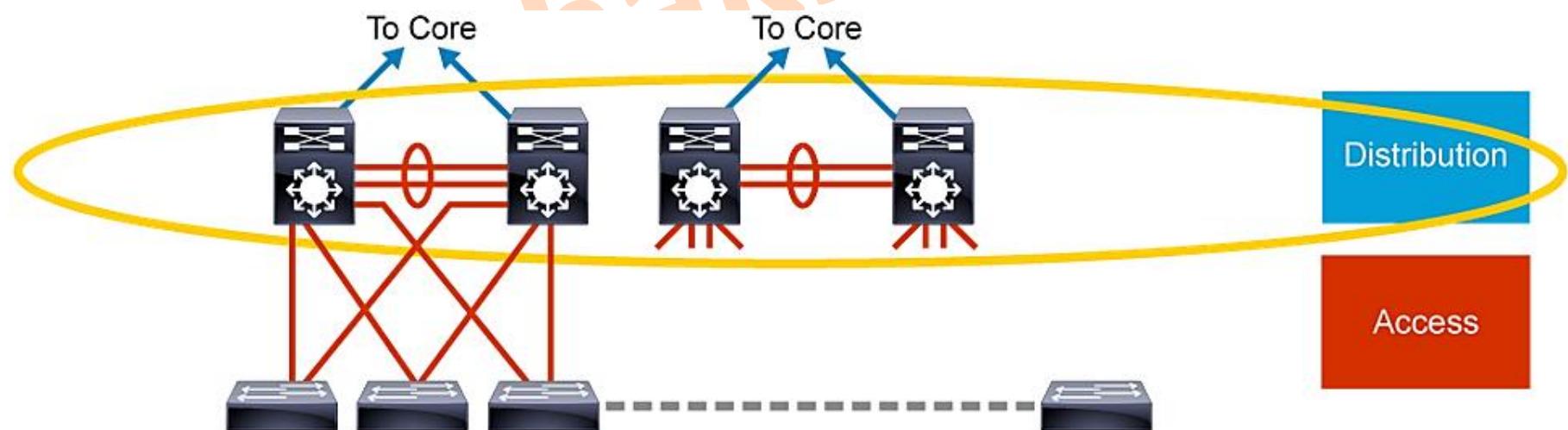
Access Layer

- The access layer **aggregates** end users and **provides uplinks** to the distribution layer.
 - Provide access and aggregation for users
 - Supports convergence for voice, wireless, and data
 - Provides security services to help control network access



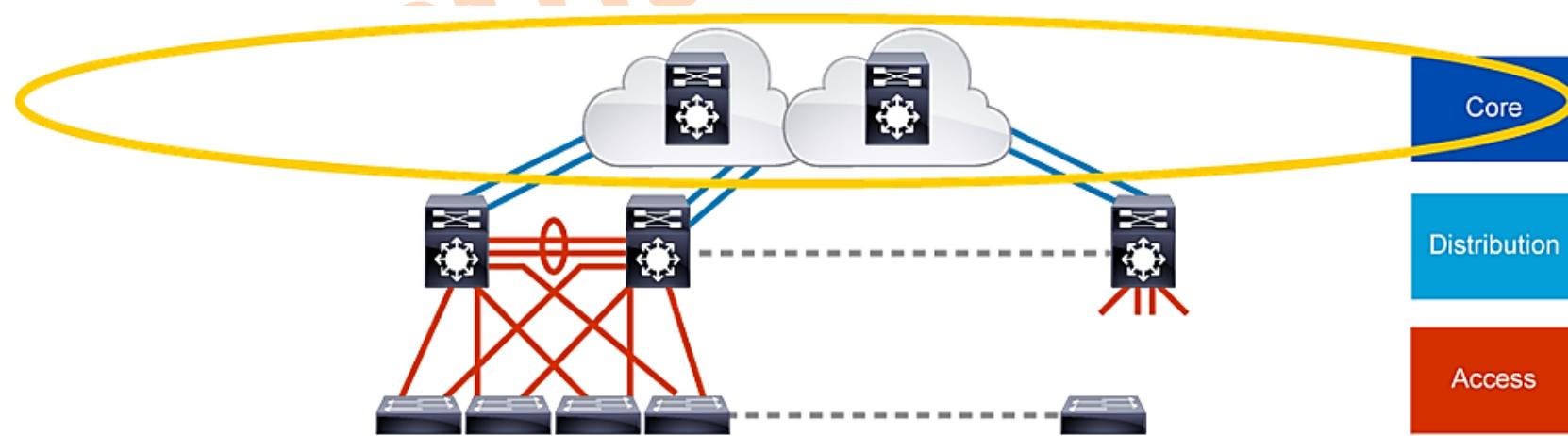
Distribution Layer

- Acts as a **services** and **control boundary** between the access layer and the core.
- The distribution layer serves multiple purposes
 - Aggregates access nodes and uplinks
 - Provides redundant connections and devices for high availability
 - Offers routing and implements policies (filtering, security, QoS)

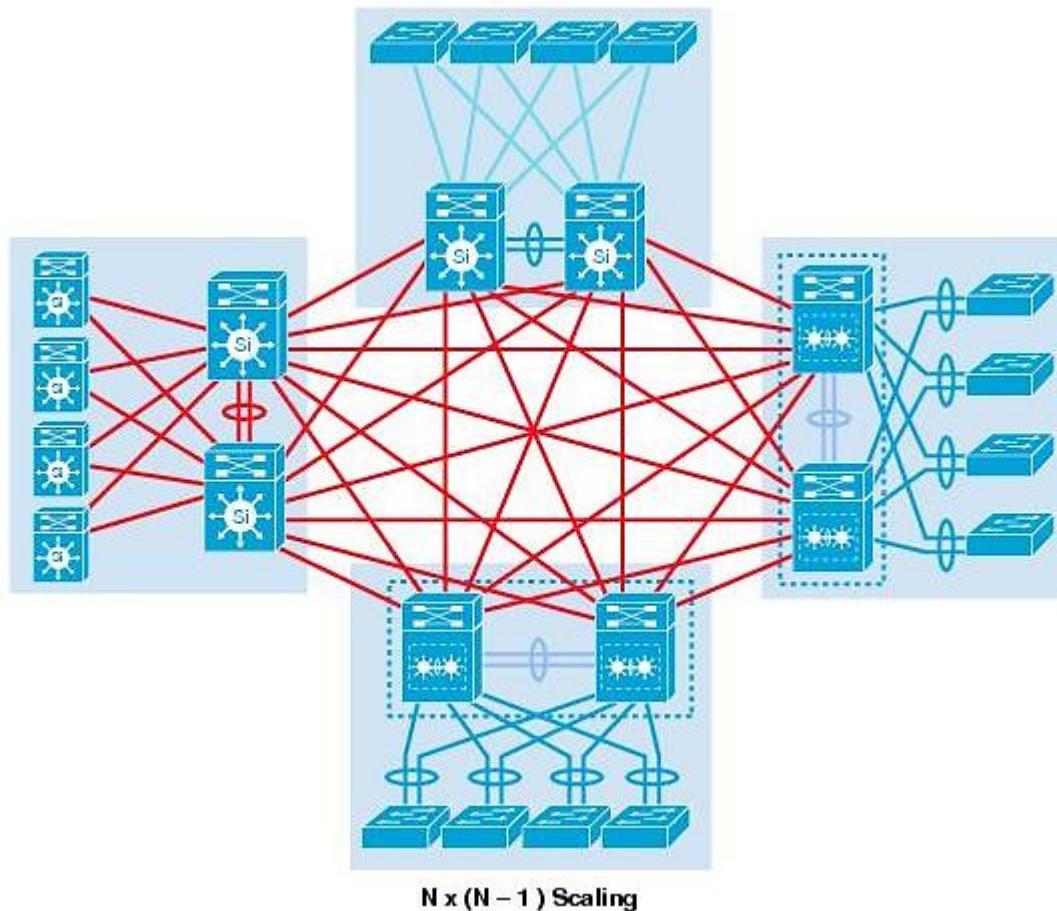


Core Layer

- The core layer is the **aggregation point** for the other layers and modules in the Cisco Enterprise Campus Architecture.
- The core must provide a high level of redundancy and adapt to changes very quickly.
 - It provides a limited set of services and is designed to be highly available and requires 100 percent uptime.
 - In large enterprises, the core of the network must operate as a nonstop, always-available service.

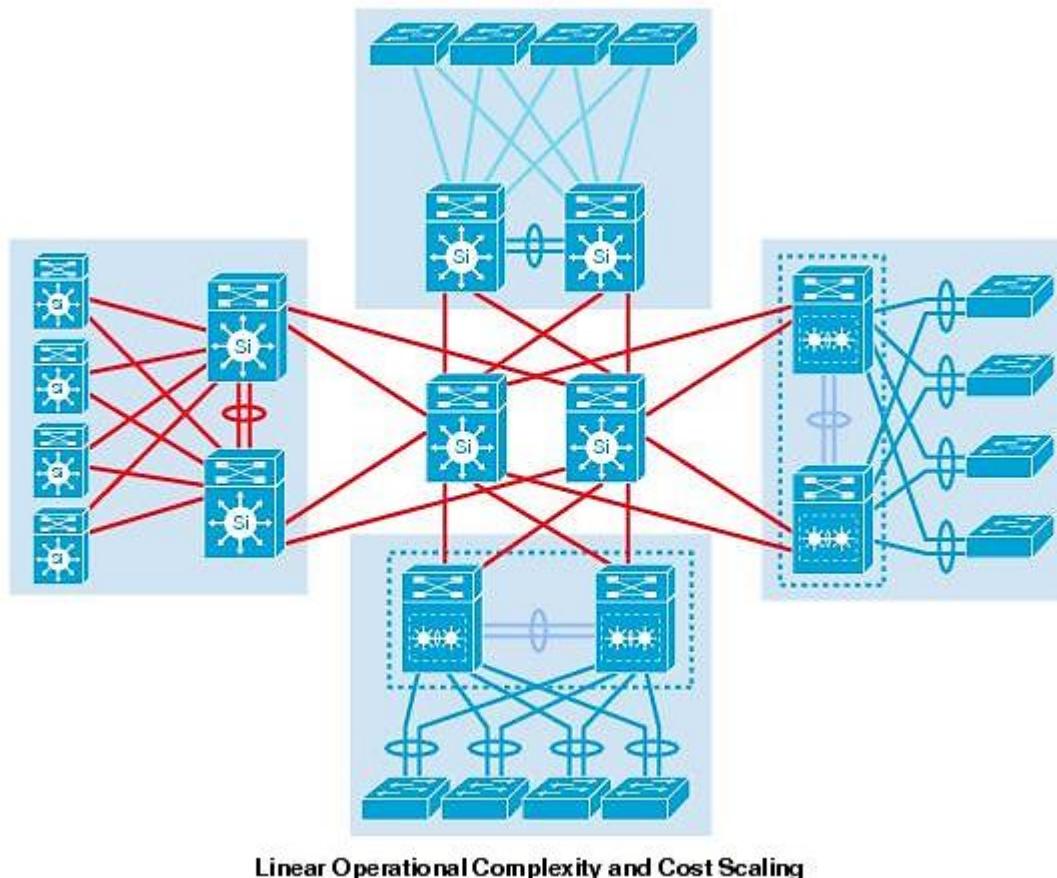


Is a Core Layer Needed?



- In environments where the campus is contained within a single building - or multiple adjacent buildings with the appropriate amount of fiber - it is possible to collapse the core into distribution switches.

Is a Core Layer Needed?



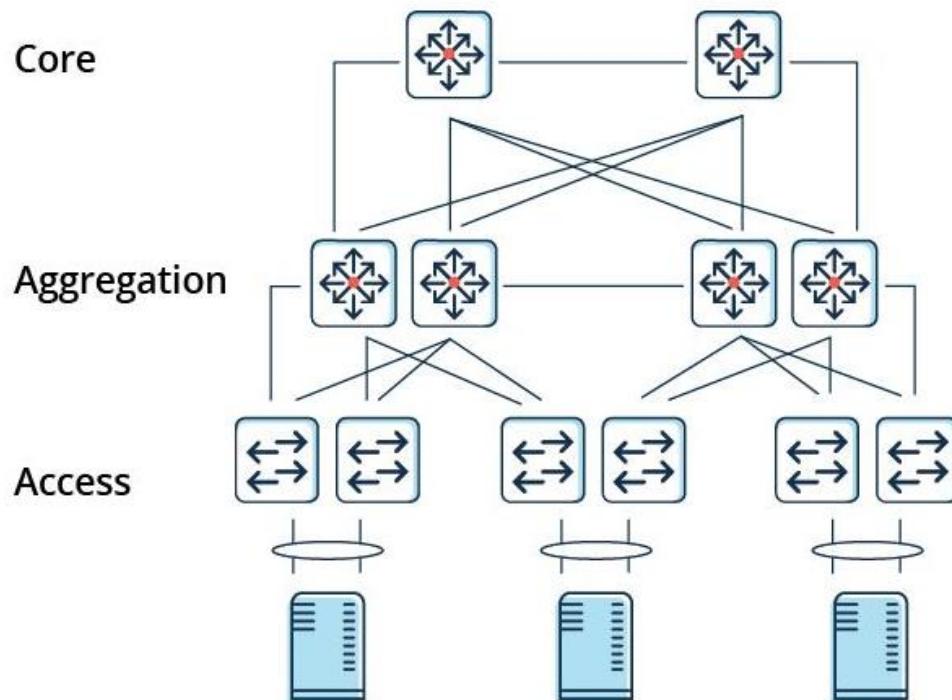
- Having a **dedicated core** layer allows the campus to accommodate this growth without compromising the design of the distribution blocks, the data center, and the rest of the network.
- In a larger, more complex campus, the core provides the capacity and scaling capability for the campus as a whole.

The hierarchical LAN design is more appropriate for **north-south traffic** flows, such as endpoints communicating with the WAN edge, data center, Internet, or network services blocks.

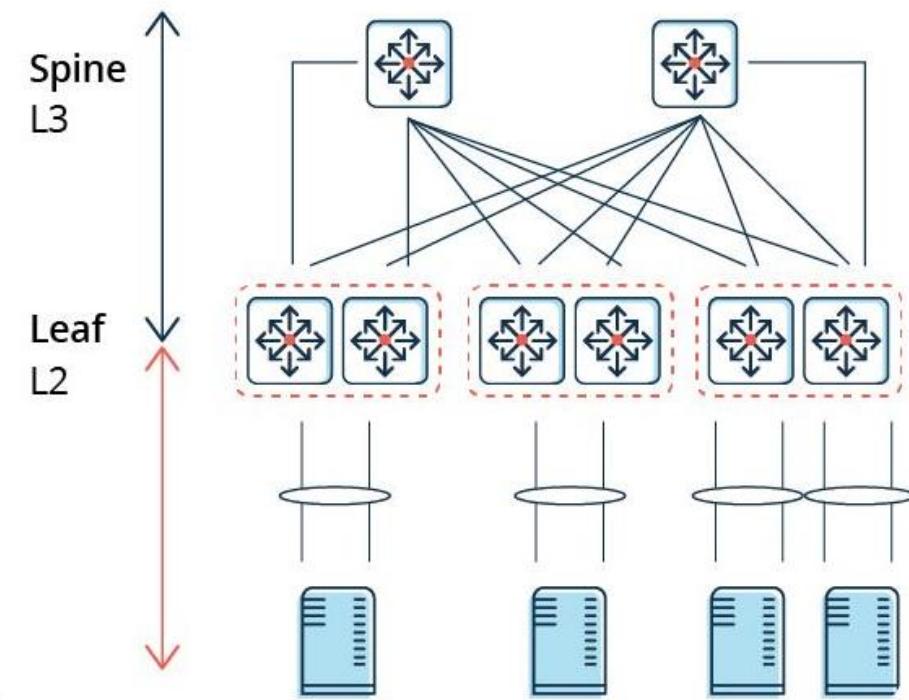
Spine and Leaf Architecture

- The data center block is using the newer **leaf-spine design**, an alternative to the three-tier design when traffic is predominantly **east-west** between servers within the data center.

Traditional 3-Tier Architecture

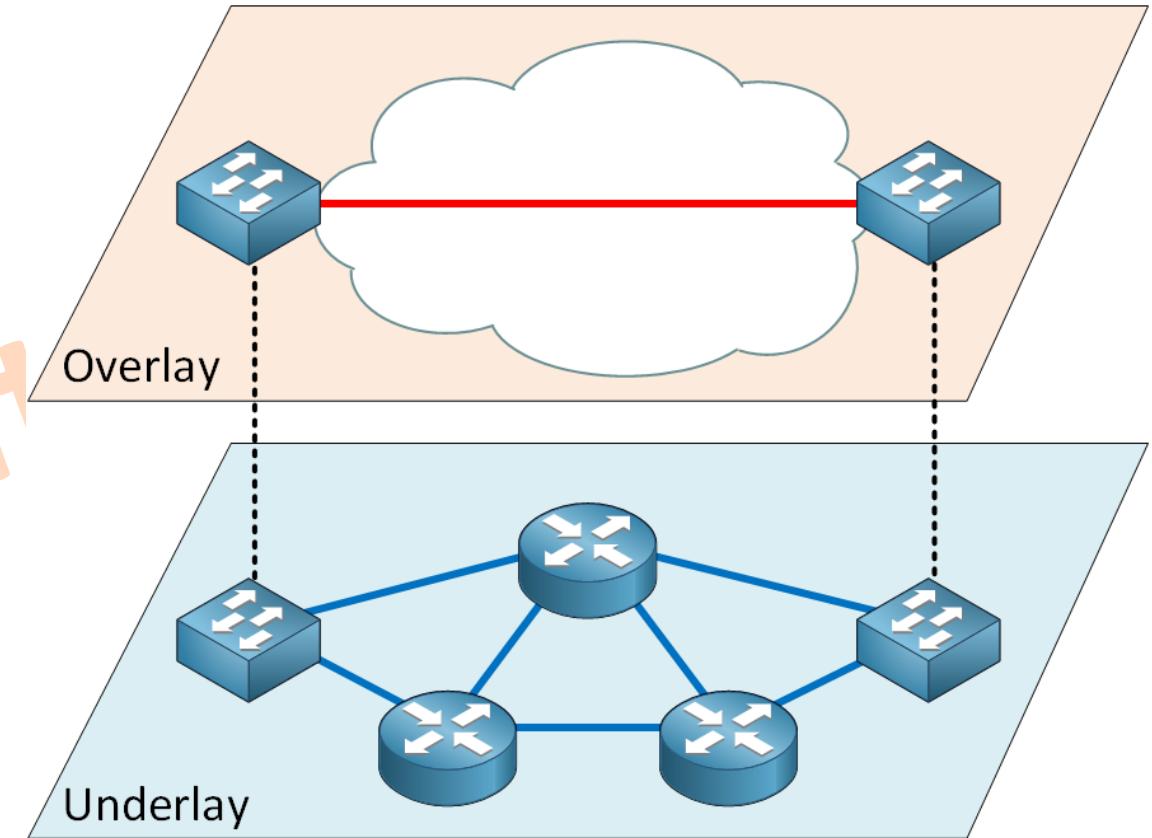


2-Tier Spine-Leaf Architecture



Underlay vs Overlay Network

- **Underlay Network** is physical infrastructure above which overlay network is built.
 - It is the underlying network responsible for delivery of packets across networks.
- **Overlay Network** is a virtual network that is built on top of underlying network infrastructure (Underlay Network).
 - Actually, “Underlay” provides a “service” to the overlay



<https://ipwithease.com/difference-between-underlay-network-and-overlay-network>

Internet service providers (ISPs)



[DOWNLOAD INTERNETMAP](#)

สหปัสดิ์เครือข่ายอินเทอร์เน็ตประจำเดือน
มีนาคม-2024 วันที่ 31 พฤษภาคม 2567

THAILAND INTERNET EXCHANGE

 บริษัท โทรคมนาคมแห่งชาติ จำกัด (มหาชน): CAT	 บริษัท โทรคมนาคมแห่งชาติ จำกัด (มหาชน): TOT	 บริษัท ทรู อินเทอร์เน็ต คอร์ปอเรชั่น จำกัด	 บริษัท ซี เอส ล็อกชันโน้ฟ จำกัด (มหาชน)	 บริษัท ยูไนเต็ด อินฟอร์เมชั่น ไซเบอร์ จำกัด
 กลุ่มบริษัท ทีซีซี เทคโนโลยี (TCCT)	 บริษัท ทรีบีลท์ บروعด์แบนด์ จำกัด (มหาชน)	 บริษัท ชิมไฟฟ์ คอมมูนิเคชั่น จำกัด (มหาชน)	 บริษัท แอดวานซ์ไวร์เลสเน็ทเวอร์ค จำกัด	 บริษัท จัสเทล เน็ทเวิร์ค จำกัด
 บริษัท มีเดนิกซ์ จำกัด	 บริษัท โทรคมนาคม-เทคโนโลยี (DTAC)	 บริษัท อินเตอร์เน็ตแน็ป เกทเวย์ จำกัด		

[ดูข้อมูล จาก กทช. ทั้งหมด อ้างอิงจาก ส่านักงานกิจการโทรคมนาคมแห่งชาติ](#)

-----โปรดเลือกผู้ให้บริการอินเทอร์เน็ต (ขณะนี้มีทั้งหมด : 31 ISPs) -----

<http://internet.nectec.or.th>

Panthakit Totid

Panthakit Totid



Basic Network For Trainee

Chapter 2

The OSI and TCP/IP Reference Model

Pantfekit Totid



ISO OSI Reference Model

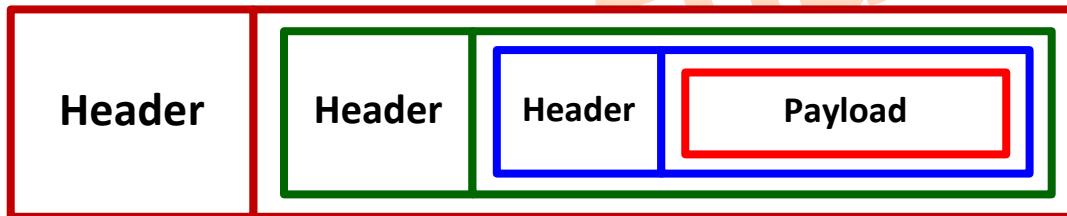
- Back in 1977, the **International Organization for Standardization (ISO)** developed a subcommittee to focus on the interoperability of multivendor communications systems.
 - It's a worldwide agreement on international standards
- An ISO standard that covers all aspects of network communications is the **Open Systems Interconnection (OSI) reference model** (also referred to as the **OSI model** or the **OSI stack**).
 - An **open system** is a set of protocols that allows any two different systems to communicate regardless of their underlying architecture.
 - The purpose of the OSI model is to show **how to facilitate communication between different systems** without requiring changes to the logic underlying hardware and software.



International
Organization for
Standardization

ISO OSI Reference Model

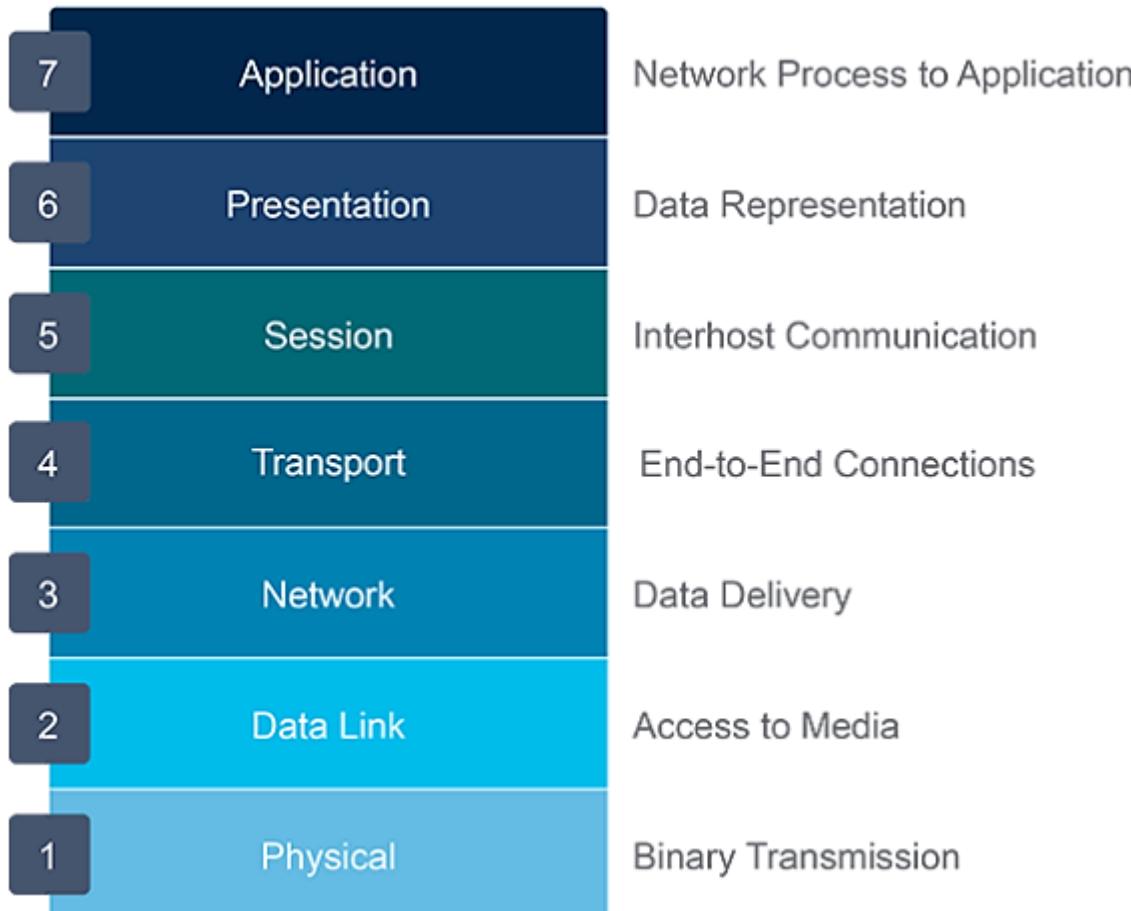
- Basically, a **reference model** is a conceptual blueprint of how communications should take place.
 - It addresses all the processes required for effective communication and divides these processes into logical groupings called **layers**.
 - When a communication system is designed in this manner, it's known as **layered architecture**.
- Each layer in reference model defines **structure of package (Header + Payload)**



*Networks use multiple protocols.
The packet from one protocol can be wrapped within the packet from another (encapsulation).*



The OSI Model

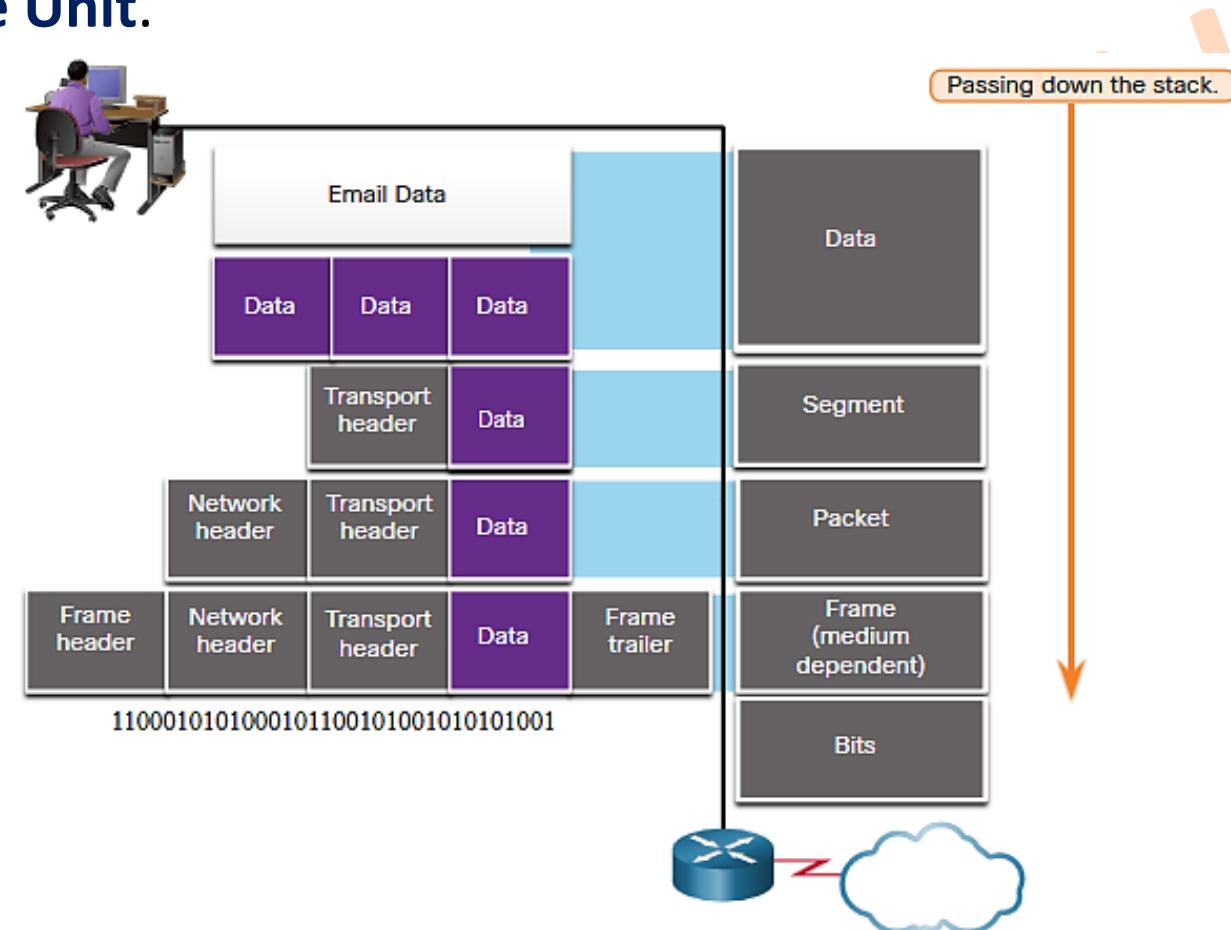


Roughly, the model layers can be grouped into upper and lower layers.

- **Layer 5 to 7**, or **upper layers**, are concerned with user interaction and the information that is communicated, its presentation, and how the communication proceeds.
- **Layer 1 to 4**, the **lower layers**, are concerned with how this content is transferred over the network

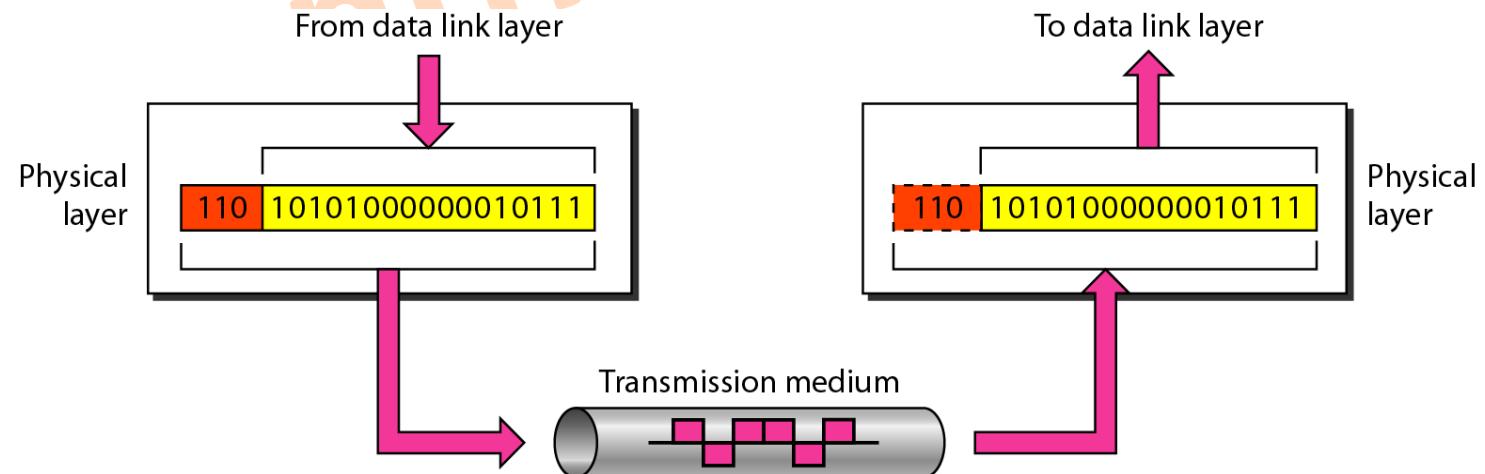
The OSI Model

- Engineers tend to use the term **packet** generically to refer to these **PDUs (Protocol Data Unit)** or **Data Service Unit**.



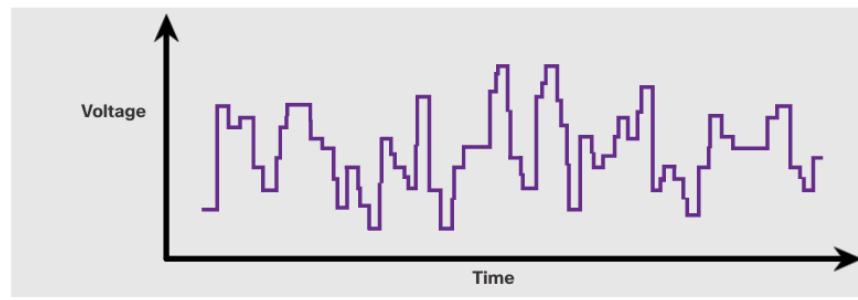
Layer 1: The Physical Layer

- The physical layer defines **electrical, mechanical, procedural, and functional specifications** for activating maintaining, and deactivating the physical link between devices.
- This layer deals with the electromagnetic **representation of bits of data** and their transmission.
- The physical layer specifications define **line encoding, voltage levels, the timing of voltage changes, physical data rates, maximum transmission distances, physical connectors, and other attributes**.
- This layer is the only layer implemented solely in hardware.

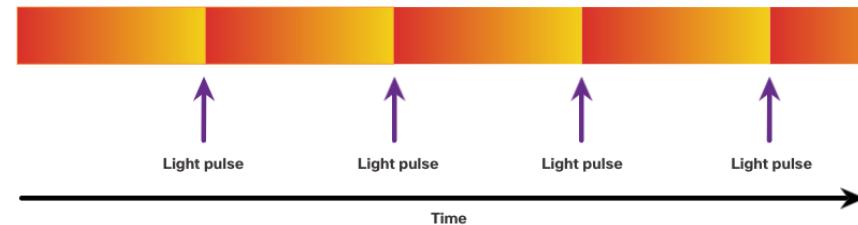


Layer 1: The Physical Layer

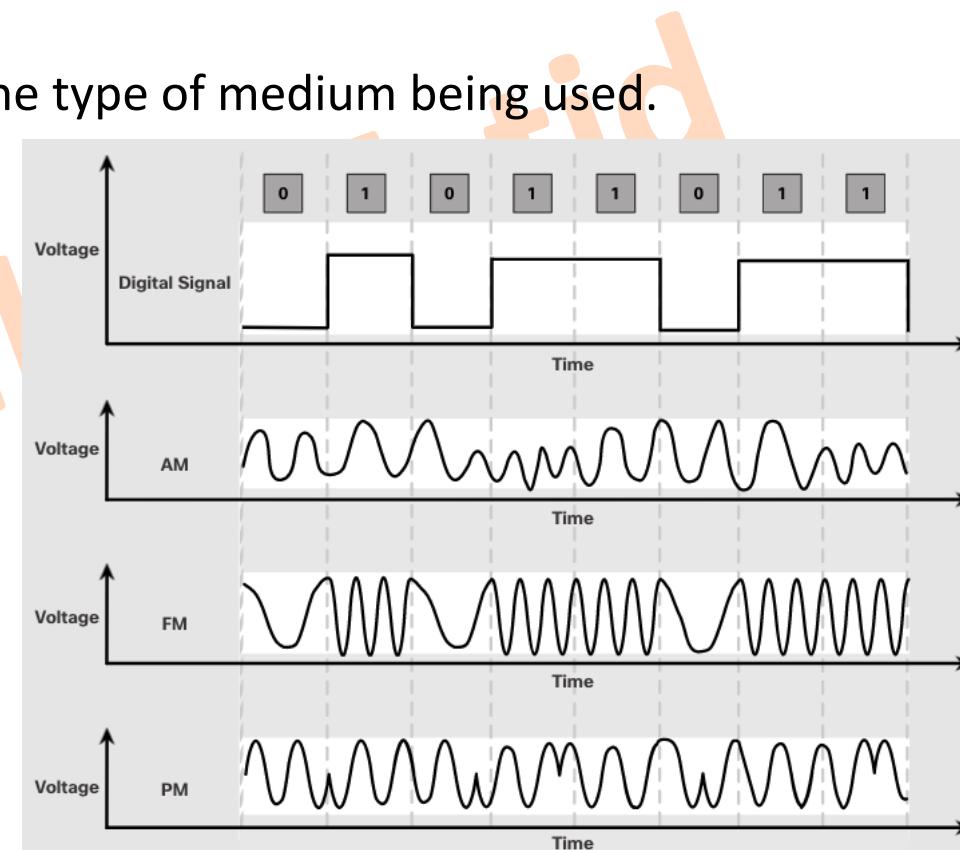
- The **signaling** method is how the bit values, “1” and “0” are represented on the physical medium.
 - The method of signaling will vary based on the type of medium being used.



Electrical Signals Over Copper Cable



Light Pulses Over Fiber-Optic Cable



Microwave Signals Over Wireless

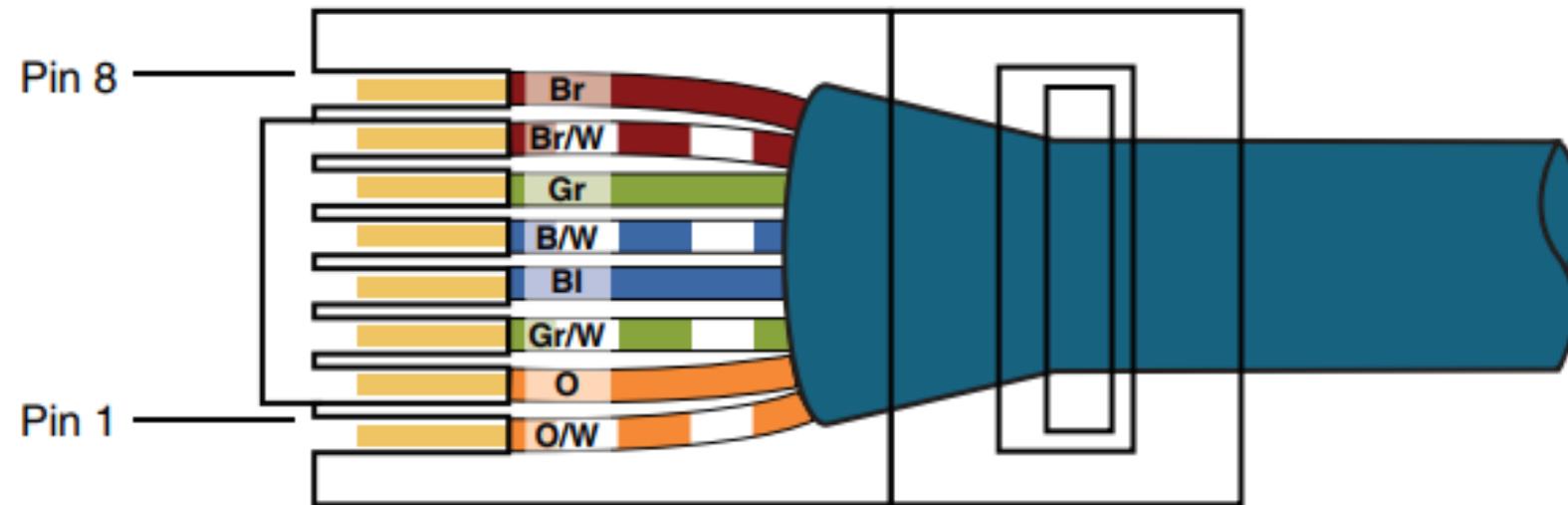
Layer 1: The Physical Layer

- **Bandwidth** is the capacity at which a medium can carry data.
- Digital bandwidth measures the **amount of data that can flow** from one place to another in a given amount of time; how many bits can be transmitted in a second.
- **Physical media properties, current technologies**, and the **laws of physics** play a role in determining available bandwidth.

Unit of Bandwidth	Abbreviation	Equivalence
Bits per second	bps	1 bps = fundamental unit of bandwidth
Kilobits per second	Kbps	1 Kbps = 1,000 bps = 10^3 bps
Megabits per second	Mbps	1 Mbps = 1,000,000 bps = 10^6 bps
Gigabits per second	Gbps	1 Gbps – 1,000,000,000 bps = 10^9 bps
Terabits per second	Tbps	1 Tbps = 1,000,000,000,000 bps = 10^{12} bps

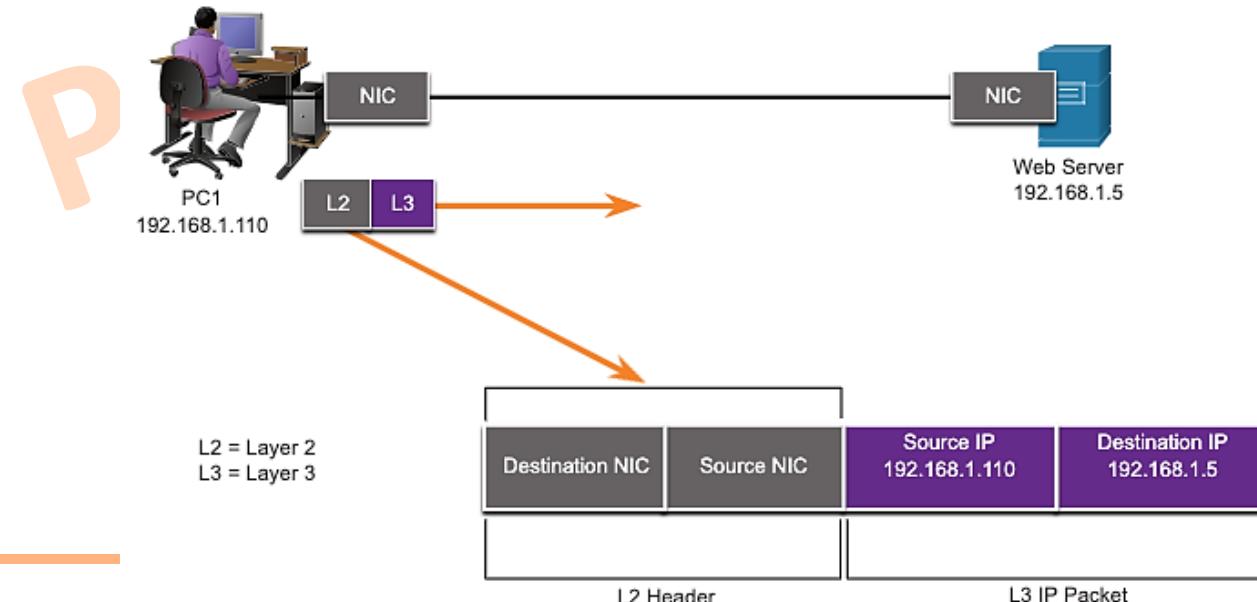
Layer 1: The Physical Layer

- **Wiring standards for connectors and jacks:** Defines the characteristics of the interface between the devices and the transmission medium (type of transmission medium also).
 - For example, the **TIA/EIA-568-B** standard describes how to wire an RJ-45 connector for use on a 100BASE-TX Ethernet network



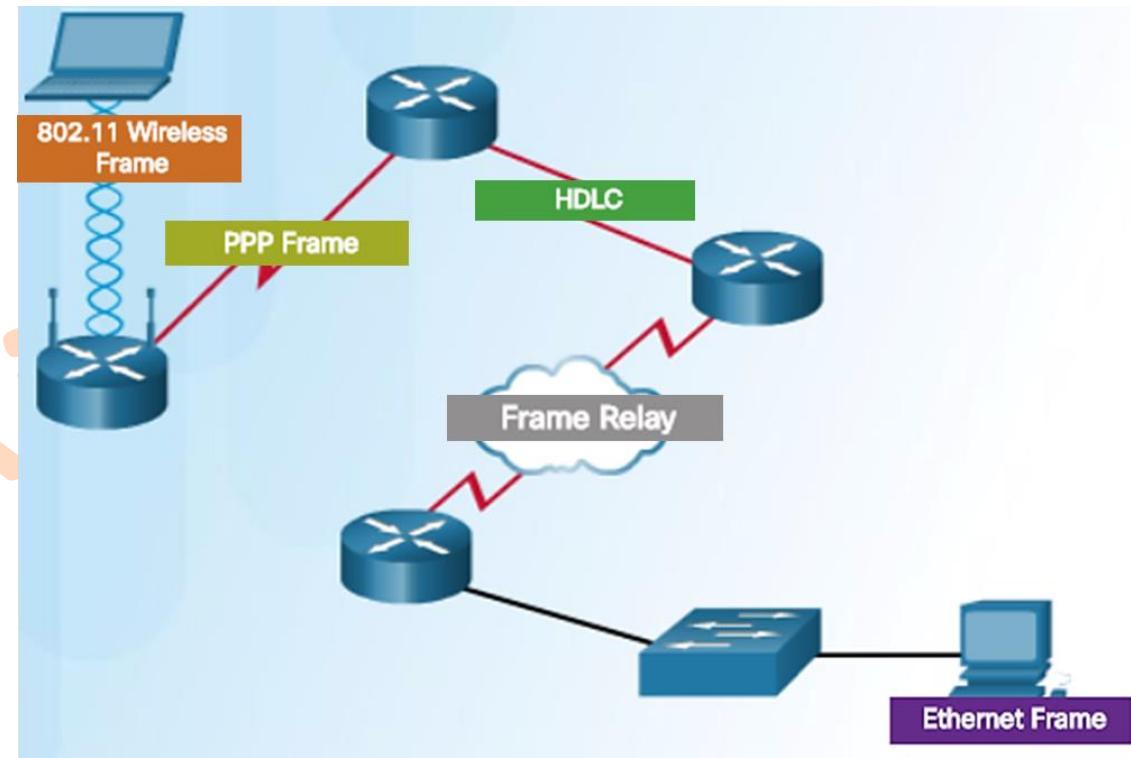
Layer 2: The Data Link Layer

- The data link layer defines **how data is formatted** for transmission and **controlled access** to physical media. This layer typically includes **error detection and correction** to ensure reliable data delivery.
- The data link layer involves **network interface** controller to network interface controller (NIC-to-NIC) communication within the same network or subnet.
- The Data Link layer is responsible for **communications between end-device network interface cards**.



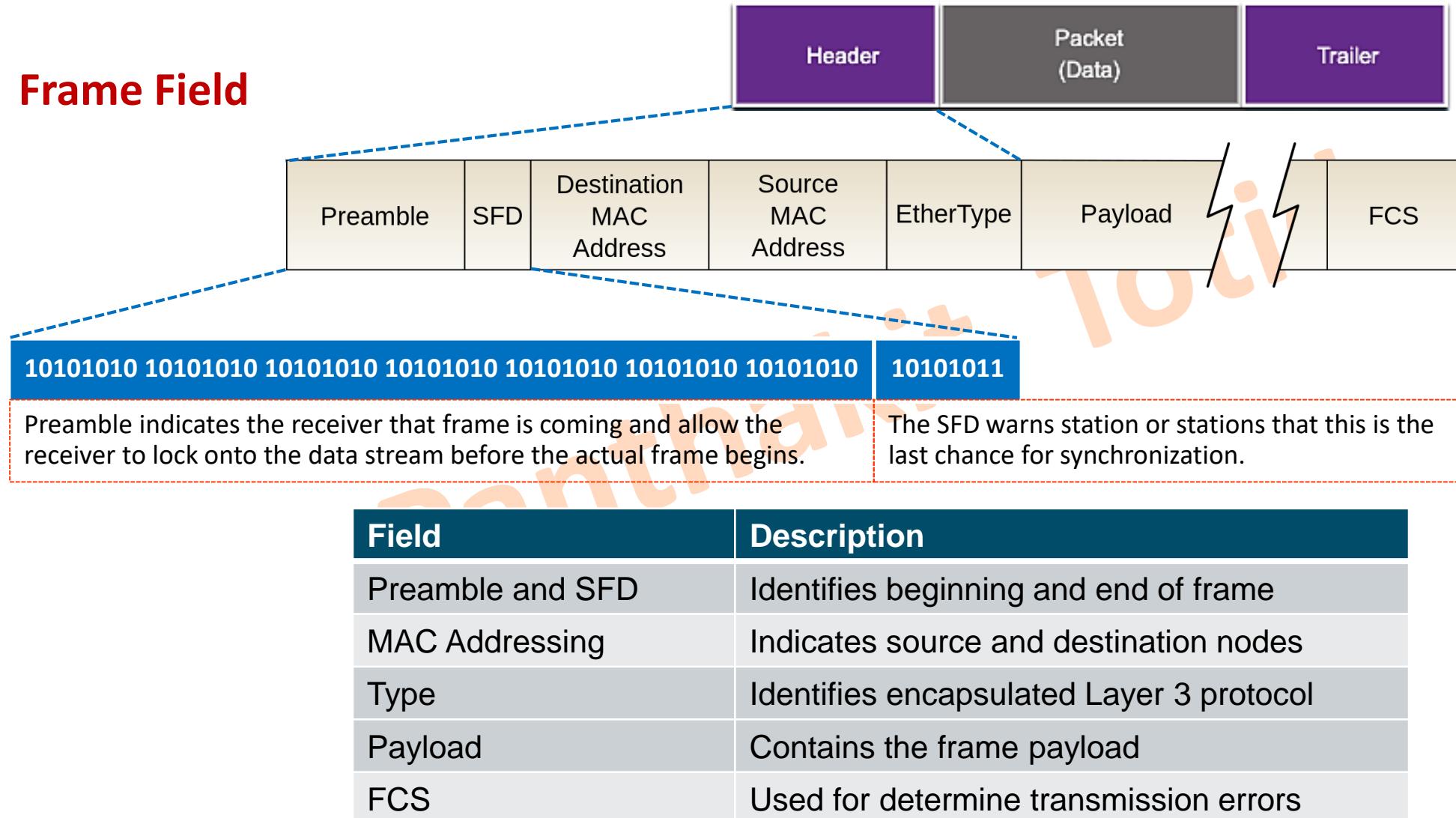
Layer 2: The Data Link Layer

- Data is encapsulated by the data link layer with a header and a trailer to form a **frame**.
- Layer 2 protocol used for a topology is determined by the technology.
- A data link frame has three parts:
 - **Header**
 - **Data**
 - **Trailer**
- The fields of the header and trailer vary according to data link layer protocol.



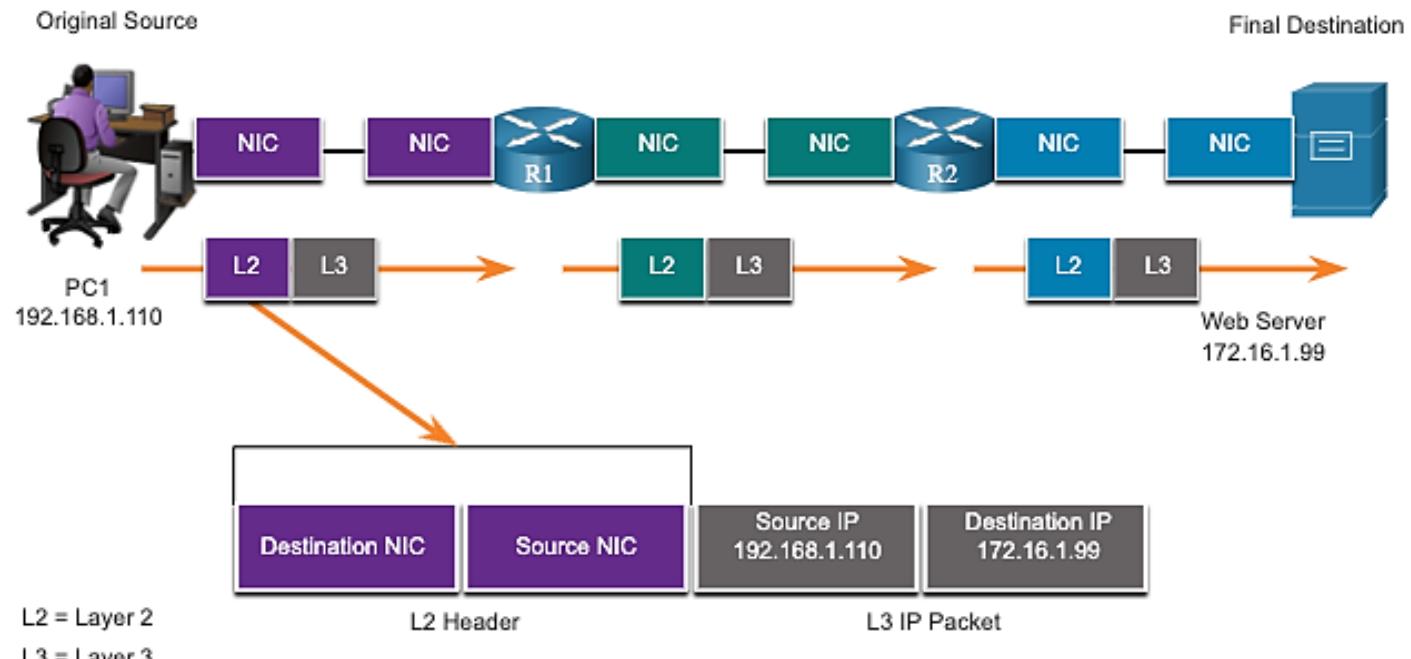
Layer 2: The Data Link Layer

- **Frame Field**



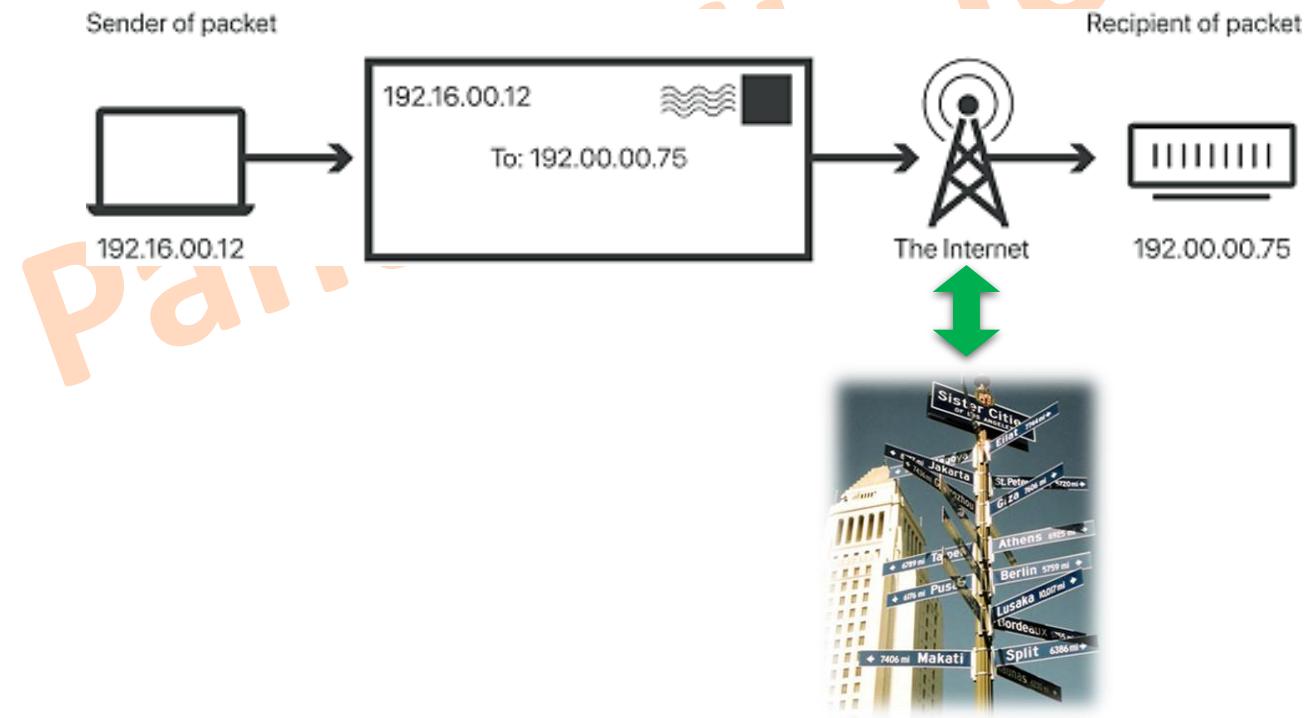
Layer 2: The Data Link Layer

- Layer 2 Addressing: **MAC address**
 - Also referred to as a **physical address**.
 - Contained in the frame header.
 - Used only for **local delivery** of a frame on the link.

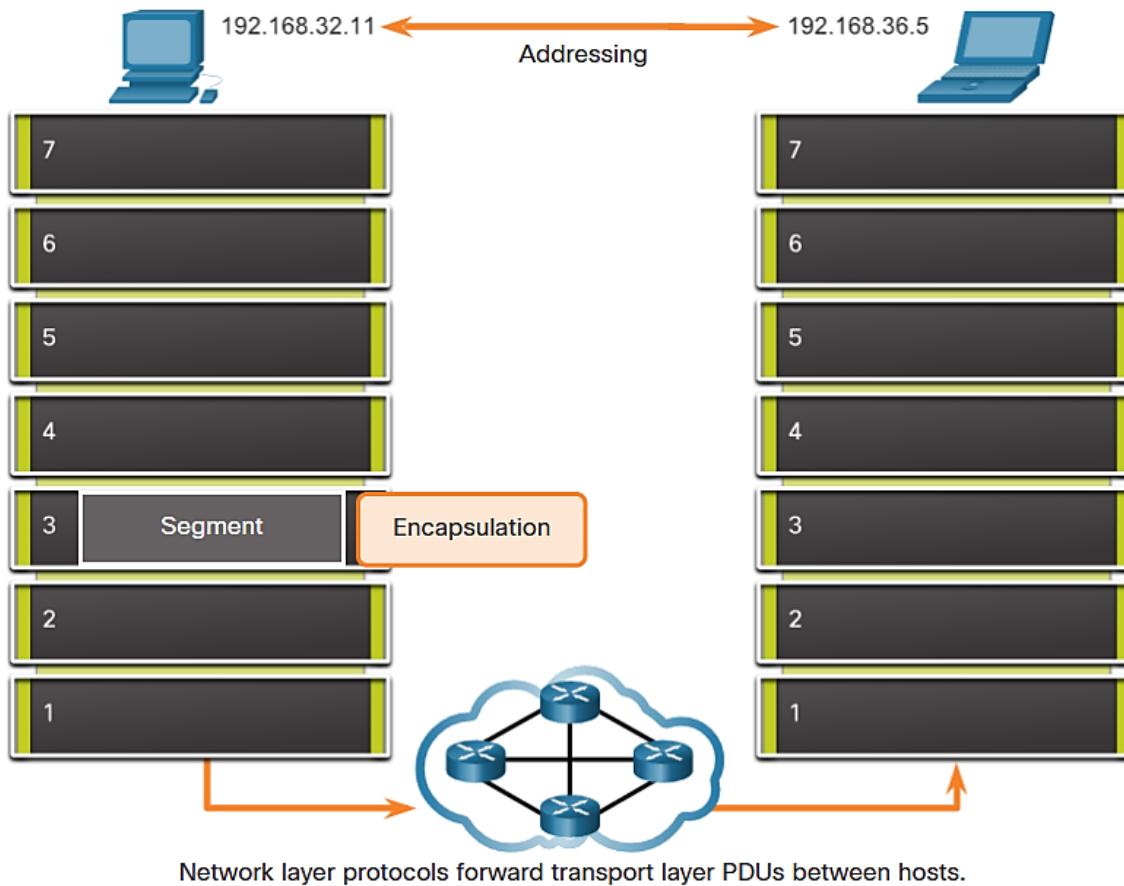


Layer 3: The Network Layer

- The network layer provides **connectivity and path selection** beyond the local segment, all the way from the source to the final destination.
- The network layer uses **logical addressing** to manage connectivity. In networking, the logical address is used to identify the sender and the recipient.



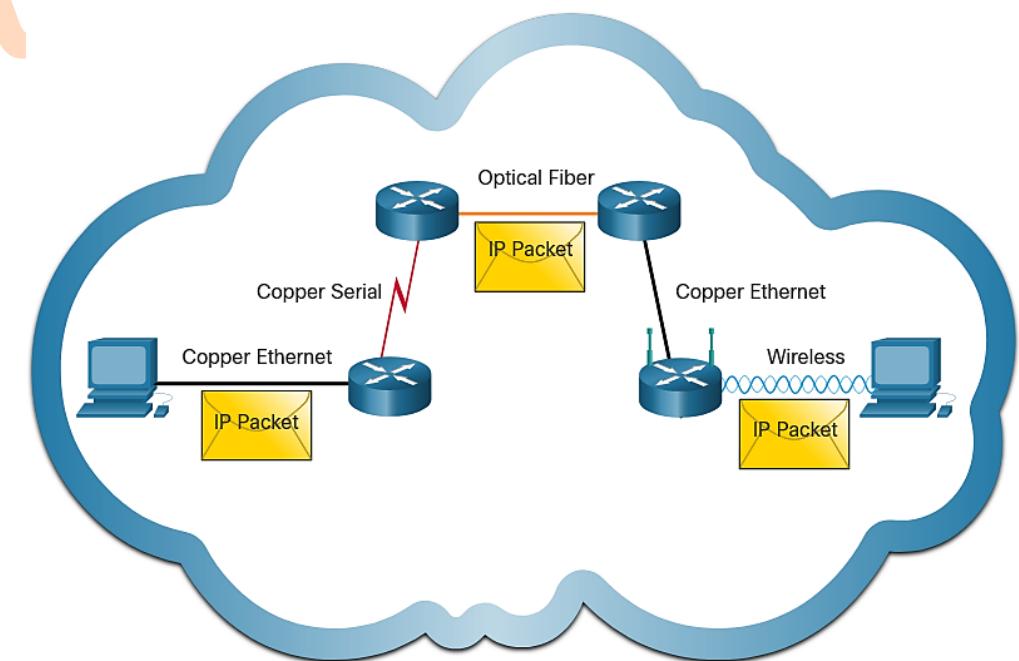
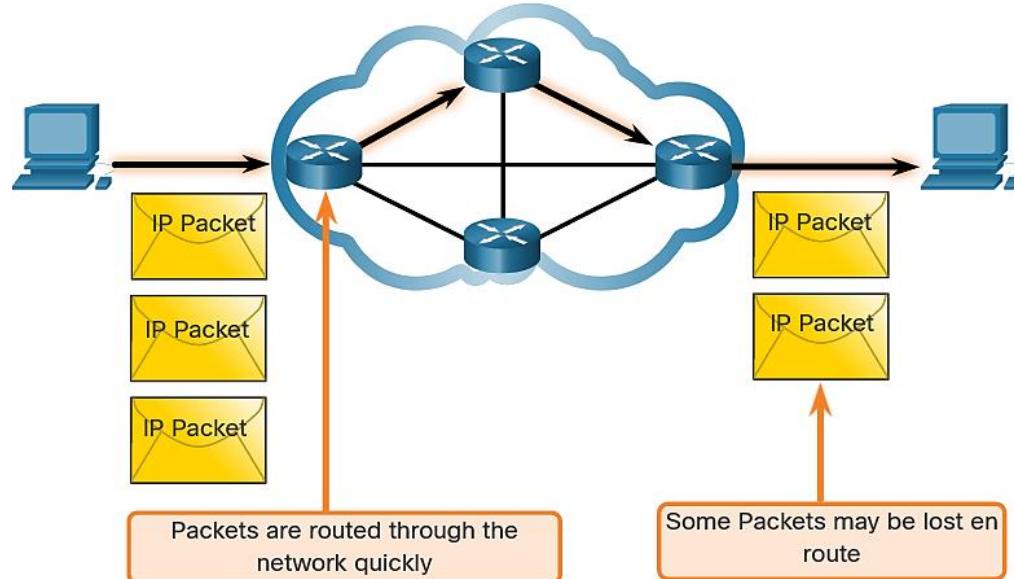
Layer 3: The Network Layer



- Provides services to allow **end devices to exchange data across a network**
- **IP version 4 (IPv4)** and **IP version 6 (IPv6)** are the principle network layer communication protocols.
- The network layer performs four basic operations:
 - **Addressing end devices**
 - **Encapsulation**
 - **Routing**
 - **De-encapsulation**

Layer 3: The Network Layer

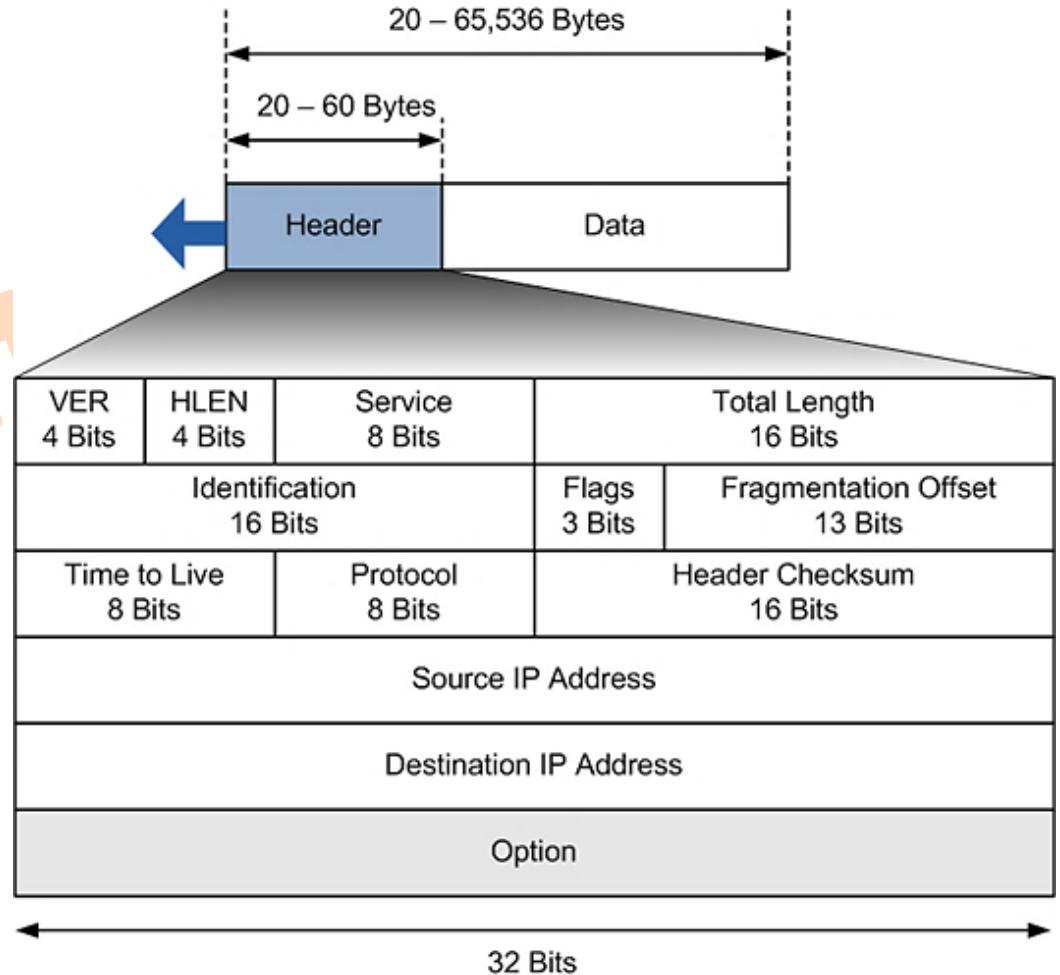
- IP is meant to have low overhead and may be described as:
 - Connectionless**
 - Best Effort**
 - Media Independent**



Layer 3: The Network Layer

IPv4 Packet Header Fields

- Packets used by the IP are called **datagrams**
- A datagram is a variable-length packet consisting of two parts: **header** and **payload (data)**.
 - Contains several fields of information
 - Diagram is read from left to right, 4 bytes per line
 - **The two most important fields are the source and destination**



Layer 3: The Network Layer

MAC vs IP

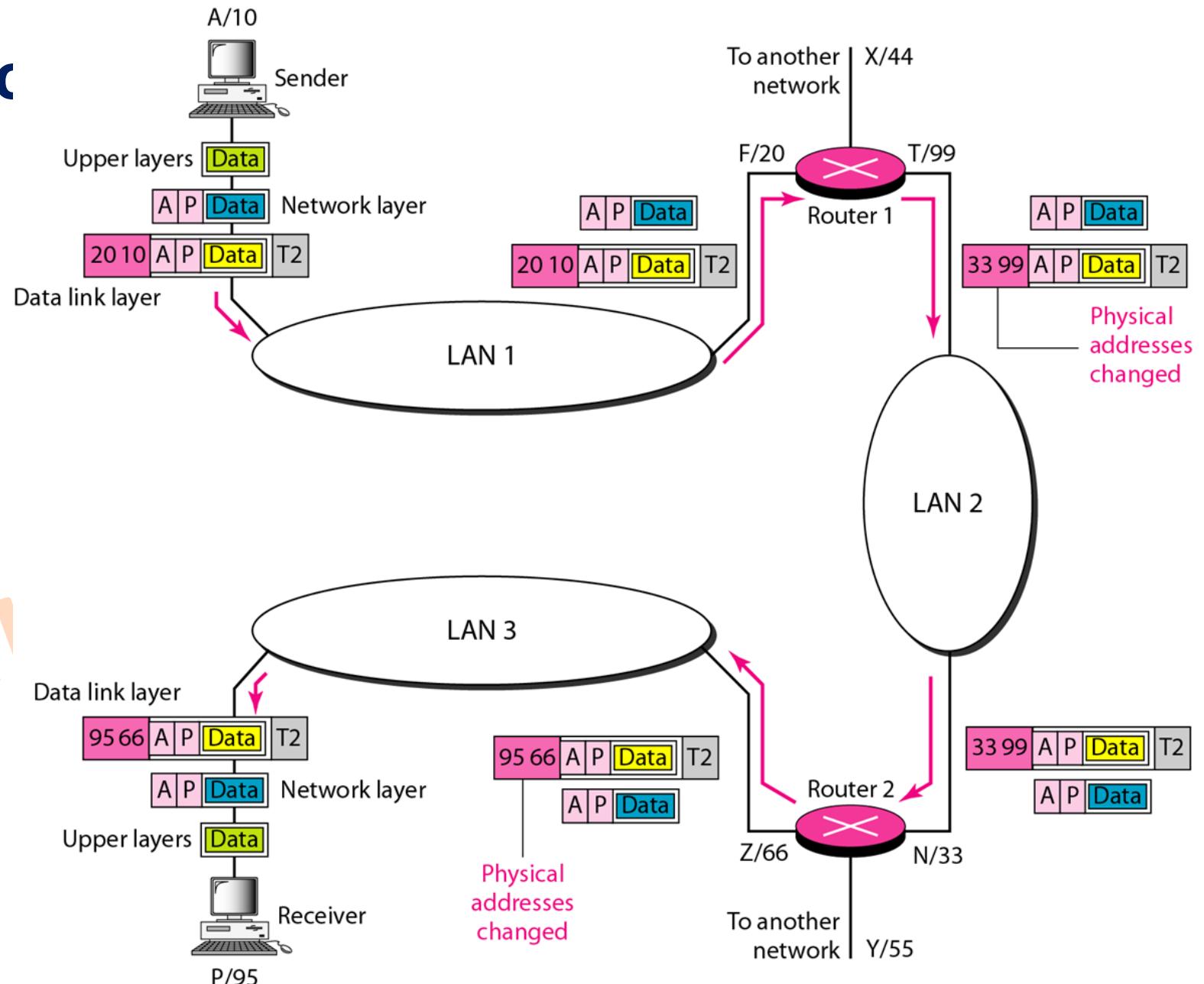
- **MAC Address**

- Physical Address
- [Change]
- Hop-to-Hop

- **IP Address**

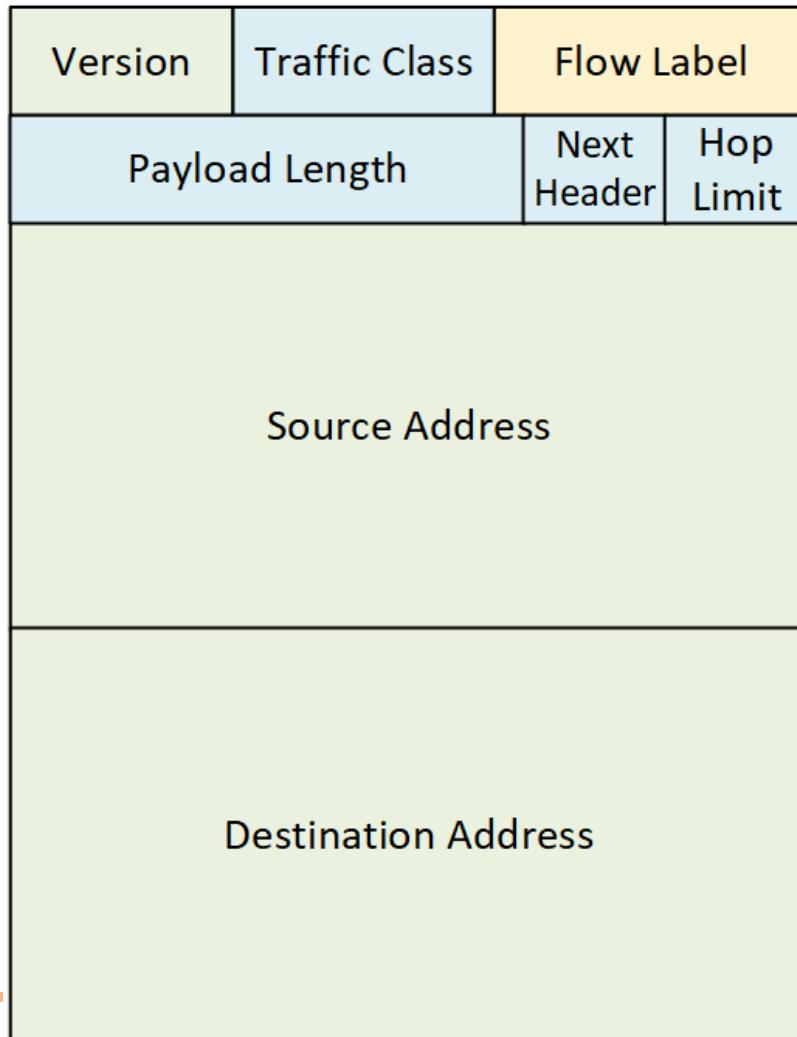
- Logical Address
- [Unchange]

Source-to-Destination



Layer 3: The Network Layer

IPv6 Header



IPv4 Header

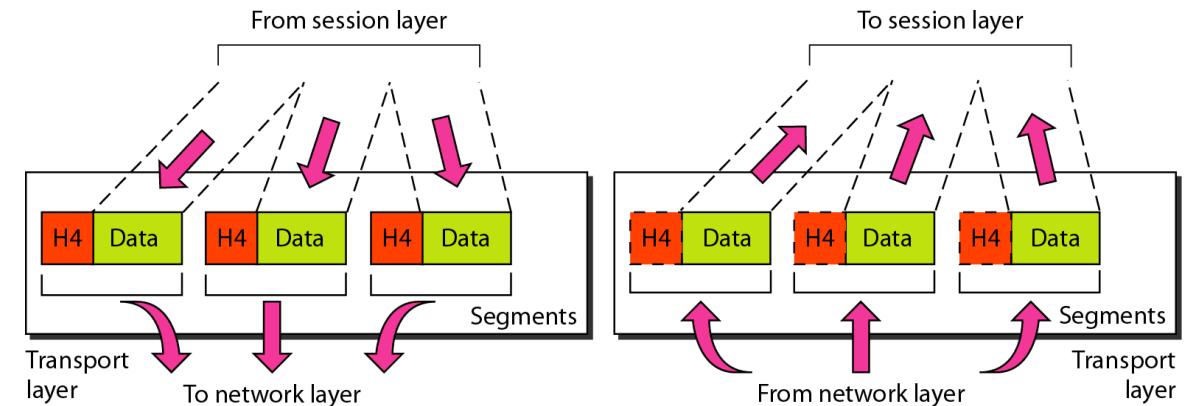
Version	IHL	Type of Service	Total Length			
Identification		Flags	Fragment Offset			
TTL	Protocol	Header Checksum				
Source Address						
Destination Address						
Options		Padding				

Legend

- Fields **kept** in IPv6
- Fields **kept** in IPv6, but name and position changed
- Fields **not kept** in IPv6
- Fields that are **new** in IPv6

Layer 4: The Transport Layer

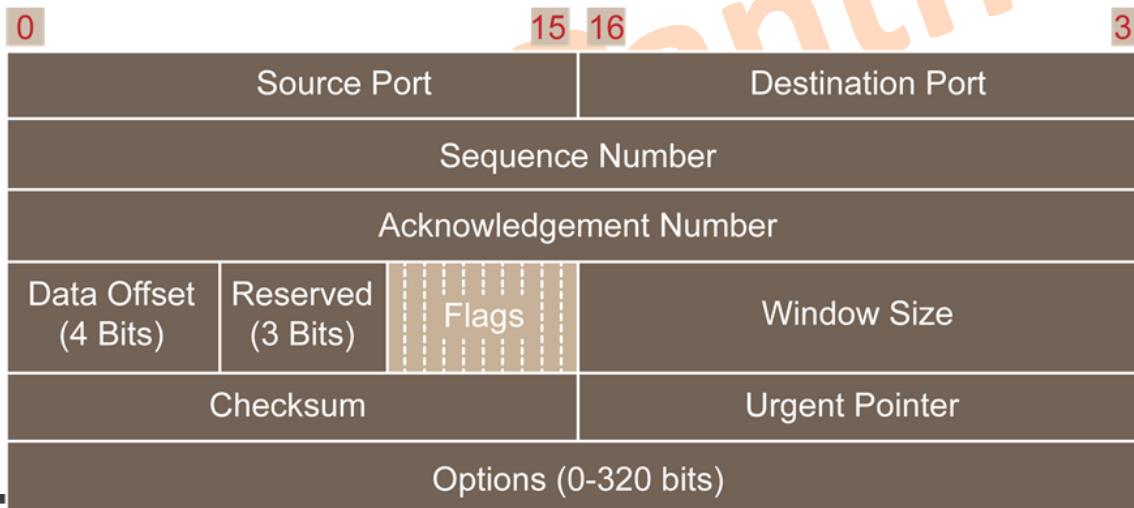
- The transport layer defines **segmenting and reassembling** of data belonging to multiple individual communications, defines the **flow control**, and defines the mechanisms for reliable transport if required.
- The transport layer serves the upper layers, which in turn interface with many user applications. To distinguish between these application processes, the transport layer uses its own addressing. This addressing is valid locally, within one host, unlike addressing at the network layer.
- The transport services can be **reliable** or **unreliable**. The selection of the appropriate service depends on application requirements.



Layer 4: The Transport Layer

Two common transport layer protocols are TCP and UDP:

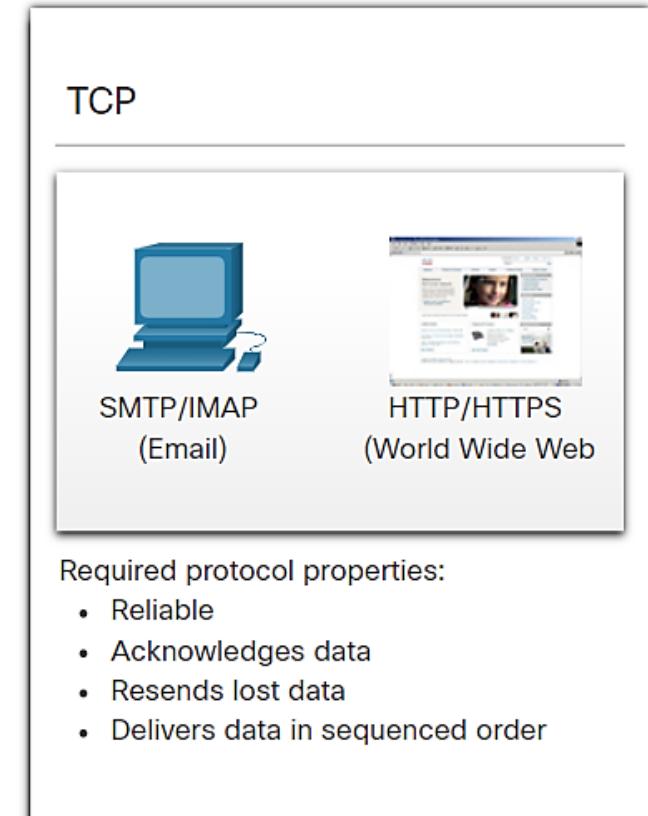
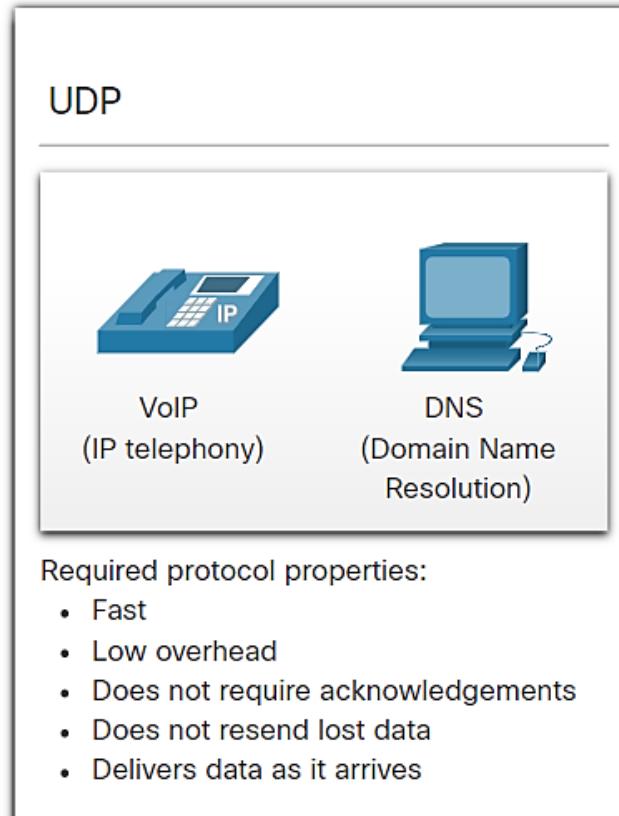
- **Transmission Control Protocol (TCP):** TCP is a **connection-oriented** transport protocol. Connection-oriented transport protocols offer reliable transport, in that if a segment is dropped, the sender can detect the drop and retransmit the dropped segment.
- **User Datagram Protocol (UDP):** UDP is a **connectionless** transport protocol. Connectionless transport protocols offer unreliable transport, in that if a segment is dropped, the sender is unaware of the drop, and no retransmission occurs.



Layer 4: The Transport Layer

The Right Transport Layer Protocol for the Right Application

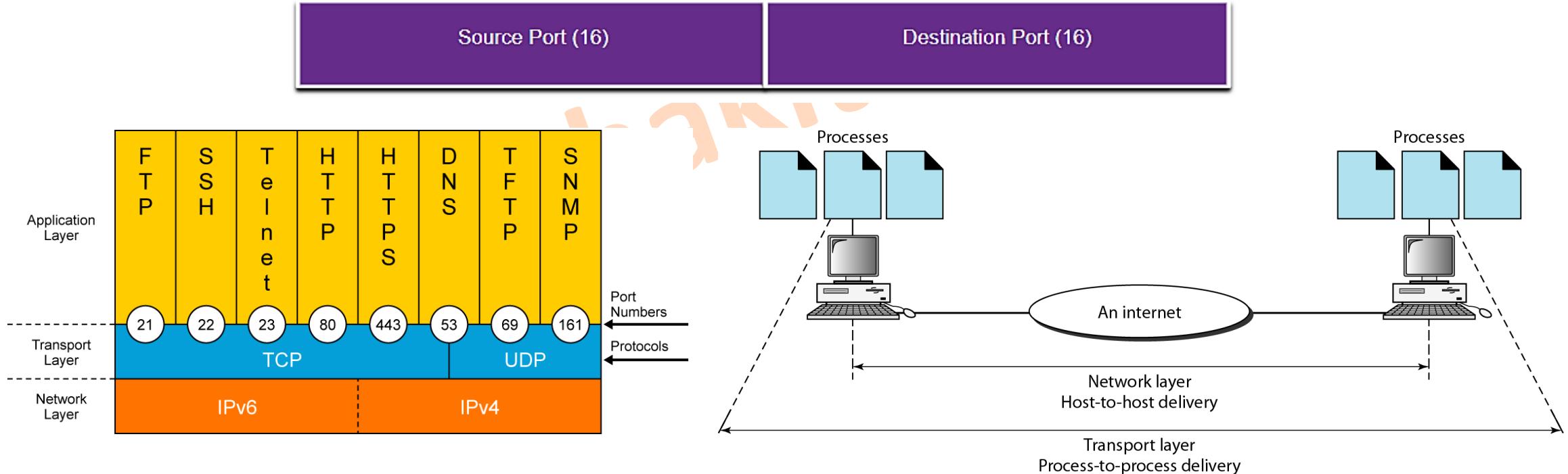
- **TCP** - databases, web browsers, and email clients require that all data that is sent arrives at the destination in its original condition.
- **UDP** - if one or two segments of a live video stream fail to arrive, if disruption in the stream, may not be noticeable to the user.



Layer 4: The Transport Layer

Port Addressing

- TCP and UDP transport layer protocols use port numbers to manage multiple, simultaneous conversations.



Layer 4: The Transport Layer

Port Addressing

Port Number Range	Port Group
0 to 1023	Well-known Ports
1024 to 49151	Registered Ports
49152 to 65535	Private and/or Dynamic Ports

- **Well-known Ports** - These numbers are reserved for services and applications.
- **Registered Ports** - These port numbers are assigned by IANA to a requesting entity to use with specific processes or applications.
- **Dynamic or Private Ports** - Usually assigned dynamically by the client's OS and used to identify the client application during communication.

Layer 4: The Transport Layer

Port Addressing - Well Known Port Numbers

Port Number	Protocol	Application
20	TCP	File Transfer Protocol (FTP) - Data
21	TCP	File Transfer Protocol (FTP) - Control
22	TCP	Secure Shell (SSH)
23	TCP	Telnet
25	TCP	Simple Mail Transfer Protocol (SMTP)
53	UDP, TCP	Domain Name Service (DNS)
67	UDP	Dynamic Host Configuration Protocol (DHCP) - Server
68	UDP	Dynamic Host Configuration Protocol - Client
69	UDP	Trivial File Transfer Protocol (TFTP)
80	TCP	Hypertext Transfer Protocol (HTTP)
110	TCP	Post Office Protocol version 3 (POP3)
143	TCP	Internet Message Access Protocol (IMAP)
161	UDP	Simple Network Management Protocol (SNMP)
443	TCP	Hypertext Transfer Protocol Secure (HTTPS)

<http://tcp-udp-ports.com>

Layer 4: The Transport Layer

The netstat Command

- Unexplained TCP connections can pose a major security threat. Netstat is an important tool to verify connections.

C:\> **netstat**

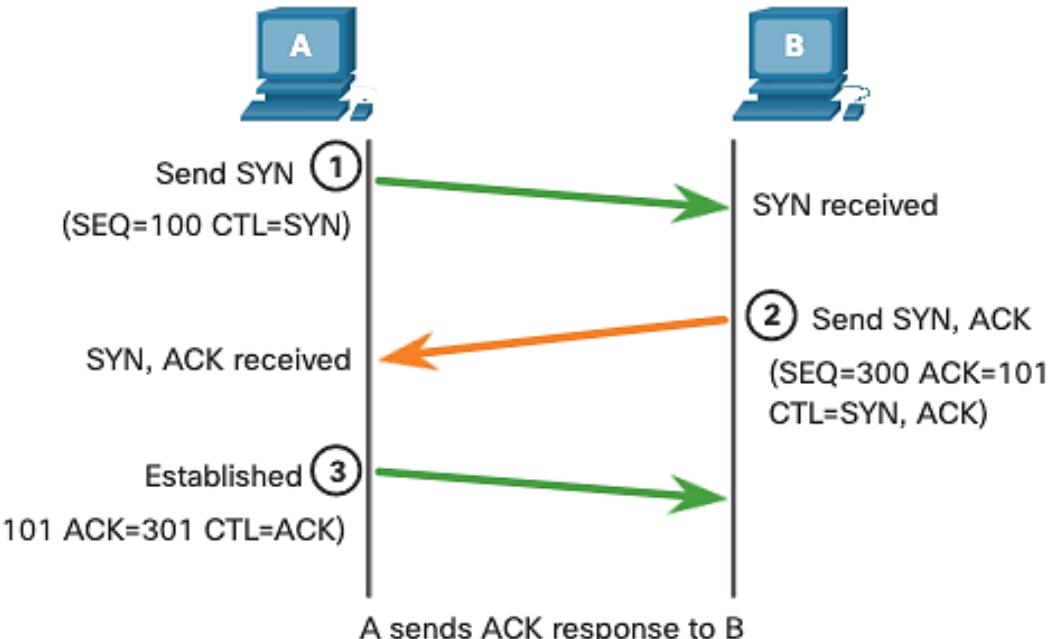
Active Connections

Proto	Local Address	Foreign Address	State
TCP	192.168.1.124:3126	192.168.0.2:netbios-ssn	ESTABLISHED
TCP	192.168.1.124:3158	207.138.126.152:http	ESTABLISHED
TCP	192.168.1.124:3159	207.138.126.169:http	ESTABLISHED
TCP	192.168.1.124:3160	207.138.126.169:http	ESTABLISHED
TCP	192.168.1.124:3161	sc.msn.com:http	ESTABLISHED
TCP	192.168.1.124:3166	www.cisco.com:http	ESTABLISHED

TCP Communication Process

TCP Connection Establishment

- **Step 1:** The initiating client requests a client-to-server communication session with the server.
- **Step 2:** The server acknowledges the client-to-server communication session and requests a server-to-client communication session.
- **Step 3:** The initiating client acknowledges the server-to-client communication session.

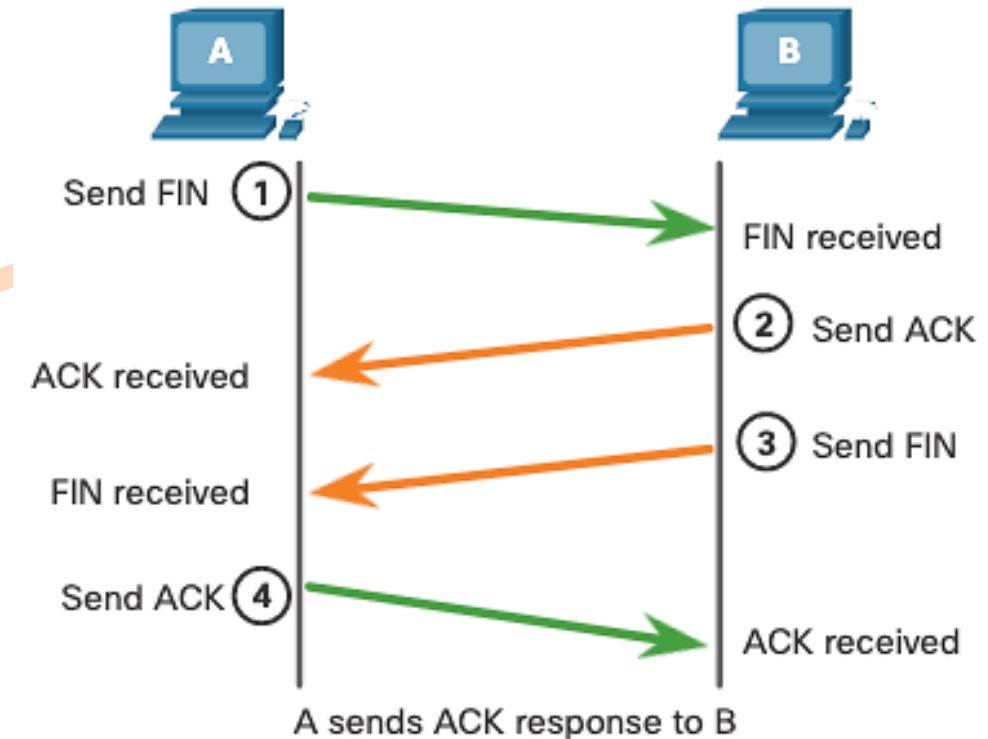


Source	Destination	Protocol	Length	Info
10.10.1.246	23.33.16.170	TCP	66	54779 > https [SYN] Seq=0
23.33.16.170	10.10.1.246	TCP	66	https > 54779 [SYN, ACK] Seq=0 Ack=1
10.10.1.246	23.33.16.170	TCP	54	54779 > https [ACK] Seq=1 Ack=1

TCP Communication Process

Session Termination

- **Step 1:** When the client has no more data to send in the stream, it sends a segment with the FIN flag set.
- **Step 2:** The server sends an ACK to acknowledge the receipt of the FIN to terminate the session from client to server.
- **Step 3:** The server sends a FIN to the client to terminate the server-to-client session.
- **Step 4:** The client responds with an ACK to acknowledge the FIN from the server.



Upper Layers

- The upper layers of the OSI model are less clearly associated with distinct real world protocols. These layers collect various functions that provide useful interfaces between software applications and the transport layer.

- Layer 5—Session**

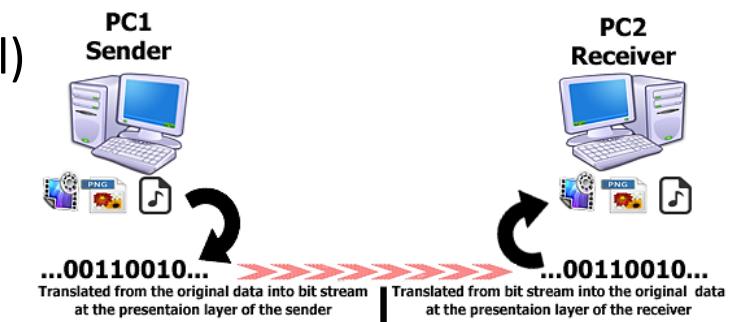
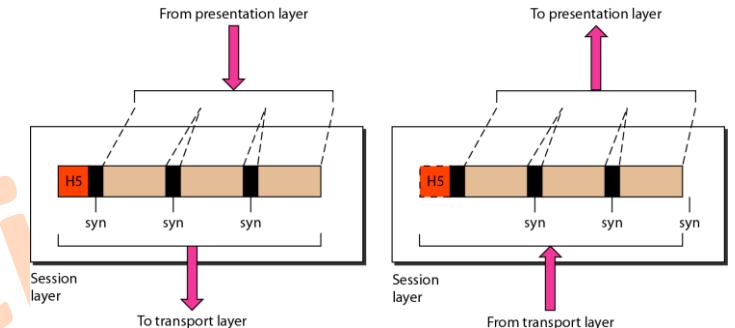
- Establish *rules for exchange* of messages and sequencing (dialog control)

- Layer 6—Presentation**

- Establish *data formats* (such as character sets)

- Layer 7—Application**

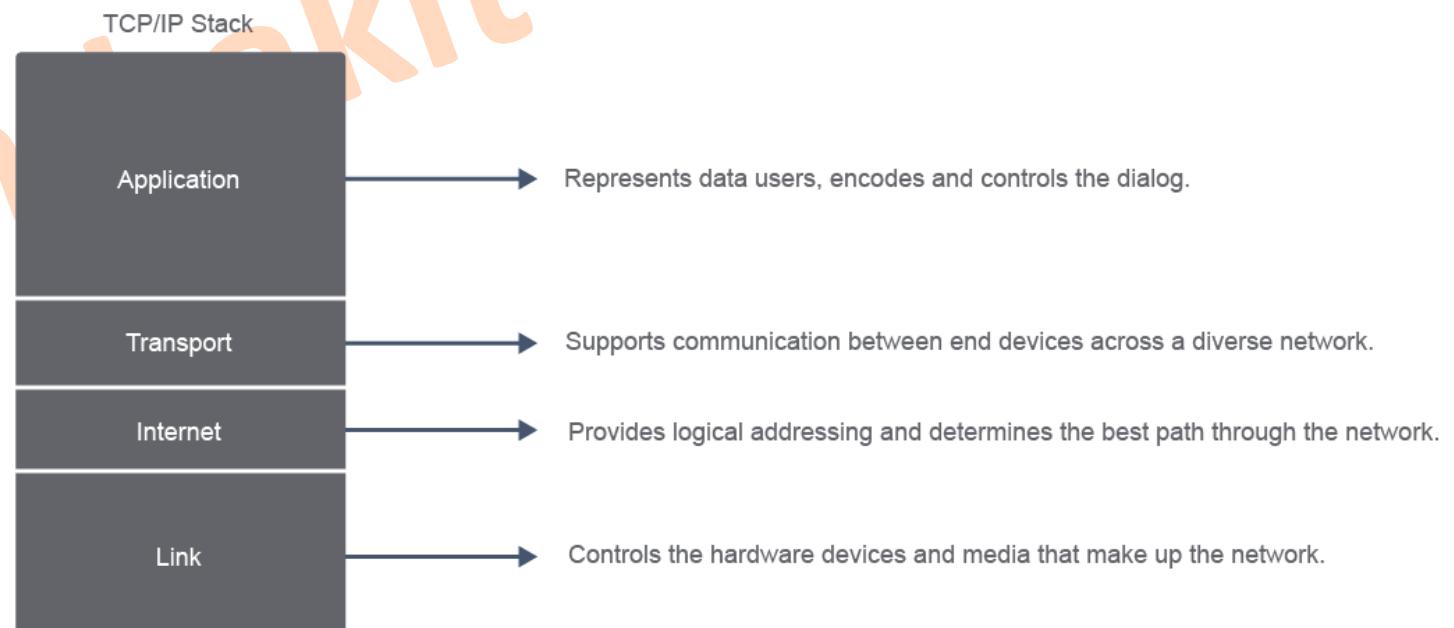
- Present *requests and responses* from server or client software with structured headers and data payload



TCP/IP Protocol Suite

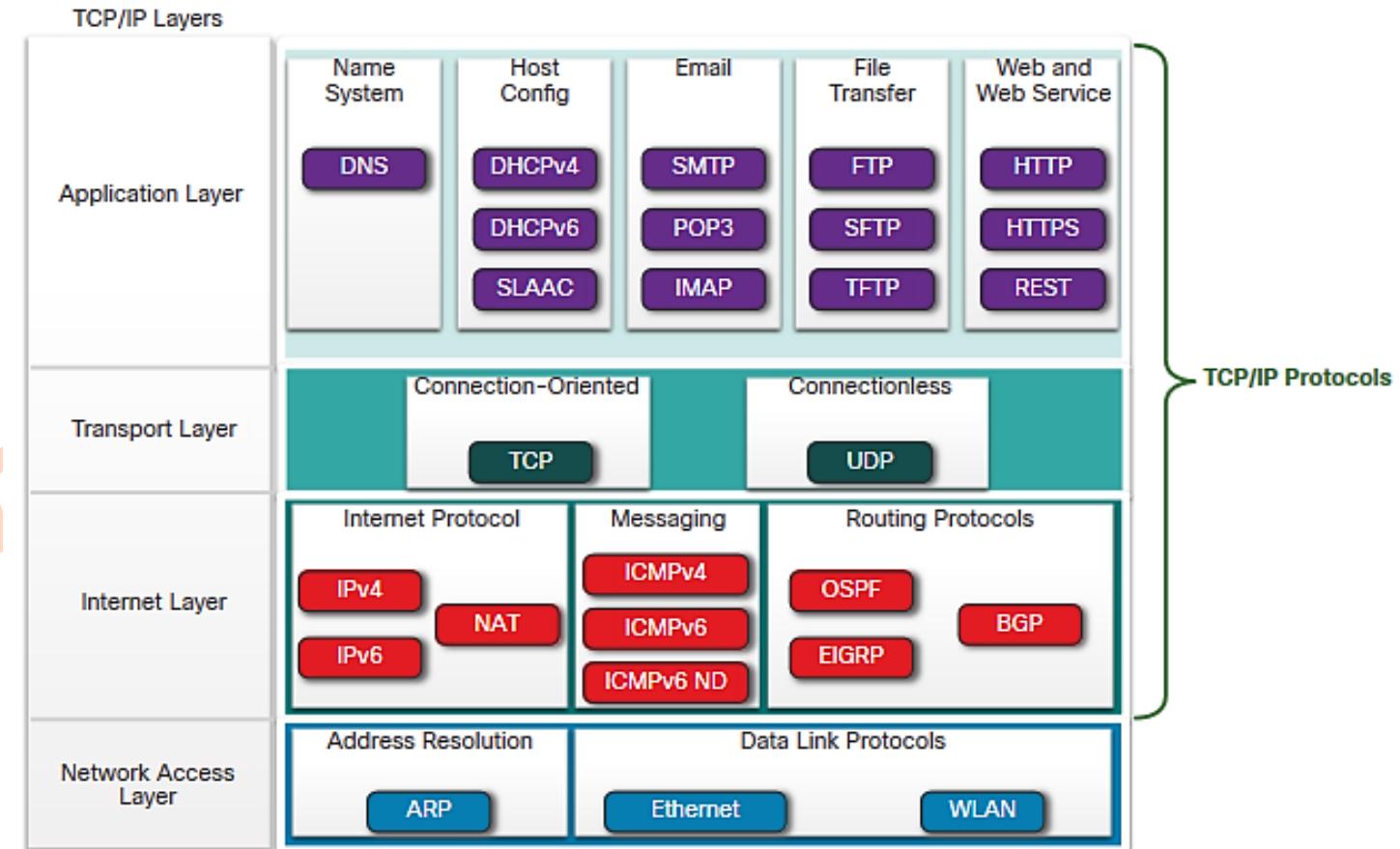
- The TCP/IP model represents a protocol suite. It is similar to the ISO OSI model in that it uses layers to organize protocols and explain which functions they perform. TCP/IP protocols are **actively used** in actual networks today.
- The TCP/IP model defines and describes requirements for the implementation of host systems. These include **standard protocols** that these systems should use.

Although this course refers to the TCP/IP protocol stack or protocol suite, it is common in the industry to shorten this term to "IP stack."

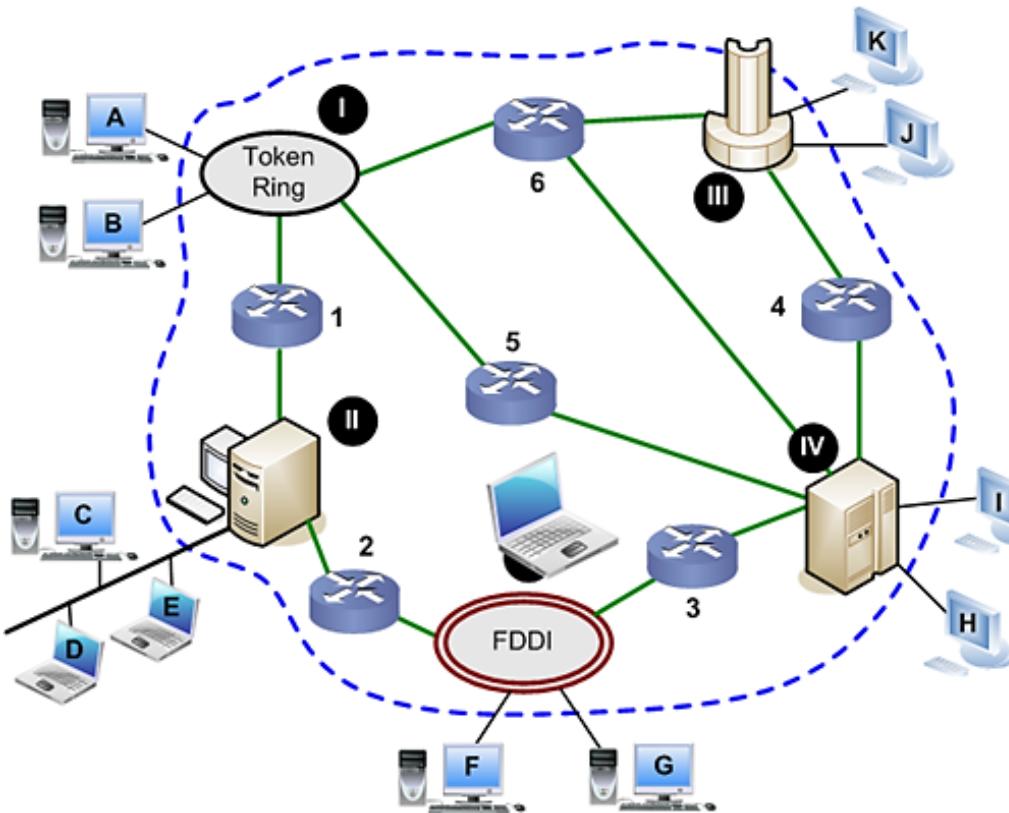


TCP/IP Protocol Suite

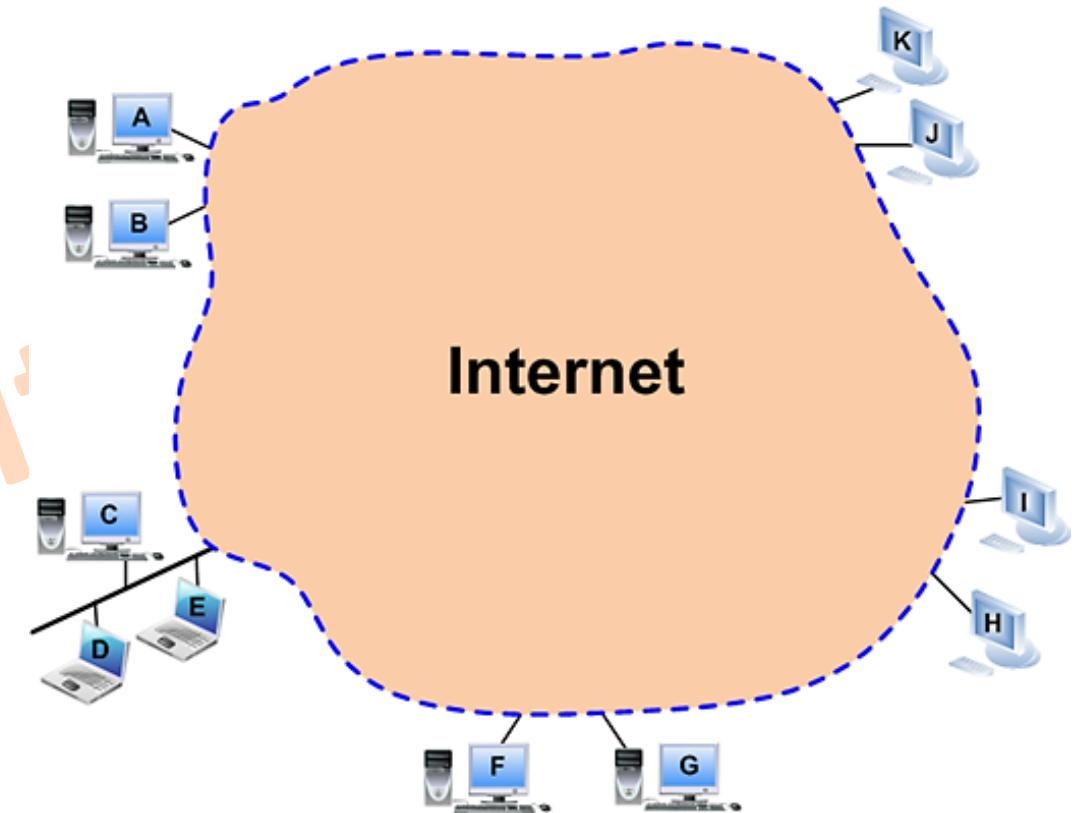
- TCP/IP is the protocol suite used by the internet and includes many protocols.
- TCP/IP is:
 - An **open standard protocol suite** that is freely available to the public and can be used by **any vendor**
 - A standards-based protocol suite that is endorsed by the networking industry and approved by a standards organization to ensure **interoperability**



The Internet with TCP/IP



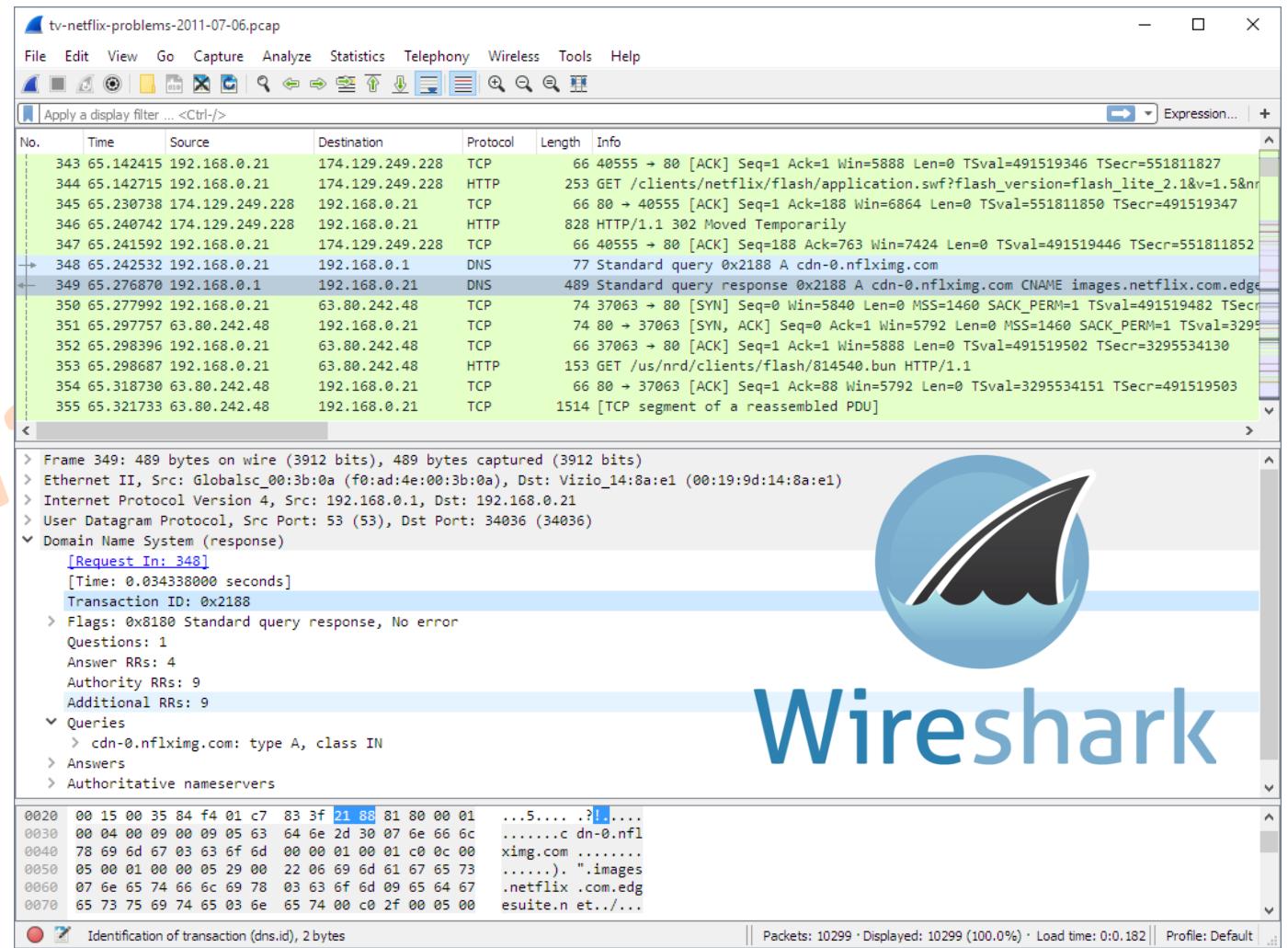
The Internet in physical networks aspect



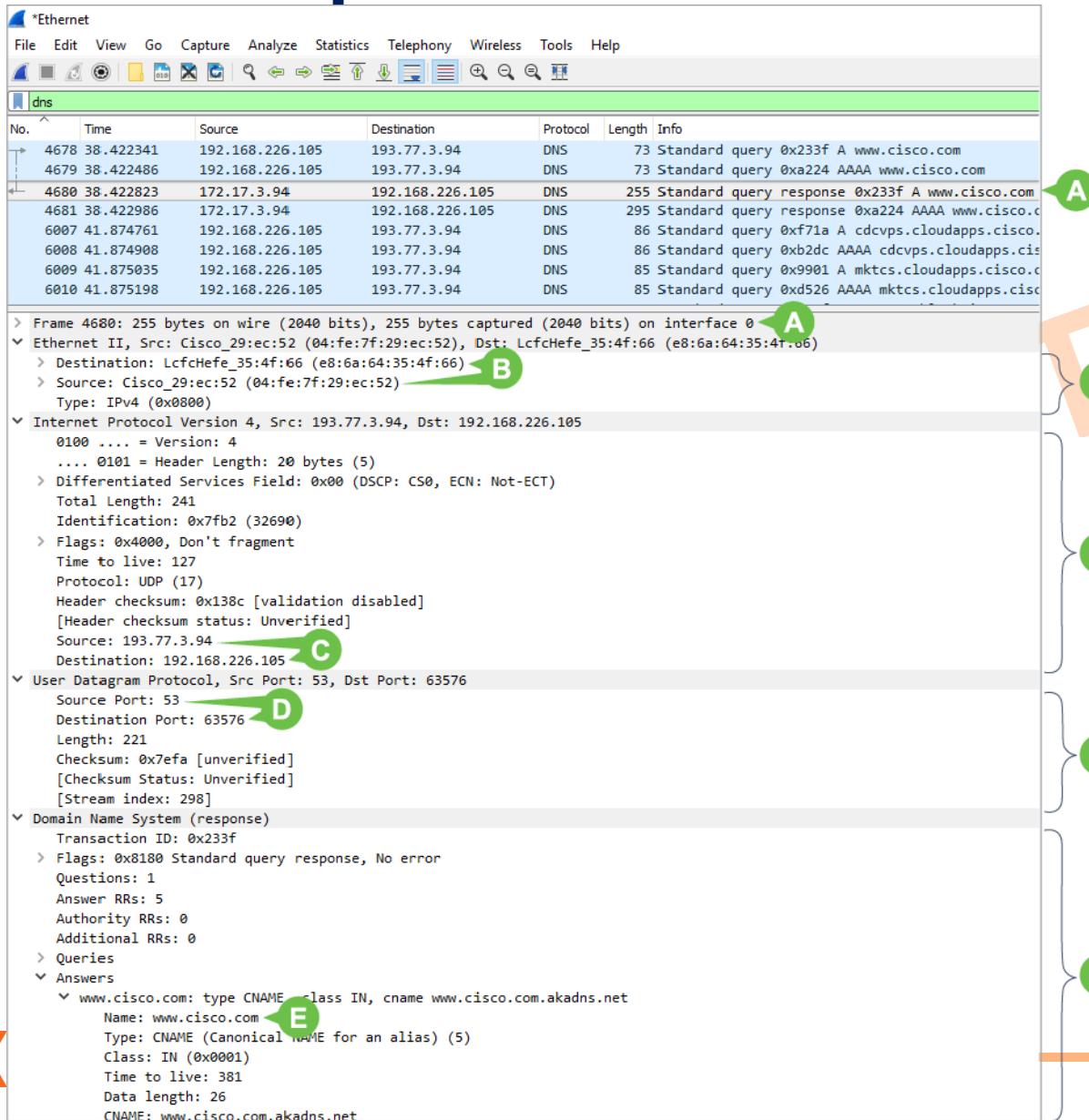
The Internet in TCP/IP aspect
(one logical network)

TCP/IP PDU Deep Dive

- To look into PDUs, you can use a **packet analyzer**, such as [Wireshark](#), which is a free and open-source packet analyzer.
- Packet analyzers capture all the PDUs on a selected interface. They then examine their content, interpret it and display it in text or using a graphical interface.
- Packet analyzers, sometimes also called **sniffers**, are used for network troubleshooting, analysis, software and communications protocol development, and education.



TCP/IP PDU Deep Dive



- A** The 4680th Frame Captured Carries Application Data.
- B** Link Layer Addresses
- C** Internet Layer Addresses
- D** Transport Layer Applications' Identifications
- E** User's Input
www.cisco.com
- F** Data Added by Link Layer Protocol to Form a Frame
- G** Data Added by Internet Layer Protocol to Form a Packet
- H** Data Added by Transport Layer Protocol to Form a Segment
- I** Application Protocol Data



Basic Network For Trainee

Module 3

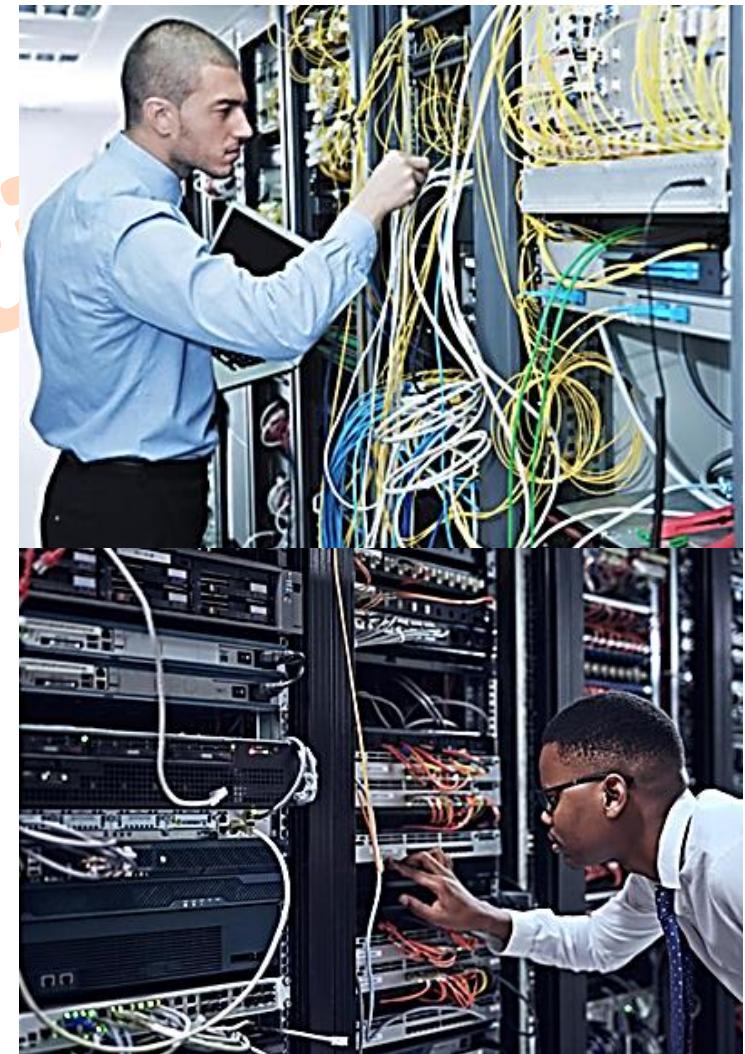
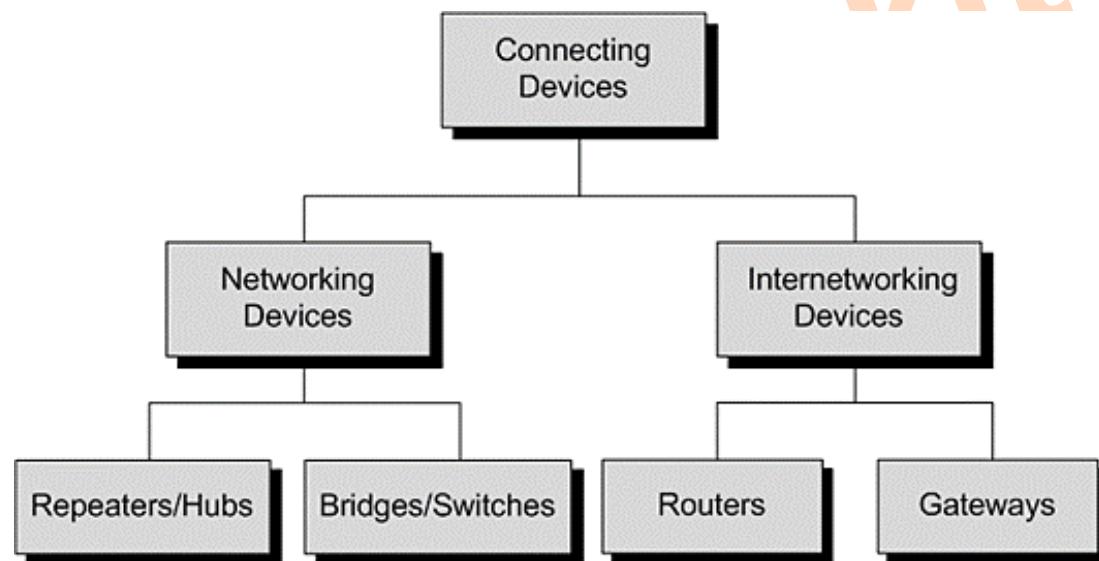
Networking Devices

Panthakit Totid



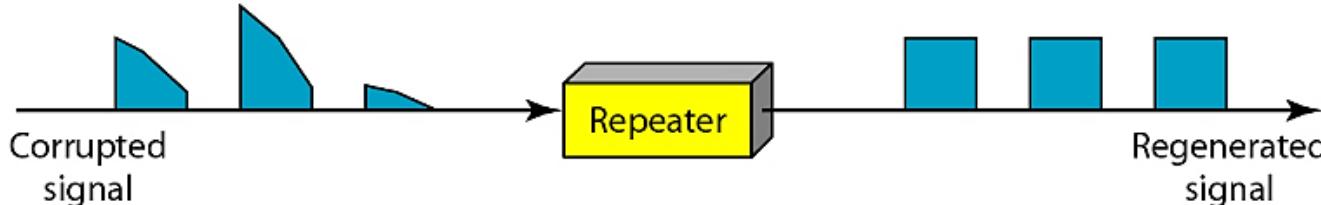
Networking Devices

- Hosts or LANs do not normally operate in isolation. They are connected to one another (internetwork) or to the Internet. To connect hosts or LANs, we use connecting devices.
- In this section, we divide connecting devices into five different categories **based on the layer in which they operate in a network.**



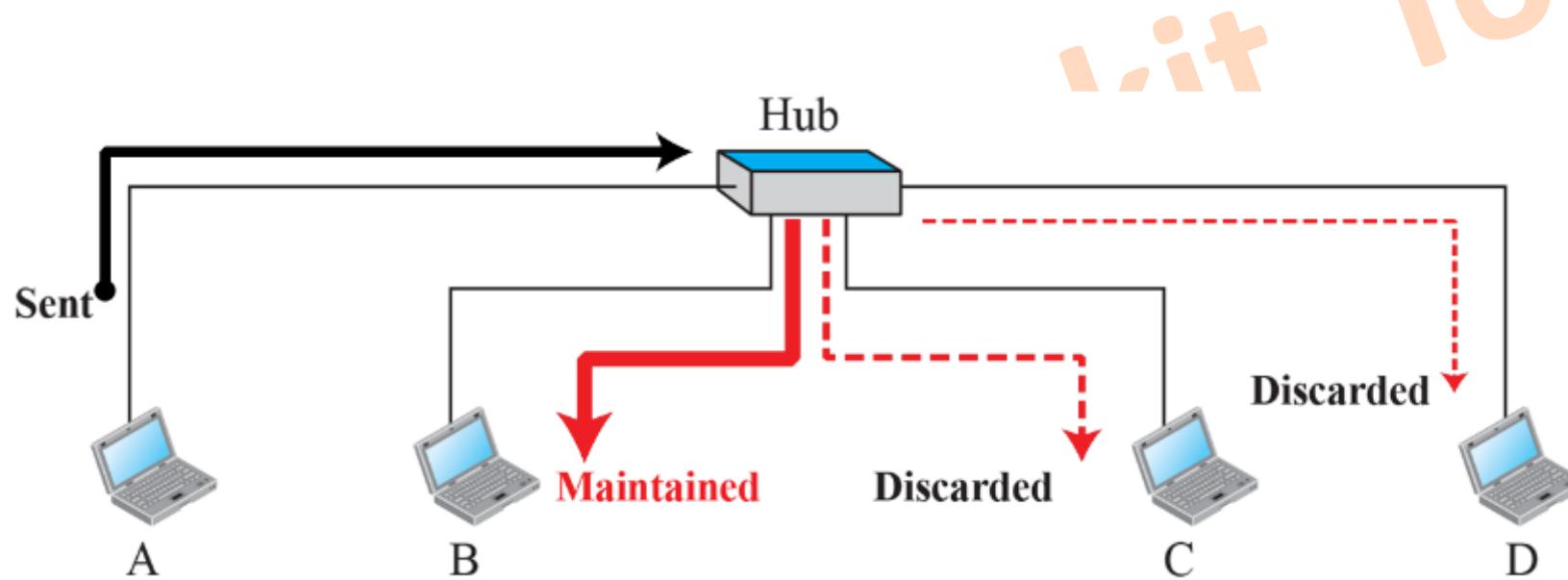
Repeater

- Operate only in physical layer
- Connects two **segments** of the same LAN
 - Both segments must be of the same protocol
- Only forwards frames; **does not filter!**
- Solves attenuation issues by **extending the physical length** of the network.
- Receives signal before too weak or corrupted, regenerates the original pattern, sends a refreshed copy.



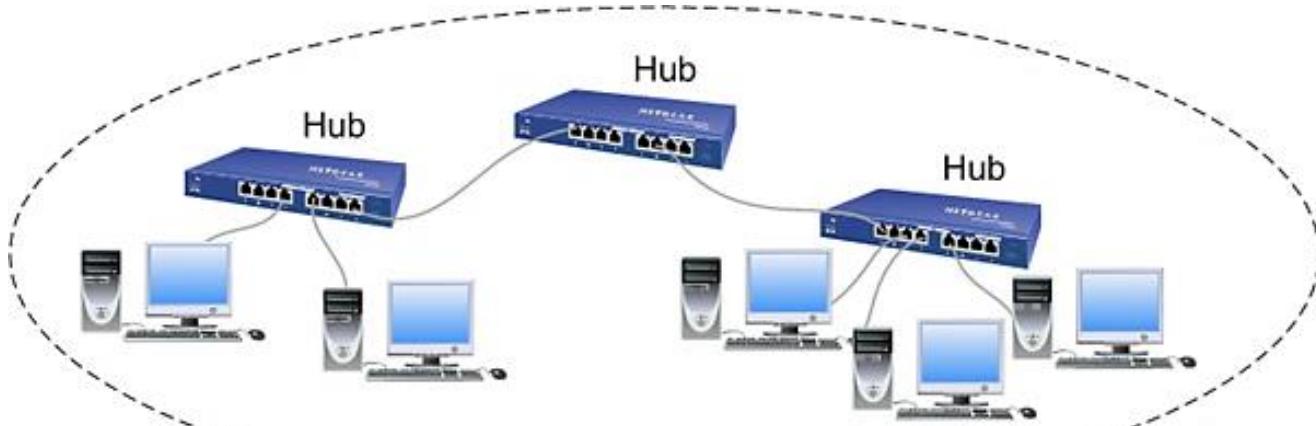
Active Hubs

- Actually a **multiport repeater**.
- Connects stations in a physical star topology.
- In a star topology, a repeater is a multiport device, often called a **hub**, that can be used to serve as the connecting point and at the same time function as a repeater.

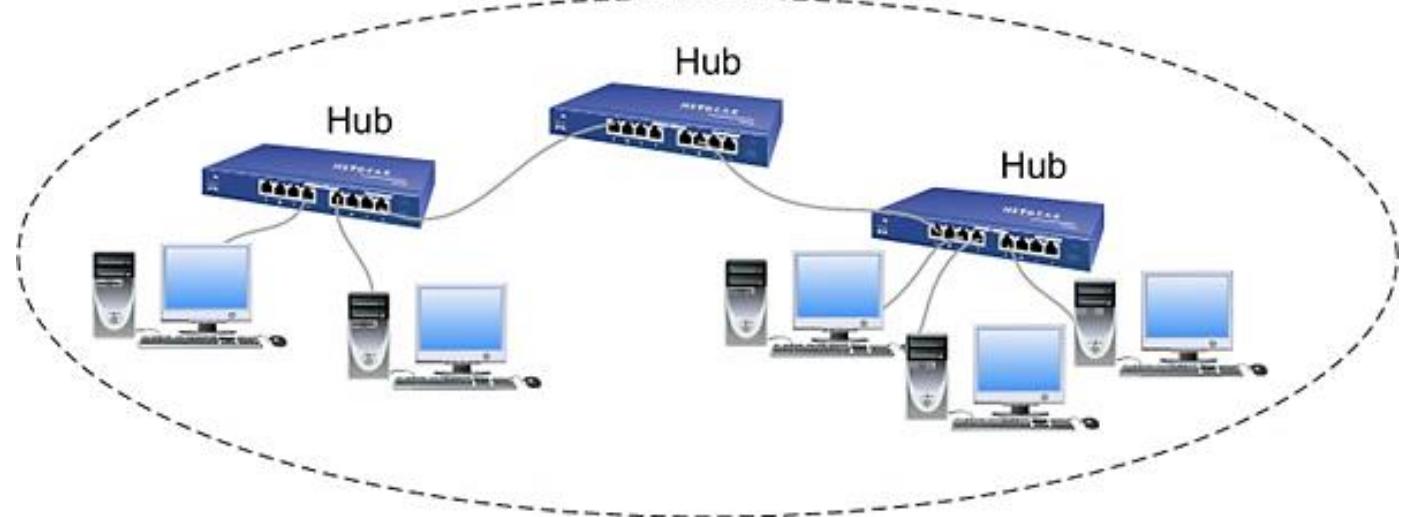


The figure definitely shows that a **hub does not have a filtering capability**; it does not have the intelligence to find from which port the frame should be sent out.

Active Hubs (Cont.)



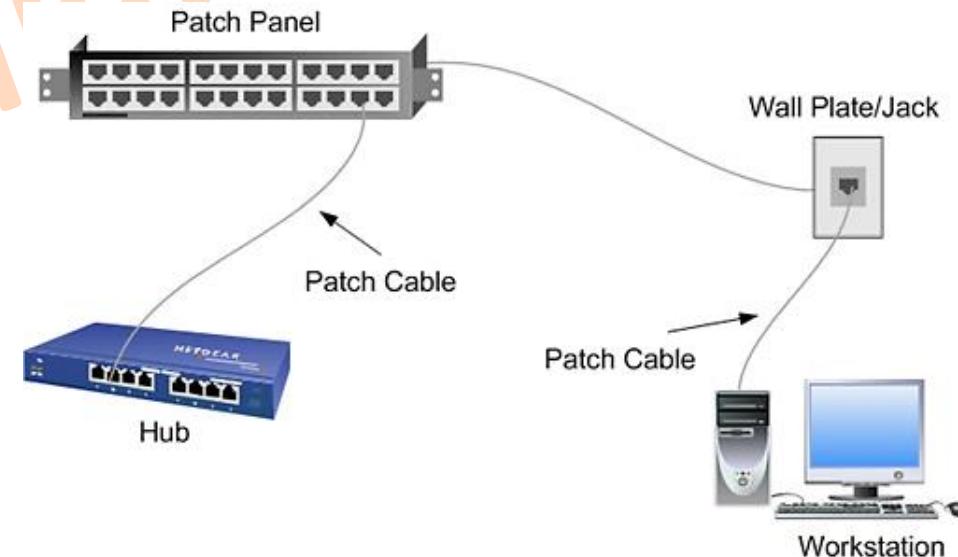
Collision Domain
Pair



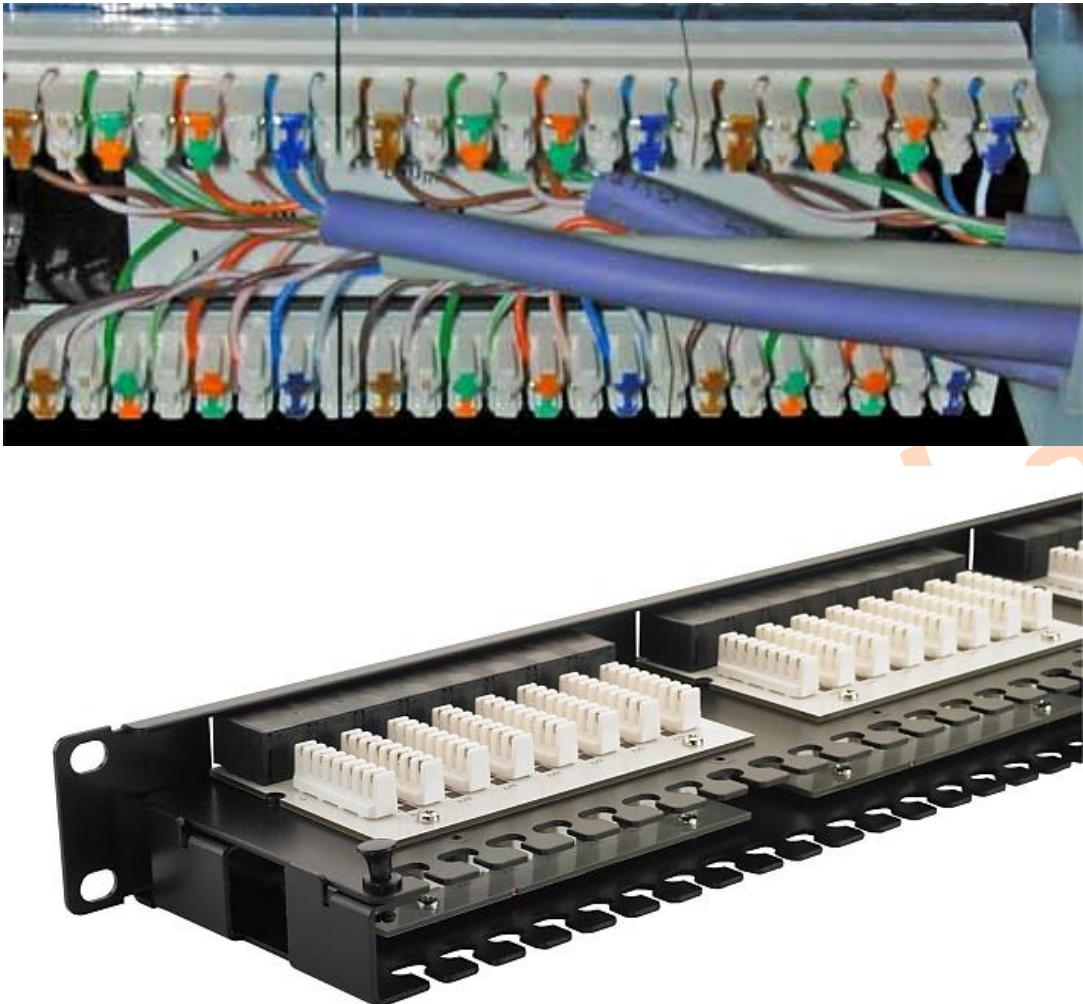
Collision Domain

Passive Hubs (Patch Panel)

- It's just a **connector!** (Junction Point)
- Connects the wires coming from different branches.
 - In a star-topology Ethernet LAN, a passive hub is just a point where the signals coming from different stations collide; the hub is the collision point.
- This type of hub is **part of the media**; its location in the Internet model is below the physical layer.

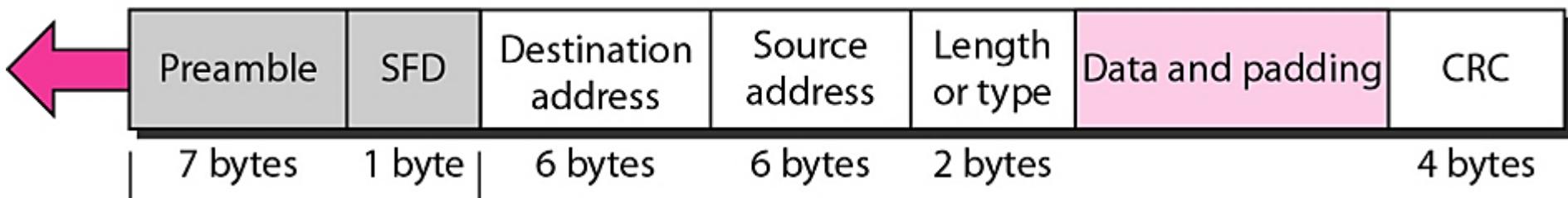


Passive Hubs (Patch Panel) (Cont.)



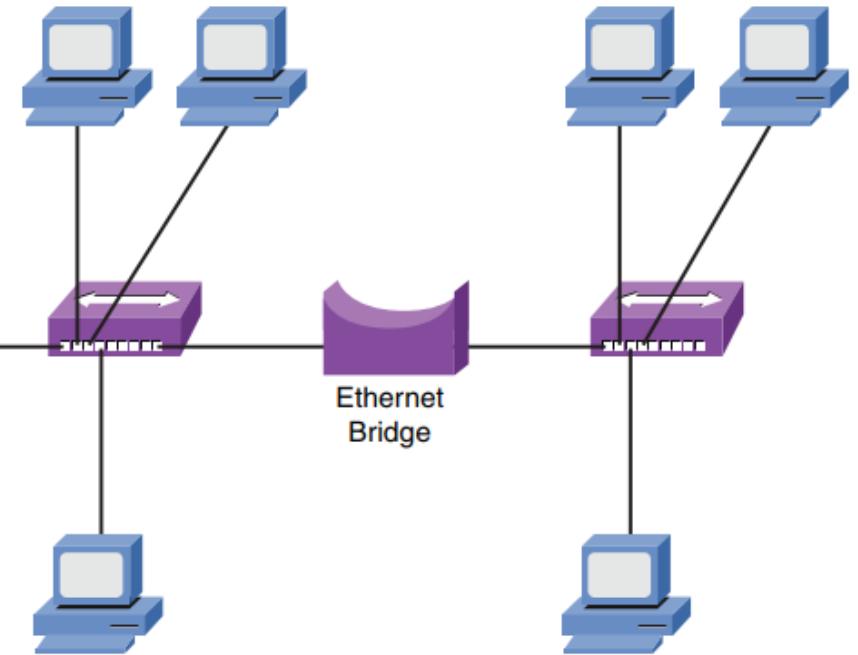
Bridge

- Operate in both physical and data link layers.
- Used to divide a network into **smaller segments**.
 - May also **relay frames between separate LANs**.
 - Keeps traffic from each segment separate; useful for **controlling congestion** and **provides isolation**, as well as **security**.
- Checks address of frame and only forwards to segment to which address belongs.



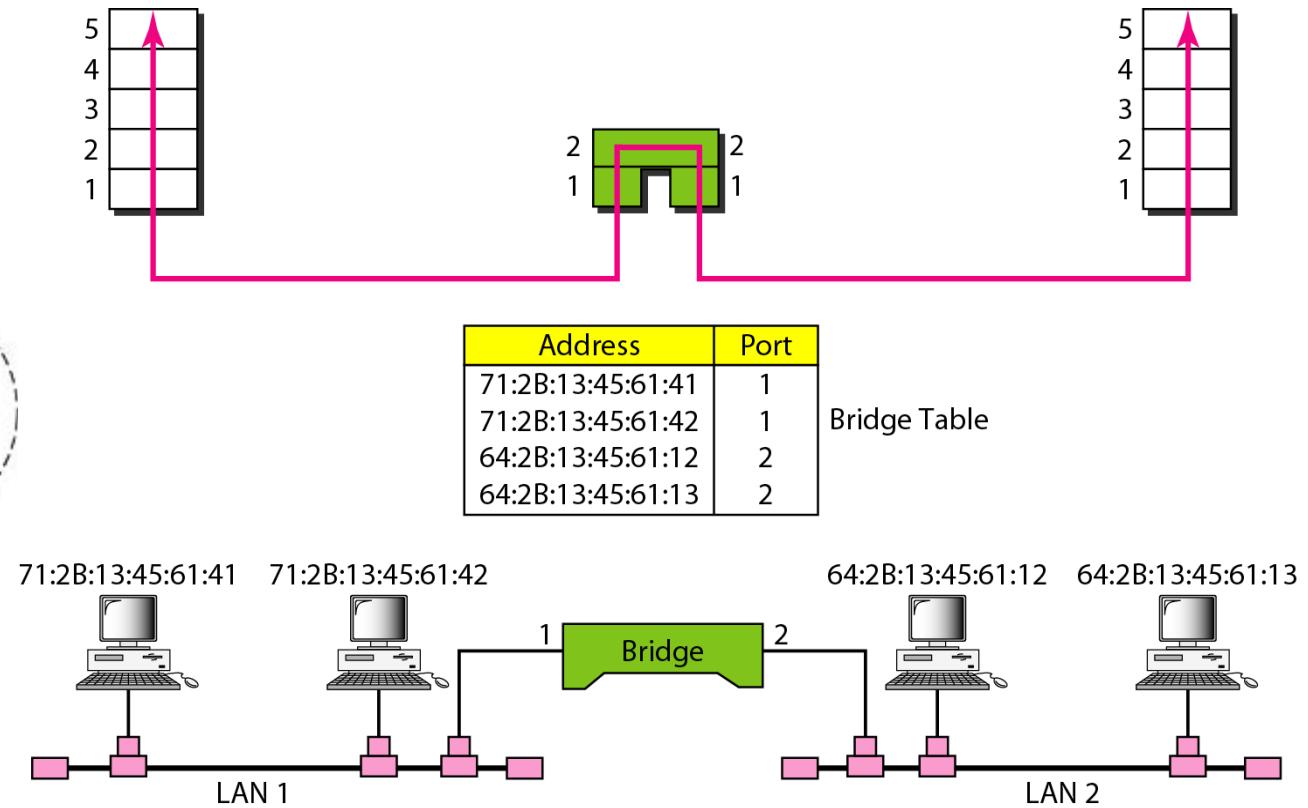
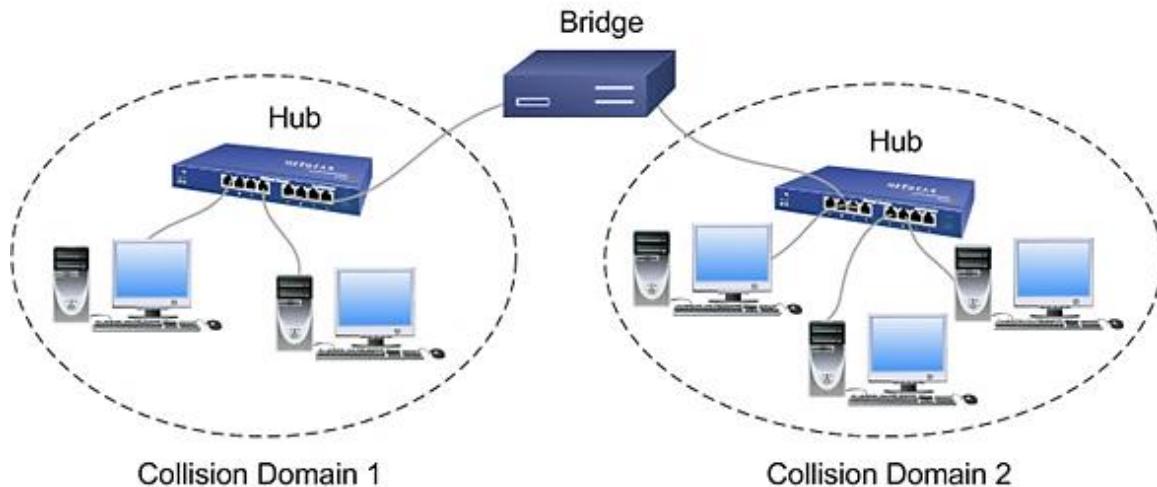
Bridge (Cont.)

- A **bridge joins two or more** Ethernet LAN segments. Each LAN segment is in a separate collision domain. As a result, an Ethernet bridge can be used to scale Ethernet networks to a larger number of attached devices.
- The simplest bridge conveys all traffic that reaches it as a repeater does; it manages collisions by **storing the frame it receives** from one side and waiting for the network to be clear before sending it on the others.



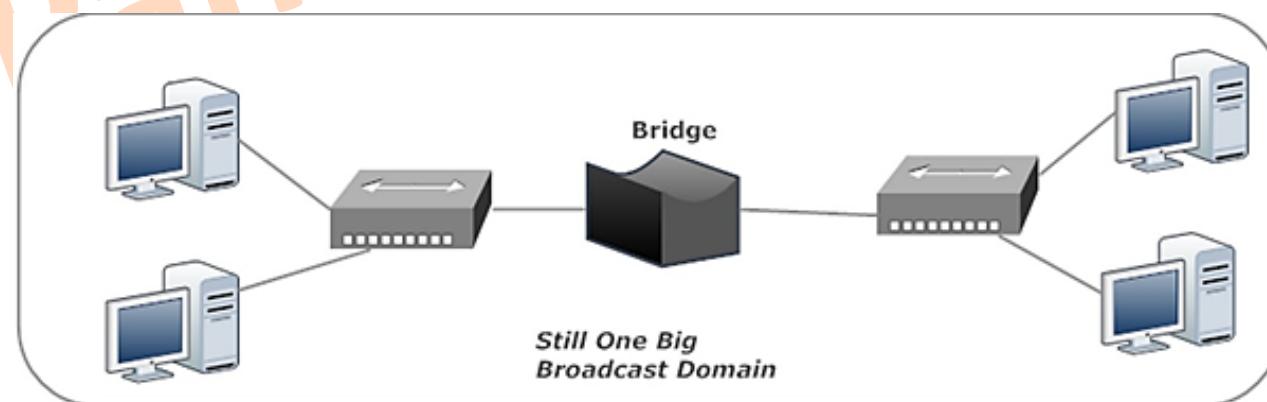
**Two collision domains
One broadcast domain**

Bridge (Cont.)



Bridge (Cont.)

- Although a bridge segments a LAN into multiple collision domains, **all ports on a bridge belong to the same broadcast domain**.
- To understand this concept, think about the destination MAC address in a **broadcast frame**. At Layer 2, the destination MAC address of a broadcast frame is **FFFF.FFFF.FFFF** in hexadecimal notation.
- Because no device on a network will have the MAC address FFFF.FFFF.FFFF, a bridge will never enter that MAC address in its MAC address table. As a result, broadcast frames are **flooded out** all bridge ports other than the port that received the frame.



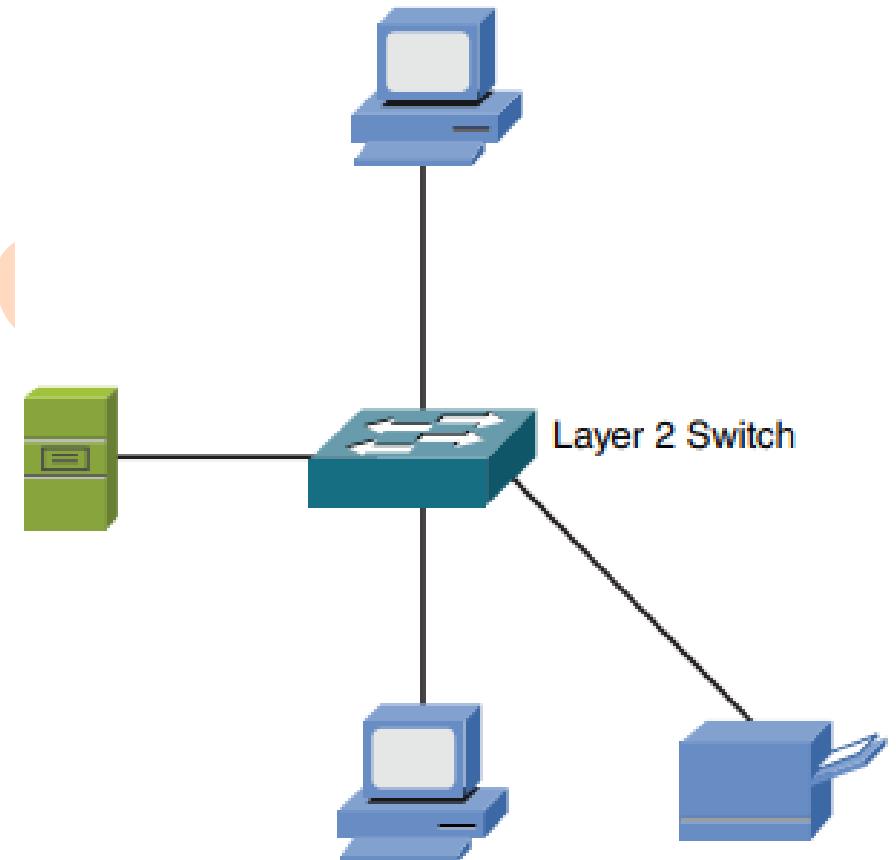
Layer 2 Switch

- Bridges were popular in the mid- to late 1980s and early 1990s, but they have largely been replaced with switches due to price, performance, and features.
- Like a bridge, a switch can **dynamically learn the MAC addresses** attached to various ports by looking at the source MAC addresses on frames coming into a port.
- All or most of switch's ports are usually for the same network standard (e.g. Gigabit Ethernet). One or more ports might be at a different speed or use a different medium, and can be auto-sensing.



Layer 2 Switch (Cont.)

- As on a bridge, each port on a switch represents a **separate collision domain**.
- Also, **all ports on a switch belong to the same broadcast domain**, with one exception: when the ports on a switch have been divided into separate virtual LANs (VLANs).
 - **Each VLAN** represents a **separate broadcast domain**, and for traffic to travel from one VLAN to another, that traffic must be routed by a Layer 3 device.

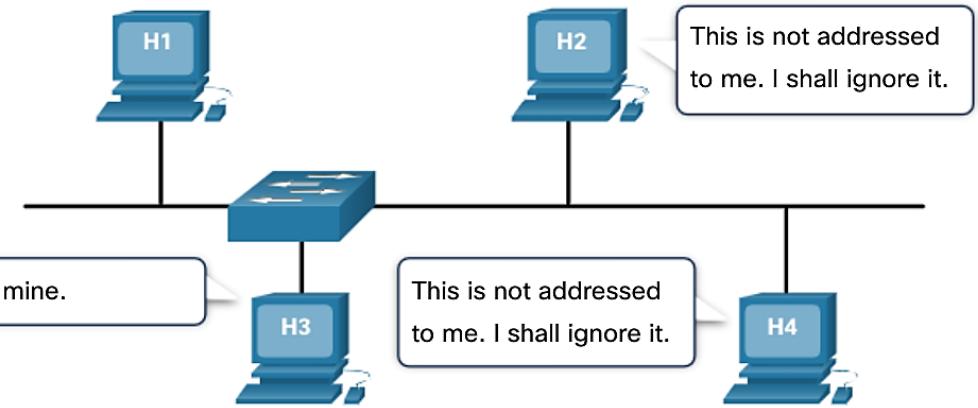


**Four collision domains
One broadcast domain**

Frame Processing

- When a device is forwarding a message to an Ethernet network, the Ethernet header include a **Source MAC address** and a **Destination MAC address**.
- When a NIC receives an Ethernet frame, it examines the destination MAC address to see if it matches the physical MAC address that is stored in RAM.
 - If there is no match, the device discards the frame.
 - If there is a match, it passes the frame up the OSI layers, where the de-encapsulation process takes place.

Destination Address	Source Address	Data
CC:CC:CC:CC:CC:CC	AA:AA:AA:AA:AA:AA	Encapsulated data
Frame Addressing		



Any device that is the source or destination of an Ethernet frame, will have an Ethernet NIC and therefore, a MAC address. This includes workstations, servers, printers, mobile devices, and routers.

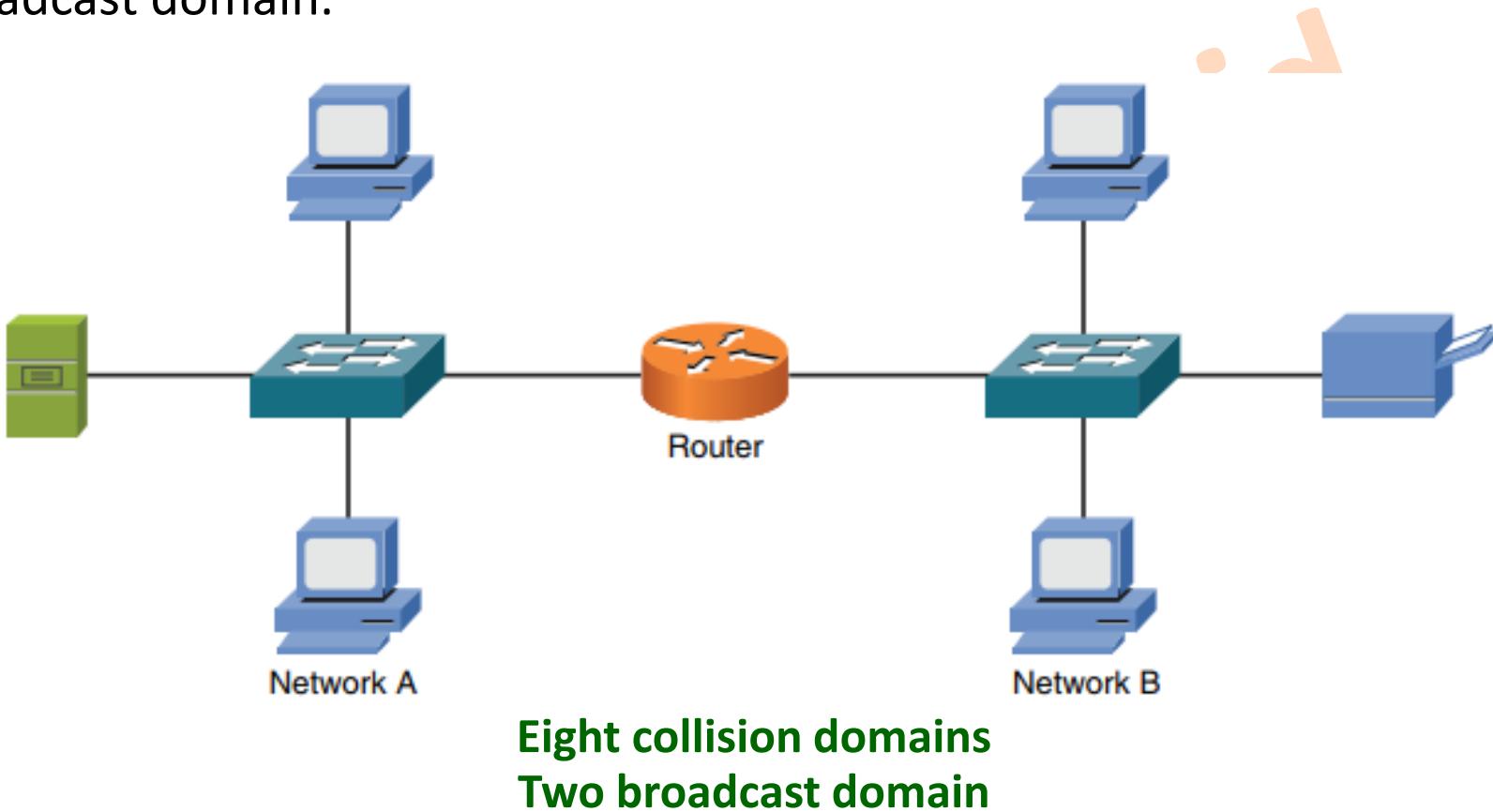
Router

- A router is a Layer 3 device, which means it makes **forwarding decisions based on logical network address** information.
- Normally router connects **LANs and WANs** in the Internet and has a routing table that is used for making decision about the route.
 - The routing tables are normally dynamic and are updated using **routing protocols**.
- Although a router is considered to be a Layer 3 device, like a multilayer switch, it has the capability to consider **high-layer traffic parameters**, such as quality of service (QoS) settings, in making forwarding decisions.



Router

- As shown in following figure, each port on a router is a separate collision domain and a separate broadcast domain.



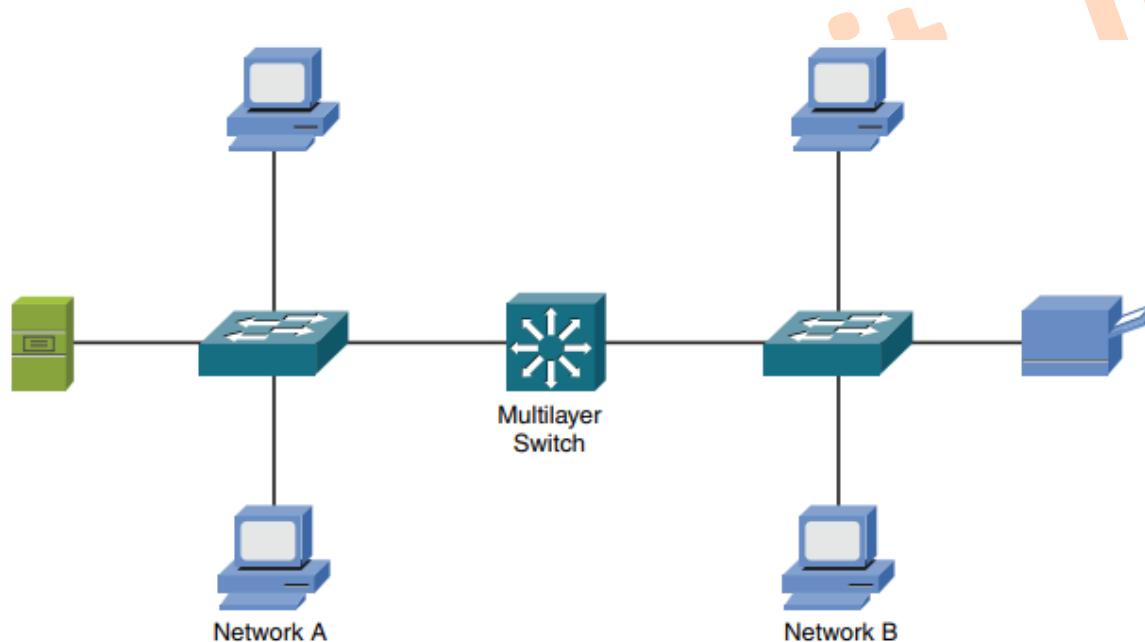
Layer 3 Switch

- Whereas a Layer 2 switch makes forwarding decisions based on MAC address information, a **multilayer switch** can make forwarding decisions based on upper-layer information.
 - For example, a multilayer switch could function as a router and make forwarding decisions based on destination IP address information.
- Many multilayer switches have **policy-based routing** features that allow upper-layer information (for example, application port numbers) to be used in making forwarding decisions.



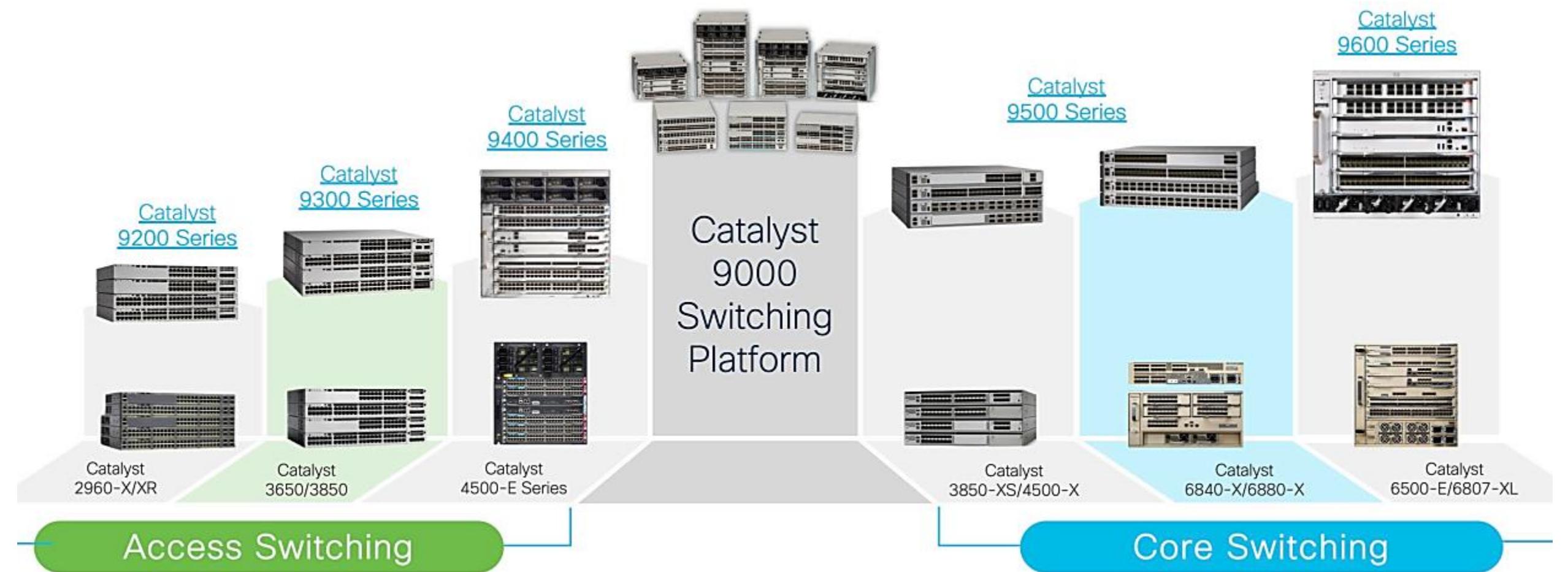
Layer 3 Switch (Cont.)

- The multilayer switch can be used to interconnect not just network segments but entire networks. For traffic to **travel between two networked devices** that belong to different networks, that traffic must be **routed**.
 - A device, such as a multilayer switch, has to make a forwarding decision based on Layer 3 information.



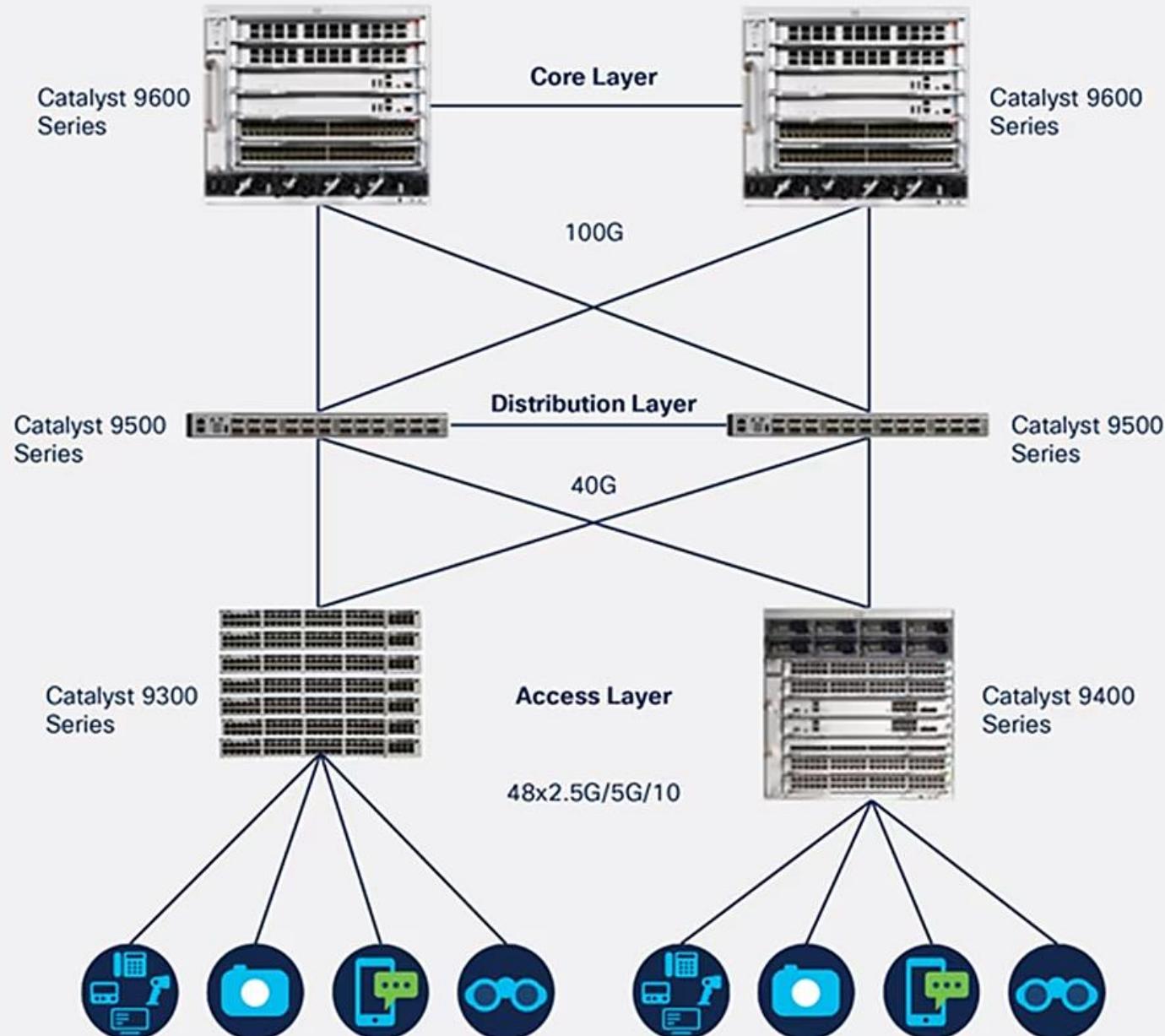
Eight collision domains
Two broadcast domain

Layer 3 Switch (Cont.)



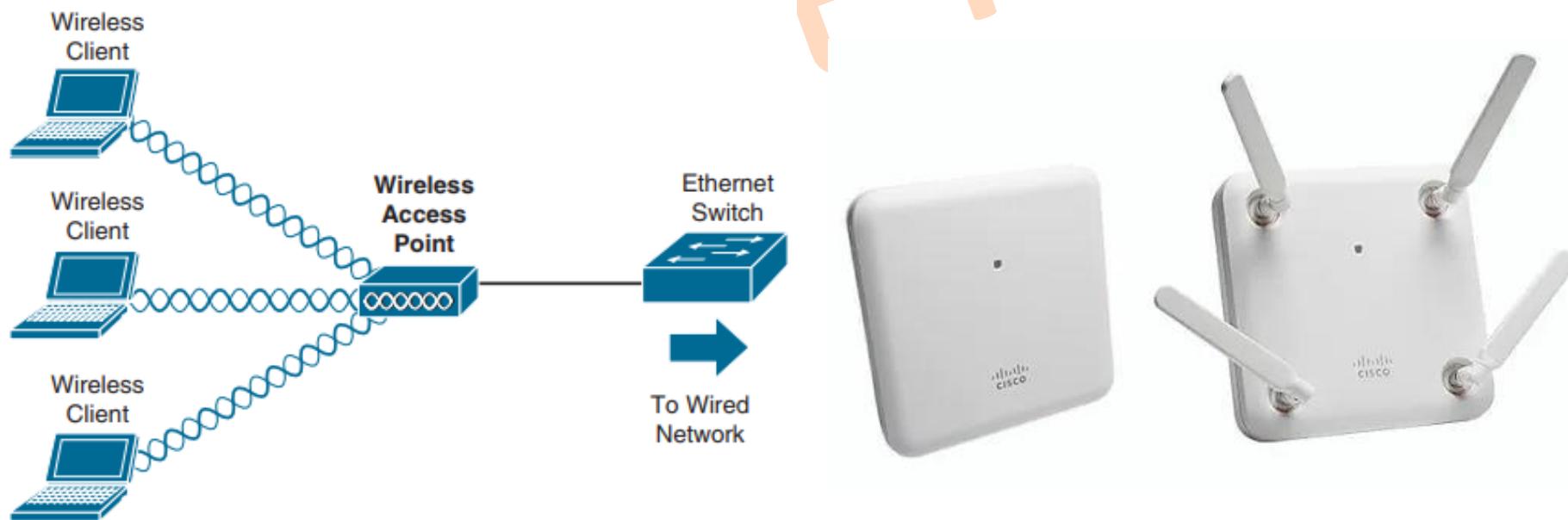
Layer 3 Switch

Higher bandwidth 40G/100G



Access Points

- In wireless LAN (WLAN), wireless clients gain access to a wired network by communicating via radio waves with a wireless **access point (AP)**.
 - All wireless devices connecting to the same AP are considered to be on the same **shared network segment**, which means that only one device can send data to and receive data from an AP at any one time.

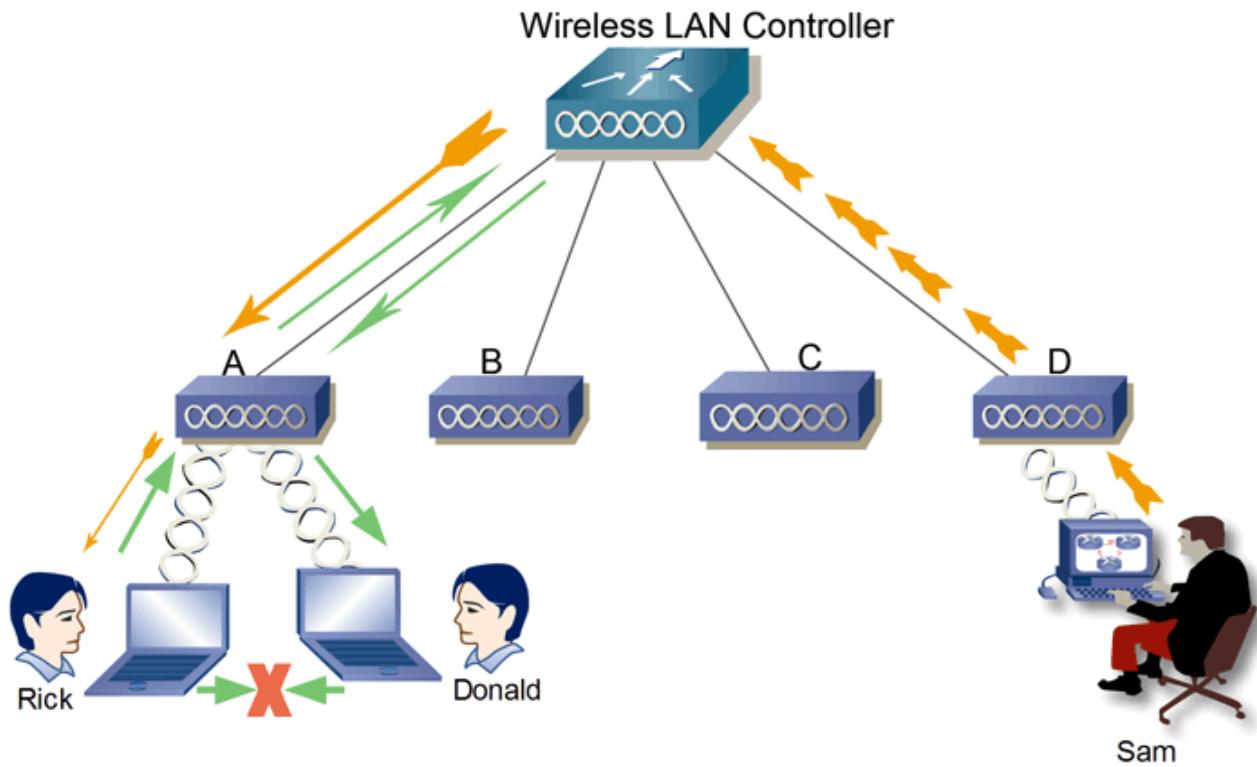


Wireless LAN Controller

- Two major types of AP: *autonomous* and *lightweight*.
- Lightweight access points do not have the control plane intelligence built in to perform their functions for the network. These devices are controlled by **wireless LAN controllers (WLCs)**.
 - WLCs are specialized network devices that permit the central control and management of large numbers of lightweight access points.
 - WLCs simplify the administration of your access points, and they can also assist you dramatically in the monitoring and ongoing maintenance of the wireless infrastructure.



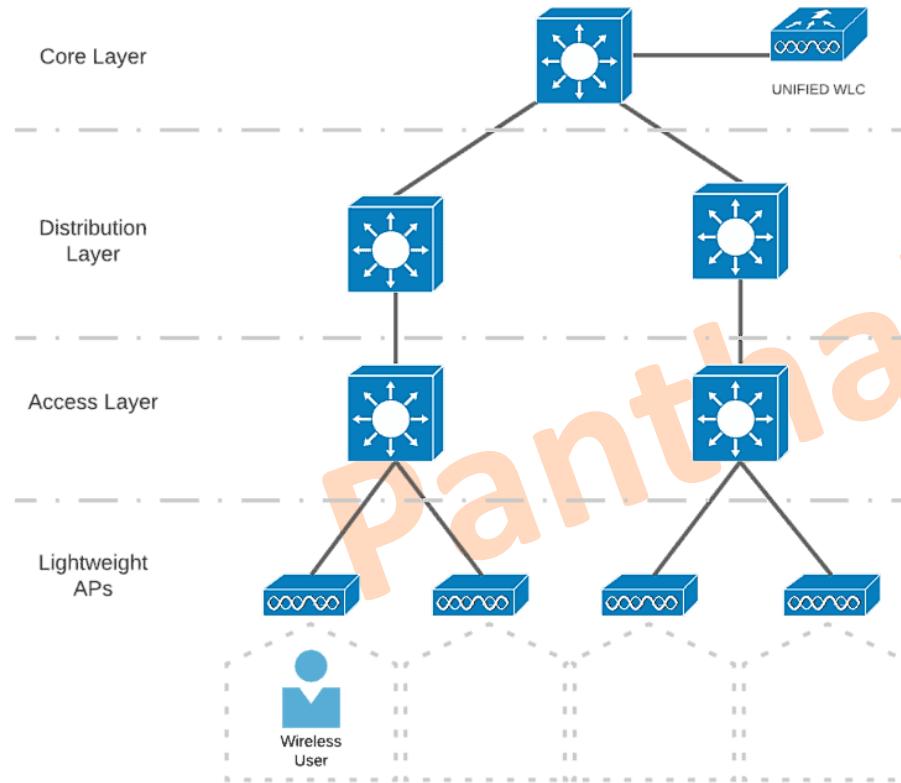
Wireless LAN Controller (Cont.)



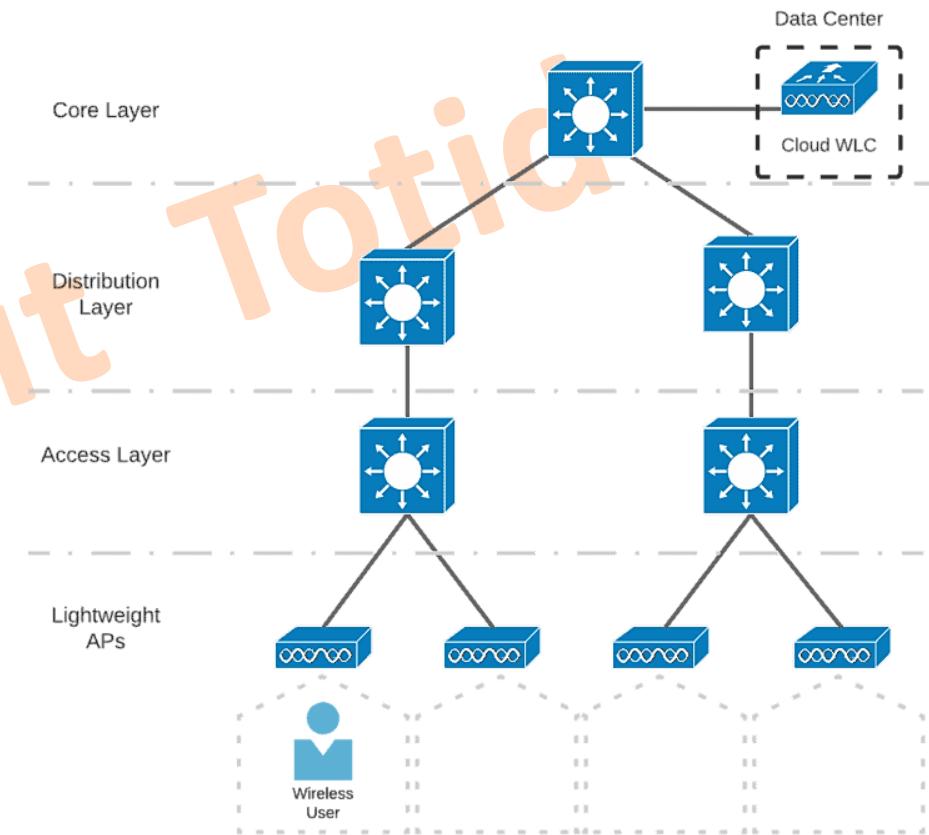
- WLCs are specialized network devices that permit the **central control and management** of large numbers of lightweight access points.
- WLCs simplify the administration of your access points, and they can also assist you dramatically in the monitoring and ongoing maintenance of the wireless infrastructure.

Wireless LAN Controller (Cont.)

- Comparing WLC Deployments



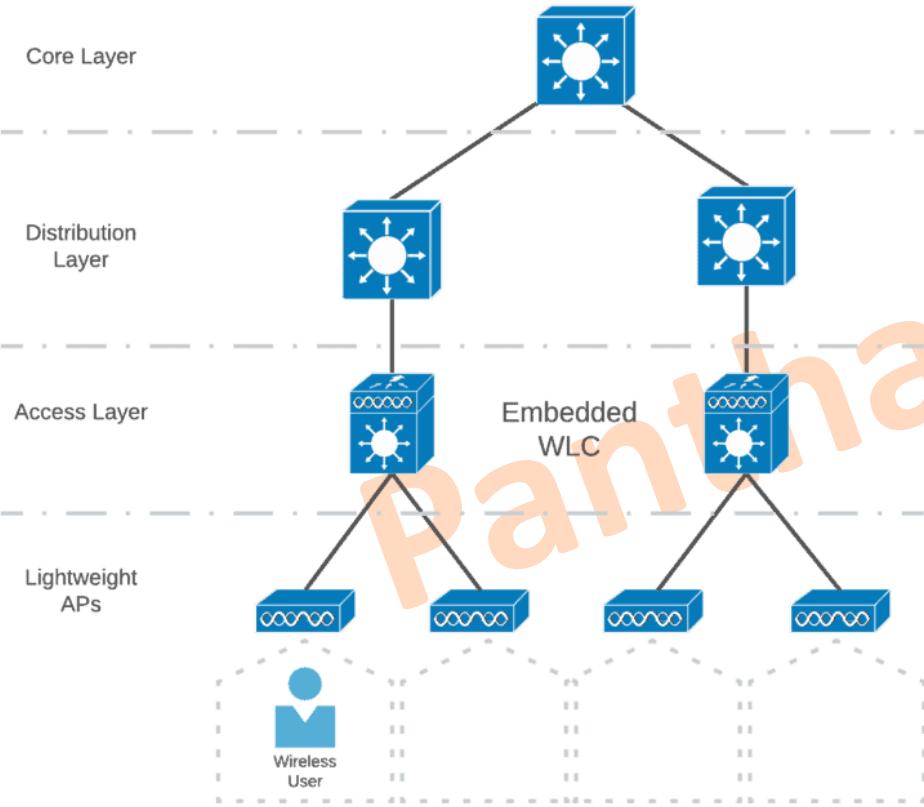
Unified or Centralized Deployment



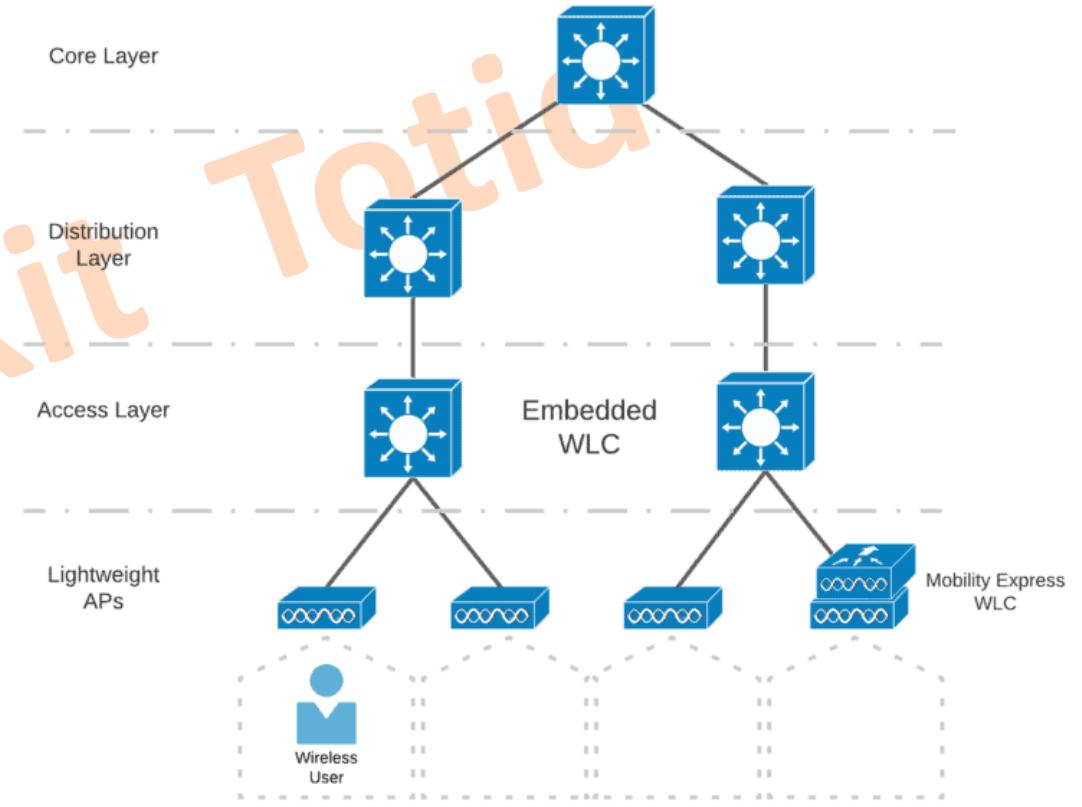
Cloud-Based Deployment

Wireless LAN Controller (Cont.)

- Comparing WLC Deployments



Embedded Deployment



Mobility Express Deployment

Wireless LAN Controller (Cont.)

- Summary of WLC Deployment Modes

Deployment Model	WLC Location	AP Supported	Clients Supported	Typical Use
Unified	Central	6,000	64,000	Large Enterprise
Cloud	DC	3,000	32,000	Private Cloud
Embedded	Access	200	4,000	Small Campus
Mobility Express	Other	100	2,000	Branch Location
Autonomous	N/A	N/A	N/A	N/A

Panthakit Totid

Panthakit Totid



Basic Network

Module 4

IPv4 Address and Subnetting

Panthekit Totid



Notational System

- In computing, a **Notational system** is one used to represent different quantities or characters. Notational systems used to represent values and quantities include **decimal**, **binary**, and **hexadecimal**.

Decimal Notation

- For most people around the world, when they want to count something, they use a numbering system based on decimal. The decimal system (or **base 10**) is based on the principle of expressing ten different numbers using a single digit in the **range 0 to 9**

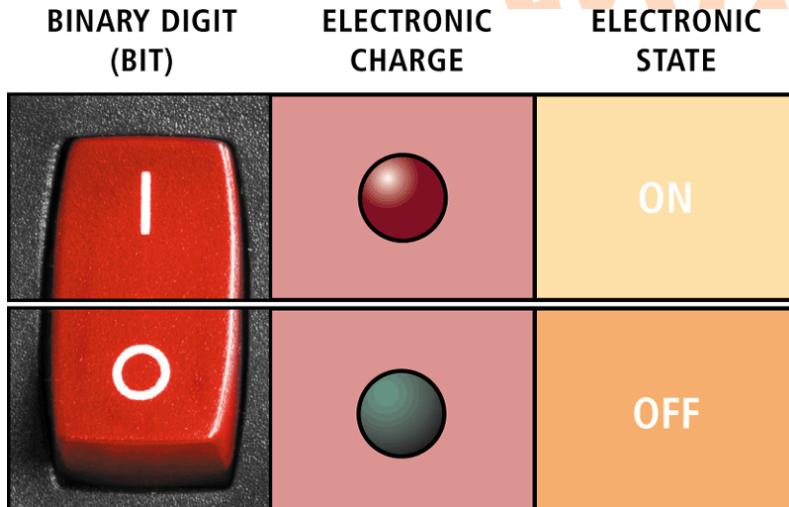
Place Value	1000	100	10	1
	0	2	9	6
	$1000*0$	$100*2$	$10*9$	$1*6$
	0	200	90	6

$$0 + 200 + 90 + 6 = 296$$

Notational Systems (Cont.)

Binary Notation

- **Computers** use binary to calculate and process information. Binary is a numbering system where each single digit can express **only two values**, in the **range 0 to 1**.
- The reason binary works well with computers is that these two values can represent the off/on states of the transistors that make up computer memory. Binary can also be referred to as **base 2**.



2 Bits = 4 States

00	01	10	11
----	----	----	----

3 Bits = 8 States

000	001	010	011
100	101	110	111

Notational Systems (Cont.)

Binary Notation

- IP addresses are represented as series of binary digits.

- IPv4 address: 32 bits

11011011 00011111 10101100 00110101 → ?

- IPv6 address: 128 bits

00100000 00000001 11011011 10000000 → ?

10101011 11001101 11111110 10010101

00000000 00000000 00000000 00000000

00000000 01101010 11111111 11111111

- You need to be able to convert between the decimal representation of a number and that number's binary equivalent.

Notational Systems (Cont.)

Binary Notation

- The column headings are the powers of 2, from 0 to 7, beginning in the rightmost column.

Place Value	128	64	32	16	8	4	2	1
	1	0	0	1	0	1	1	0
	128*1	64*0	32*0	16*1	8*0	4*1	2*1	1*0
	128	0	0	16	0	4	2	0

$$128 + 0 + 0 + 16 + 0 + 4 + 2 + 0 = 150$$

Notational Systems (Cont.)

Question 1: Converting a Binary Number to a Decimal Number

	128	64	32	16	8	4	2	1	Answer
	1	1	0	0	0	1	1	1	
	0	0	1	1	1	1	0	0	
	0	1	0	0	1	1	1	1	
	1	1	1	0	0	1	0	0	
	0	1	0	0	1	1	0	0	
	1	0	0	1	0	0	1	1	

Notational Systems (Cont.)

Question 2: Converting a Decimal Number to a Binary Number

Answer

128	64	32	16	8	4	2	1	
						1	0	114
						1	1	67
						1	1	127
						1	0	143
						1	1	389
						1	0	273
						1	1	1022

Notational Systems (Cont.)

Hexadecimal Notation

- When handling **large values**, expressing them in binary can consume many digits. You can express large numbers more efficiently by using hexadecimal, often shortened to "**hex**."
- The hex notation system enables you to **express 16 different** numbers using a single digit in the **range 0 to F**.
- Hex is used in **programming**. You will also encounter this numbering system when you plan and implement **Internet Protocol (IP)** networks.

Place Value	4096	256	16	1
0	4	D	2	
4096*0	256*4	16*13	1*2	
0	1024	208	2	

$$0 + 1024 + 208 + 2 = \mathbf{1234}$$

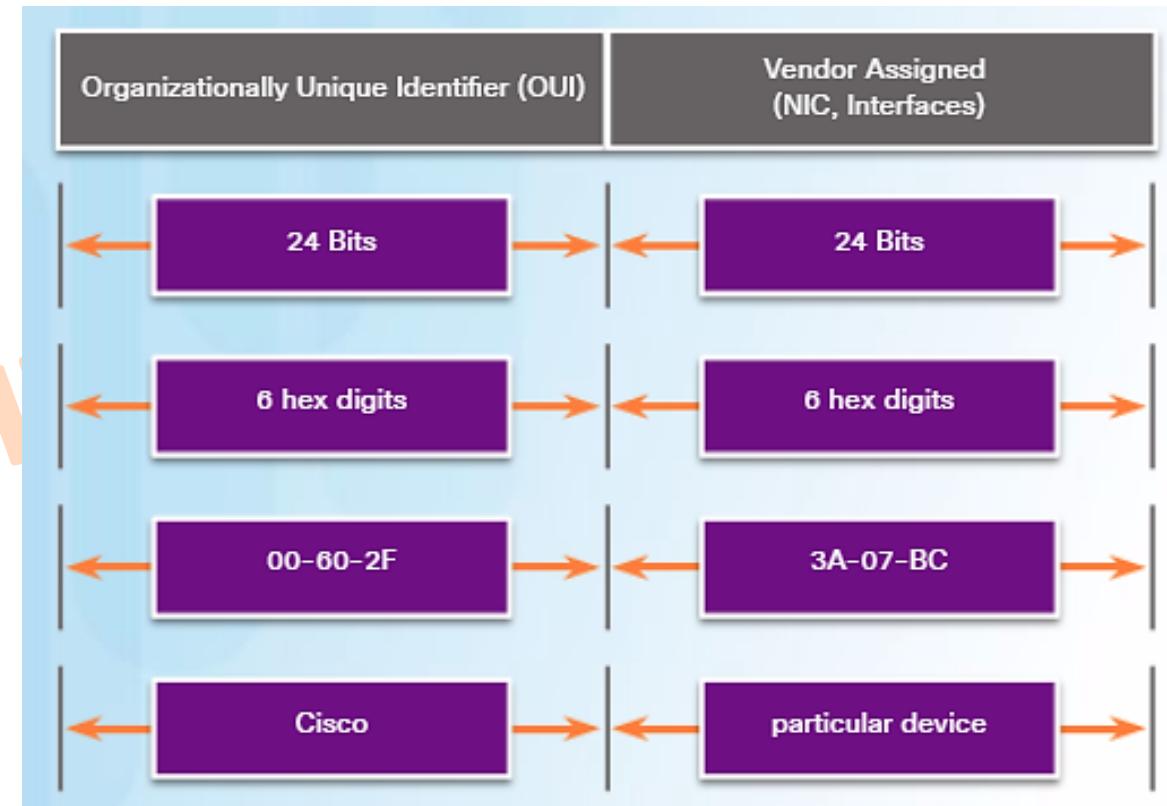
MAC Address

- In an Ethernet LAN, every network device is connected to the same, shared media. MAC addressing provides a method for device identification at the data link layer of the OSI model.
- An Ethernet MAC address is a **48-bit** binary value expressed as 12 hexadecimal digits (4 bits per hexadecimal digit).

Decimal	Binary	Hexadecimal
0	0000	0
1	0001	1
2	0010	2
3	0011	3
4	0100	4
5	0101	5
6	0110	6
7	0111	7
8	1000	8
9	1001	9
10	1010	A
11	1011	B
12	1100	C
13	1101	D
14	1110	E
15	1111	F

MAC Address (Cont.)

- MAC addresses were created to identify the actual source and destination.
 - The MAC address rules are established by IEEE.
 - The IEEE assigns the vendor a 3-byte (24-bit) code, called the **Organizationally Unique Identifier (OUI)**.
 - All MAC addresses assigned to a NIC or other Ethernet device must use that vendor's assigned OUI as the first 3 bytes.
 - All MAC addresses with the same OUI must be assigned a unique value in the last 3 bytes.



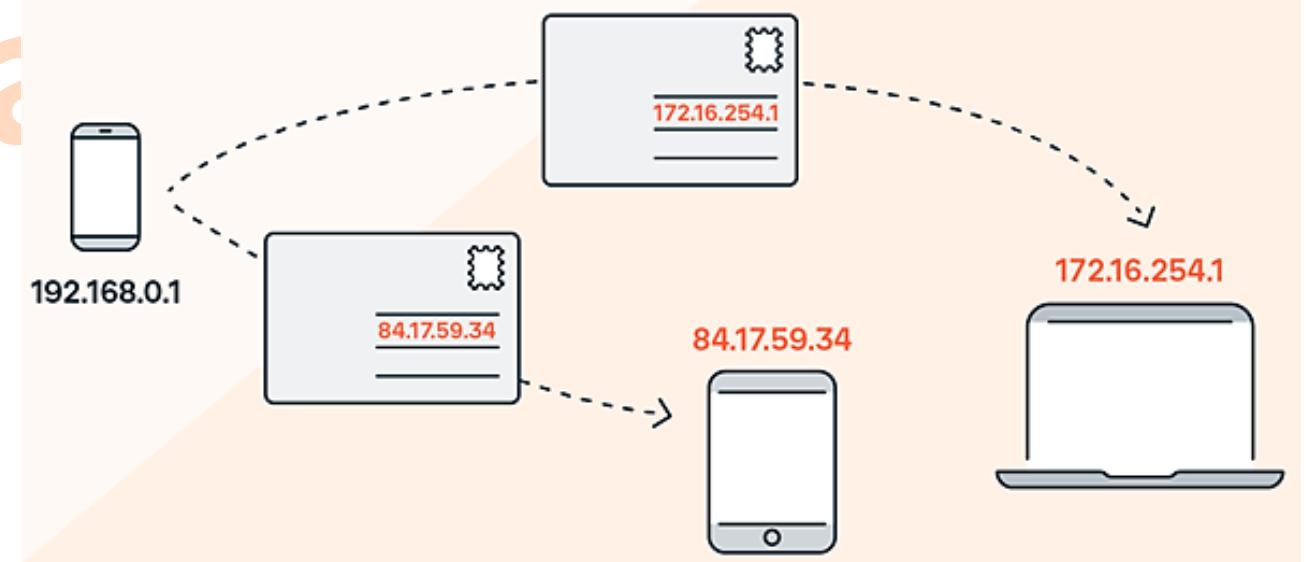
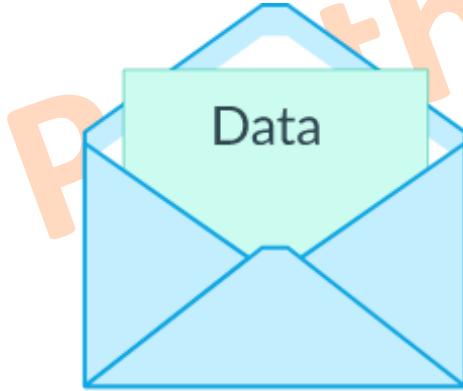
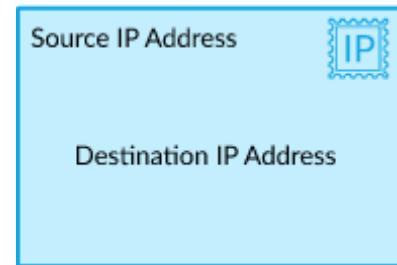
Internet Assigned Numbers Authority (IANA)

- Currently run by **Internet Corporation for Assigned Names and Numbers (ICANN)**
- Manages allocation of IP addresses and maintenance of the top-level domain space.
- **Regional Internet Registries (RIRs)** and **ISPs**



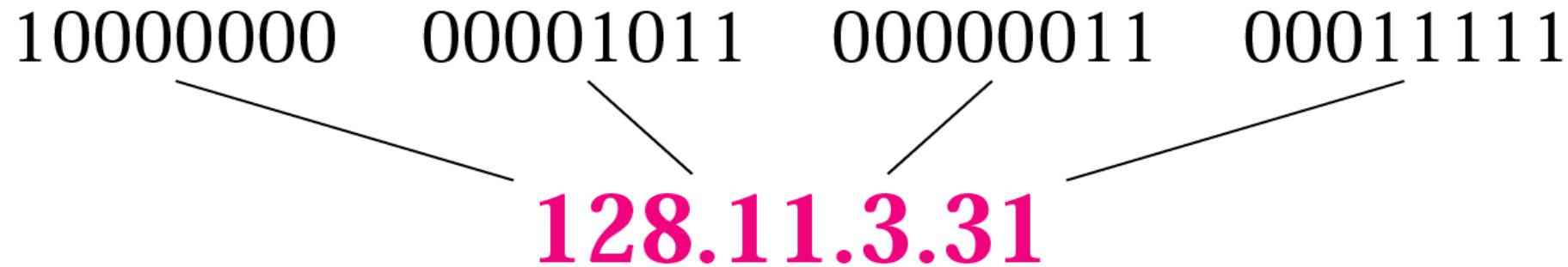
IPv4 Addressing

- **IPv4** is the most popular Layer 3 addressing scheme in today's networks.
 - The term **IPv4 address** is used interchangeably with the more generic term **IP address**.
- Devices on an IPv4 network use unique IP addresses to communicate with one another.
 - you can relate this to sending a letter through the postal service.



IPv4 Address Structure

- An IP address is used to **logically identify each device (host)** on a given network.
 - An IP address is a **32-bit** binary value.
 - To make this value easier to enter in configuration dialogs, it is expressed as **four decimal numbers** separated by periods.
 - Each number represents a byte value, that is, an **eight-character binary value**, also called an **octet**, or a decimal value between **0 and 255**. This is referred to as **dotted decimal notation**.



IPv4 Address Structure (Cont.)

- RFC 790 (1981) allocated IPv4 addresses in classes:

A B C D E

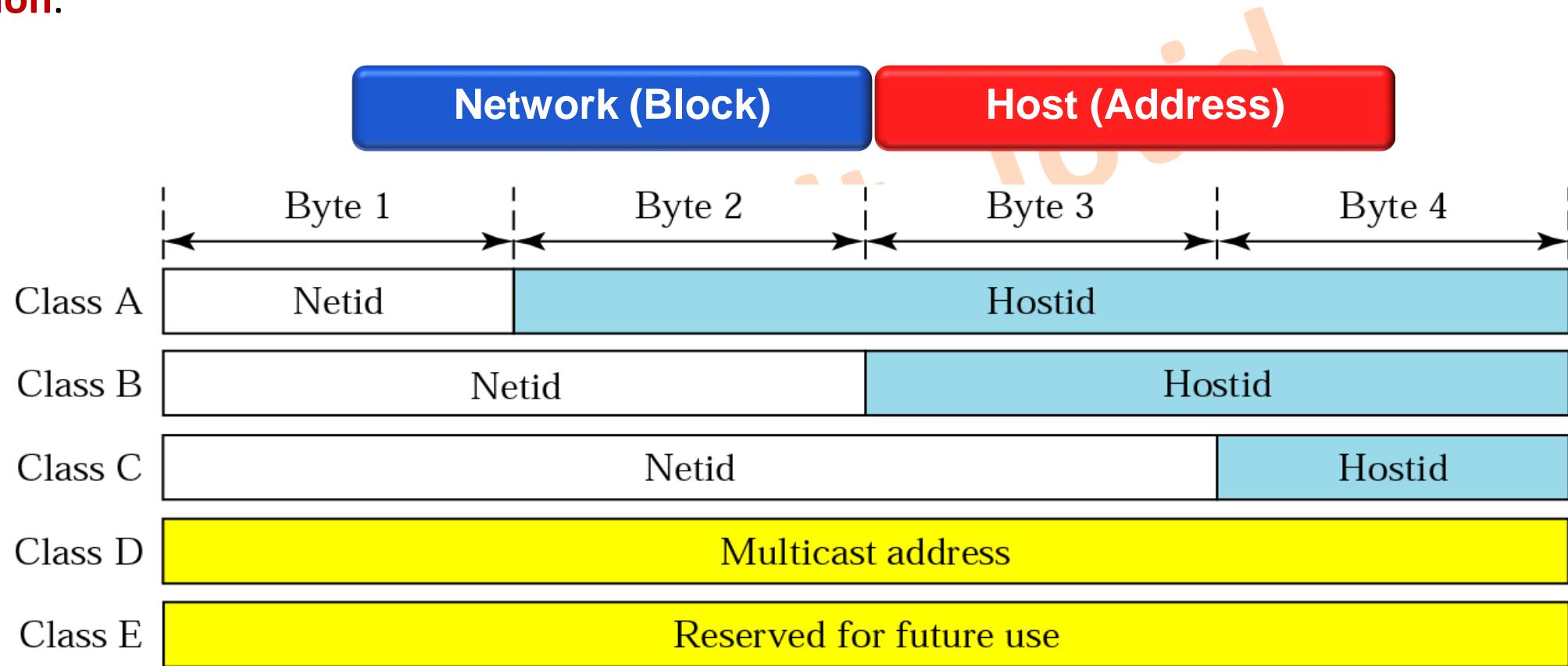
- First byte of address identifies class
 - **A B C : Unicast**
 - **D : Multicast**
 - **E : Reserved**

	First byte	Second byte	Third byte	Fourth byte
Class A	0			
Class B	10			
Class C	110			
Class D	1110			
Class E	1111			

	First byte	Second byte	Third byte	Fourth byte
Class A	0 to 127			
Class B	128 to 191			
Class C	192 to 223			
Class D	224 to 239			
Class E	240 to 255			

IPv4 Address Structure (Cont.)

- An IPv4 address is a 32-bit hierarchical address that is made up of a **network portion** and a **host portion**.

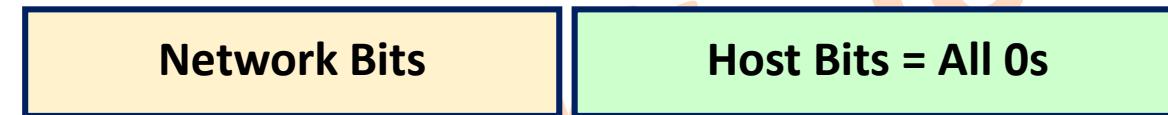


IPv4 Addressing

There are 2 IP addresses we cannot use on our network.

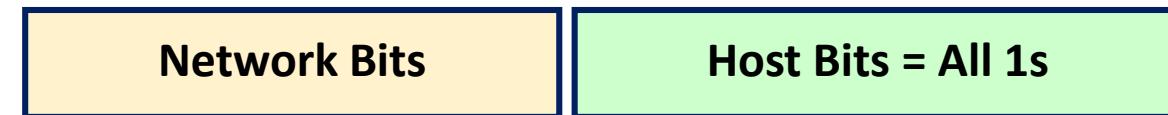
- **Network Address**

- The network address cannot be used on a computer as an IP address because it's being used to “**define**” the network.



- **Broadcast Address**

- The broadcast address cannot be used on a computer as an IP address because it's used by **broadcast applications**.
- A broadcast is an IP packet that will be received by **all devices** in your network.



IPv4 Addressing (Cont.)

Subnet Masks

- In order to **distinguish the network ID and host ID portions** within an address, each host must also be configured with a **subnet mask** (or **network prefix length**).

Class	Classful Mask (Dotted Decimal)	Classful Mask (Binary)	Classful Mask (Prefix Notation)
A	255.0.0.0	11111111 00000000 00000000 00000000	/8
B	255.255.0.0	11111111 11111111 00000000 00000000	/16
C	255.255.255.0	11111111 11111111 11111111 00000000	/24

IPv4 Addressing (Cont.)

Subnet Masks

- The actual process used to identify the network and host portions is called **ANDing**.

Logical AND Operation

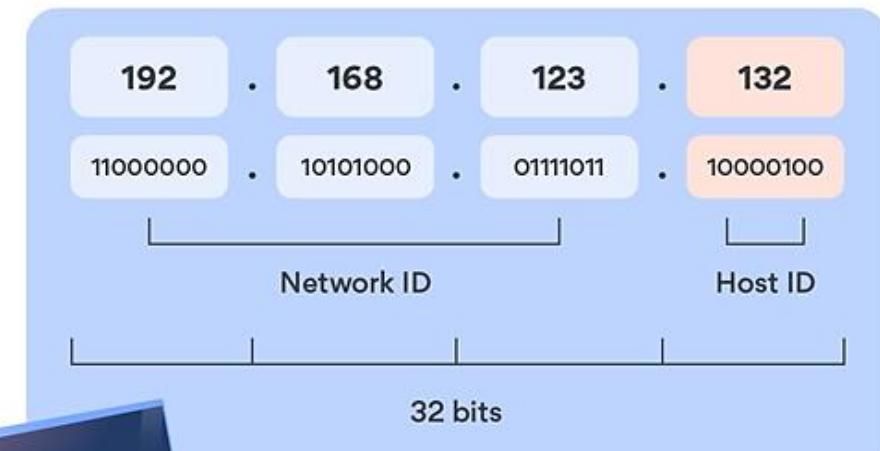
$$0 \text{ AND } 0 = 0$$

$$0 \text{ AND } 1 = 0$$

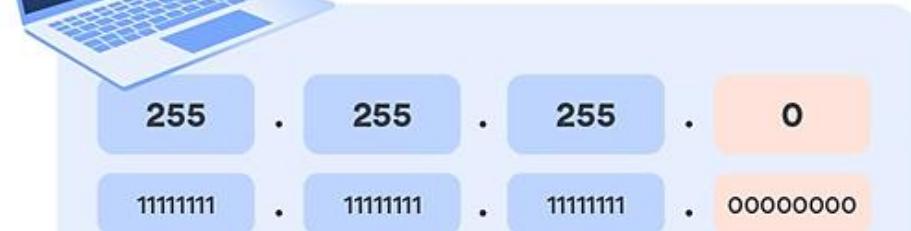
$$1 \text{ AND } 0 = 0$$

$$1 \text{ AND } 1 = 1$$

IP address explained



Subnet mask



Subnet \rightarrow 192.168.123.0

Device in the subnet \rightarrow 192.168.123.132

IPv4 Addressing (Cont.)

Question 3: Given the following address, find the class, the network address, and broadcast address.

IP Address	Class	Network Address	Broadcast Address
136.67.13.9			
201.34.12.72			
15.32.56.7			
187.34.94.126			
120.111.99.160			
202.188.192.168			

IPv4 Addressing (Cont.)

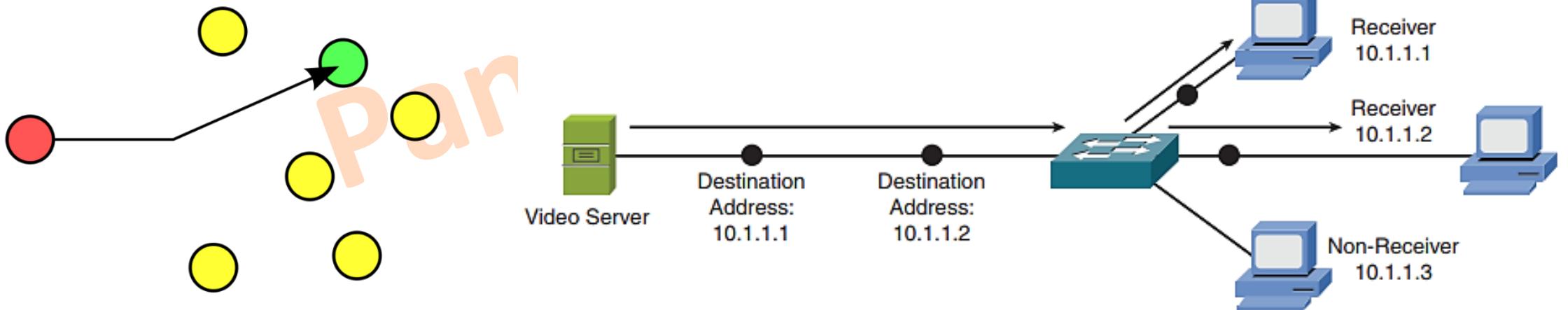
Question 4: Given the following address, find the first and the last of usable IP addresses (host ranges address).

IP Address	Usable IP Addresses
218.155.230.14	
10.250.1.60	
158.98.70.88	
126.202.231.41	
199.155.11.22	

Type of Addresses

Unicast

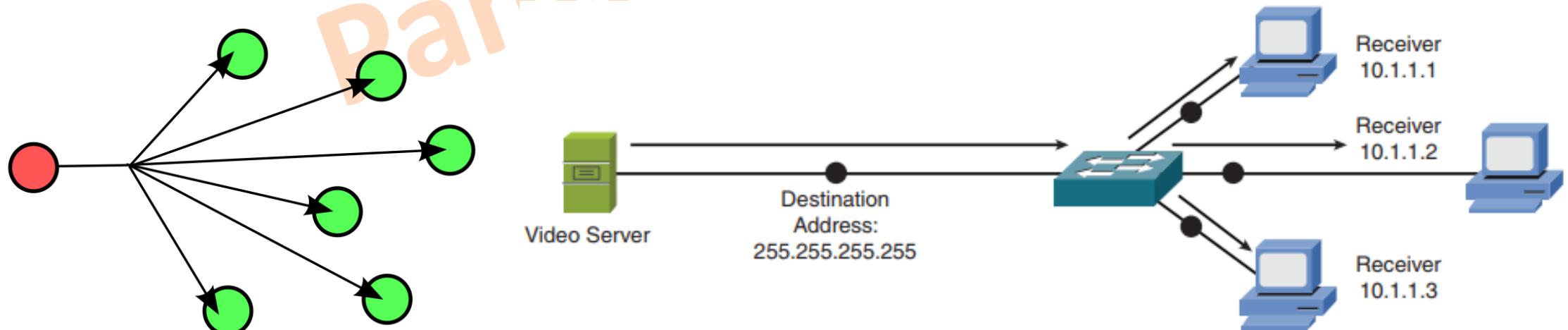
- This is an address for a **single interface**, and these are used to send packets to a **single destination host**.
 - Most network traffic is unicast in nature, meaning that traffic travels from a single source device to a single destination device.



Type of Addresses (Cont.)

Broadcast (L3)

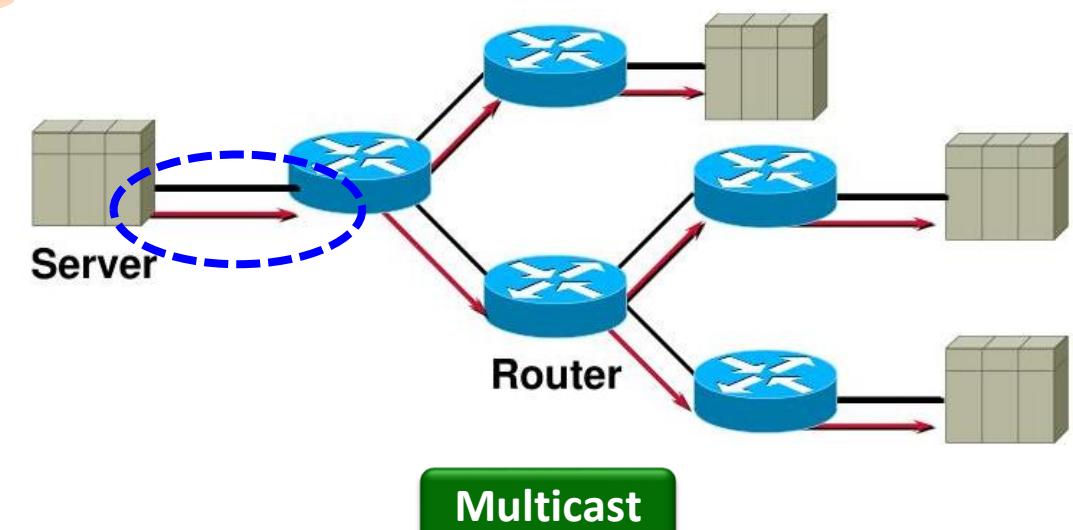
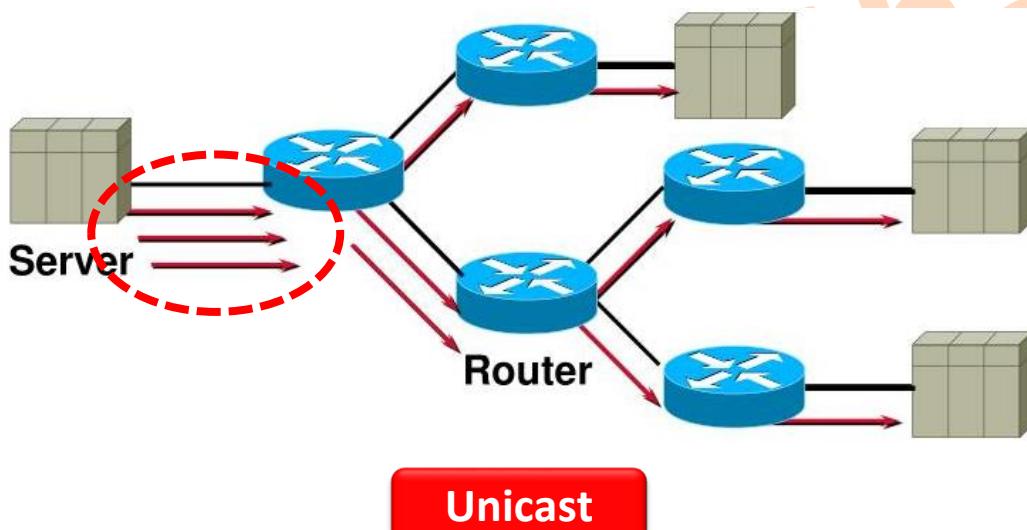
- Broadcast traffic travels from a single source to all destinations on a network (a **broadcast domain**)
 - **Local Broadcast** (255.255.255.255)
 - Send packet to every host on local IP network
 - **Directed Broadcast** (broadcast ip of each network/subnet)
 - Send packet to every host on local IP network
 - Send packet to every host on remote IP network



Type of Addresses (Cont.)

Multicast

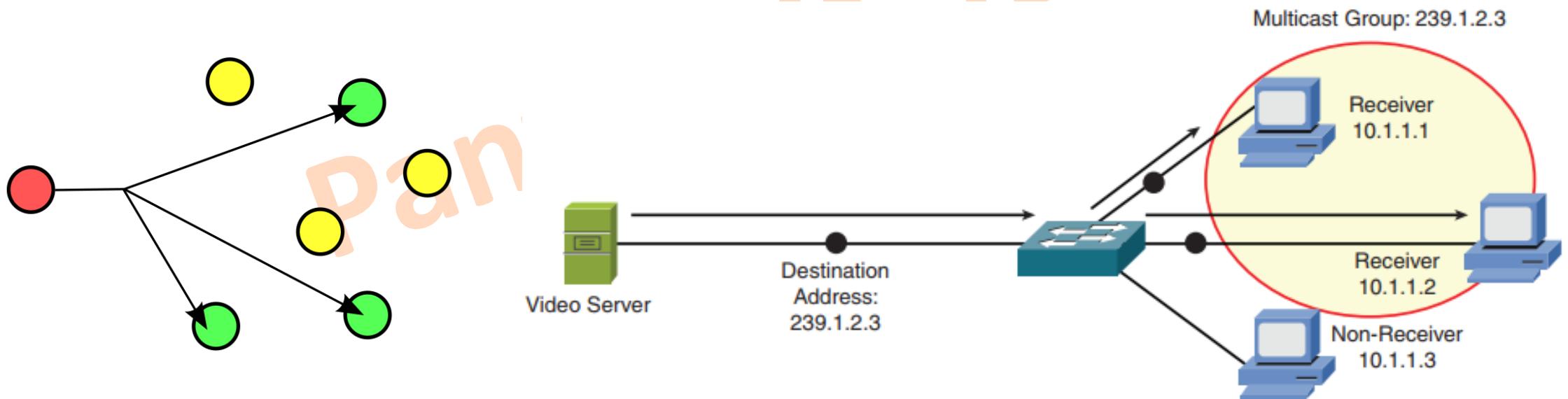
- Multicast technology offers an **efficient mechanism** for a single host to send traffic to multiple specific destinations. For example, a network has 100 users, and 20 of those users want to receive a video stream from a video server.
 - Multicast allowing the video server to send the video stream **only once** and sending the video stream only to devices on the network that want to receive the stream.



Type of Addresses (Cont.)

Multicast

- A Class D address, such as 239.1.2.3, represents the address of a **multicast group**.
 - The video server could send a **single copy** of each video stream packet destined for 239.1.2.3. Devices wanting to receive the video stream could join the multicast group.



Types of IPv4 Addresses

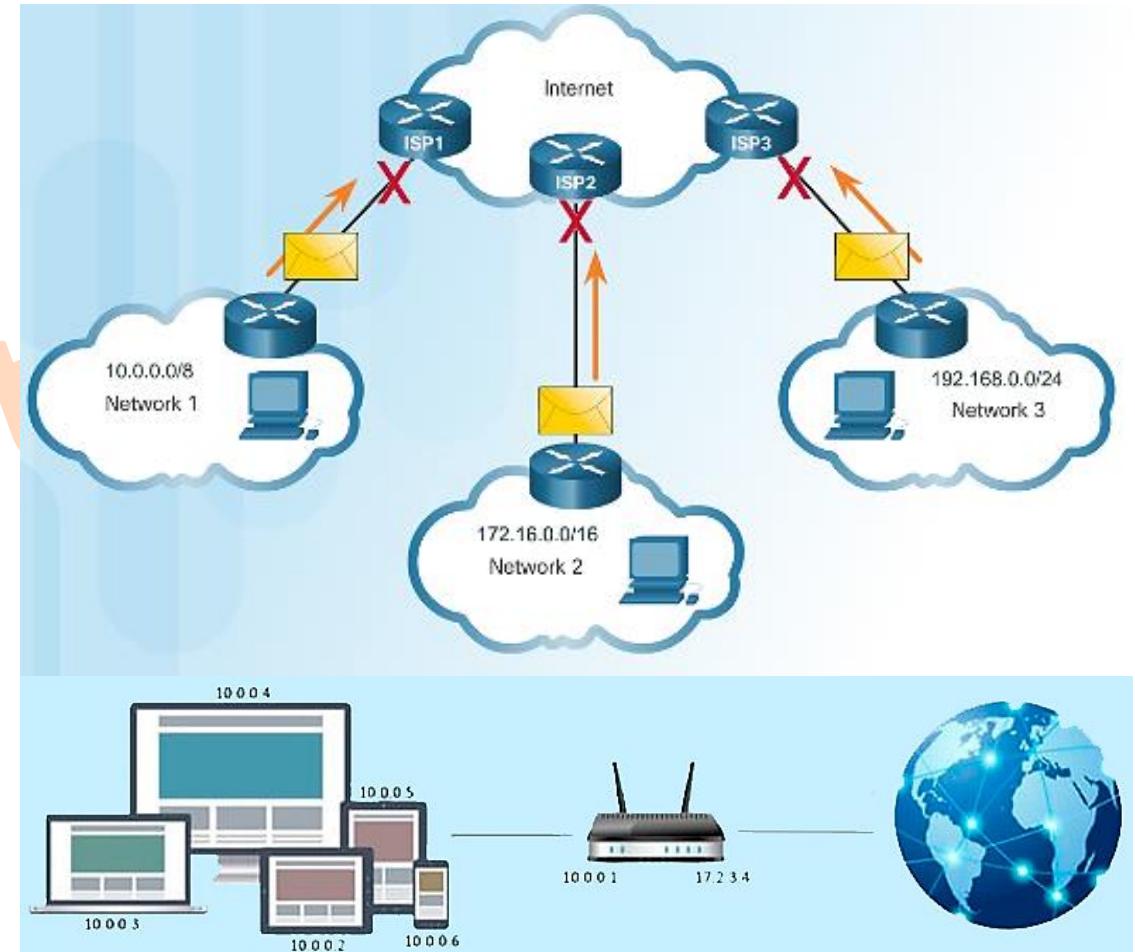
Public and Private IPv4 Addresses

- **Private Addresses**

- Not routable
- Introduced in mid 1990s due to depletion of IPv4 addresses
- Used only in internal networks.
- Must be translated to a public IPv4 to be routable.
- Defined by RFC 1918

- Private Address Blocks

- **10.0.0.0 /8** (10.0.0.0 - 10.255.255.255)
- **172.16.0.0 /12** (172.16.0.0 - 172.31.255.255)
- **192.168.0.0 /16** (192.168.0.0 - 192.168.255.255)



Types of IPv4 Addresses (Cont.)

Special User IPv4 Addresses

- **Loopback addresses**

- **127.0.0.0/8** (127.0.0.1 - 127.255.255.254)
- Commonly identified as only 127.0.0.1
- Used on a host to test if TCP/IP is operational.

```
C:\Users\NetAcad> ping 127.0.0.1
Pinging 127.0.0.1 with 32 bytes of data:
Reply from 127.0.0.1: bytes=32 time<1ms TTL=128
Reply from 127.0.0.1: bytes=32 time<1ms TTL=128
```

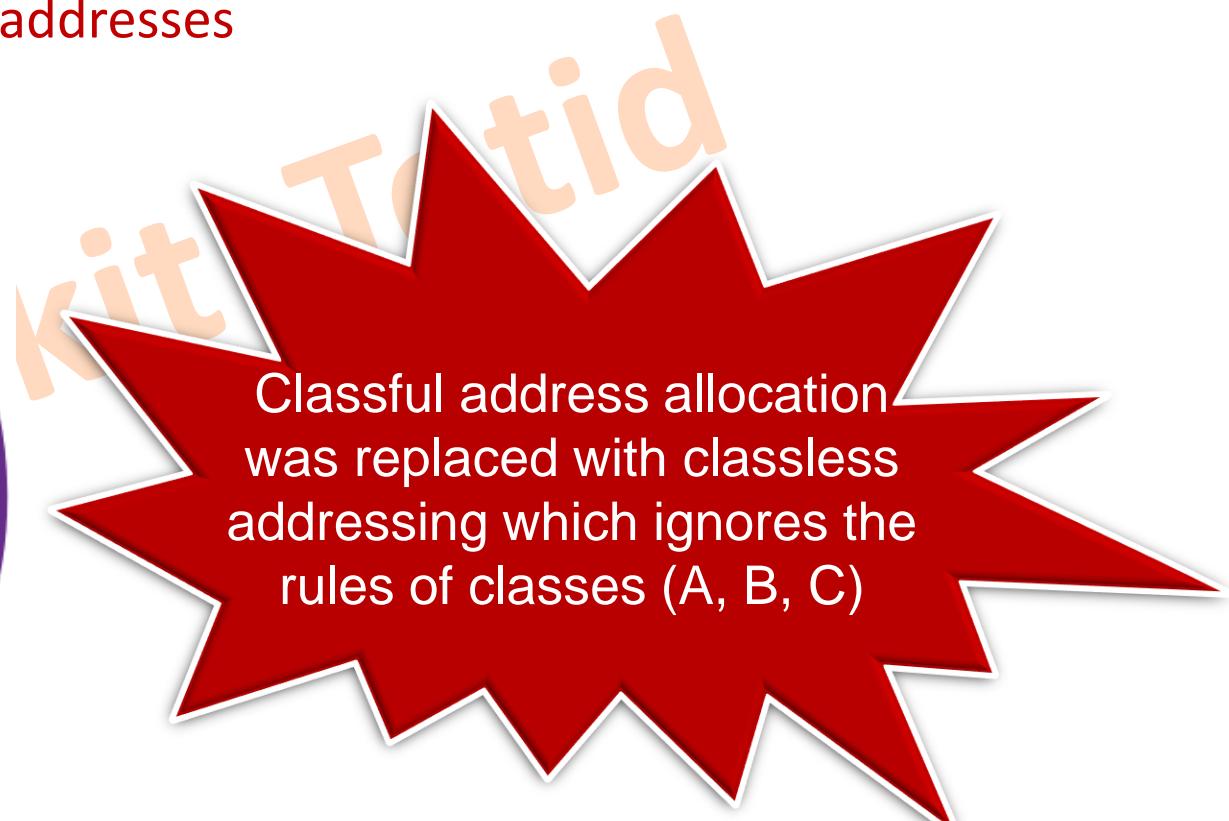
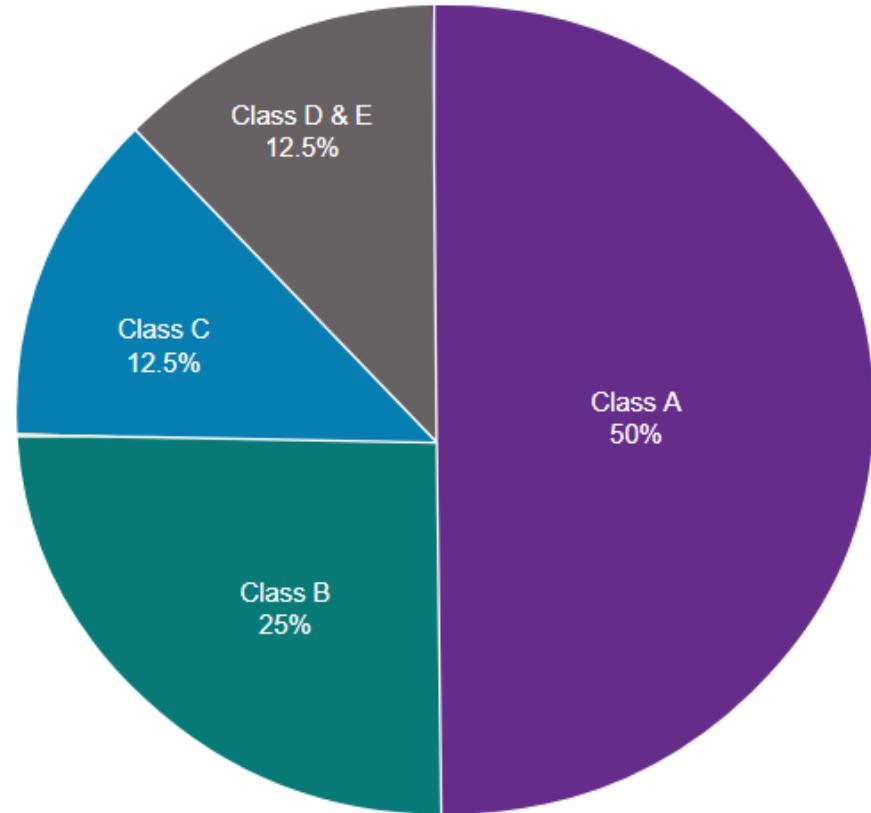
- **Link-Local addresses**

- **169.254.0.0/16** (169.254.0.1 - 169.254.255.254)
- Commonly known as the Automatic Private IP Addressing (APIPA) addresses or self-assigned addresses.
- Used by Windows DHCP clients to self-configure when no DHCP servers are available.

Subnetting

Some issues about legacy classful addressing

- Legacy classful addressing wasted many IPv4 addresses

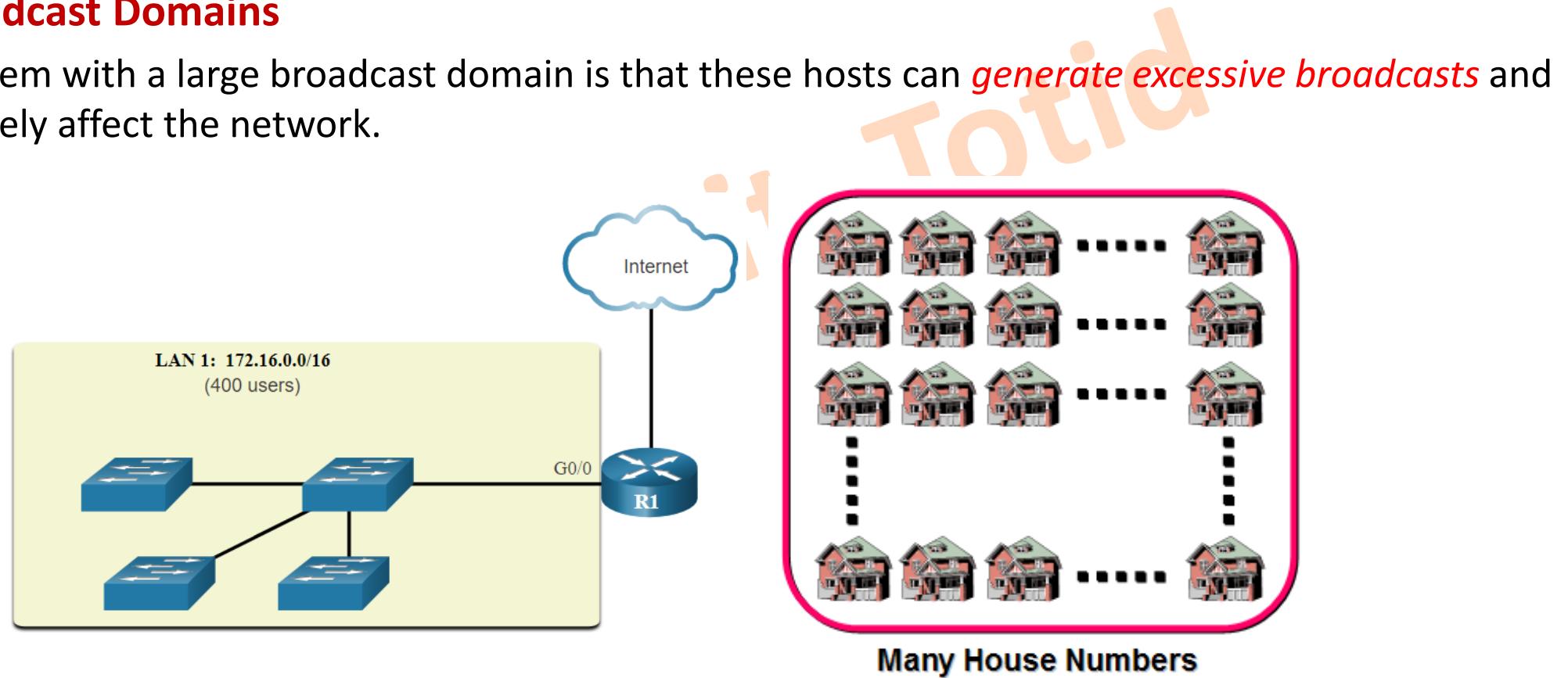


Subnetting (Cont.)

Some issues about legacy classful addressing

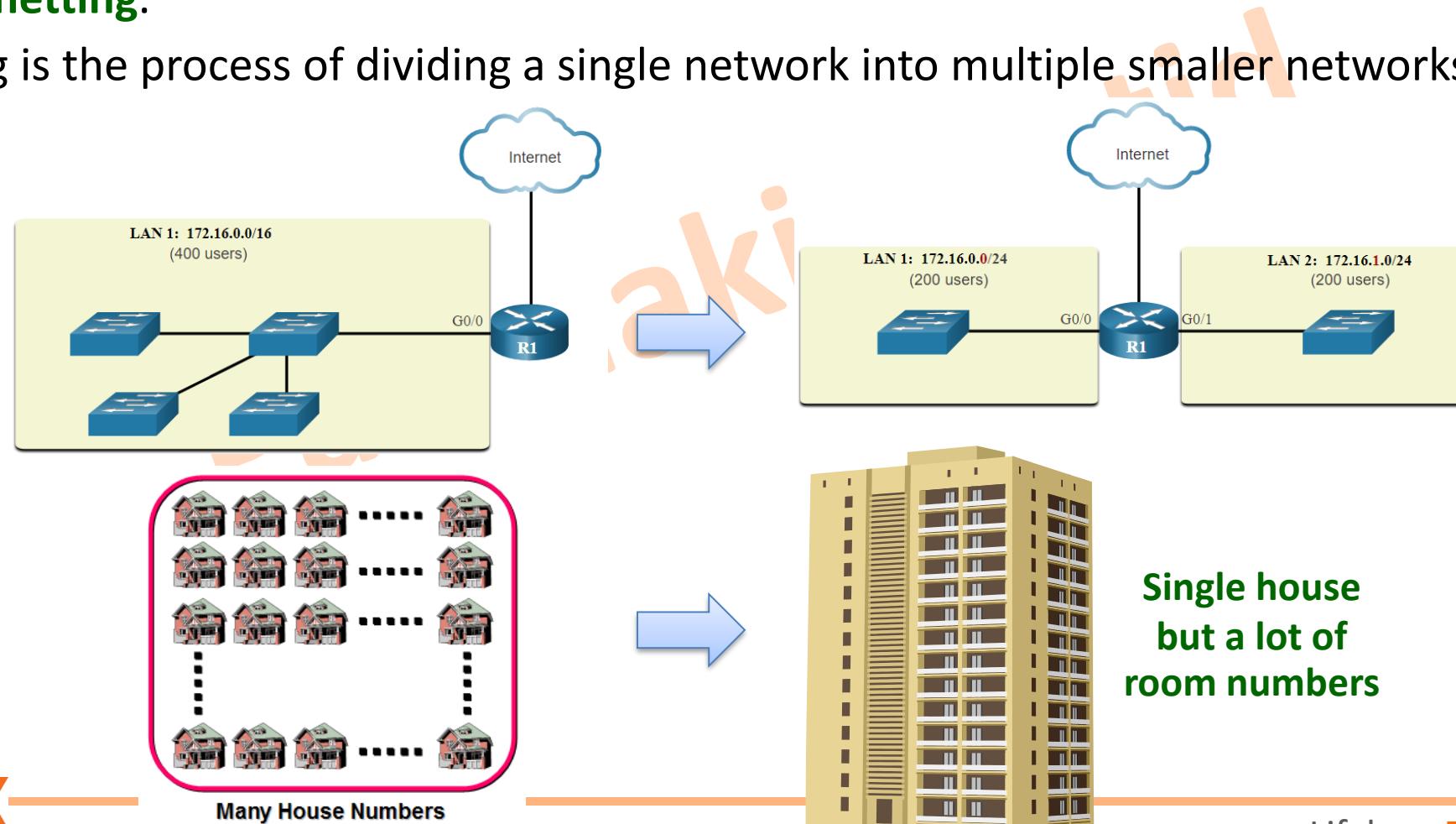
- **Large Broadcast Domains**

- A problem with a large broadcast domain is that these hosts can *generate excessive broadcasts* and negatively affect the network.



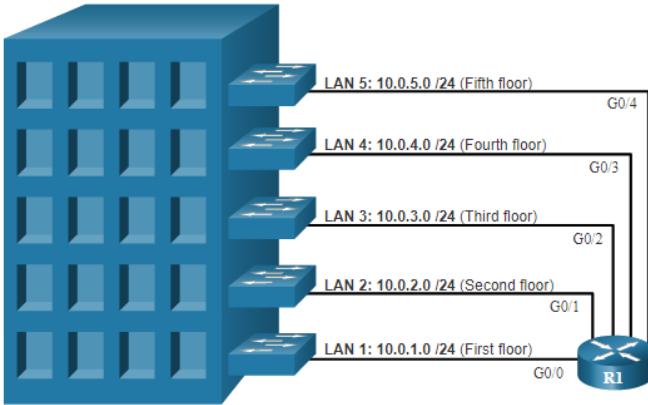
Subnetting (Cont.)

- The solution is to reduce the size of the network to create *smaller broadcast domains* in a process called **subnetting**.
- Subnetting is the process of dividing a single network into multiple smaller networks.

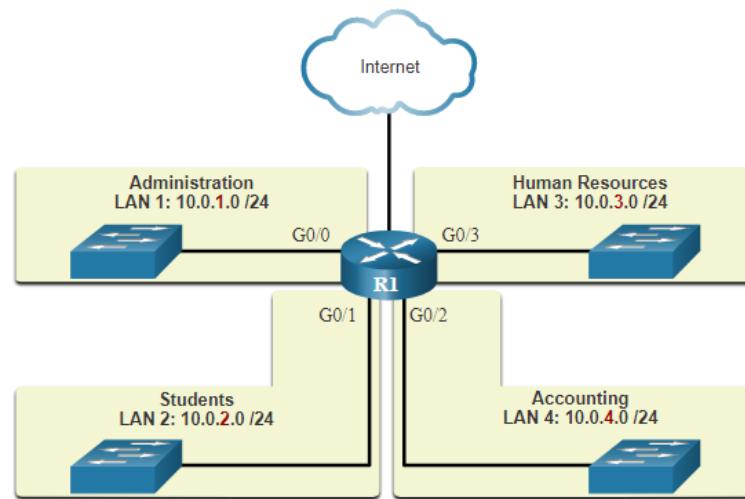


Subnetting (Cont.)

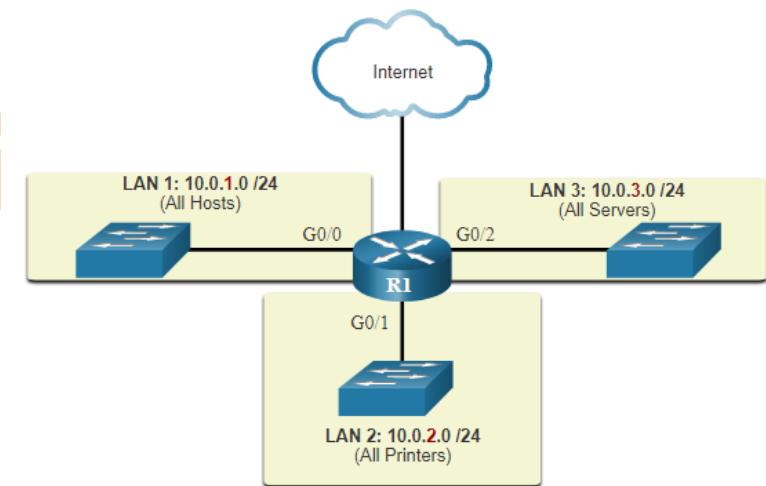
- Subnets are used for a variety of reasons including by:



Location



Group / Function



Device Type

Subnetting (Cont.)

- Subnetting is done based on these two requirements:

**Requirement of
Subnets**

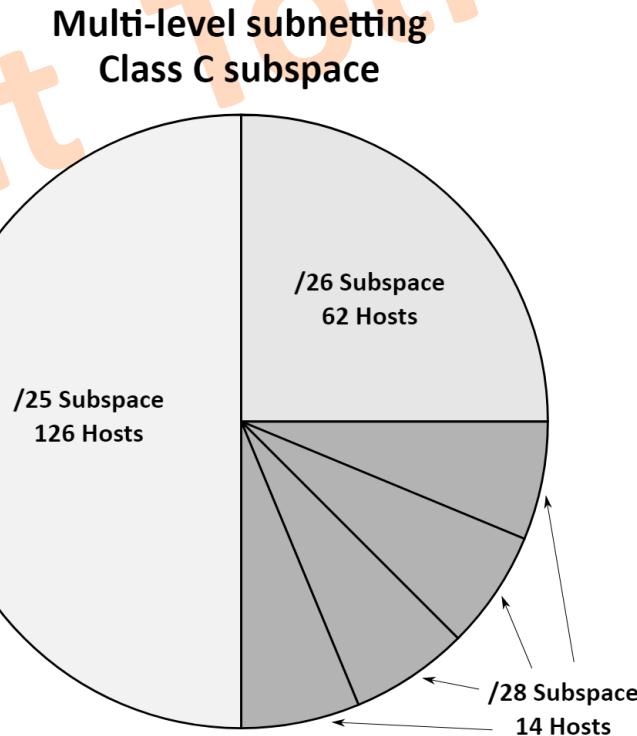
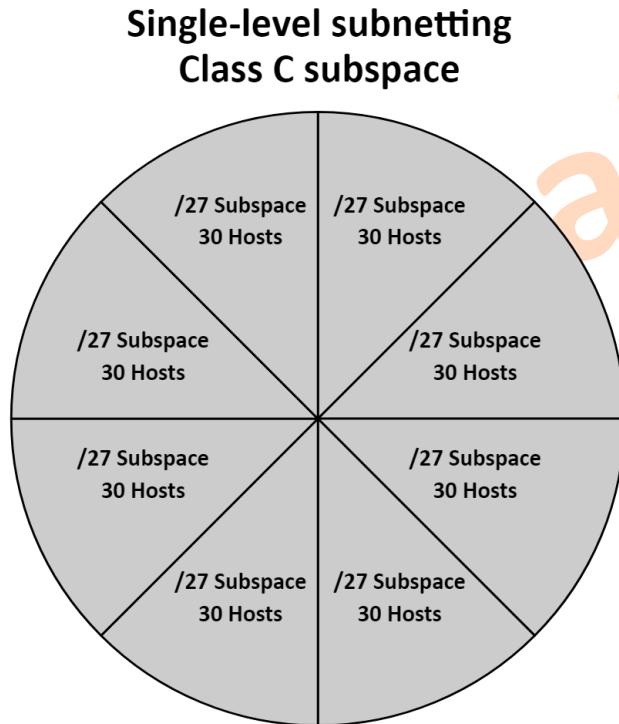
$2^N \geq \text{Requirement}$, N is Network bits required

**Requirement of
Hosts per Subnet**

$2^H - 2 \geq \text{Requirement}$, H is Host bits required

Subnetting (Cont.)

- Subnetting can be performed in two ways:
 - **FLSM (Fixed Length Subnet Mask)**
 - **VLSM (Variable Length Subnet Mask)**



Subnetting (Cont.)

- **Question 5:** Assume ABC company has a following block of addresses **192.168.100.0/24**, and this company decides to divide this block into **4 subnets**. Determine the network and broadcast address and the usable addresses for each subnets.

- **192.168.100.0/24** → 192.168.100.**0** → 8 host bits (192.168.100.**NN** **HHHHHH**)
255.255.255.**0**
- **4 Subnet required** → Need **2 network bits** (borrowed bits)
- 8 host bits - 2 network bits = 6 host bits left

192.168.100.**NN** **HHHHHH**

192.168.100.**00** **HHHHHH**

192.168.100.**01** **HHHHHH**

192.168.100.**10** **HHHHHH**

192.168.100.**11** **HHHHHH**

New Subnet Mask for each subnet:

255.255.255.11000000

255.255.255.192

Subnetting (Cont.)

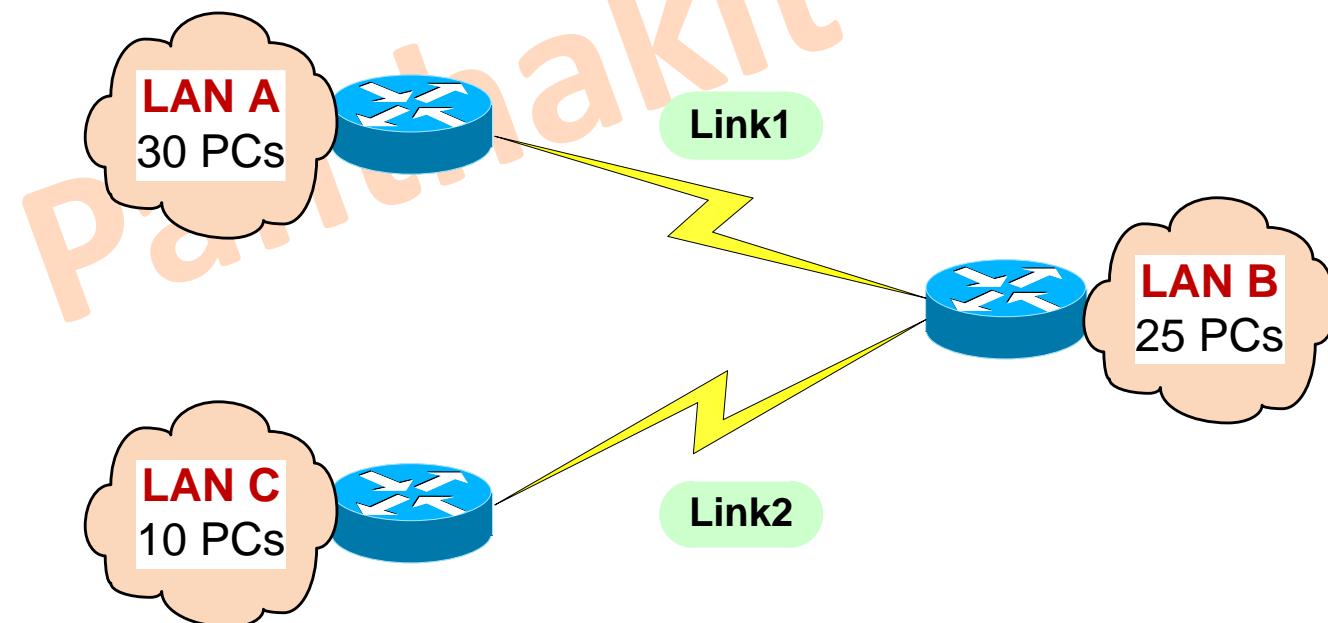
- Question 5:

	Network Add & Broadcast Add	Host (Usable) Address Range
1 st Subnet - 192.168.100. 00 000000 111111	192.168.100.0 192.168.100.63	192.168.100.1 - 192.168.100.62
2 nd Subnet - 192.168.100. 01 000000 111111	192.168.100.64 192.168.100.127	192.168.100.65 - 192.168.100.126
3 rd Subnet - 192.168.100. 10 000000 111111	192.168.100.128 192.168.100.191	192.168.100.129 - 192.168.100.190
4 th Subnet - 192.168.100. 11 000000 111111	192.168.100.192 192.168.100.255	192.168.100.193 - 192.168.100.254

There are 5 host bits left → $2^6 - 2 = 64 - 2 = 62$ IPs/Subnet

Subnetting (Cont.)

- **Question 6:** An organization is granted the block **192.168.90.0/24**. The administrator wants to create subnets from the following network diagram. For each subnet, Find
 - Network address (or Subnet address)
 - Broadcast address
 - Subnet mask (in dotted decimal and slash notations).



Subnetting (Cont.)

- **Question 7:**

- Network Address - **120.88.0.0/16**
- Number of needed usable hosts: **500 hosts/subnet**

Number of bits borrowed : _____

Maximum number of subnets can created : _____

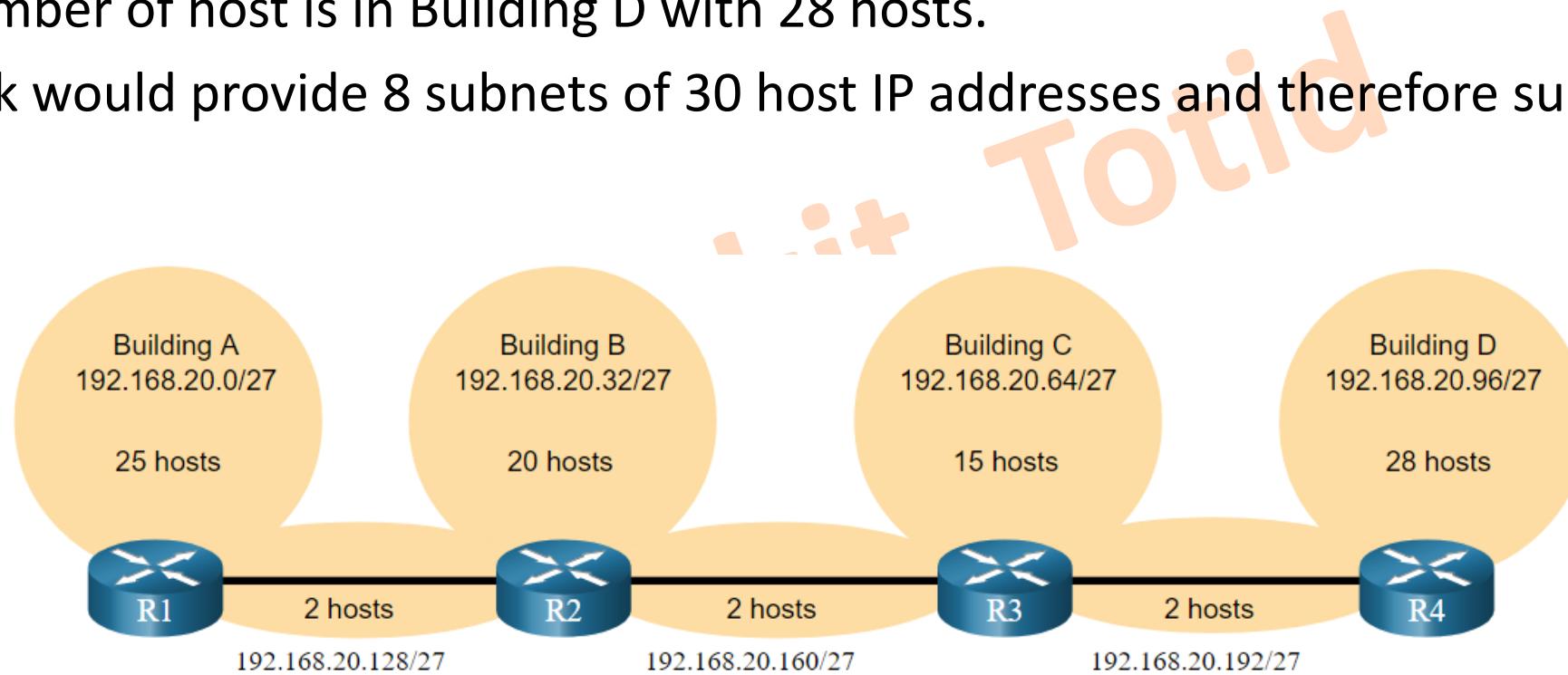
Total number of address/subnet : _____

Total number of hosts/subnet : _____

New Subnet Mask : _____

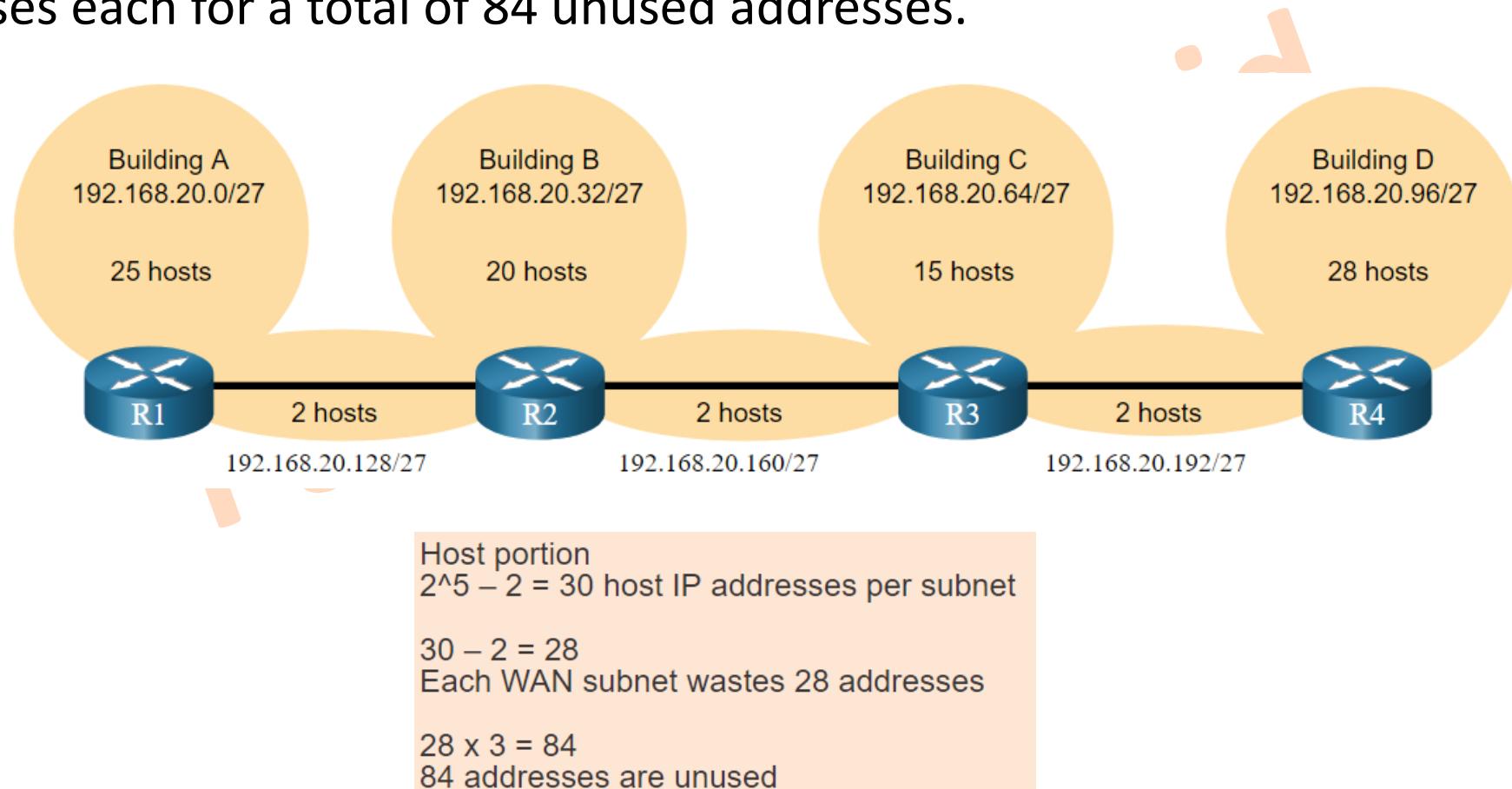
Subnetting Issue

- Given the topology, **7 subnets** are required (i.e, **four LANs** and **three WAN links**) and the largest number of host is in Building D with 28 hosts.
- A /27 mask would provide 8 subnets of 30 host IP addresses and therefore support this topology.



Subnetting Issue (Cont.)

- However, the **point-to-point WAN links** only **require two addresses** and therefore waste 28 addresses each for a total of 84 unused addresses.



Implementing VLSM

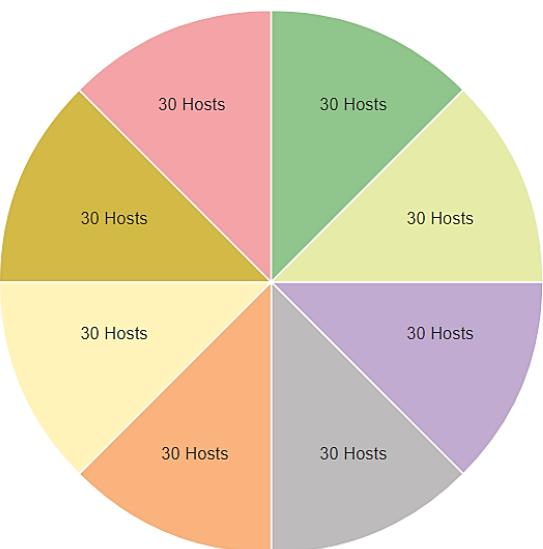
- Applying a traditional subnetting scheme to this scenario is not very efficient and is wasteful.
 - It is a “**one-size-fits-all**” design.
- VLSM is the process of “**subnetting a subnet**” and using different subnet masks for different networks in your IP plan.

What you have to remember is that you need to make sure that there is no overlap in any of the addresses.

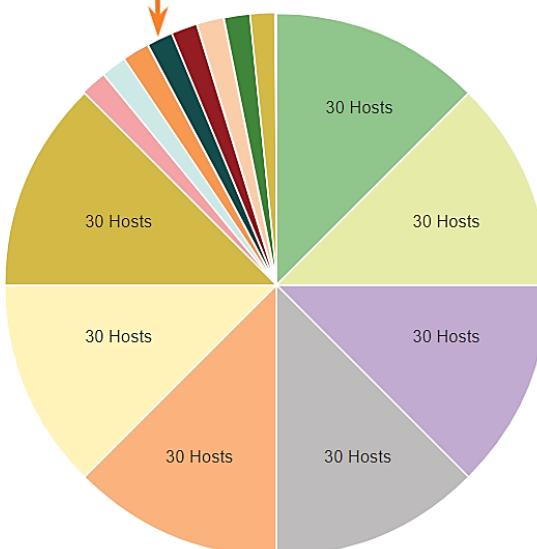
Implementing VLSM (Cont.)

- When using VLSM, always begin by satisfying the host requirements of the largest subnet and continue subnetting until the host requirements of the smallest subnet are satisfied.

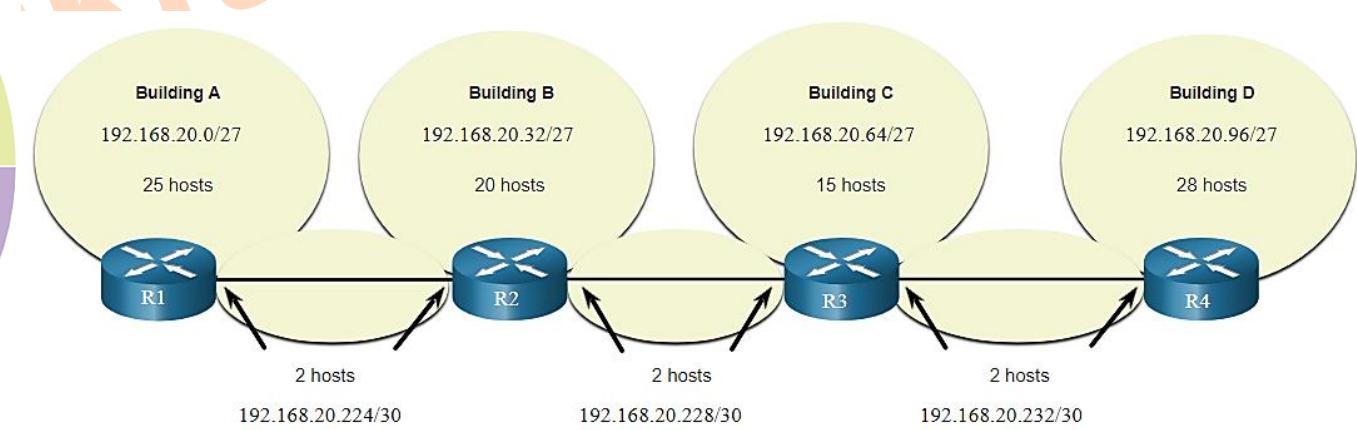
Traditional Subnetting Creates Equal Sized Subnets



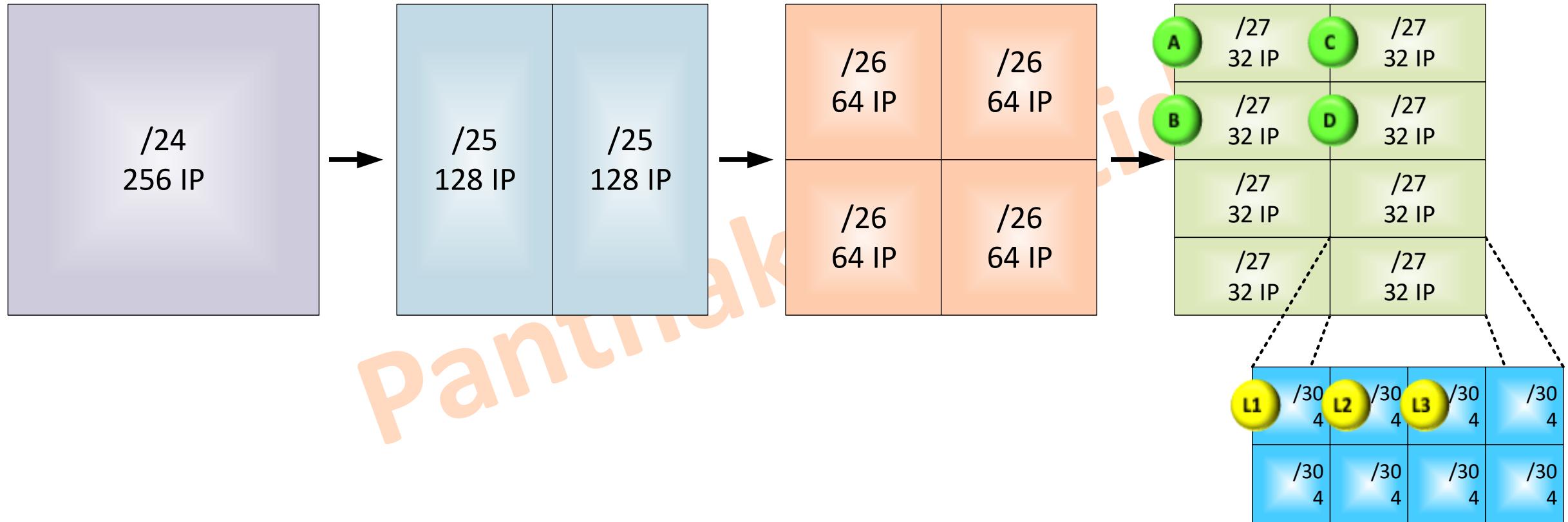
Subnets of Varying Sizes



The resulting topology with VLSM applied.



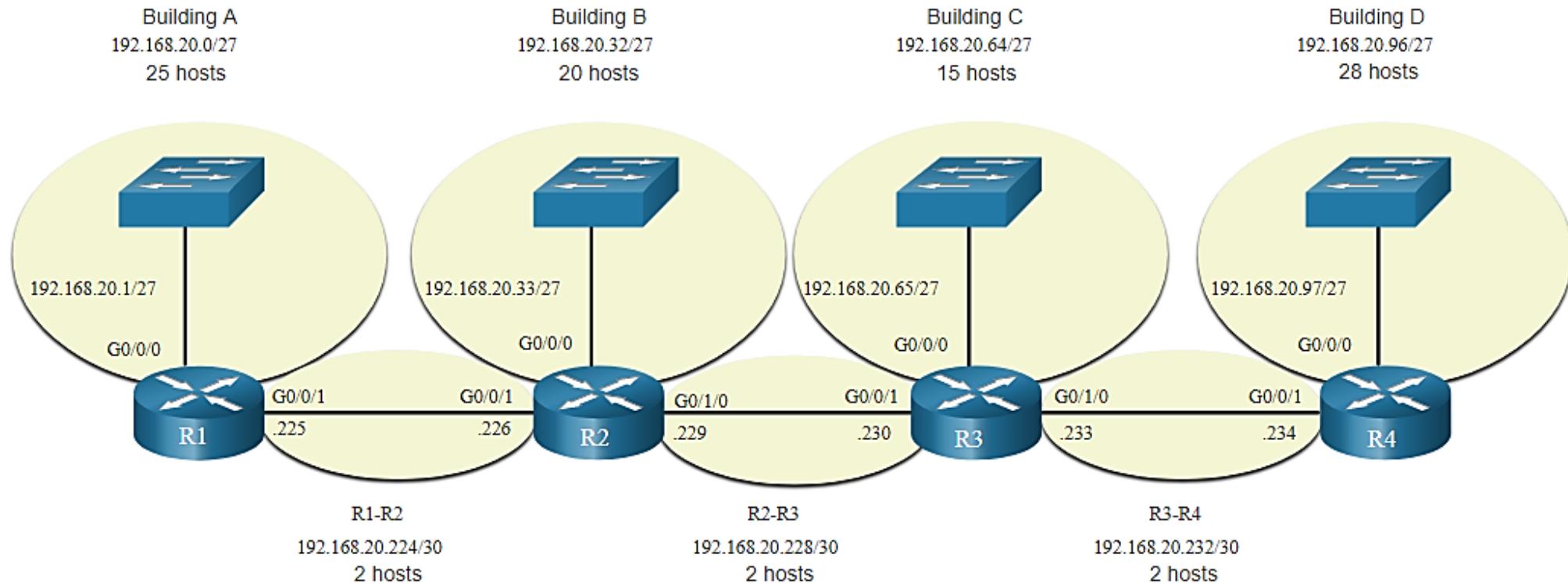
Implementing VLSM (Cont.)



Implementing VLSM (Cont.)

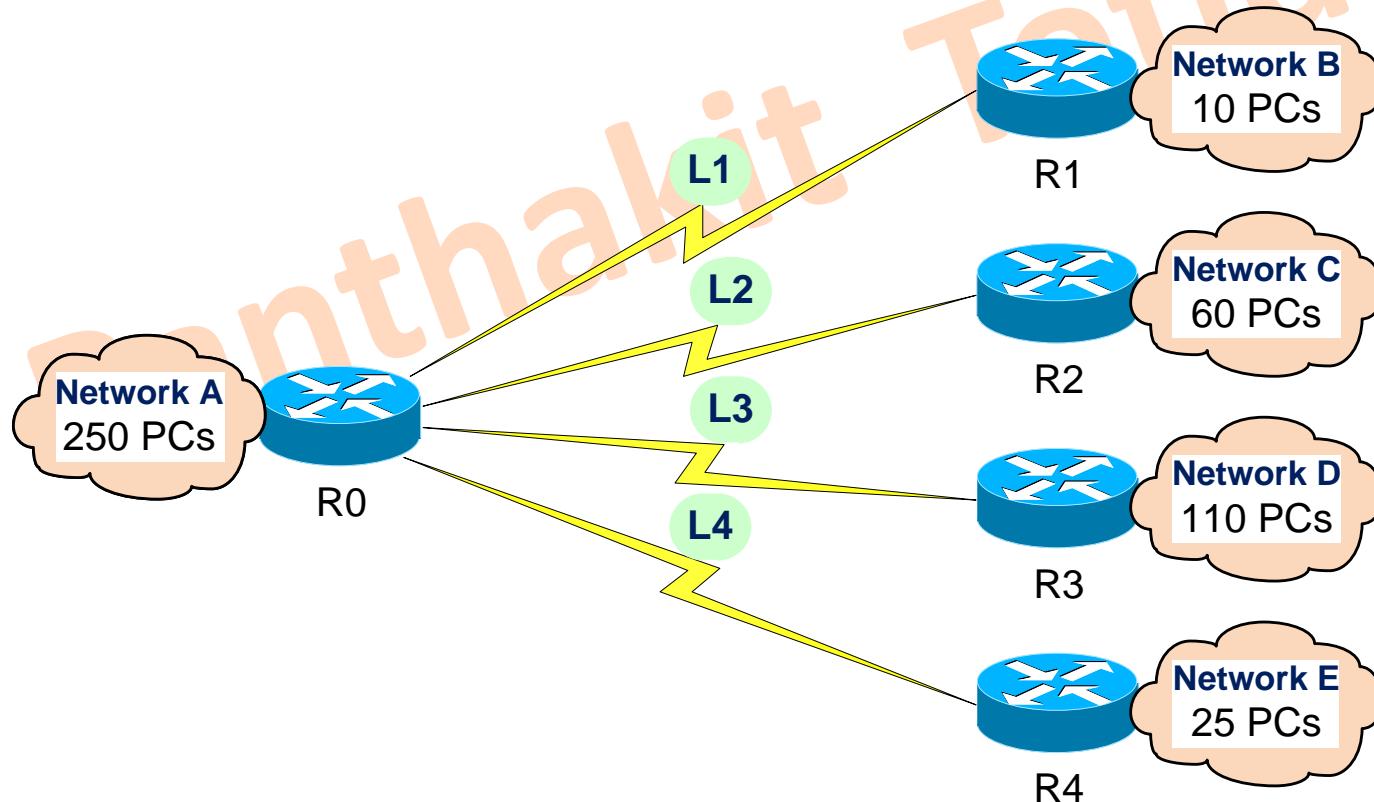
- **VLSM Topology Address Assignment**

- Using VLSM subnets, the LAN and inter-router networks can be addressed without unnecessary waste as shown in the logical topology diagram.



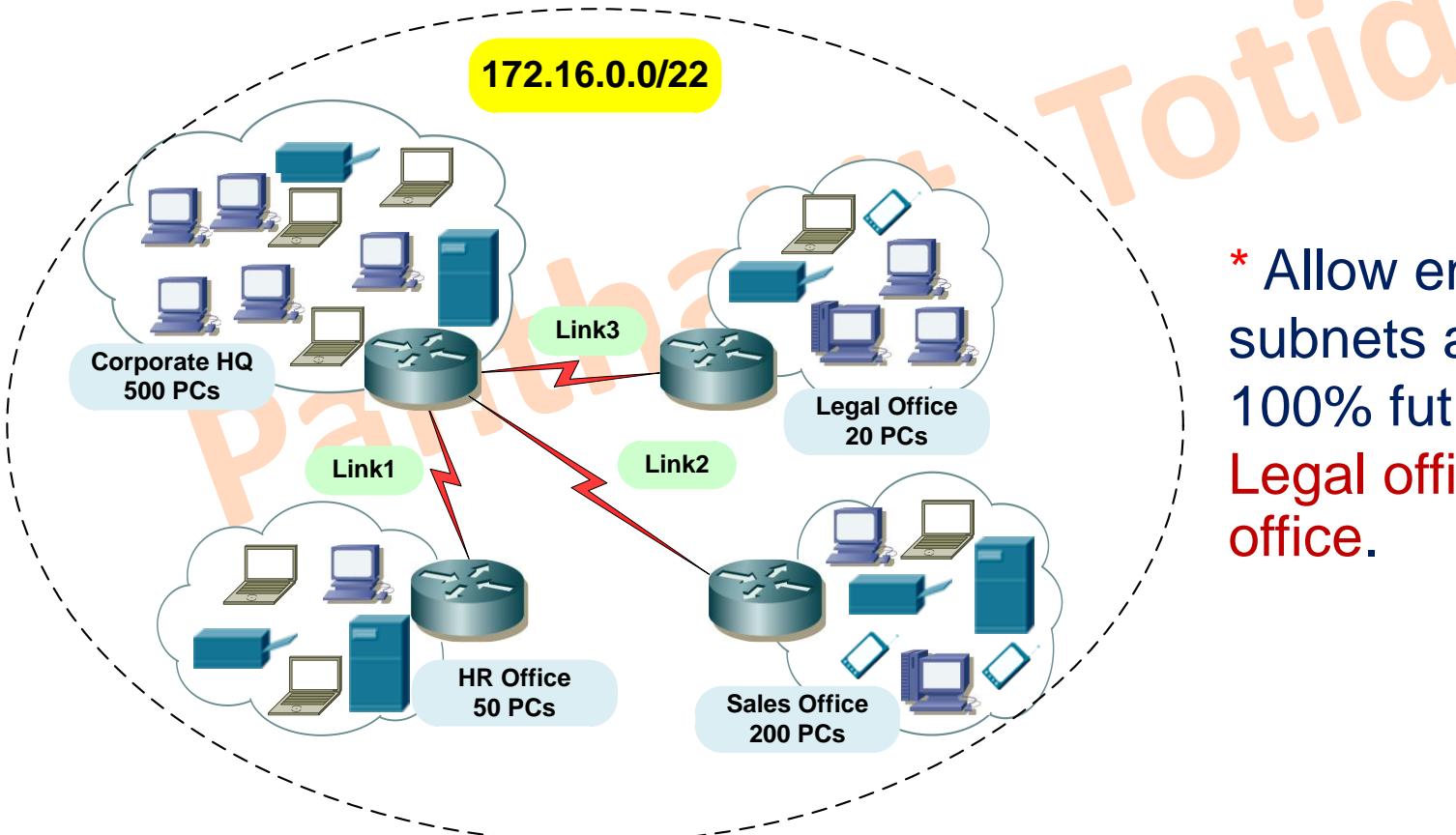
Question 8

- From the following network diagram. Given a block of addresses starting with **190.199.50.0/23**. Design the subblocks and find **network address**, **broadcast address**, and **subnet mask** (in dotted-decimal and slash notation) for each subnet.



Question 9

- From the following network diagram. Design the subblocks and find **network address**, **broadcast address**, and **subnet mask** (in dotted-decimal and slash notation) for each subnet.



* Allow enough extra subnets and host for 100% future growth in Legal office and HR office.

Question 10

- Find the **network address**, the **broadcast address**, the **host addresses**, the **subnet mask** (in dotted-decimal notation) and the **number of IP addresses** of the following IP addresses.
- A. 205.16.37.142/28
 - B. 123.56.77.32/10
 - C. 172.16.67.30/22
 - D. 169.234.16.0/14
 - E. 192.82.220.144/18
 - F. 192.82.220.144/21
 - G. 192.82.220.144/23
 - H. 192.82.220.144/25
 - I. 192.82.220.144/29

Panthakit Totid



Basic Network For Trainee

Module 5

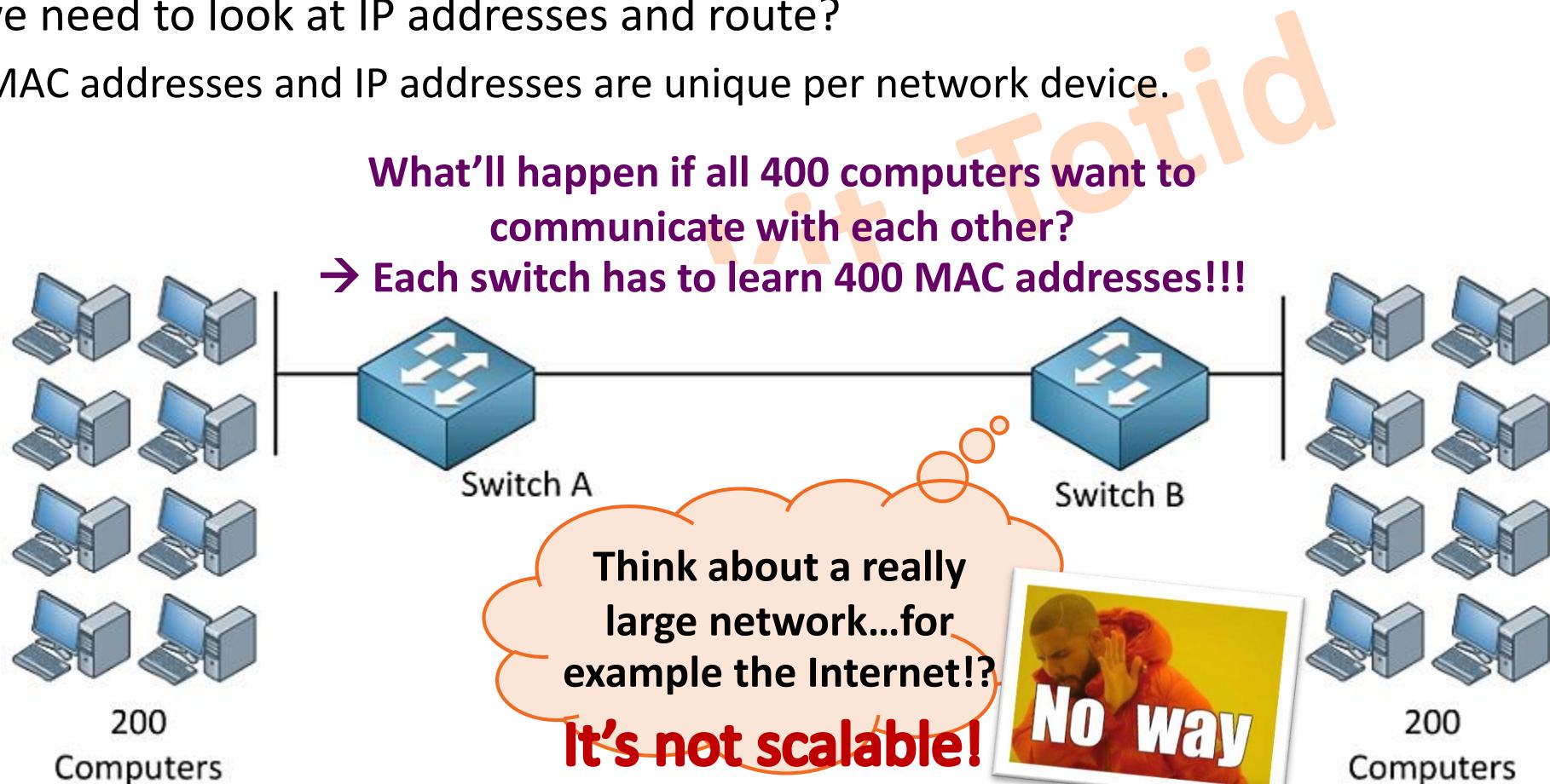
Function of Routing

Panthakit Totid



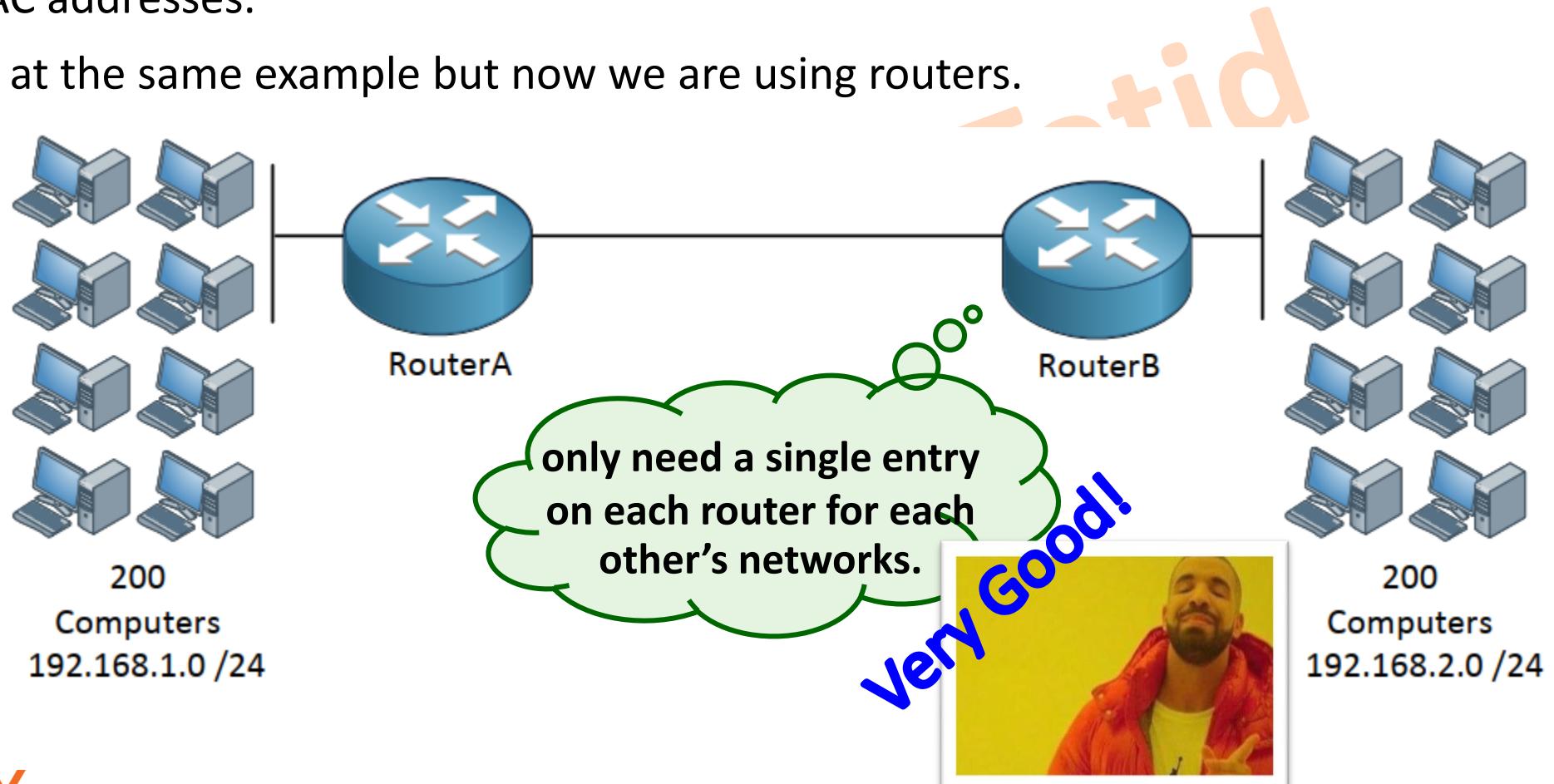
Why We Need Routing

- Why don't we use MAC addresses everywhere and switch?
- Why do we need to look at IP addresses and route?
 - Both MAC addresses and IP addresses are unique per network device.



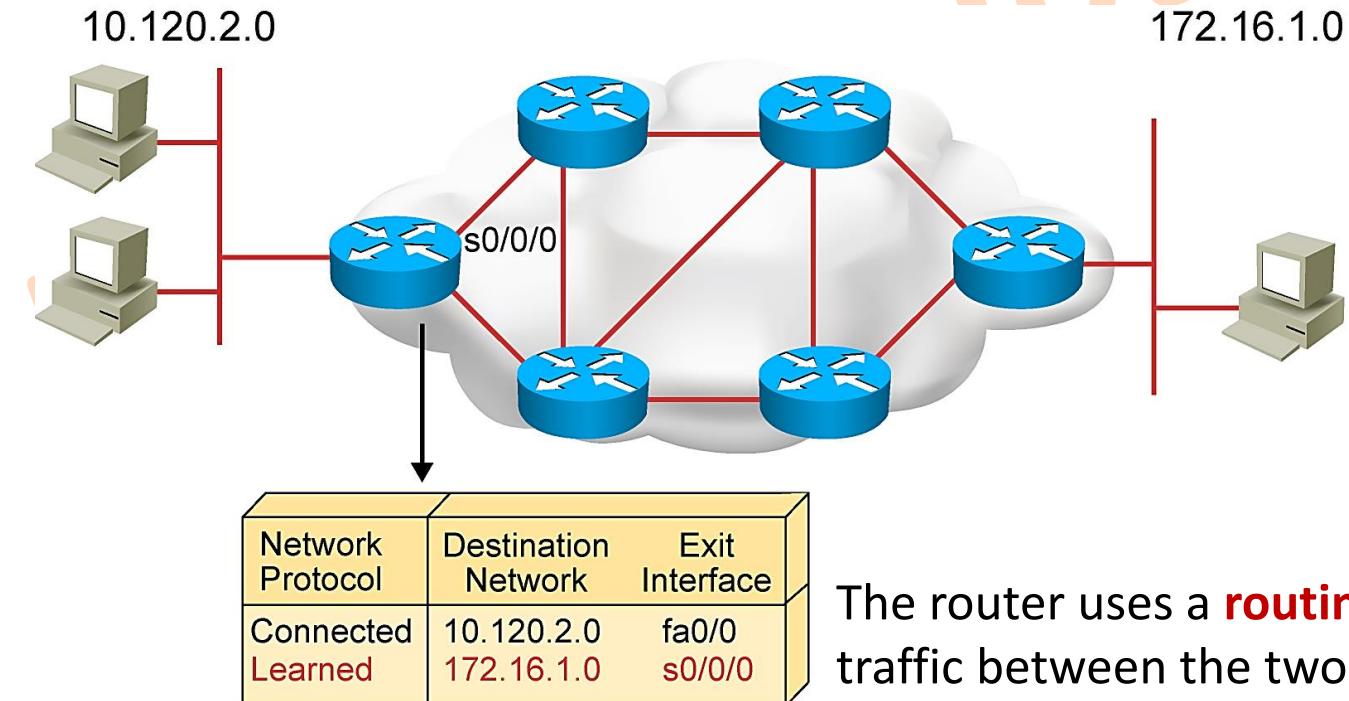
Why We Need Routing (Cont.)

- The problem with switching is that it's not scalable; we don't have any hierarchy just flat 48-bit MAC addresses.
- Let's look at the same example but now we are using routers.



Role of a Router

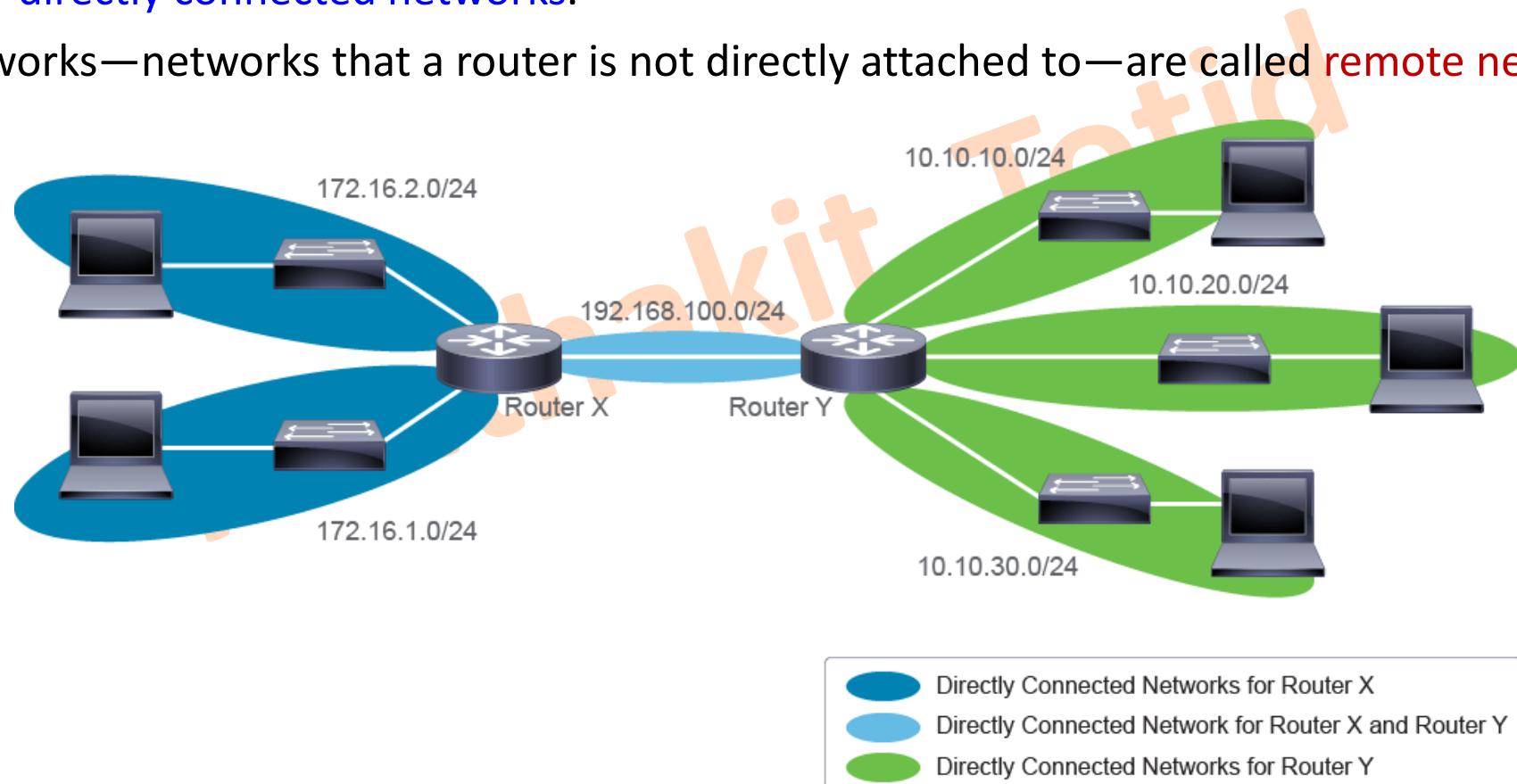
- A router is a networking device that forwards packets between different networks.
- While switches exchange data frames between segments to enable communication within a single network, routers are required to reach hosts that are not in the same network. Routers enable internetwork communication by connecting interfaces in multiple networks.



The router uses a **routing table** to route traffic between the two networks.

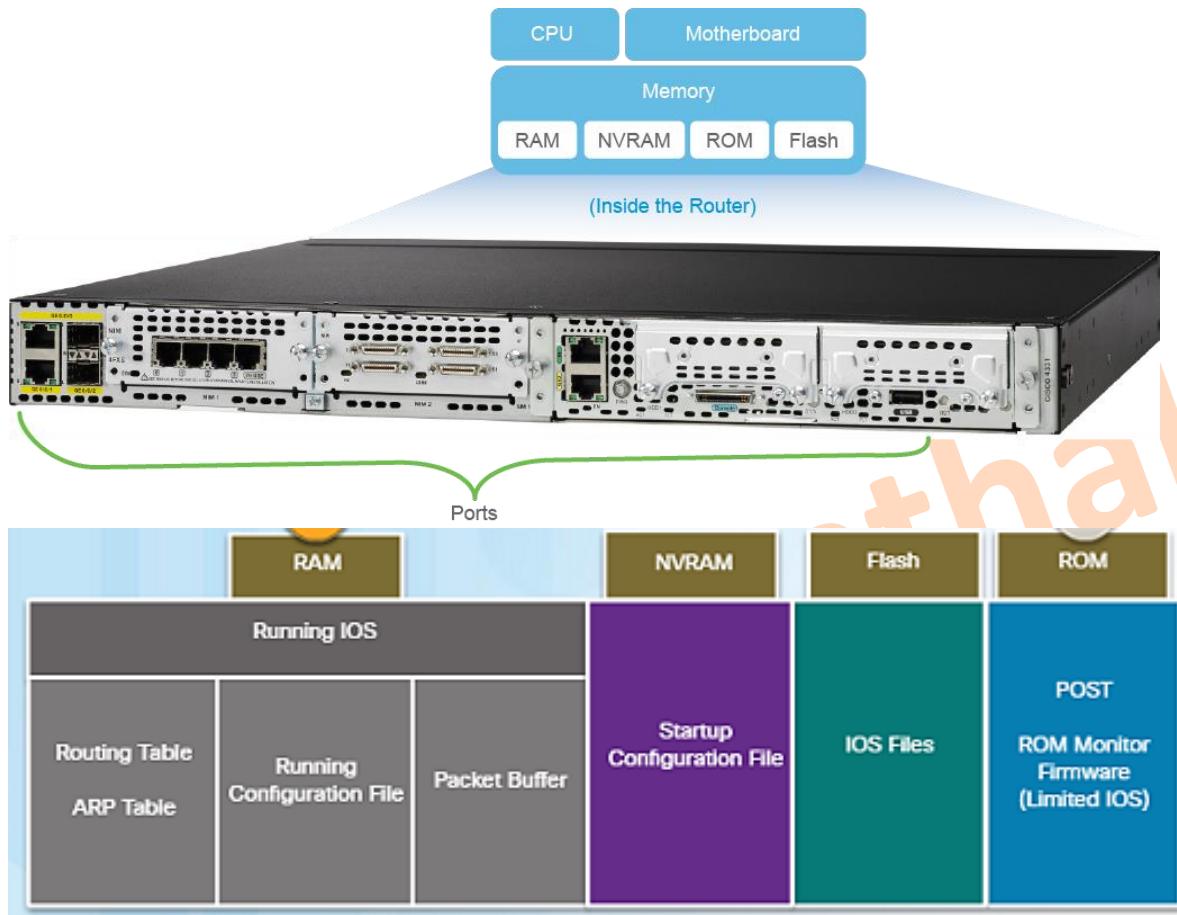
Role of a Router (Cont.)

- This figure illustrates another important routing concept. Networks to which the router is attached are called **local** or **directly connected networks**.
- All other networks—networks that a router is not directly attached to—are called **remote networks**.



Anatomy of a Router

Router's Memory

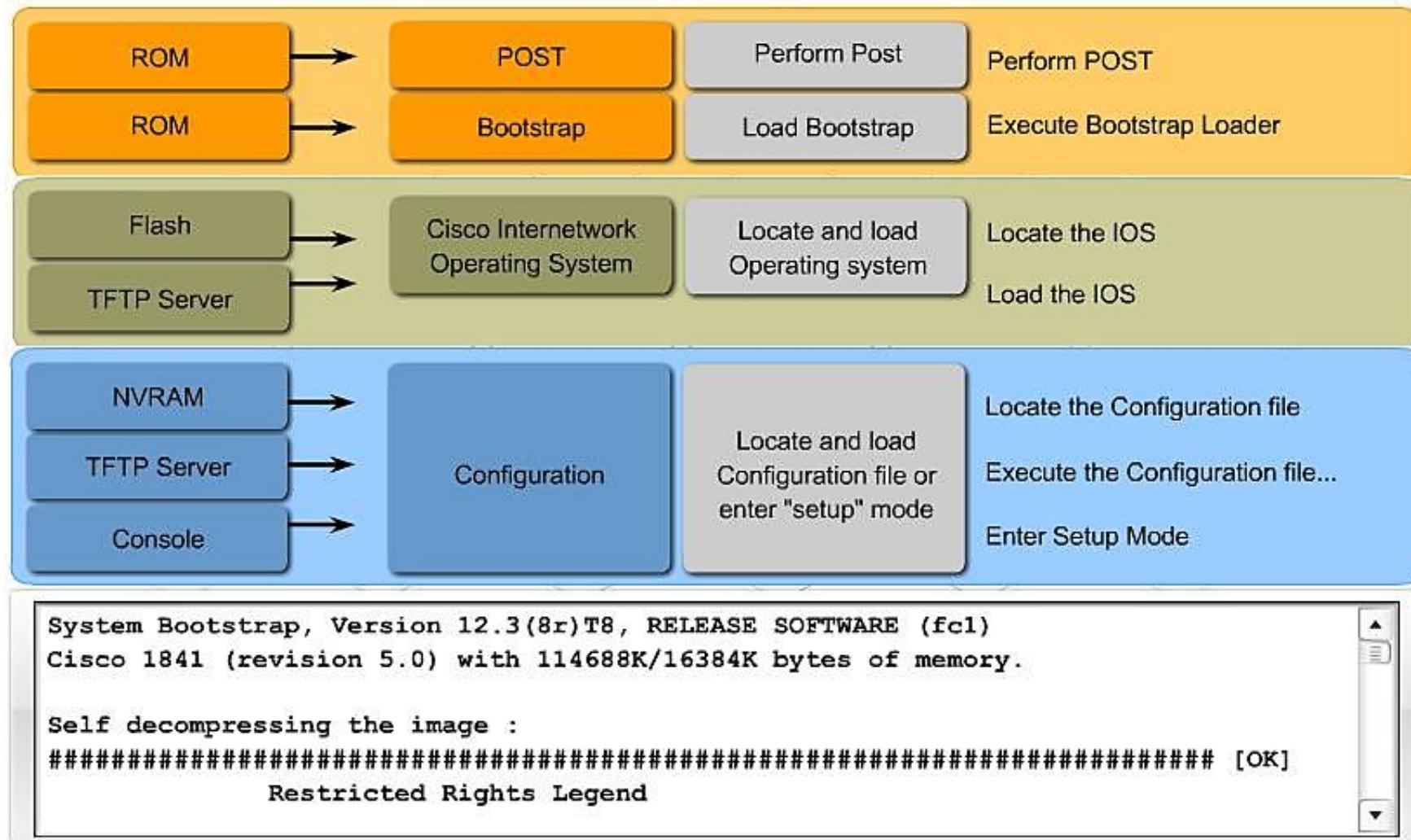


A router uses four types of memory:

- **RAM** - Volatile memory used to store applications, processes, and data needed to be executed by the CPU.
- **ROM** - Non-volatile memory used to store crucial operational instructions and a limited IOS. ROM is firmware embedded on an integrated circuit inside of the router.
- **NVRAM** - Non-volatile memory used as permanent storage for the startup configuration file.
- **Flash** - Non-volatile memory used as permanent storage for the IOS and other operating system files such as log or backup files.

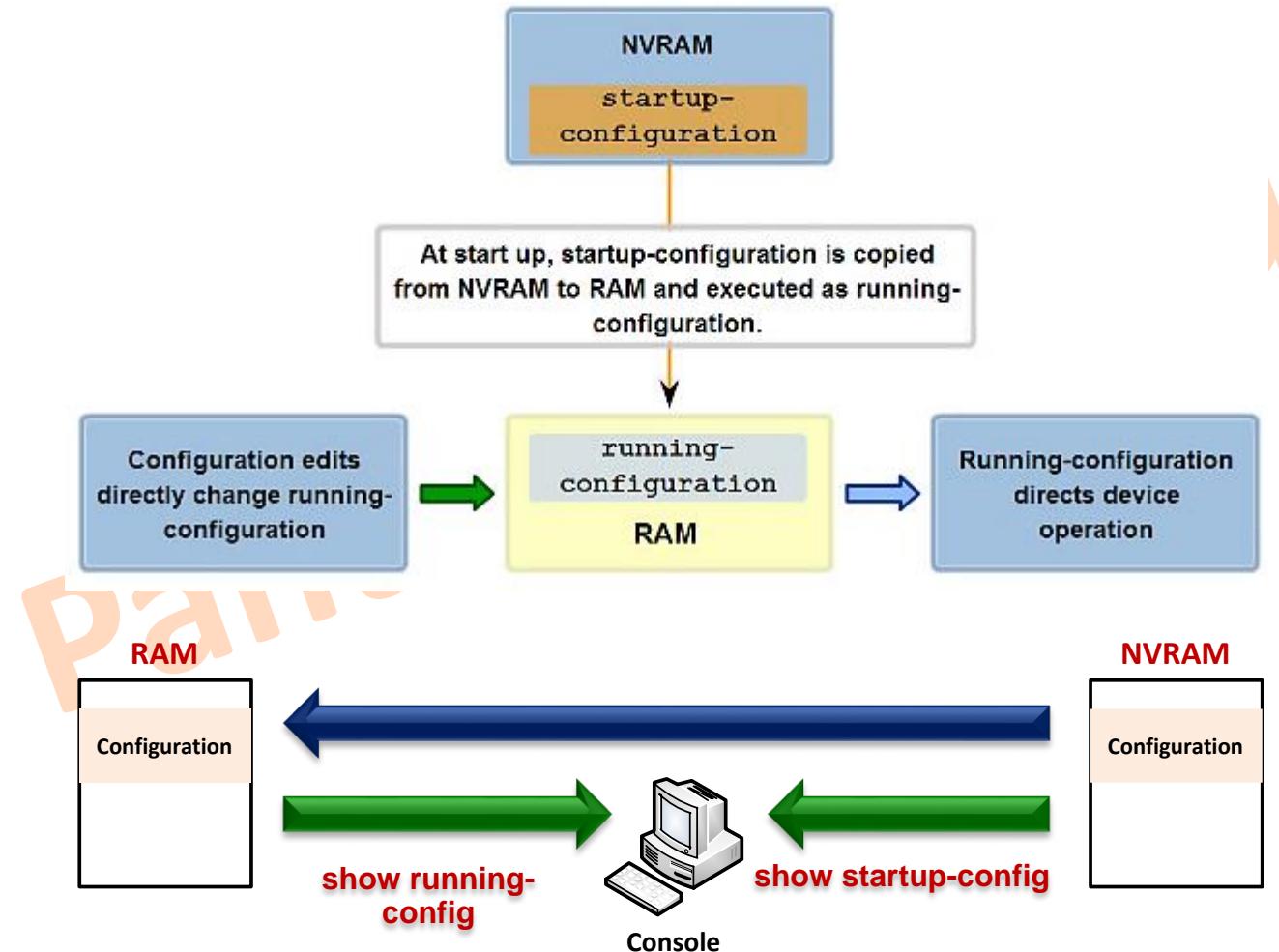
Anatomy of a Router (Cont.)

Router Bootup Process



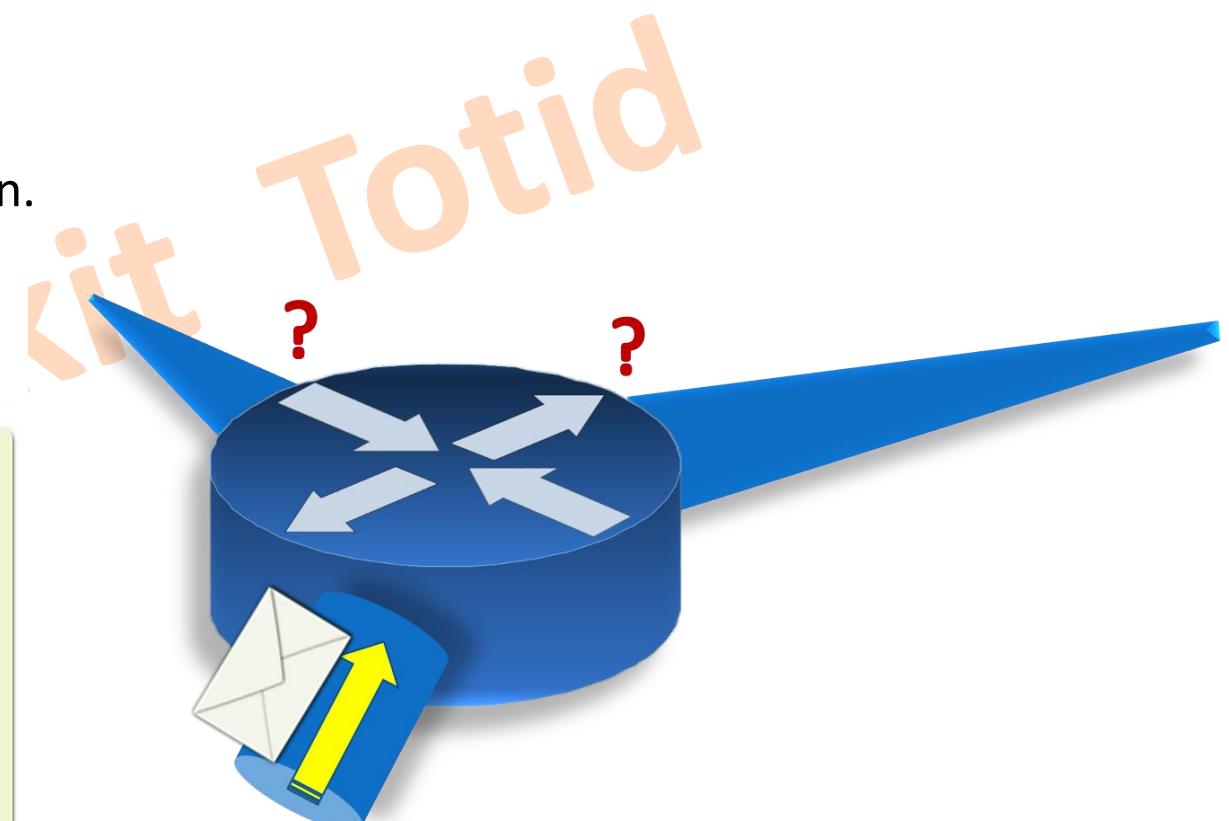
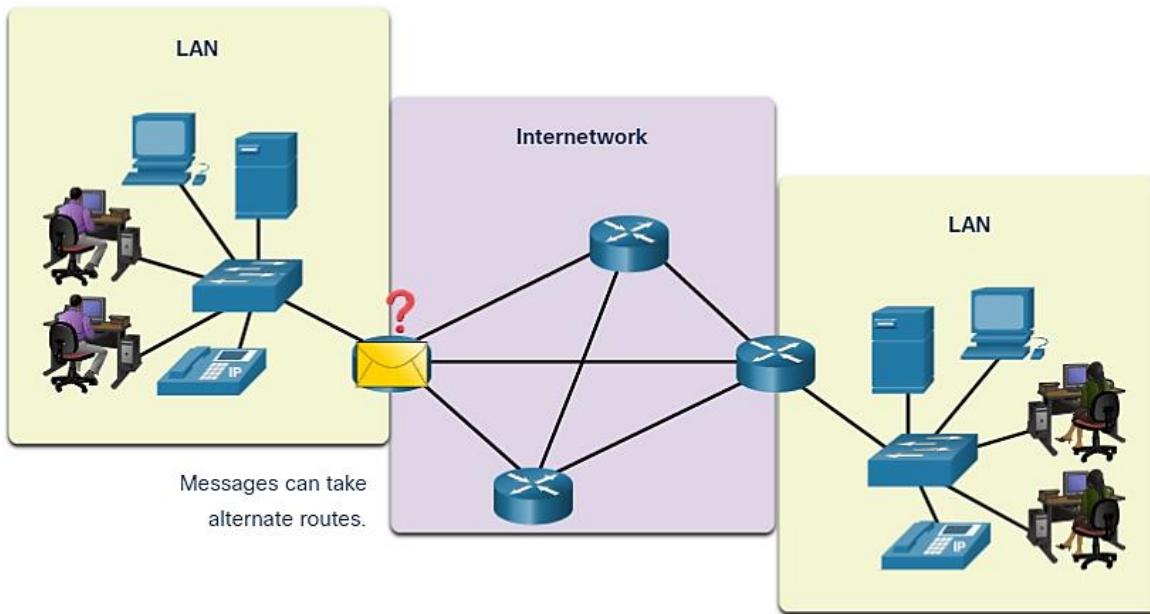
Anatomy of a Router (Cont.)

Configuration File



Two Functions of a Router

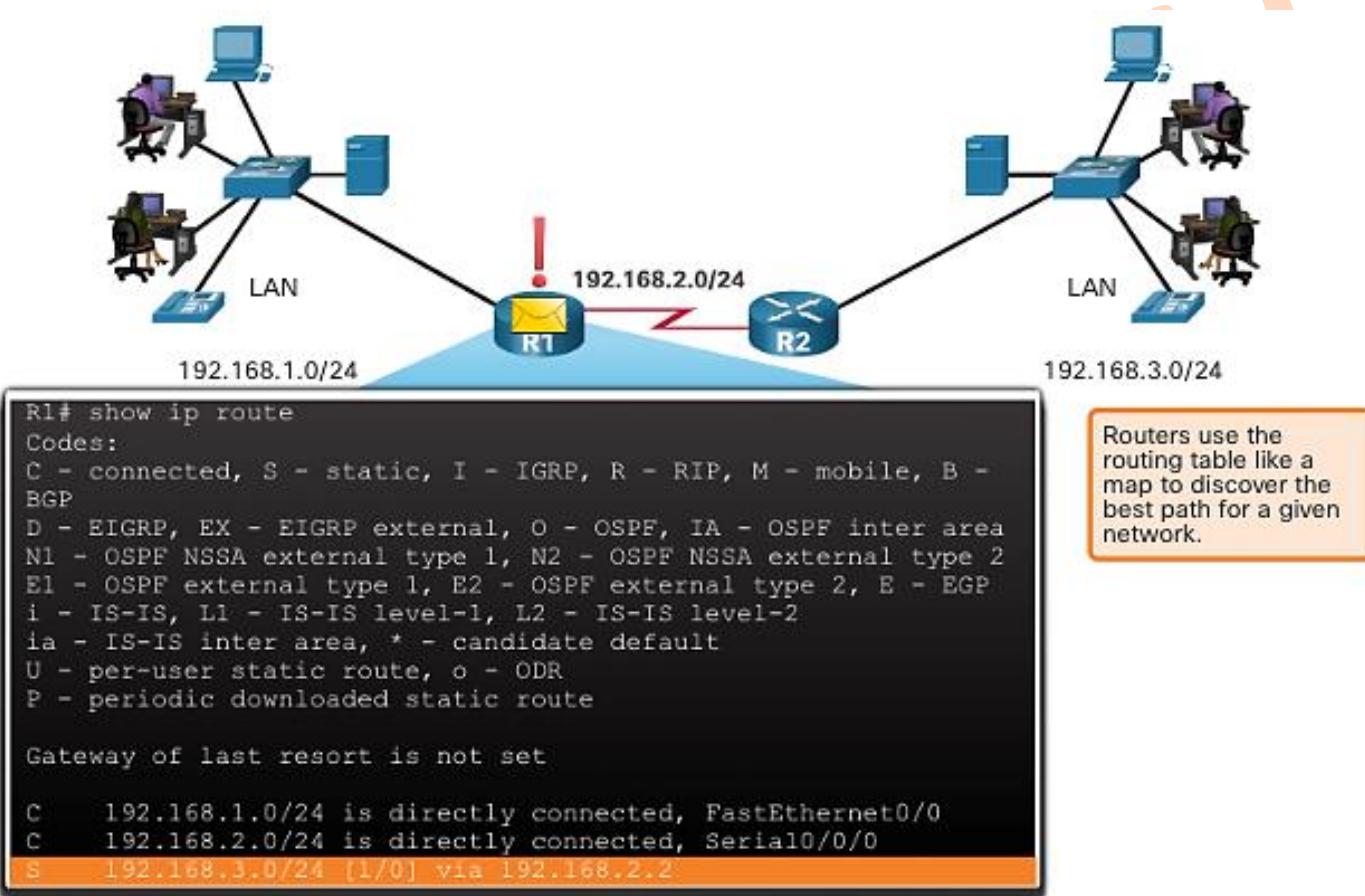
- **Path Determination**
 - How routers determine the best path.
- **Packet Forwarding**
 - How routers forward packets to the destination.



Path Determination

Router Functions Example

- The router uses its IP routing table to determine which path (route) to use to forward a packet. R1 and R2 will use their respective IP routing tables to first determine the best path, and then forward the packet.



Path Determination (Cont.)

Best Path Equals Longest Match

- The **best path in the routing table** is also known as the **longest match**.
- The longest match is the **route in the routing table** that has the greatest number of far-left matching bits with the destination IP address of the packet. The longest match is always the preferred route.

Note: The term **prefix length** will be used to refer to the network portion of both IPv4 and IPv6 addresses.

Path Determination (Cont.)

Longest Match Example

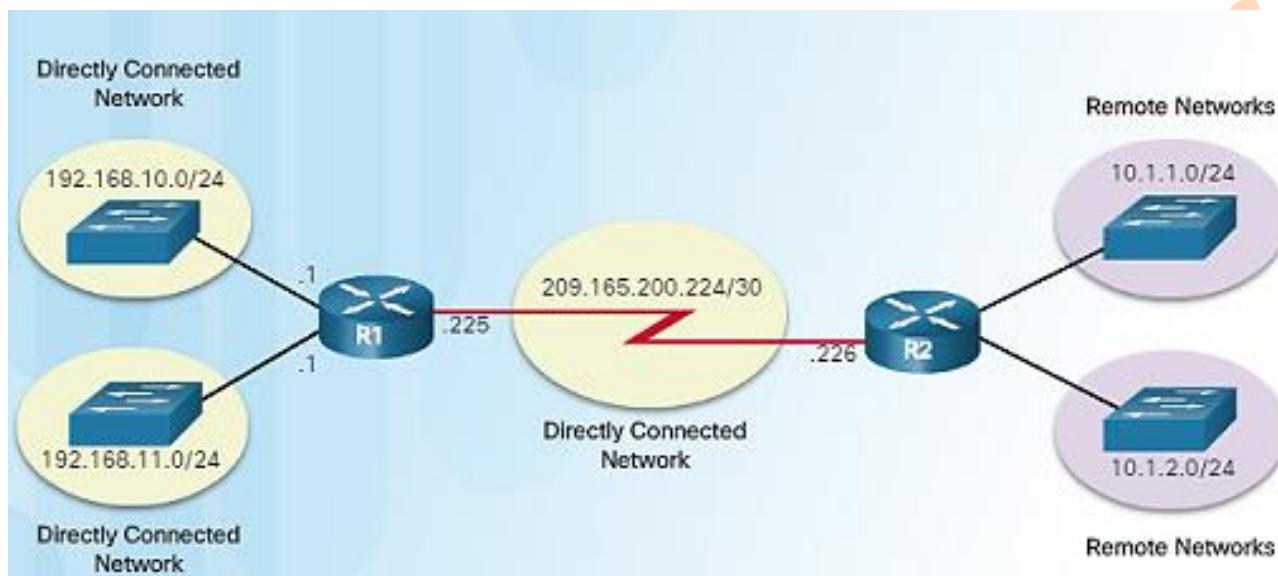
- In the table, an IPv4 packet has the destination IPv4 address 172.16.0.10. The router has three route entries in its IPv4 routing table that match this packet: 172.16.0.0/12, 172.16.0.0/18, and 172.16.0.0/26.
- Of the three routes, 172.16.0.0/26 has the longest match and would be chosen to forward the packet. For any of these routes to be considered a match there must be at least the number of matching bits indicated by the subnet mask of the route.

Destination IPv4 Address		Address in Binary
172.16.0.10		10101100.00010000.00000000.00 001010
Route Entry	Prefix/Prefix Length	Address in Binary
1	172.16.0.0/12	10101100.00010000.00000000.0000 1010
2	172.16.0.0/18	10101100.00010000.00 000000.00001010
3	172.16.0.0/26	10101100.00010000.00000000.00 001010

IP Routing Table

The Routing Table

- The routing table of a router stores information about:
 - **Directly connected routes** – Obtained from the active router interfaces.
 - **Remote routes** – These are remote networks connected to other routers that are learned from dynamic routing protocols or are statically configured.



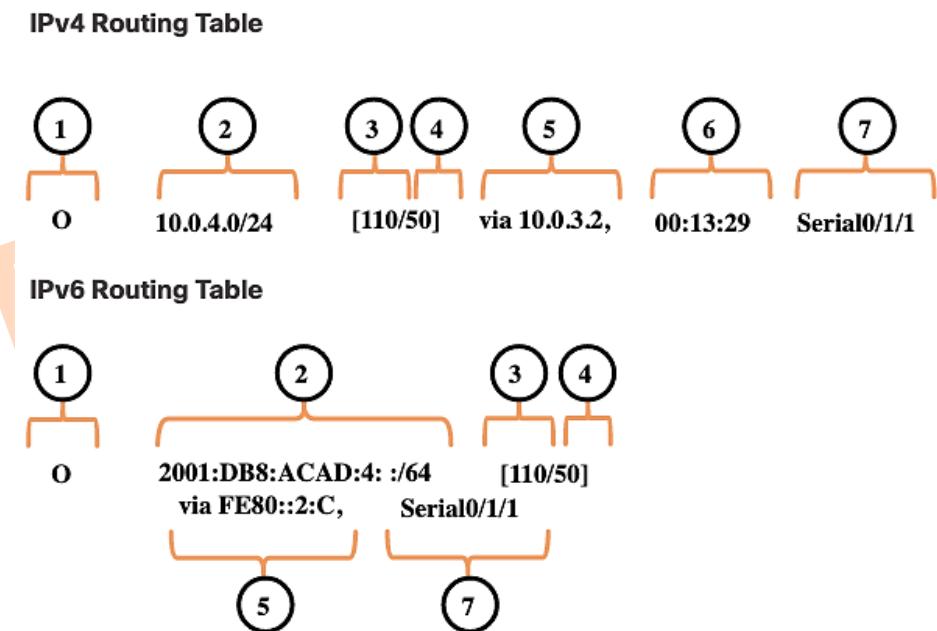
- A routing table is a data file in **RAM** that is used to store information about directly connected and remote networks.
- The routing table contains **next hop** associations for remote networks. The association tells the router what the next hop is for a destination network.

IP Routing Table (Cont.)

Routing Table Entries

In the figure, the numbers identify the following information:

- **Route source** - This identifies how the route was learned.
- **Destination network** (prefix and prefix length) - This identifies the address of the remote network.
- **Administrative distance** - This identifies the trustworthiness of the route source. Lower values indicate preferred route source.
- **Metric** - This identifies the value assigned to reach the remote network. Lower values indicate preferred routes.
- **Next-hop** - This identifies the IP address of the next router to which the packet would be forwarded.
- **Route timestamp** - This identifies how much time has passed since the route was learned.
- **Exit interface** - This identifies the egress interface to use for outgoing packets to reach their final destination.



IP Routing Table (Cont.)

Administrative Distance

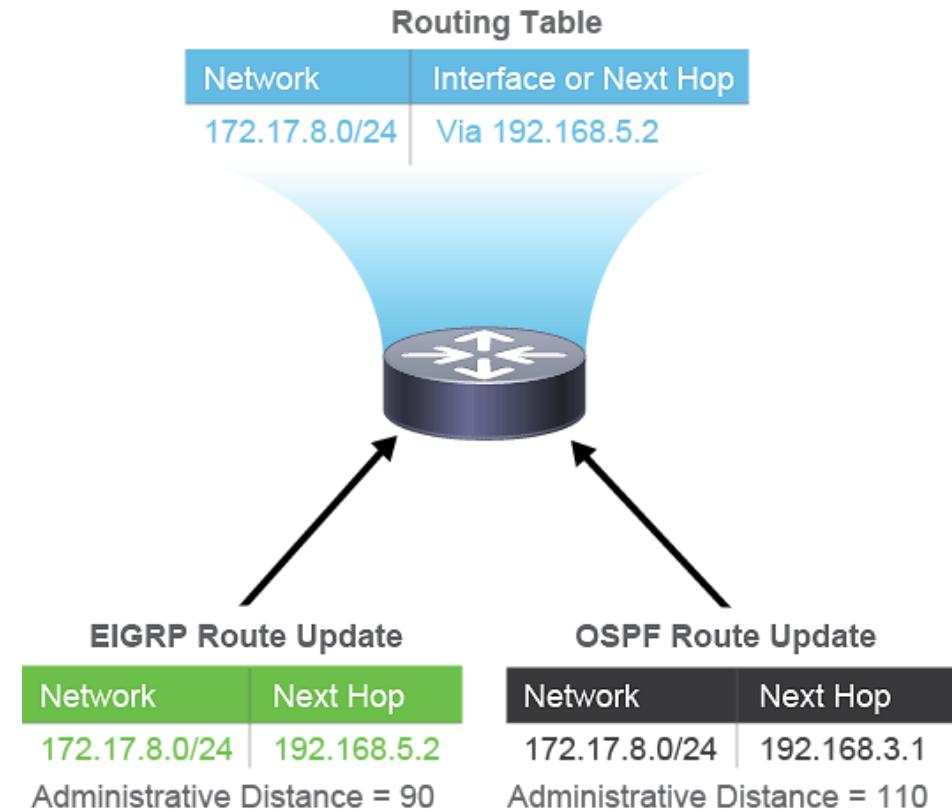
- A route entry for a specific network address (prefix and prefix length) can only appear once in the routing table.
 - However, it is possible that the routing table learns about the same network address from more than one routing source.
 - Each routing protocol may decide on a different path to reach the destination based on the metric of that routing protocol.
- This raises a few questions, such as the following:
 - How does the router know which source to use?
 - Which route should it install in the routing table?
- Cisco IOS uses what is known as the **administrative distance (AD)** to determine the route to install into the IP routing table. The AD represents the "**trustworthiness**" of the route. The lower the AD, the more trustworthy the route source.

IP Routing Table (Cont.)

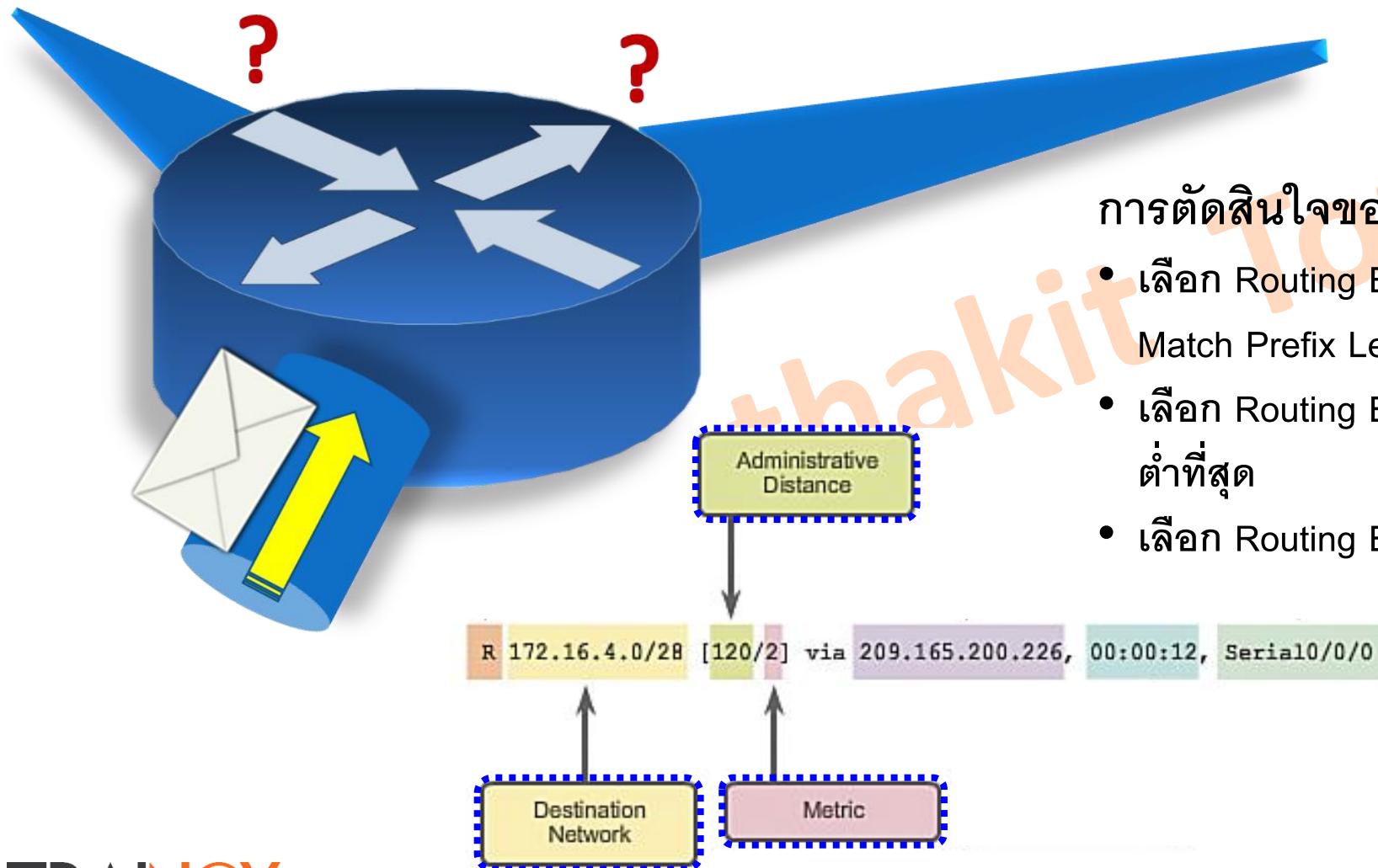
Administrative Distance

- The table lists various routing protocols and their associated ADs.

Route Source	Administrative Distance
Directly connected	0
Static route	1
EIGRP summary route	5
External BGP	20
Internal EIGRP	90
OSPF	110
IS-IS	115
RIP	120
External EIGRP	170
Internal BGP	200



Route Selection of Router

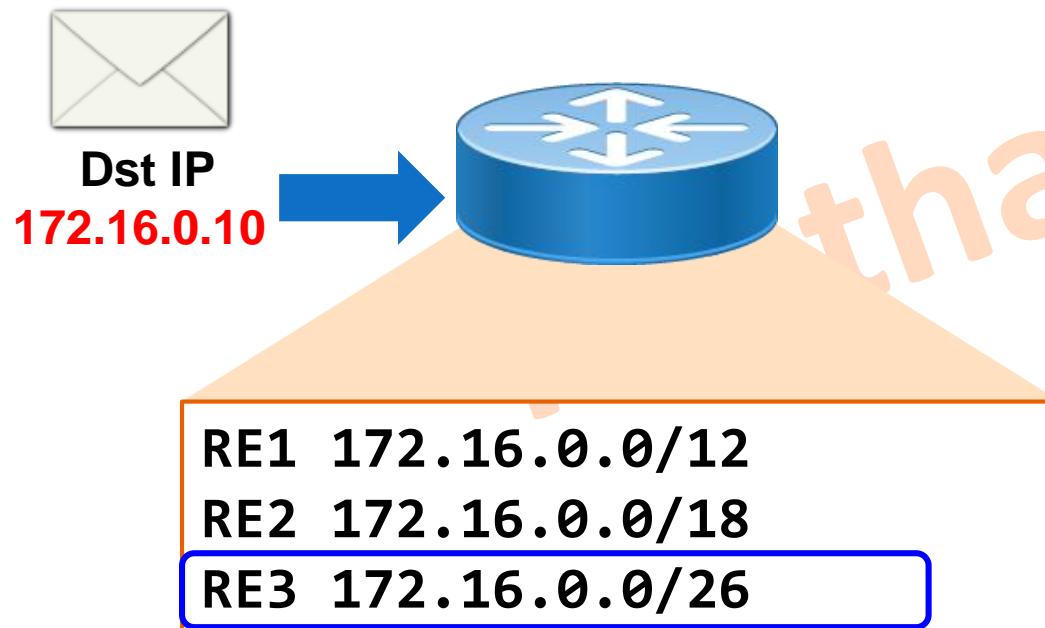


การตัดสินใจของ Router ในการเลือกเส้นทาง

- เลือก Routing Entry ที่ **Longest Match** มากที่สุด (Longest Match Prefix Length)
- เลือก Routing Entry ที่มีค่า **Administrative Distance (AD)** ต่ำที่สุด
- เลือก Routing Entry ที่มีค่า **Metric** ต่ำที่สุด

Route Selection of Router (Cont.)

1) เลือก Routing Entry ที่ **Longest Match** มากที่สุด (Longest Match Prefix Length)



- นำ IP Address ปลายทางมาเทียบกับ Routing Entry โดยดูจากจำนวนบิตด้านหน้าที่ตรงกันมากที่สุด
- หรือเป็นการหา Network ID ที่แคบที่สุด (Most Specific) สำหรับ IP Address ดังกล่าวนั้นเอง (ดู Prefix Length)

10101100.00010000.00000000.00001010

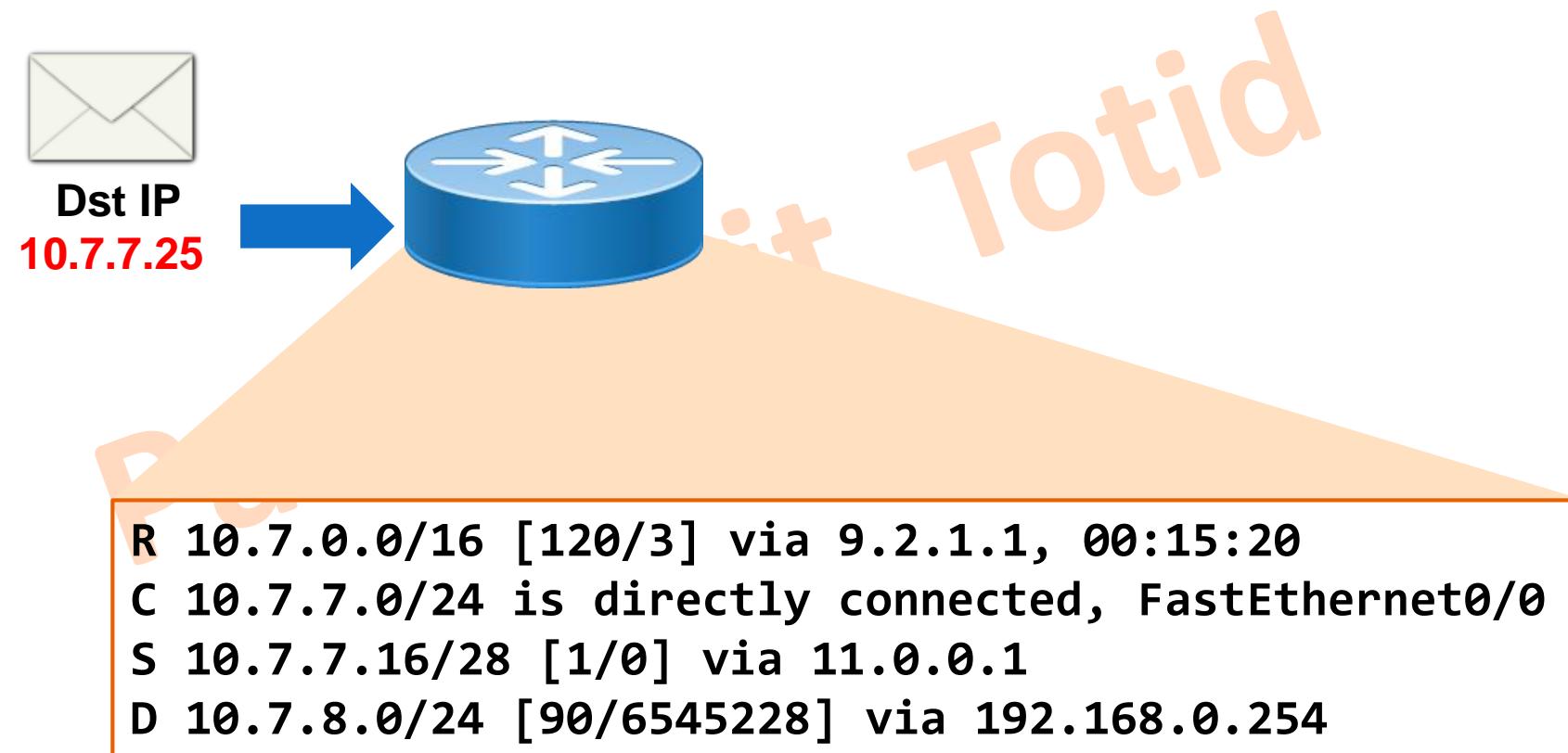
10101100.00010000.00000000.00001010

10101100.00010000.00000000.00001010

10101100.00010000.00000000.00001010

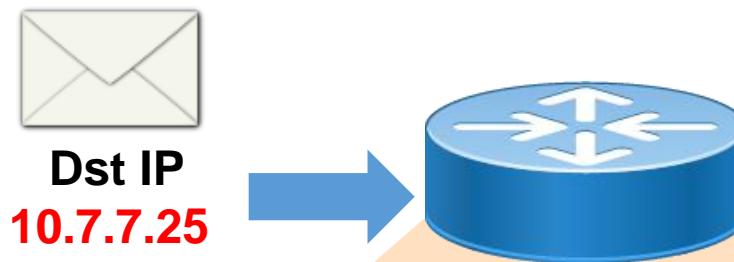
Route Selection of Router (Cont.)

- 1) เลือก Routing Entry ที่ Longest Match มากที่สุด (Longest Match Prefix Length)



Route Selection of Router (Cont.)

2) เลือก Routing Entry ที่มีค่า Administrative Distance (AD) ต่ำที่สุด



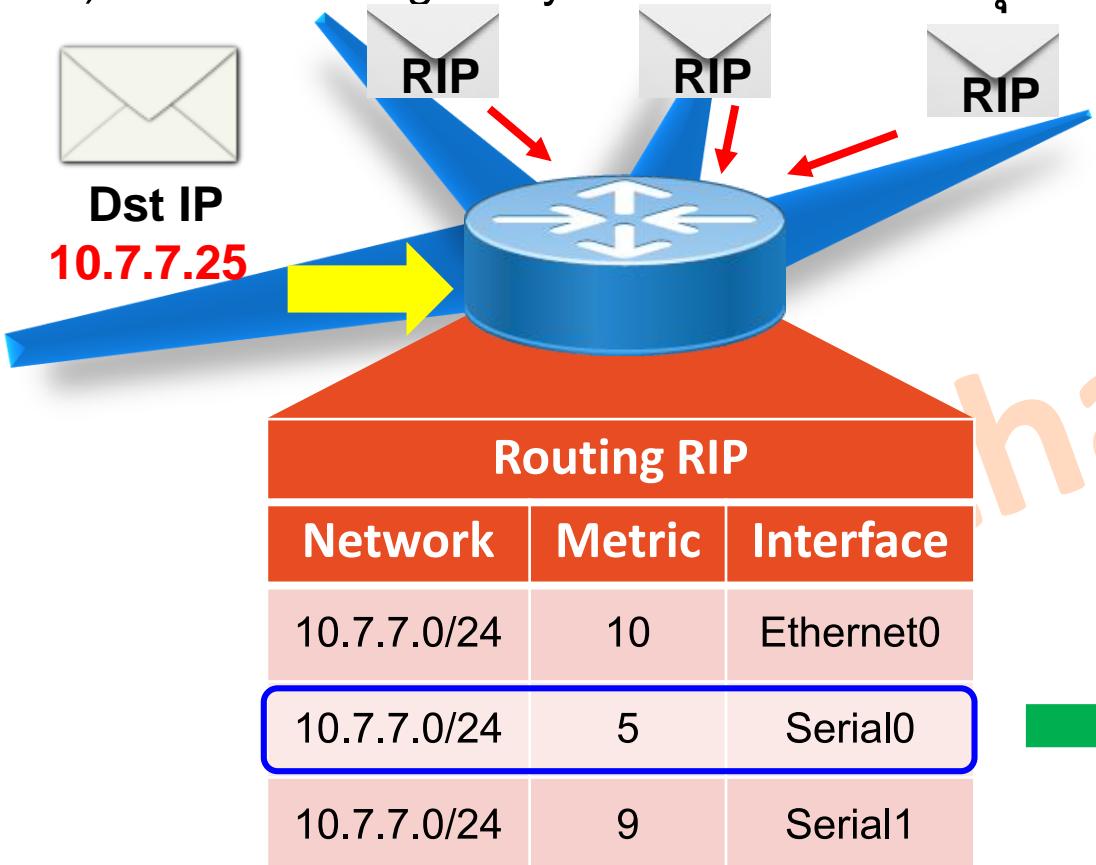
- ค่า AD บ่งบอกถึงความน่าเชื่อถือ และเสถียรภาพของเส้นทางนั้น
- เช่น เส้นทางแบบ Static มีความเสถียรสูงกว่าเส้นทางแบบ Dynamic เป็นต้น

```
R 10.7.0.0/16 [120/3] via 9.2.1.1, 00:15:20
C 10.7.7.16/28 is directly connected, FastEthernet0/0
S 10.7.7.16/28 [1/0] via 11.0.0.1
D 10.7.8.0/24 [90/6545228] via 192.168.0.254
```

Route Source	Default AD
Connected Interface	0
Static Route	1
External BGP	20
EIGRP	90
OSPF	110
IS-IS	115
RIP	120

Route Selection of Router (Cont.)

3) เลือก Routing Entry ที่มีค่า Metric ต่ำที่สุด

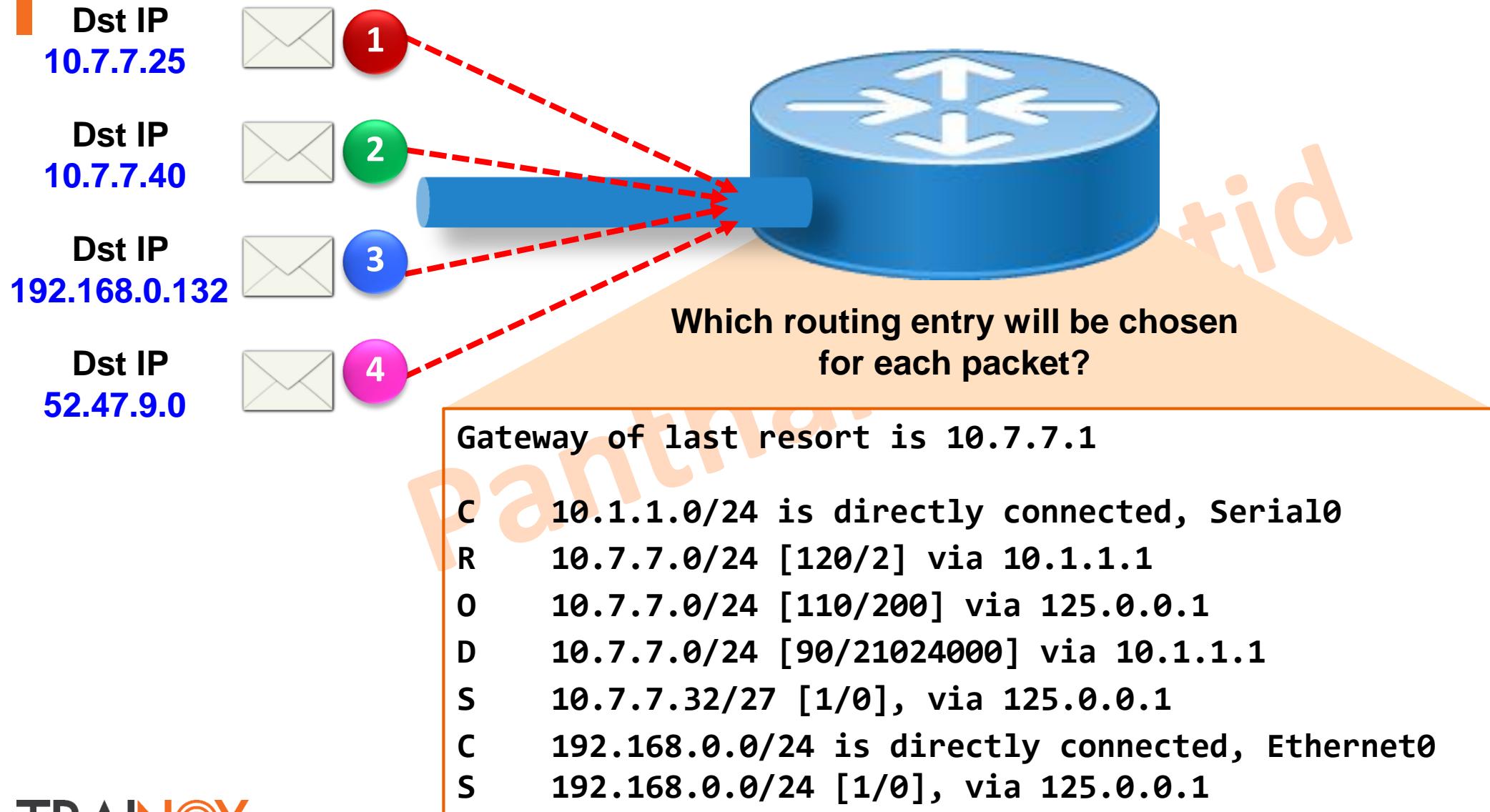


- ค่า Metric ถูกนำมาใช้คัดเลือกเส้นทางที่ดีที่สุดสำหรับแต่ละ Routing Protocol
- ค่า Metric ของแต่ละ Routing Protocol ไม่สามารถมาใช้ร่วมกันได้
RIP → Hop Count
OSPF → Bandwidth
EIGRP → Bandwidth + Delay

นำเส้นทางที่ดีที่สุดไปไว้ใน Routing Table

R 10.7.0.0/24 [120/5] via 10.1.1.1, 00:15:20

Route Selection of Router (Cont.)



Panthakit Totid

Panthakit Totid



Basic Network For Trainee

Module 6

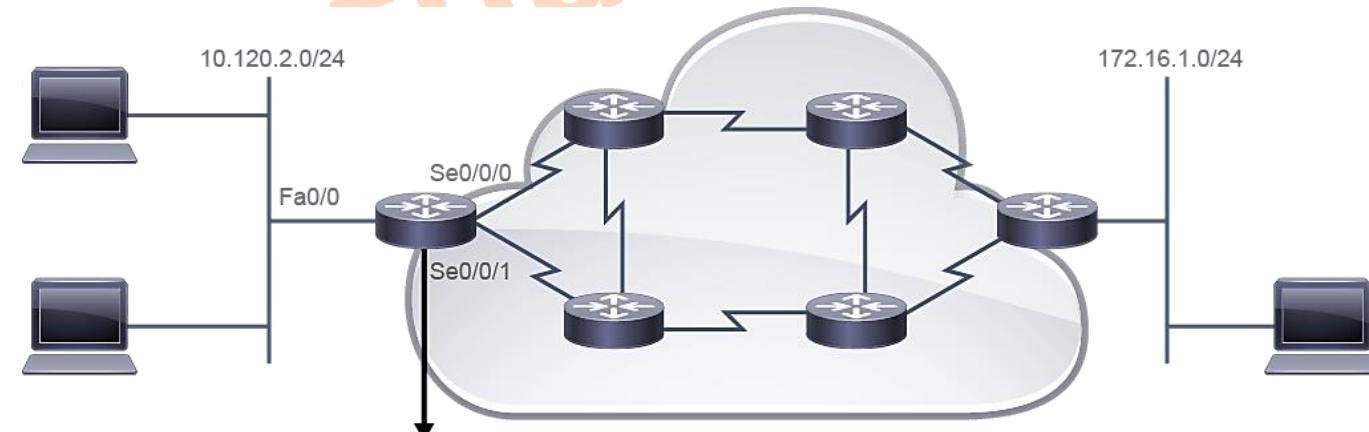
Static Routing

Panthakit Totid



Routing Operation

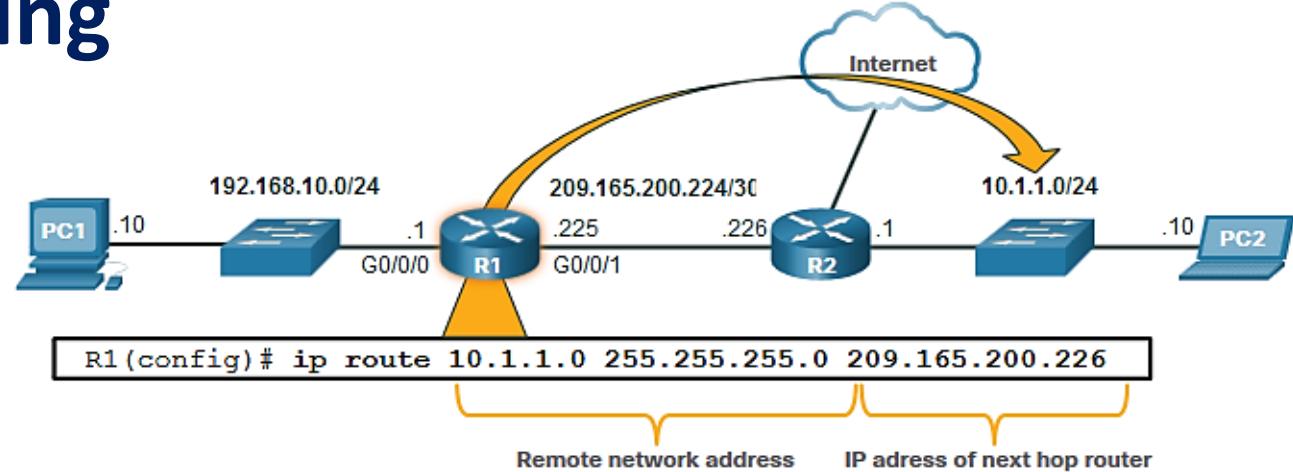
- Routing is the process of **selecting a path** to forward data that originated from one network and is destined for a different network. Routers **gather and maintain routing information** to enable the transmission and receipt of such data packets.
- Conceptually, routing information takes the form of **entries** in a routing table, with one entry for each identified route. You can **manually configure** the entries in the routing table or the router can use a routing protocol to create and maintain the routing table **dynamically** to accommodate network changes when they occur.



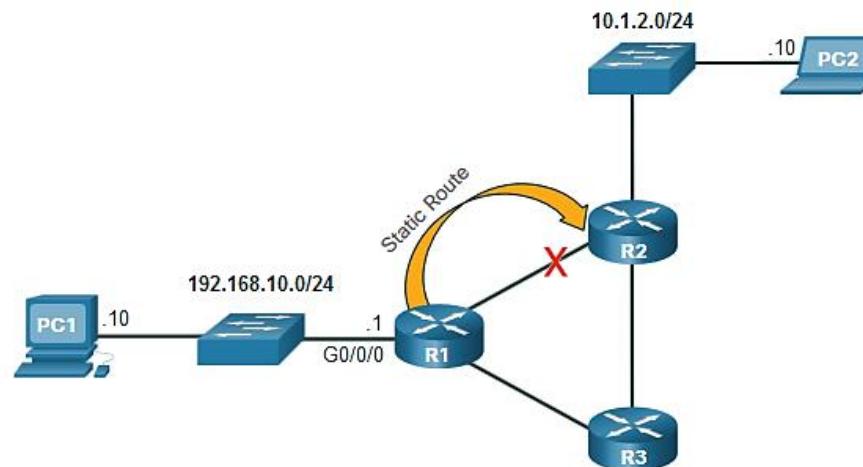
Network Protocol	Destination Protocol	Exit Interface
Connected	10.120.2.0/24	Fa0/0
Learned	172.16.1.0/24	Se0/0/0

Introduction to Static Routing

- Static Route Characteristics:
 - Must be *configured manually*
 - Must be *adjusted manually* by the administrator when there is a change in the topology
 - Good for small non-redundant networks
 - Often used in conjunction with a dynamic routing protocol for configuring a **default route**



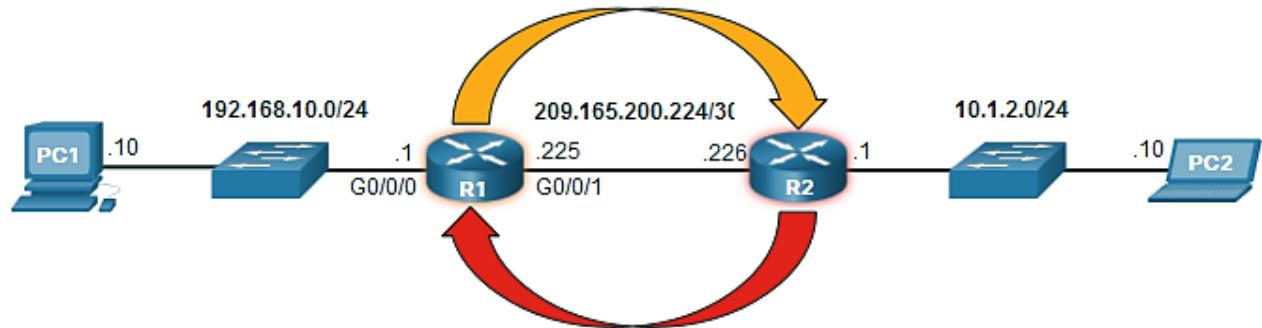
R1 is manually configured with a static route to reach the 10.1.1.0/24 network. If this path changes, R1 will require a new static route.



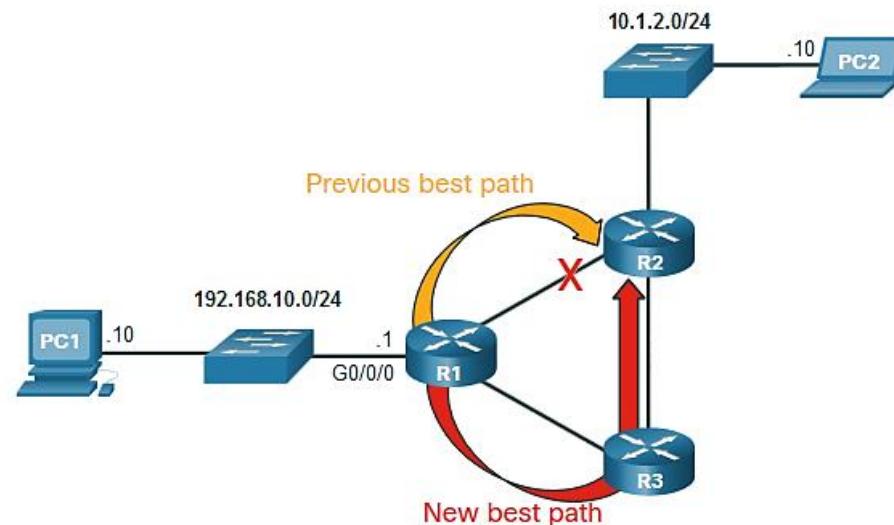
If the route from R1 via R2 is no longer available, a new static route via R3 would need to be configured. A static route does not automatically adjust for topology changes.

Introduction to Dynamic Routing

- Dynamic Routes Automatically:
 - **Discover** remote networks
 - **Maintain** up-to-date information
 - **Choose** the best path to the destination
 - **Find new** best paths when there is a topology change
- Dynamic routing can also share static default routes with the other routers.



- R1 is using the routing protocol OSPF to let R2 know about the 192.168.10.0/24 network.
- R2 is using the routing protocol OSPF to let R1 know about the 10.1.1.0/24 network.



R1, R2, and R3 are using the dynamic routing protocol OSPF. If there is a network topology change, they can automatically adjust to find a new best path.

Static or Dynamic Routing?

- The table shows a comparison of some the differences between dynamic and static routing

Feature	Dynamic Routing	Static Routing
Configuration complexity	Independent of network size	Increases with network size
Topology changes	Automatically adapts to topology changes	Administrator intervention required
Scalability	Suitable for simple to complex network topologies	Suitable for simple topologies
Security	Security must be configured	Security is inherent
Resource Usage	Uses CPU, memory, and link bandwidth	No additional resources needed
Path Predictability	Route depends on topology and routing protocol used	Explicitly defined by the administrator

When to Use Static Routing

- Static routes are best suited for small networks, such as LANs, where routes rarely change. If routes change, you need to manually update to reflect the new data transmission paths.
- **Use static routes In these situations:**
 - In a small network that requires only simple routing
 - In a hub-and-spoke network topology
 - When you want to create a quick ad hoc route
 - Common use is a default static route
- **Do not use static routes In these situations:**
 - In a large network
 - When the network is expected to scale

Static Routes

Types of Static Routes

- Static routes are commonly implemented on a network. This is true even when there is a dynamic routing protocol configured.
- Static routes can be configured for IPv4 and IPv6. Both protocols support the following types of static routes:
 - **Standard static route**
 - **Default static route**
 - **Floating static route**
 - **Summary static route**
- Static routes are configured using the **ip route** and **ipv6 route** global configuration commands.

Static Routes

Next-Hop Options and Command

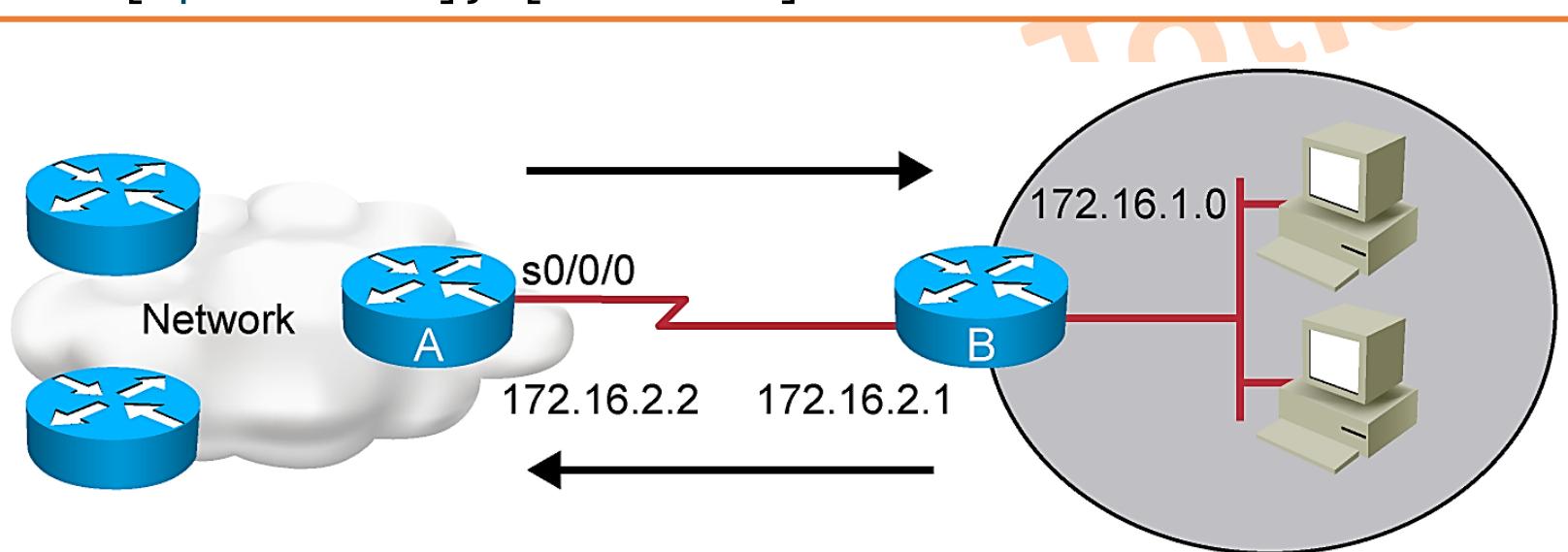
- When configuring a static route, the next hop can be identified by an IP address, exit interface, or both. How the destination is specified creates one of the three following types of static route:
 - Next-hop route** - Only the next-hop IP address is specified
 - Directly connected static route** - Only the router exit interface is specified
 - Fully specified static route** - The next-hop IP address and exit interface are specified

Static Routes Configuration

Standard Static Route

- IPv4 static routes are configured using the following global configuration command:

```
Router(config)# ip route network-address subnet-mask {ip-address |  
exit-intf [ip-address]} [distance]
```



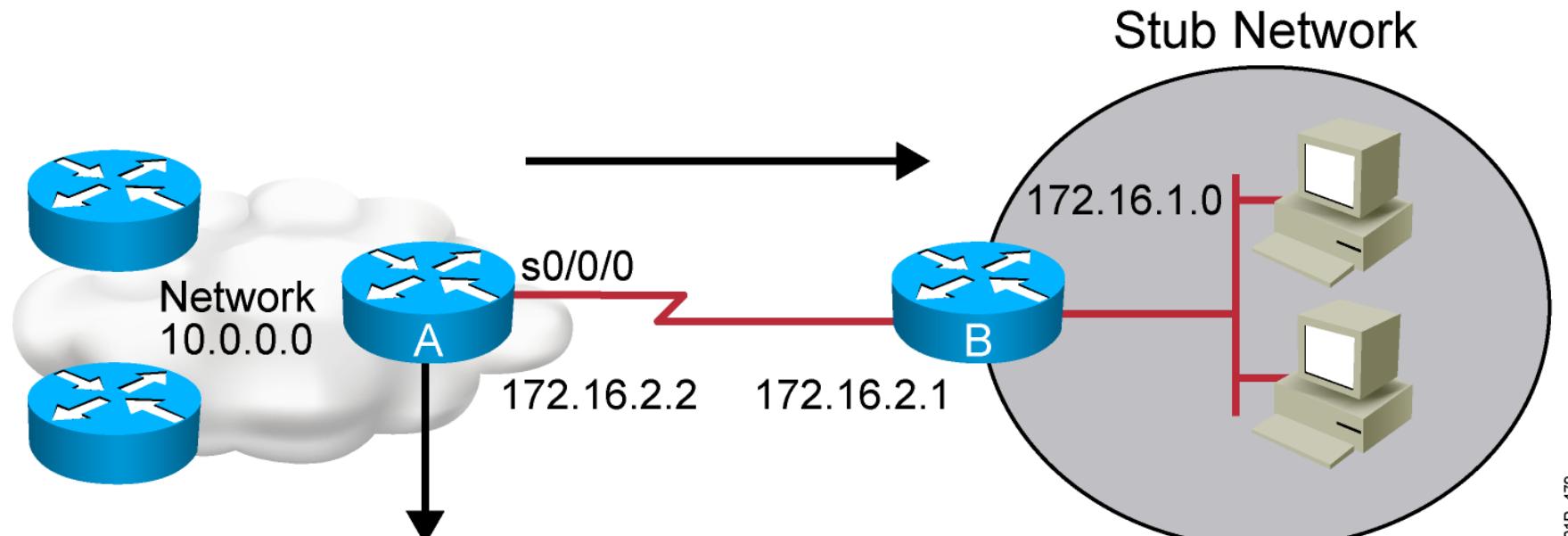
301P_478

This is a **unidirectional route**.

You must have a route configured in the opposite direction.

Static Routes Configuration

Standard Static Route



```
RouterA(config)#ip route 172.16.1.0 255.255.255.0 172.16.2.1
```

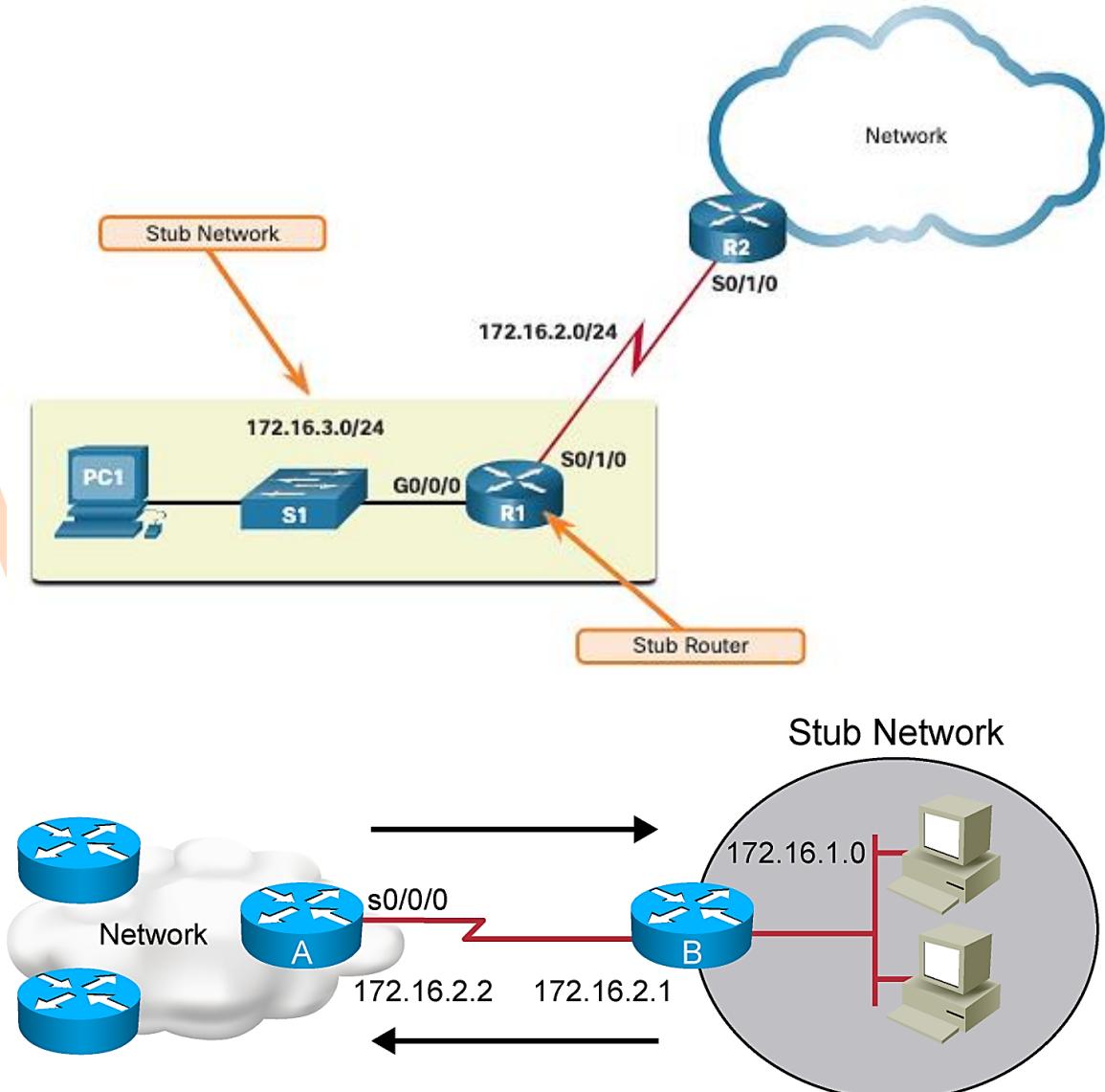
or

```
RouterA(config)#ip route 172.16.1.0 255.255.255.0 S0/0/0
```

Static Routes Configuration

Default Static Route

- A default route is a static route that **matches all packets**.
 - A single default route represents any network that is not in the routing table.
 - Used as the **Gateway of Last Resort**.
- Routers commonly use default routes that are either configured locally or learned from another router.
- Default static routes are commonly used when connecting an edge router to a service provider network, or a stub router (a router with only one upstream neighbor router).



Static Routes Configuration

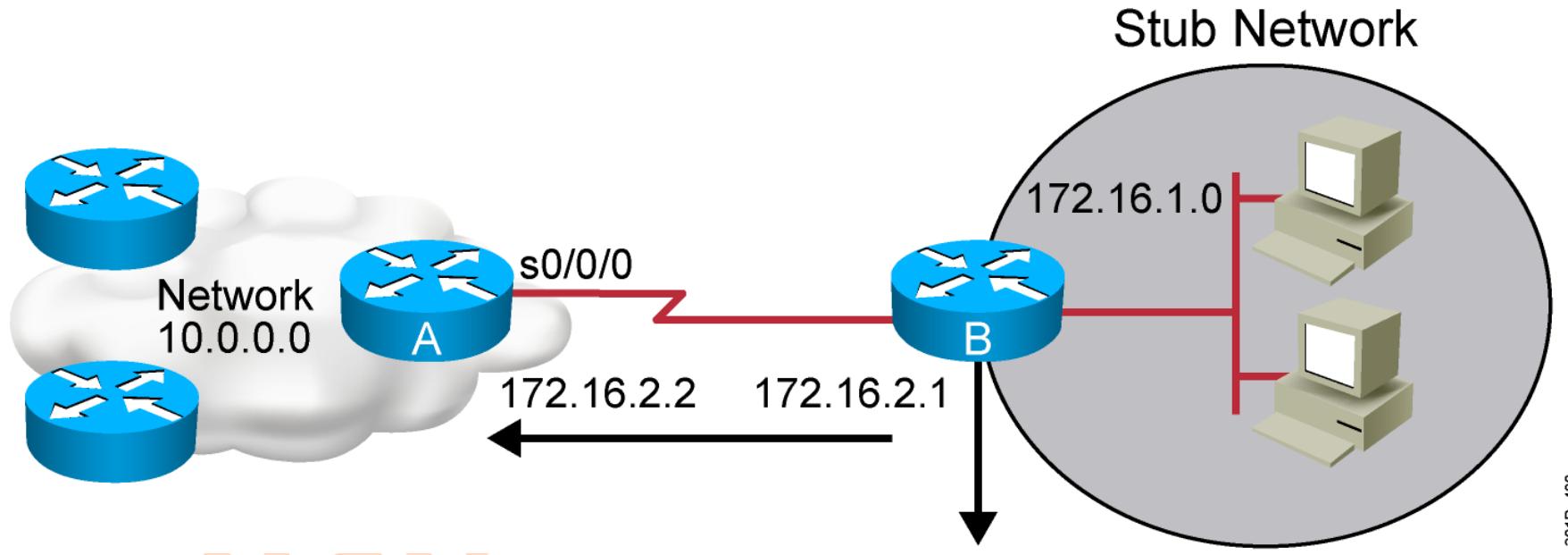
Default Static Route (Cont.)

- The command syntax for an IPv4 default static route is similar to any other IPv4 static route, except that the network address is 0.0.0.0 and the subnet mask is 0.0.0.0. The 0.0.0.0 0.0.0.0 in the route will match any network address.
- Note:** An IPv4 default static route is commonly referred to as a **quad-zero route**.
- The basic command syntax for an IPv4 default static route is as follows:

```
Router(config)# ip route 0.0.0.0 0.0.0.0 {ip-address | exit-intf}
```

Static Routes Configuration

Default Static Route (Cont.)



301P_480

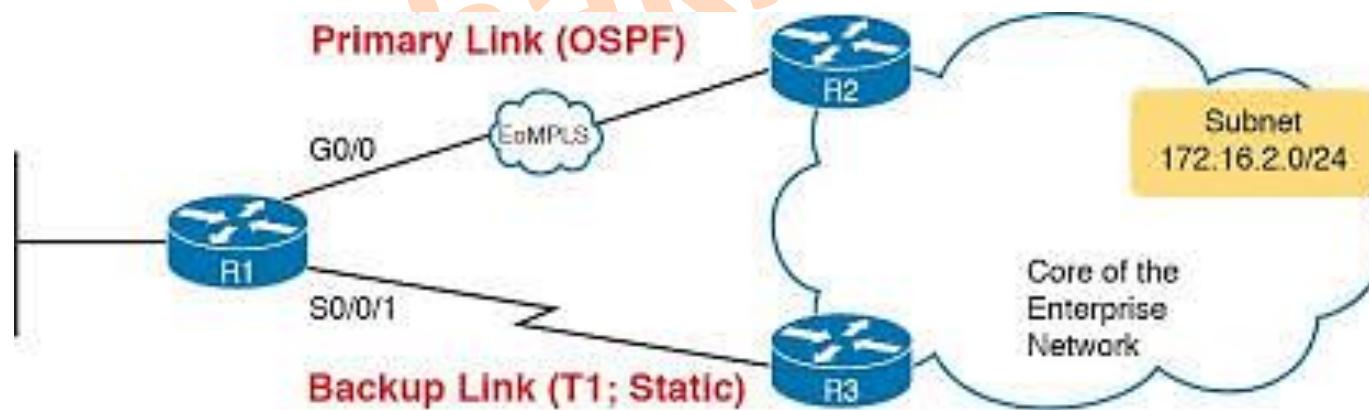
```
RouterB(config)#ip route 0.0.0.0 0.0.0.0 172.16.2.2
```

This route allows the stub network to reach
all unknown networks beyond Router A.

Static Routes Configuration

Floating Static Routes

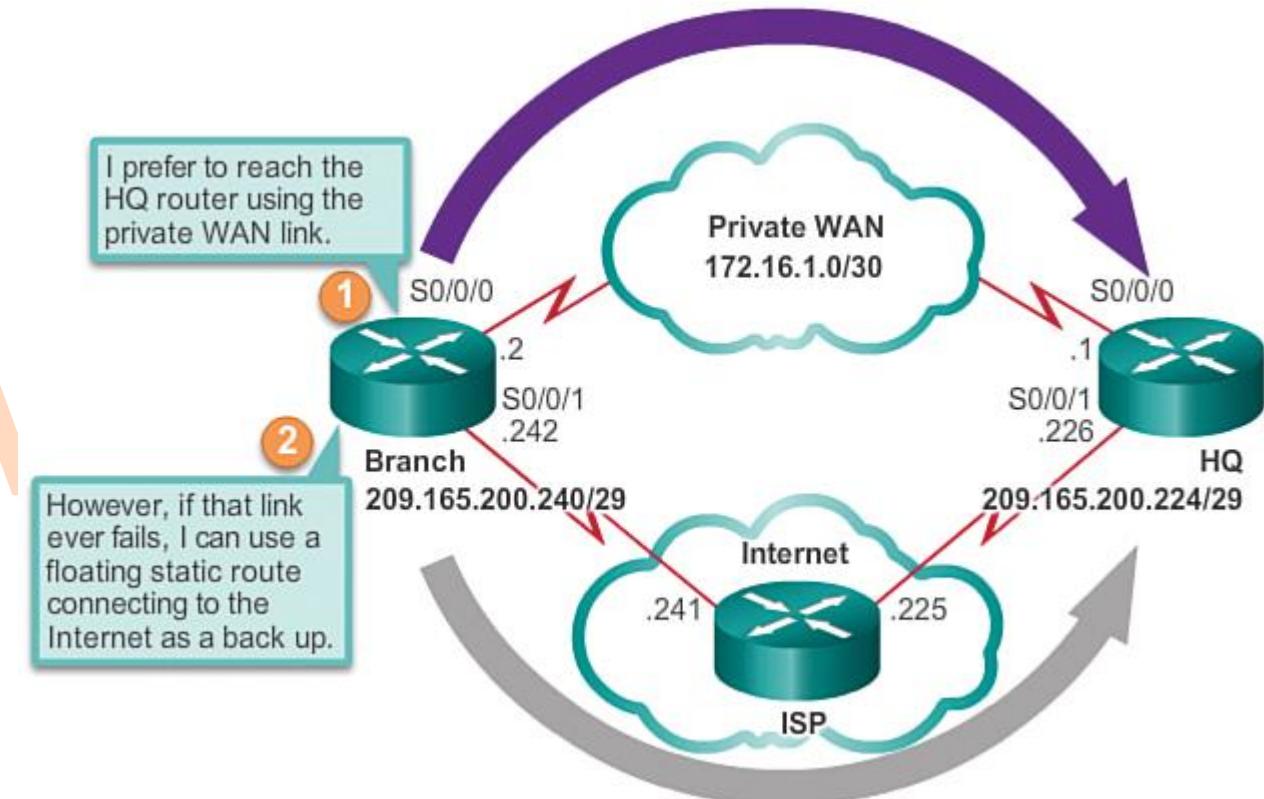
- Floating static routes are static routes that are used to provide a **backup path** to a primary static or dynamic route.
 - Only used when the primary route is not available.
 - Configured with a higher administrative distance than the primary route.
 - If multiple paths to the destination exist, the router will choose the path with the lowest administrative distance.



Static Routes Configuration

Floating Static Routes

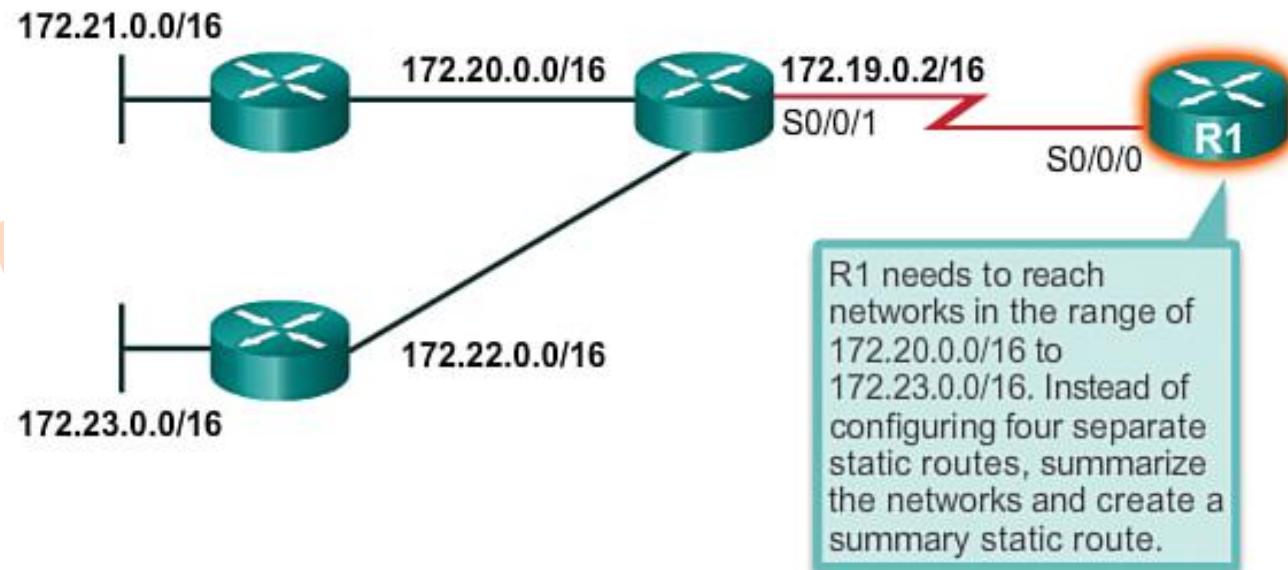
- By default, static routes have an AD of 1, making them preferable to routes learned from dynamic routing protocols.
- **Example:** EIGRP administrative distance equals 90. A floating static route with an AD of 91 or higher would serve as backup route and will be used if EIGRP route goes down.



Static Routes Configuration

Summary Static Routes

- Multiple static routes can be summarized into a single network address
 - Destination networks must be contiguous
 - Multiple static routes must use the same exit interface or next hop
 - In figure, four networks is summarized into one summary static route





Basic Network For Trainee

Module 7

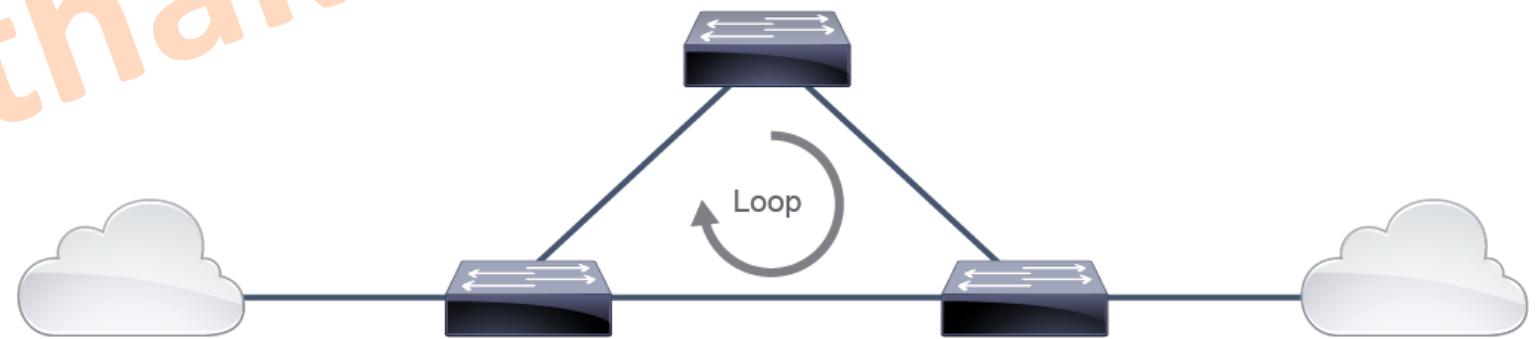
STP Concepts and EtherChannel

Panthakit Totid



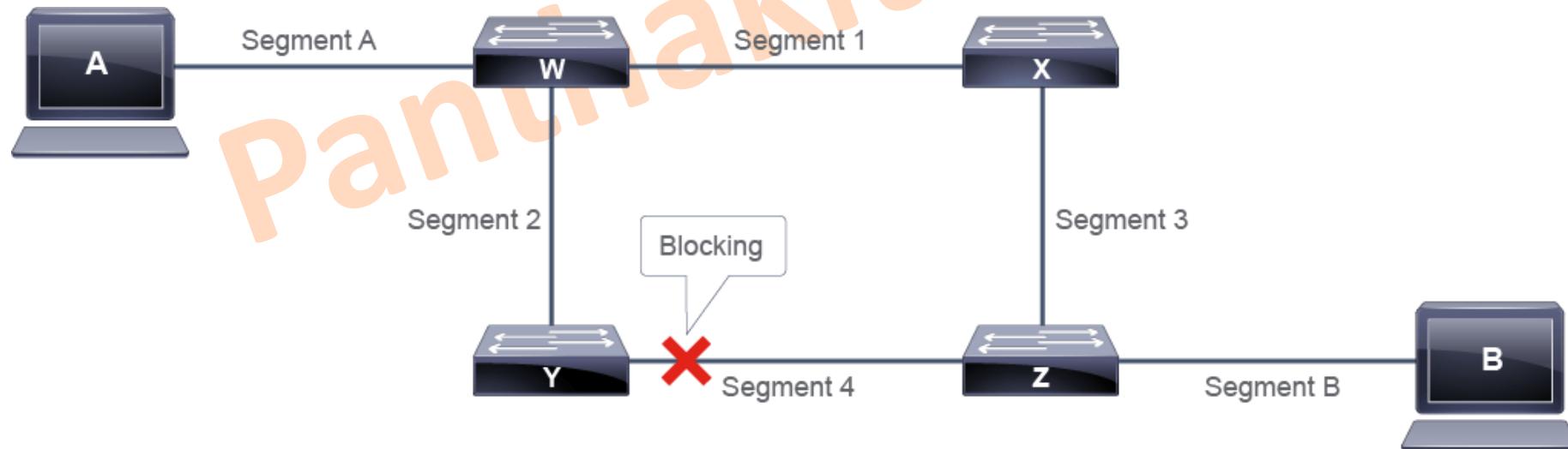
Physical Redundancy in a LAN

- Enterprise voice and data networks are designed with physical component redundancy to **eliminate the possibility of any single point of failure** causing a loss of function for an entire switched network.
- Building a reliable switched network **requires additional switches** and redundant physical links.
- However, when adding redundant physical links and additional switches, a **physical loop** is created and by spanning a single VLAN between connected switches a **Layer 2 loop** is also created.
- A redundant switch topology is vulnerable to these conditions:
 - Continuous frame duplication
 - Multiple frame transmission
 - MAC database instability



Spanning Tree Operation

- The original STP is an IEEE committee standard that is defined as **802.1D**
- STP behaves in the following way:
 - STP uses **bridge protocol data units (BPDUs)** for communication between switches.
 - STP forces certain ports into a **blocked state** so that they do not listen to, forward, or flood data frames. The overall effect is that only one path to each network segment is active at any time.



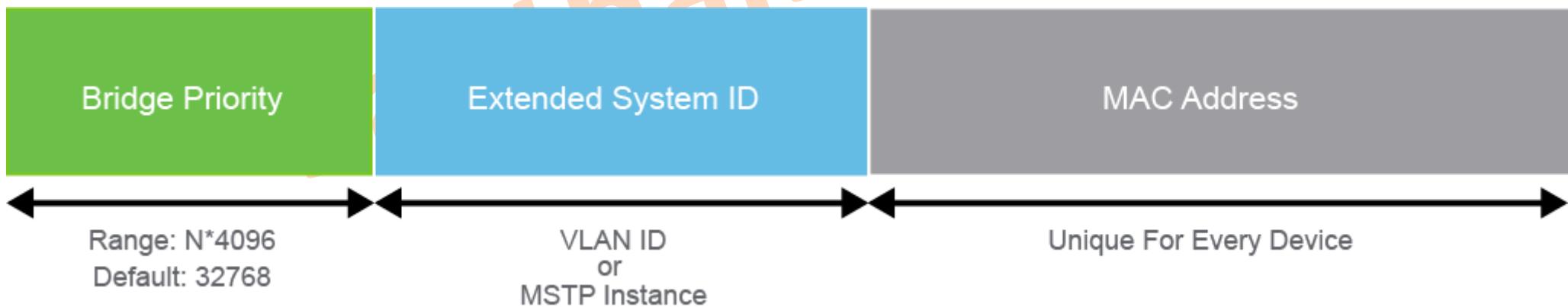
Spanning Tree Operation (Cont.)

- To prevent Layer 2 loops in a network, STP uses a reference point called **root bridge**.
 - The root bridge is the **logical center of the spanning tree topology**.
- The root bridge is chosen with an election.
 - In the original STP, each switch has a unique **64-bit bridge ID (BID)** that consists of the **16-bit bridge priority** and **48-bit MAC address**.
 - The bridge priority is a number between 0 and 65535 and the default on Cisco switches is **32768**.



Spanning Tree Operation (Cont.)

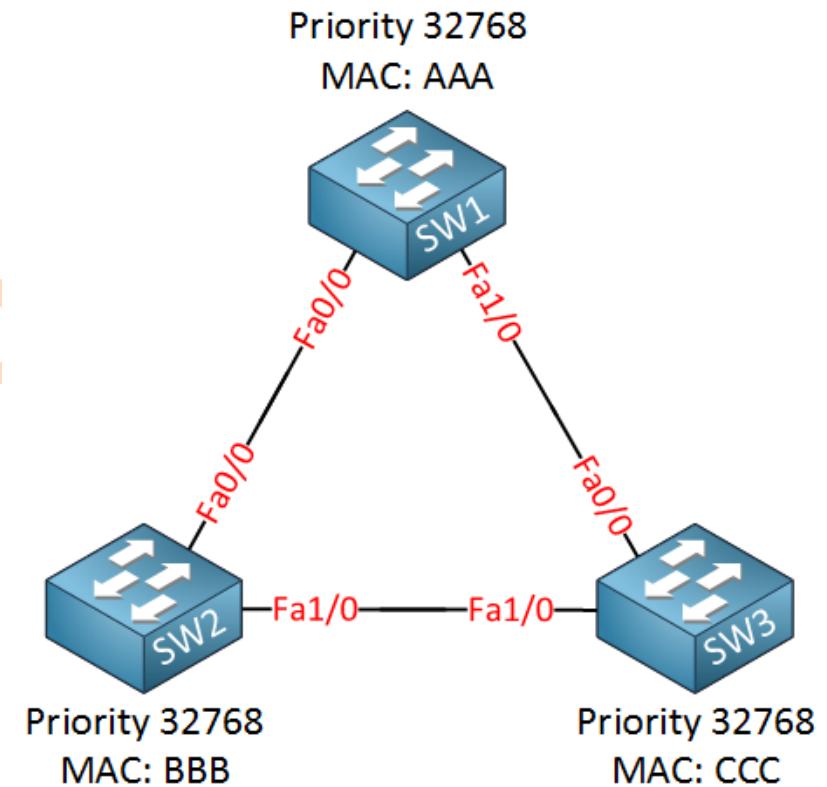
- In evolved variants of STP, like **Cisco PVST+, RSTP** or **Multiple Spanning Tree Protocol (MSTP)**, the original bridge priority field in the BID is changed to include an **Extended System ID field**.
 - This field carries information such as **VLAN ID** or instance number required for the evolved variants of STP to operate.
- The bridge priority field in this case is 4 bits and the Extended System ID field is 12 bits.
- The bridge priority is a number between 0 and 65535 in **increments of 4096**



Spanning Tree Operation (Cont.)

Steps of the spanning tree algorithm:

- 1) The switches elect a **root bridge**.
 - The root bridge is selected based on the *lowest BID* (in all STP variants). If all switches in the network have the *same bridge priority*, the switch with the *lowest MAC address* becomes the root bridge.
- 2) Each nonroot switch determines a **root port**.
 - The root port is the port with the *best path (lowest root path cost)* to the **root bridge**.
- 3) On each segment, a **designated port** is selected.
 - The designated port on a segment is on the switch with the *lowest root path cost*.
 - On root bridges, all switch ports are designated ports.
- 4) The root ports and designated ports transition to the **forwarding state** and any other ports (called nondesignated ports) stay in the **blocking state**.



To prevent Layer 2 loops while STP executes its algorithm, all ports start out in the **blocking state**.

Spanning Tree Operation (Cont.)

- The **STP path cost** depends on the speed of the link. The first table shows the default STP link costs.

Data Rate	STP Cost (802.1D-1998)	STP Cost (802.1D-2004)
4 Mbps	250	5,000,000
10 Mbps	100	2,000,000
16 Mbps	62	1,250,000
100 Mbps	19	200,000
1 Gbps	4	20,000
2 Gbps	3	10,000
10 Gbps	2	2000

Spanning Tree Operation (Cont.)

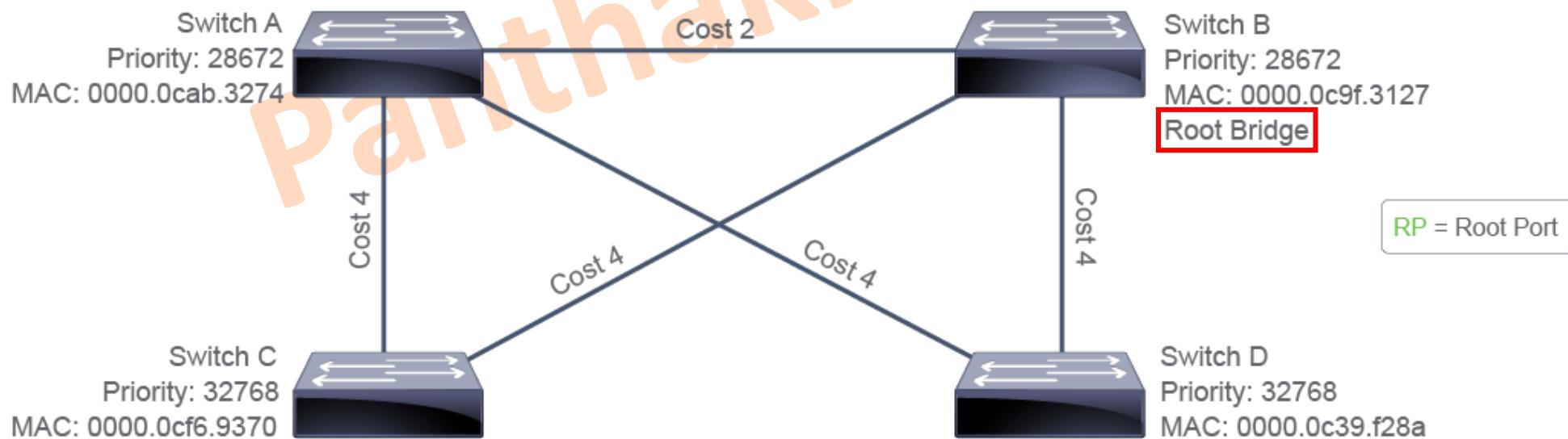
STP Port State	Receive BPDUs	Send BPDUs	Learn MAC Addresses	Receive Data	Send Data	Duration of the State
Blocking	✓	✗	✗	✗	✗	Undefined (if there is a loop)
Listening	✓	✓	✗	✗	✗	Forward Delay (15 Seconds)
Learning	✓	✓	✓	✗	✗	Forward Delay (15 Seconds)
Forwarding	✓	✓	✓	✓	✓	Undefined (as long as there is no loop)
Disabled	✗	✗	✗	✗	✗	Until administrator enables it

Spanning Tree Operation (Cont.)

Spanning Tree Operation Example

- Step 1: Elect a root bridge.

- Initially, **all switches assume that they are the root**. They start transmitting BPDUs with the Root ID field containing the same value as the bridge ID field.
- Eventually, all switches learn and record the BID of the switch that has the **lowest BID**. The switches all transmit this BID in the Root ID field of their BPDUs.

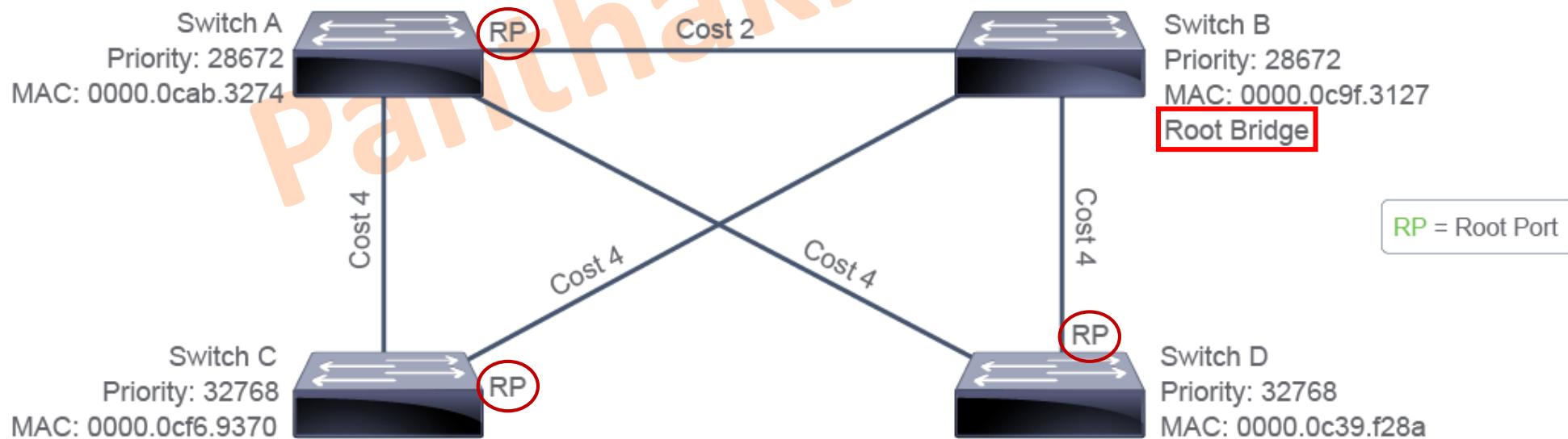


Spanning Tree Operation (Cont.)

Spanning Tree Operation Example

- Step 2: Elect a root port for each nonroot switch

- When a switch recognizes that it is not the root (because it is receiving BPDUs that have a root ID value that is lower than its own BID), it marks the port on which it is receiving those BPDUs as its **root port**.
- The decision is based on the **lowest root cost path** to the root as its root port.
- If necessary, ties are broken by **upstream BID** and **port ID values**.

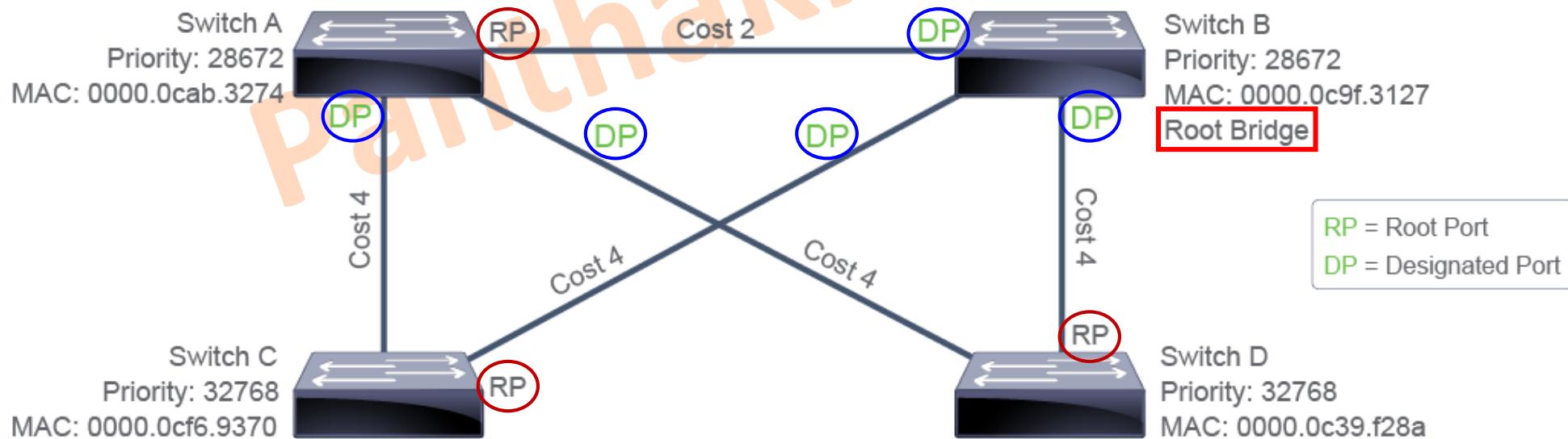


Spanning Tree Operation (Cont.)

Spanning Tree Operation Example

- Step 3: Elect a designated port for each segment

- The decision is based on the **lowest root cost path** to the root as its root port.
- If necessary, ties are broken by **upstream BID** and **port ID values**.
- The switch stops transmitting BPDUs on the port and marks it as a **nondesignated port** if the other switch has lower values.

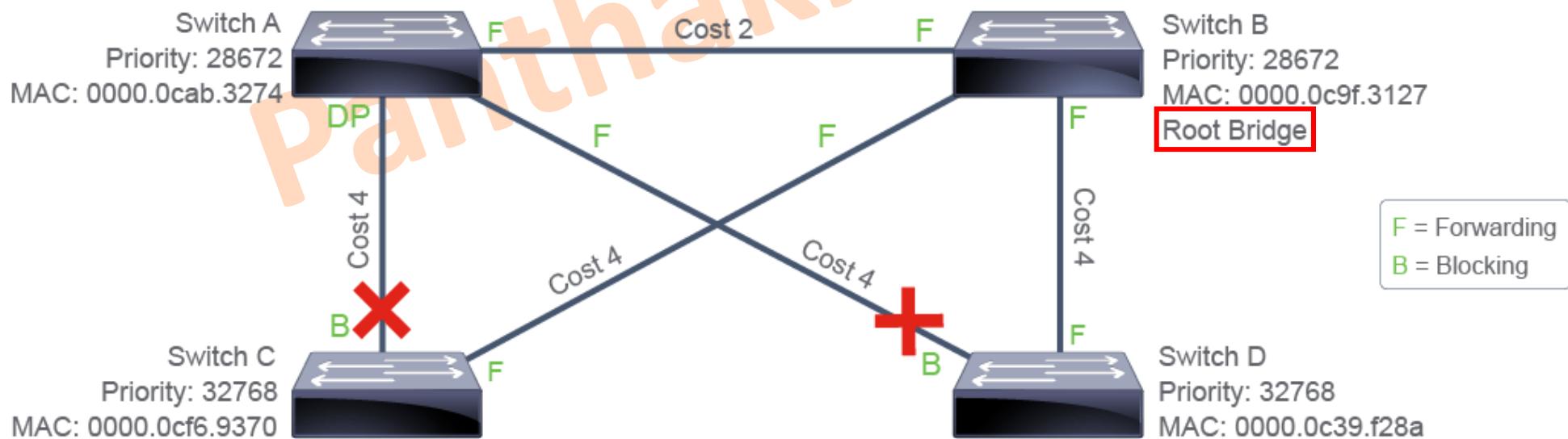


Spanning Tree Operation (Cont.)

Spanning Tree Operation Example

- Step 4: The ports transition to the forwarding or blocking state

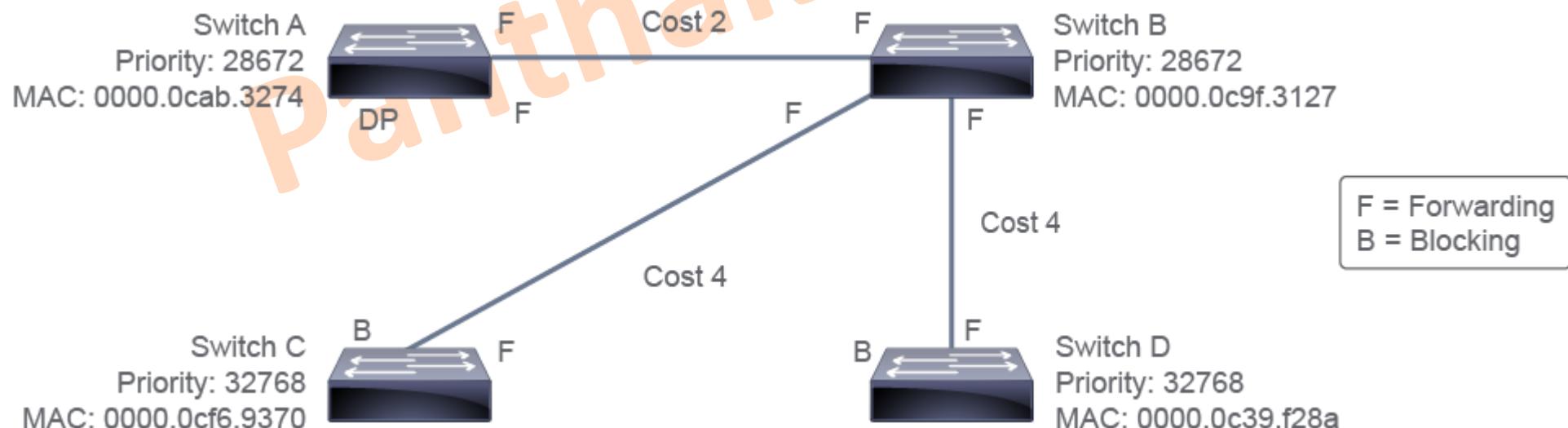
- When STP marks a port as either a root port or a designated port, the algorithm starts to transition this port to the **forwarding state** and all nondesignated ports remain in the **blocking state**.
- As the BPDUs are propagated through the network, all switches eventually have a consistent view of the topology of the network.



Spanning Tree Operation (Cont.)

Spanning Tree Operation Example

- There are **two loops** in the sample topology, meaning that two ports should be in the blocking state to break both loops.
 - The port on Switch C that is not directly connected to Switch B (root bridge) is blocked, because it is a nondesignated port.
 - The port on Switch D that is not directly connected to Switch B (root bridge) is also blocked, because it is a nondesignated port.



Types of Spanning Tree Protocols

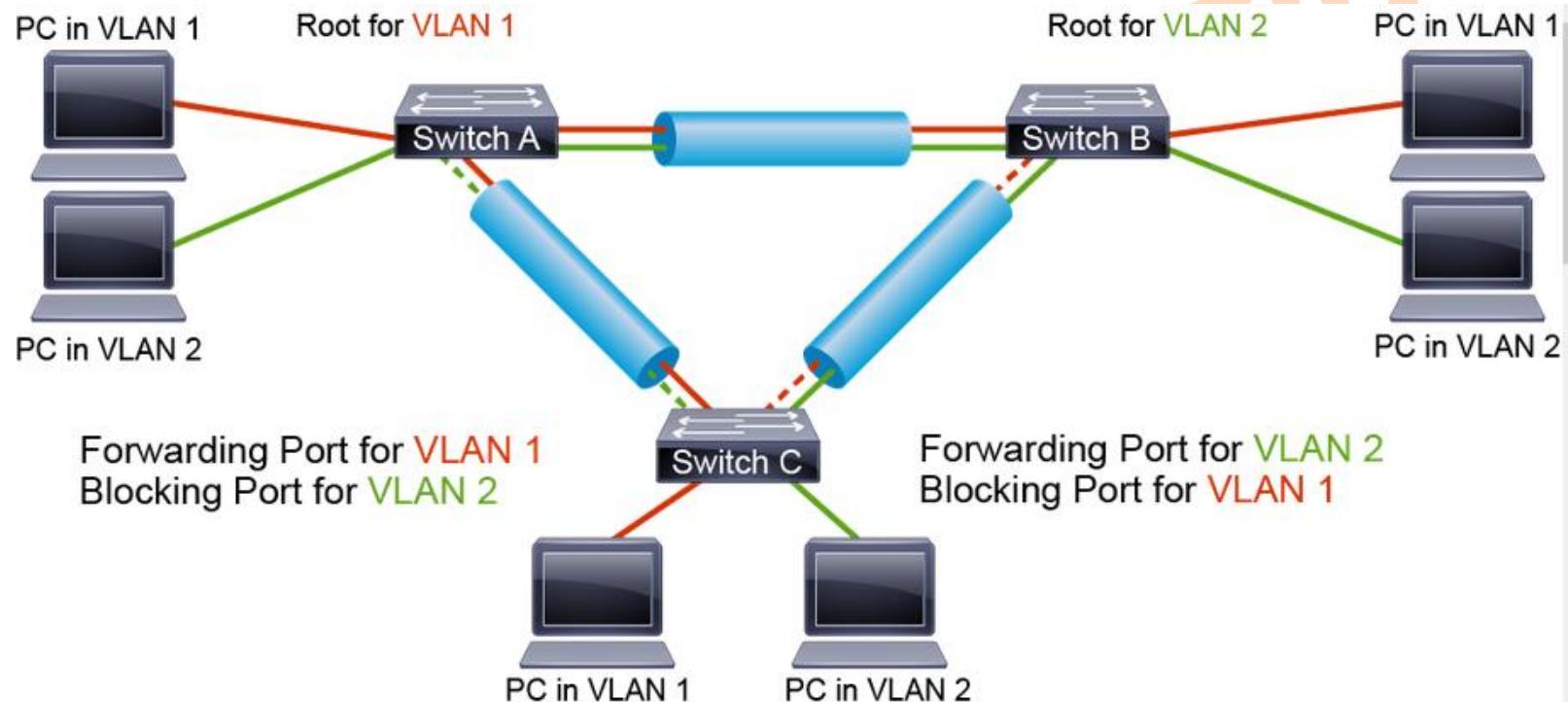
Several varieties of Spanning Tree Protocols exist:

- **STP (IEEE 802.1D)**
 - The legacy standard that provides a loop-free topology in a network with redundant links.
STP creates a **Common Spanning Tree (CST)** that assumes one spanning tree instance for the entire bridged network, regardless of the number of VLANs.
- **PVST+**
 - A Cisco enhancement of STP that provides a separate 802.1D spanning tree instance for each VLAN that is configured in the network.
- **MSTP (IEEE 802.1s)**
 - An IEEE standard that is inspired by the earlier Cisco proprietary Multi-Instance STP (MISTP) implementation.
MSTP maps *multiple VLANs* into the *same spanning tree instance*.
- **RSTP (IEEE 802.1w)**
 - An evolution of STP that provides *faster convergence* of STP. It redefines port roles and enhances BPDU exchanges.
- **Rapid PVST+**
 - A Cisco enhancement of RSTP that uses PVST+. Rapid PVST+ provides a separate instance of 802.1w per VLAN.

Types of Spanning Tree Protocols (Cont.)

- **PVST+**

- Unlike CST, PVST runs 1 spanning-tree instance for each VLAN. This condition allows you to **load-balance traffic over redundant links**, when they area assigned into a different VLAN.



Types of Spanning Tree Protocols (Cont.)

Comparison of Spanning Tree Protocols

Protocol	Standard	Resource Needed	Convergence	Number of Trees
STP	802.1D	Low	Slow	One
PVST+	Cisco	High	Slow	One for every VLAN
RSTP	802.1W	Medium	Slow	One
Rapid PVST+	Cisco	Very High	Fast	One for every VLAN
MSTP	802.1S	Medium or High	Fast	One for multiple VLANs

Default Spanning Tree Configuration

- Default spanning tree configuration for Cisco Catalyst switches includes the following characteristics:
 - PVST+
 - Enabled on all ports in VLAN 1
 - Slower convergence after topology change than with RSTP

Rapid Spanning Tree Protocol (Cont.)

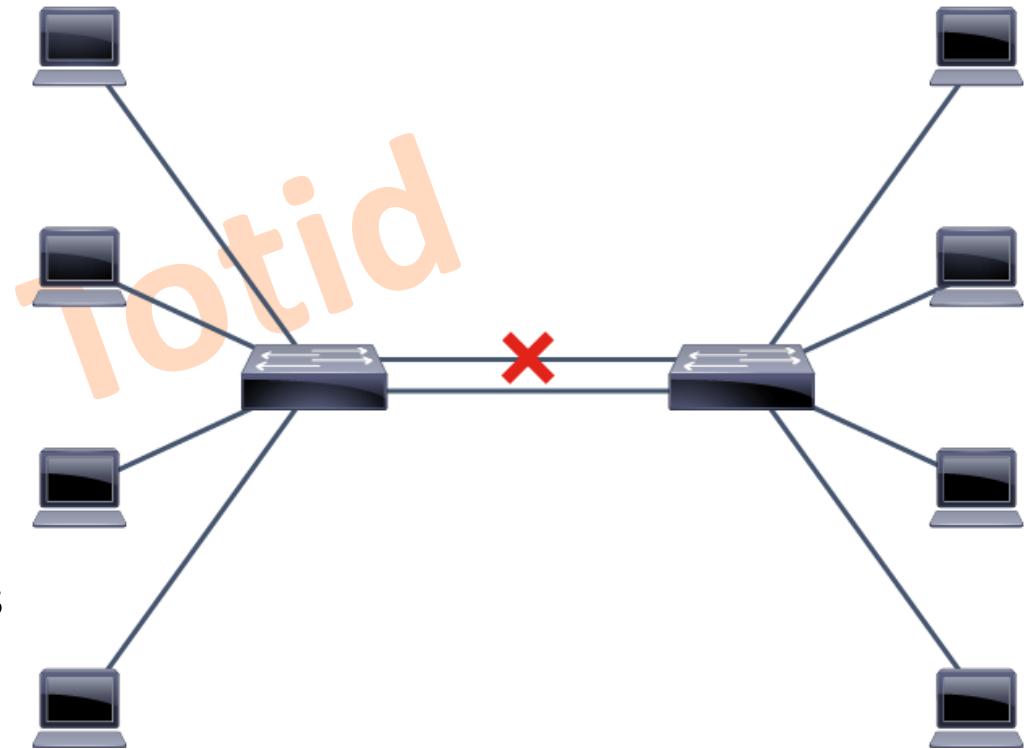
Comparison of RSTP and STP Port States

STP Port Role	STP Port State	RSTP Port Role	RSTP Port State
Root port	Forwarding	Root port	Forwarding
Designated port	Forwarding	Designated port	Forwarding
Nondesignated port	Blocking	Alternate or backup port	Discarding
Disabled	-	Disabled	Discarding
In transition	Listening Learning	<i>In transition</i>	Learning

- There is ***no listening state*** as there was with STP. The listening and blocking STP states are replaced with the ***discarding state***.
- In stable topology, RSTP ensures that every **root port** and **designated port** transit to ***forwarding***, while all **alternate ports** and **backup ports** are always in the ***discarding state***.

EtherChannel Overview

- When traffic from multiple devices is aggregated into one link, **congestion** may occur.
- Solutions** to avoid congestion include the following:
 - Upgrade links**
 - Cannot scale indefinitely
 - Can be expensive
 - Aggregate multiple links into one**
 - Control mechanisms, such as STP, might disable ports



Many other terms are used to name the aggregation concept; ***link-bundling***, ***NIC bonding***, ***NIC teaming***, ***network bonding***, and ***channel bonding***.

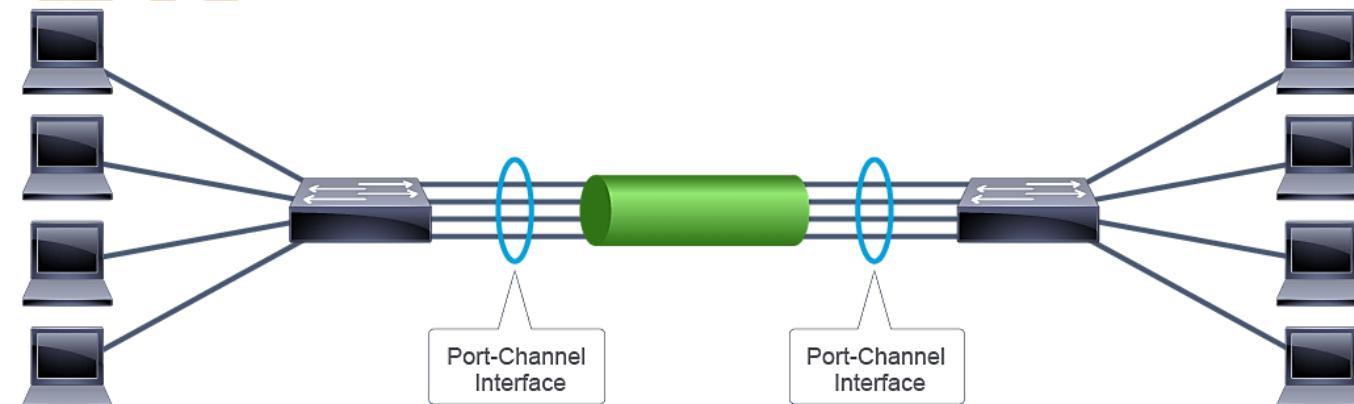
EtherChannel Overview (Cont.)

- EtherChannel enables packets to be sent over several physical interfaces as if over a single interface. EtherChannel ***logically bonds*** several physical connections into one logical connection.

- **Redundancy and Load Balancing**

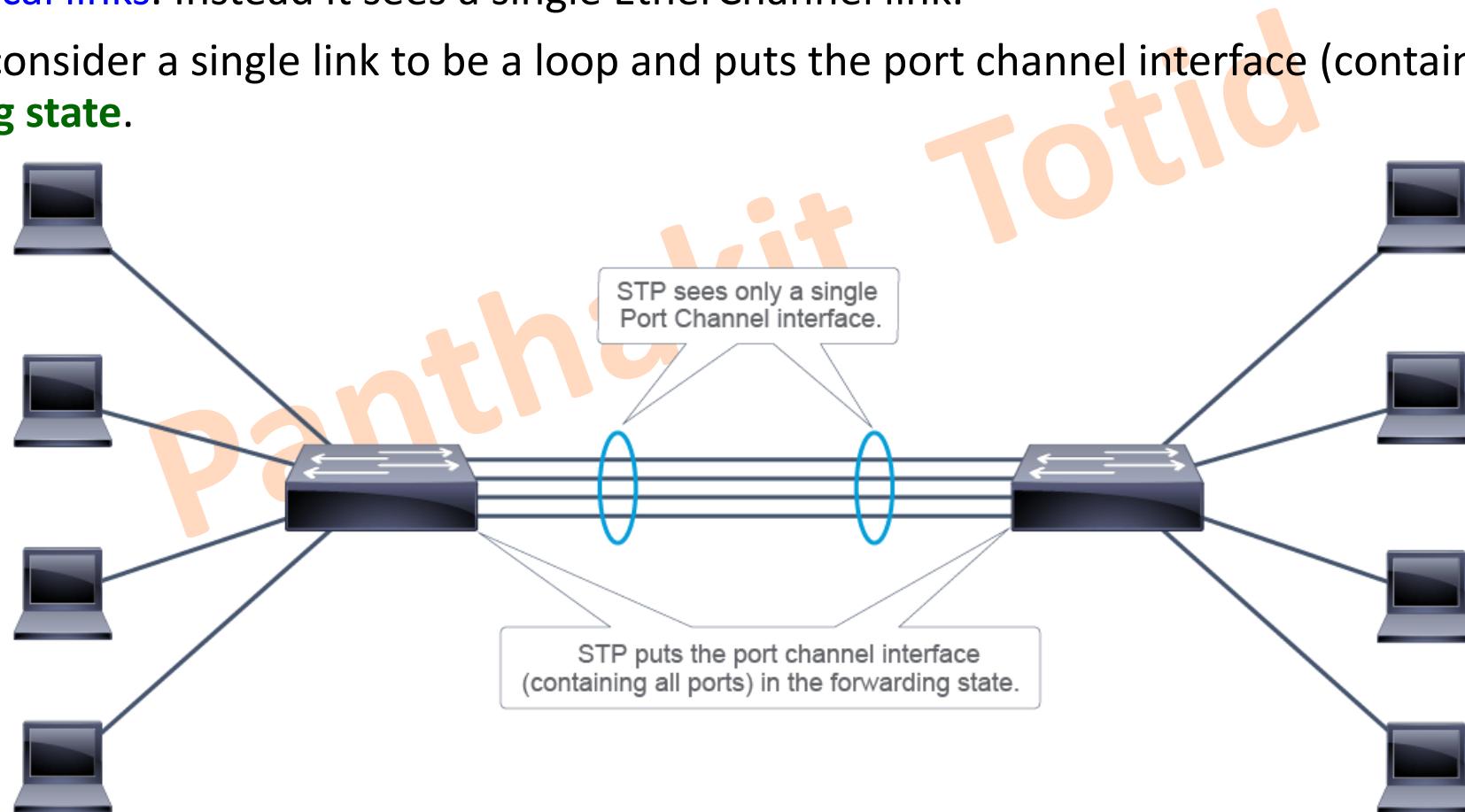


- EtherChannel bundles individual links into a channel group to create a single logical interface called a **port channel** that provides the aggregate bandwidth of several physical links.
 - Each link can be in only one port channel.
 - All the links in a port channel must be compatible. They must use the *same speed* and operate in *full-duplex* mode.



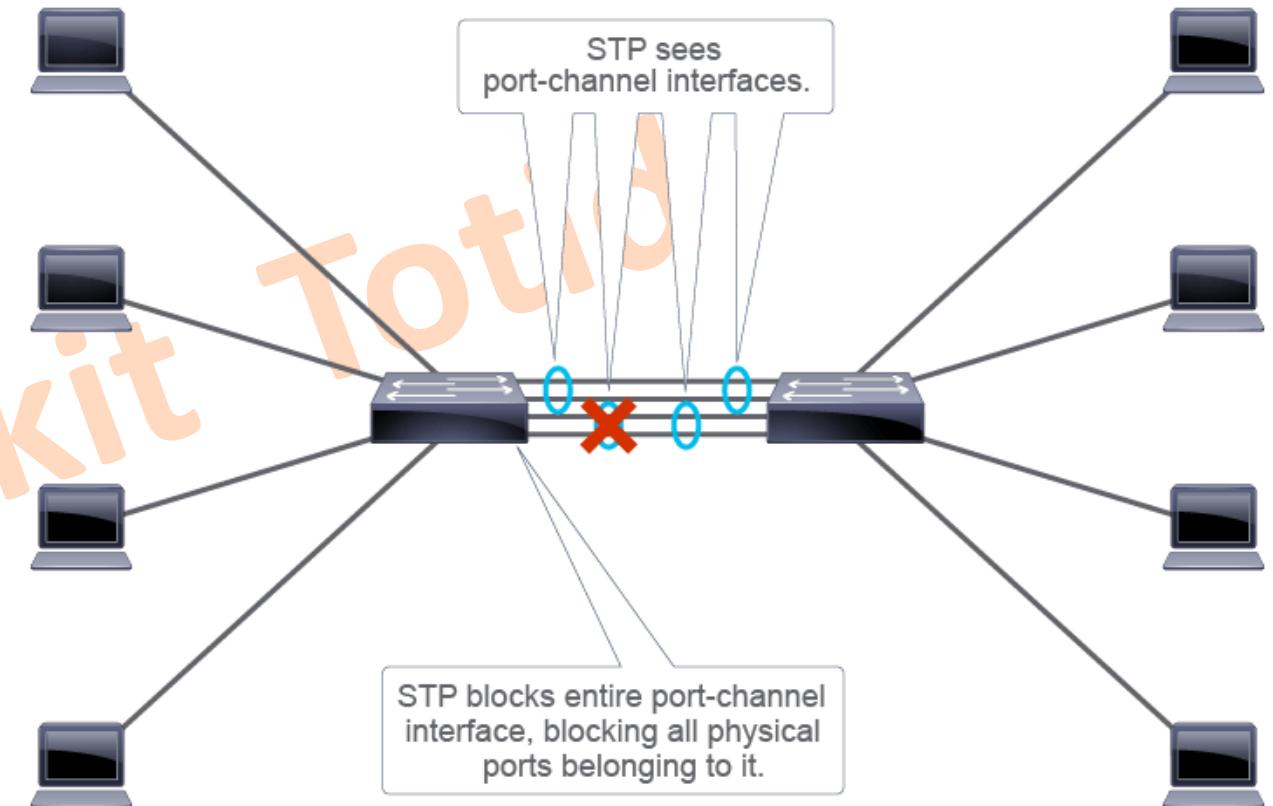
EtherChannel Overview (Cont.)

- Since the multiple physical links are bundled into a single EtherChannel, STP no longer sees them as **separate physical links**. Instead it sees a single EtherChannel link.
- STP does not consider a single link to be a loop and puts the port channel interface (containing all ports) in the **forwarding state**.



EtherChannel Overview (Cont.)

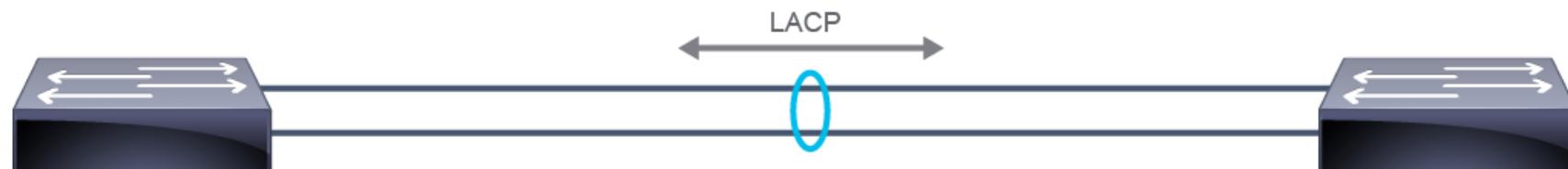
- You can also configure multiple EtherChannel links between two devices.
 - However, when several logical EtherChannel links exist between two switches, *STP detects loops*.
 - To avoid loops, STP will make **only one logical link operational**.
 - When STP blocks the redundant links, it blocks one entire EtherChannel, thus blocking all the ports belonging to that EtherChannel link.



EtherChannel Configuration Options

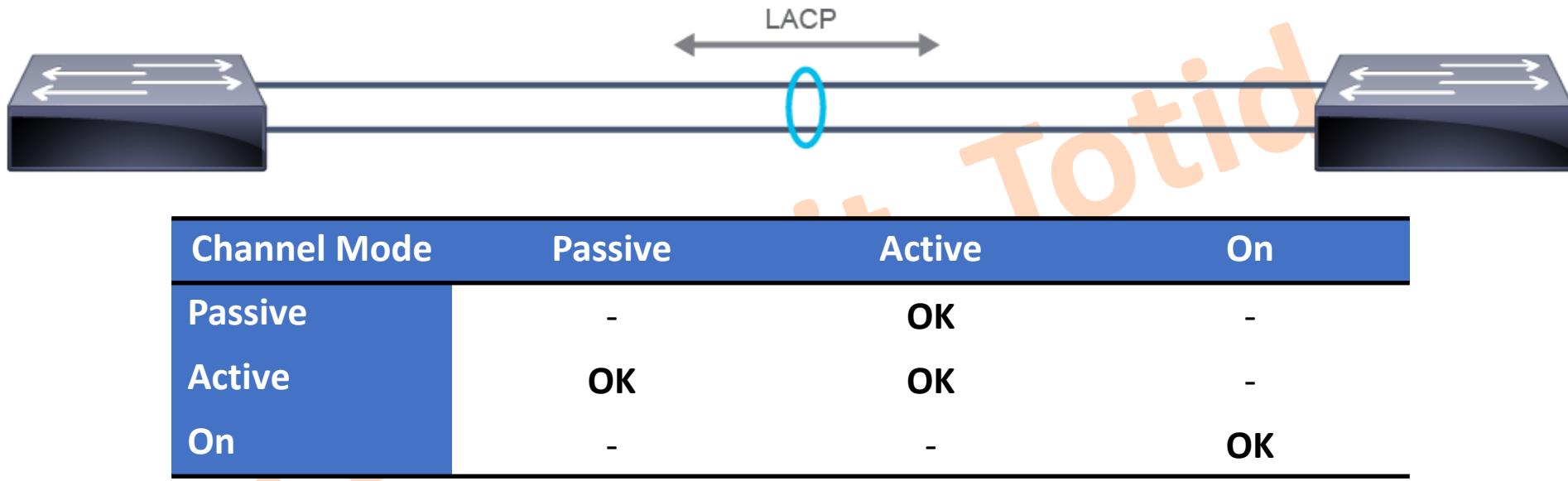
- The individual links must match on several parameters:
 - **Interface types** cannot be mixed
 - **Speed** and **duplex** settings must be the same on all the participating links.
 - **Switchport mode** and **VLAN information** must match.
- EtherChannel related physical interface configuration options are as follows:
 - **LACP modes**
 - **Passive:** Passively waiting for the other side to initiate negotiations
 - **Active:** Actively negotiating EtherChannel link establishment
 - **Static**, manual configuration mode
 - **On:** Unconditional EtherChannel member, nonegotiations performed.

First standardized of LACP is **IEEE 802.3AD**, currently defined in **IEEE 802.1AX**



EtherChannel Configuration Options (Cont.)

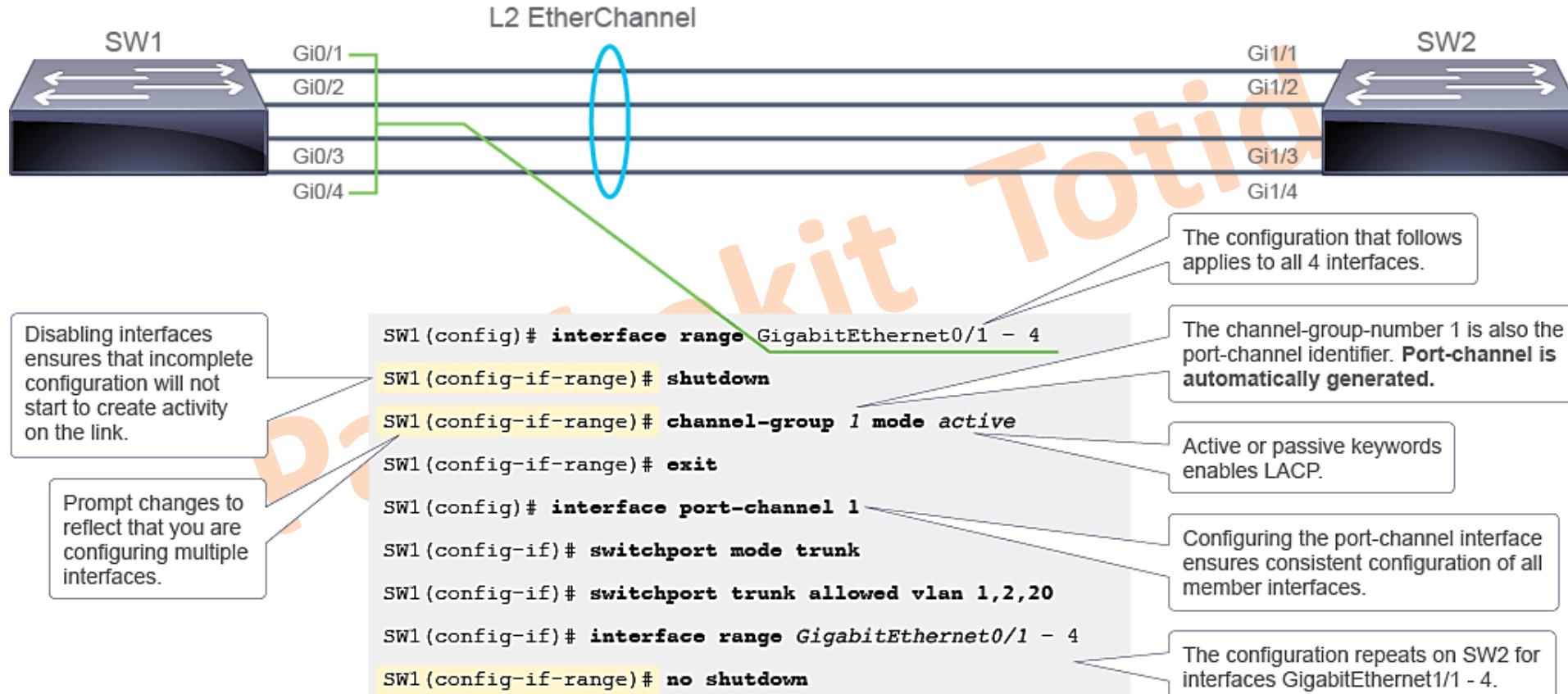
- For the EtherChannel link to form, modes on both sides of the individual links must be compatible.



- Once an EtherChannel is formed, whether by static configuration or dynamic negotiation, **if a link within the EtherChannel fails, the EtherChannel will still be functional**, as long as at least one physical link is active.

Configuring and Verifying EtherChannel

- To ensure configuration consistency of the physical interfaces, you can use the **interface range** command.



The channel-group identifier **does not need to match** on both sides of the port channel. However, it is a good practice to do so because it makes it easier to manage the configuration.

Configuring and Verifying EtherChannel (Cont.)

Verifying EtherChannel Configuration

- The `show interface port-channel` command displays the general status of the logical port channel interface that represents the aggregated link.
- In the example, the interface port-channel 1 is operational.

```
SW1# show interface Port-channel1
Port-channel1 is up, line protocol is up (connected)
  Hardware is EtherChannel, address is 000f.34f9.9182 (bia 000f.34f9.9182)
    MTU 1500 bytes, BW 200000 Kbit, DLY 100 usec,
      reliability 255/255, txload 1/255, rxload 1/255
    Encapsulation ARPA, loopback not set
<... output omitted ...>
```

Configuring and Verifying EtherChannel (Cont.)

Verifying EtherChannel Configuration

- The `show etherchannel summary` command displays one line of information per port channel and is particularly useful when several port channel interfaces are configured on the same device.
- The output of the command provides, among other, information on port channel interface status, method used for link aggregation, member interfaces, and their status.

```
SW2# show etherchannel summary
Flags:  D - down      P - bundled in port-channel
        I - stand-alone  S - suspended
        H - Hot-standby (LACP only)
        R - Layer3       S - Layer2
        U - in use       f - failed to allocate aggregator
        M - not in use, minimum links not met
        u - unsuitable for bundling
        w - waiting to be aggregated
        d - default port
Number of channel-groups in use: 1
Number of aggregators:          1
Group  Port-channel  Protocol    Ports
-----+-----+-----+
 1     Po1(SU)       LACP        Fa0/1(P)   Fa0/2(P)
```

Panthakit Totid

Panthakit Totid



Basic Network For Trainee

Module 8

VLANs and Trunks

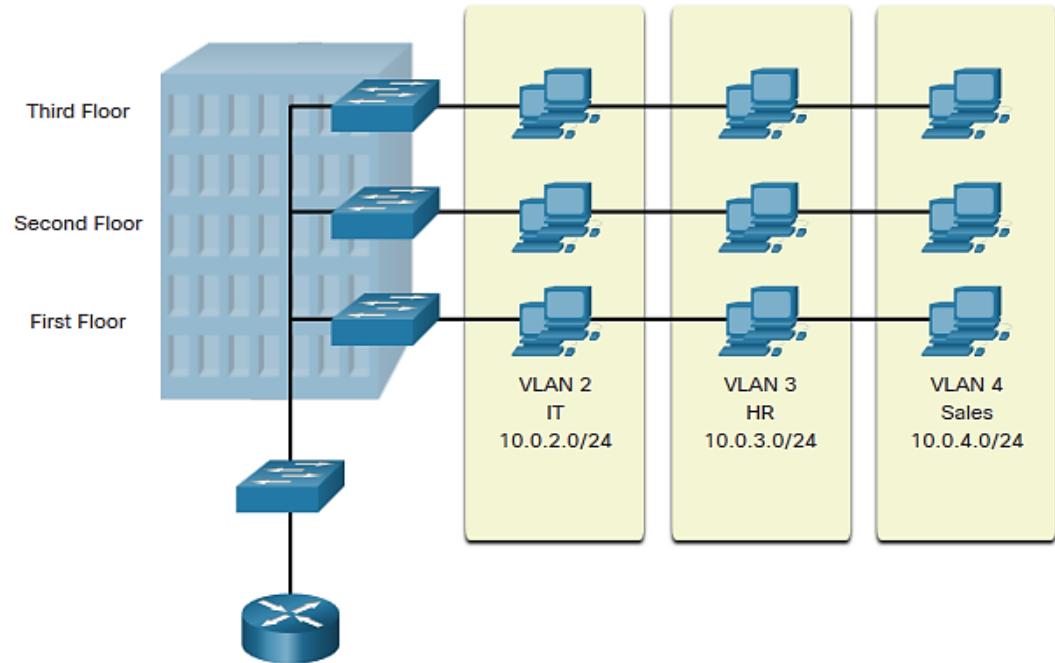
Panthakit Totid



Overview of VLANs

VLAN Definitions

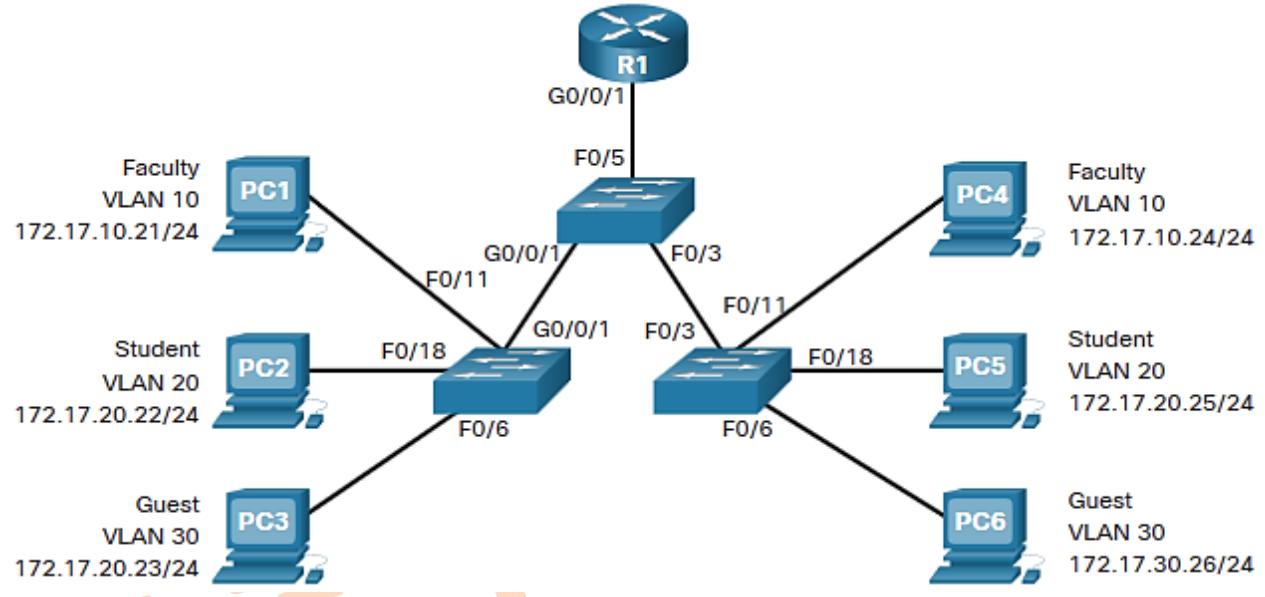
- VLANs are **logical connections** with other similar devices.
- Placing devices into various VLANs have the following characteristics:
 - Provides **segmentation** of the various groups of devices on the same switches
 - Provide organization that is **more manageable**
 - Broadcasts, multicasts and unicasts are **isolated** in the individual VLAN
 - Each VLAN will have its own unique range of IP addressing
 - Smaller broadcast domains



VLANs are mutually isolated and packets can only pass between VLANs via a router.

Overview of VLANs

Benefits of VLANs



Benefits	Description
Smaller Broadcast Domains	Dividing the LAN reduces the number of broadcast domains
Improved Security	Only users in the same VLAN can communicate together
Improved IT Efficiency	VLANs can group devices with similar requirements, e.g. faculty vs. students
Reduced Cost	One switch can support multiple groups or VLANs
Better Performance	Small broadcast domains reduce traffic, improving bandwidth
Simpler Management	Similar groups will need similar applications and other network resources

Overview of VLANs

Types of VLANs

Default VLAN

- **VLAN 1** is the following:
 - The default VLAN
 - The default Native VLAN
 - The default Management VLAN
 - Cannot be deleted or renamed

Switch# show vlan brief

VLAN Name	Status	Ports
1 default	active	Fa0/1, Fa0/2, 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, Fa0/24 Gig0/1, Gig0/2
1002 fddi-default	active	
1003 token-ring-default	active	
1004 fddinet-default	active	
1005 trnet-default	active	

Initially, all switch ports are members of VLAN 1.

- **Note:** While we **cannot delete VLAN1** Cisco will recommend that we assign these default features to other VLANs

Overview of VLANs

Types of VLANs (Cont.)

Data VLAN

- Dedicated to **user-generated traffic** (email and web traffic).
- VLAN 1 is the default data VLAN because all interfaces are assigned to this VLAN.

Native VLAN

- This is used for **trunk links only**.
- All frames are tagged on an 802.1Q trunk link except for those on the native VLAN.
- This is traffic that does not originate from a VLAN port (e.g., STP BPDU traffic exchanged between STP enabled switches)

Management VLAN

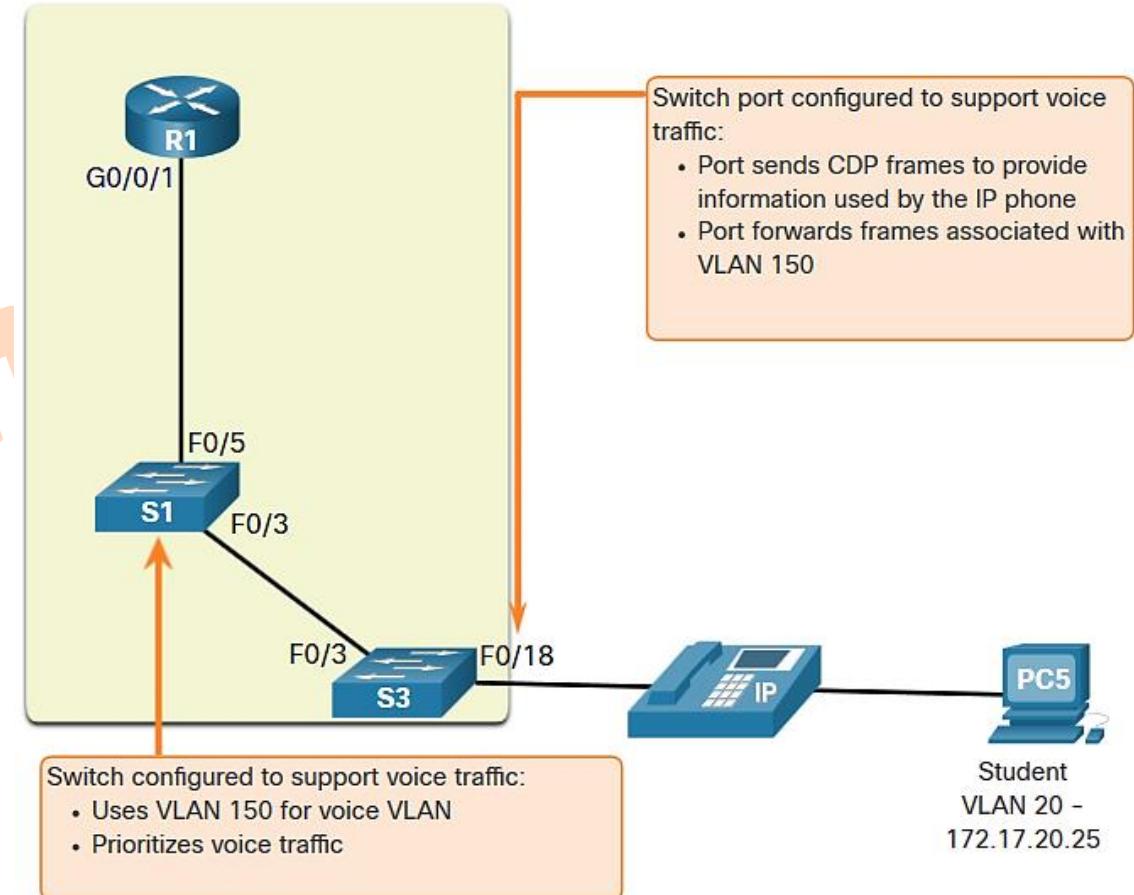
- This is used for **SSH/Telnet VTY traffic** and should not be carried with end user traffic.
- Typically, the VLAN that is the SVI for the Layer 2 switch.

Overview of VLANs

Types of VLANs (Cont.)

Voice VLAN

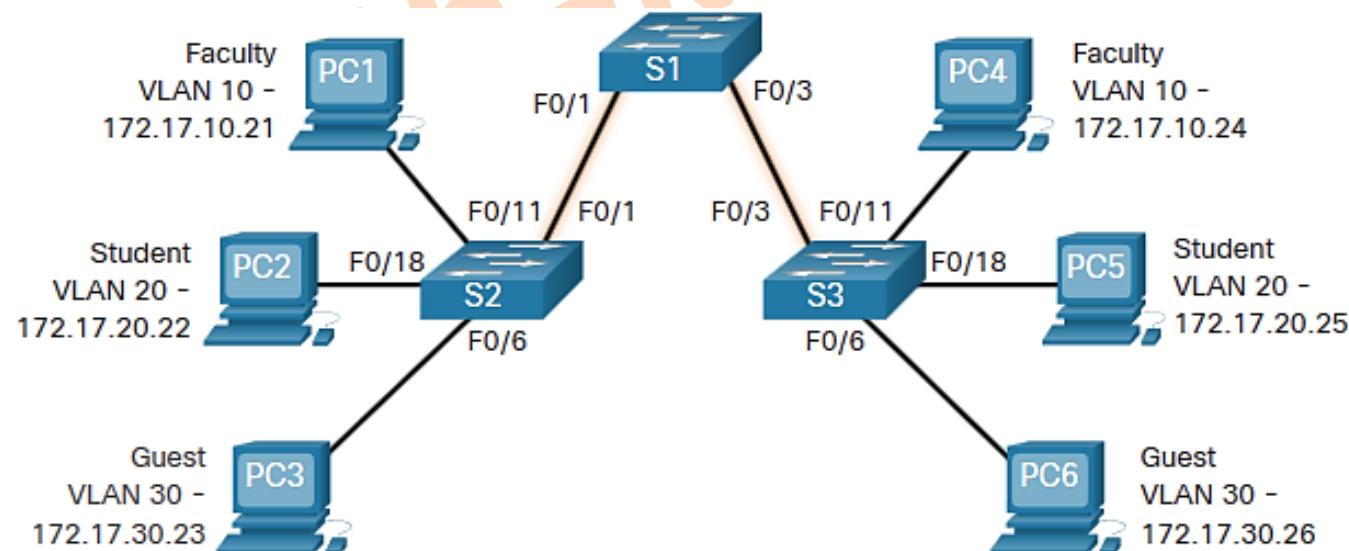
- A **separate VLAN** is required because Voice traffic requires:
 - Assured bandwidth
 - *High QoS priority*
 - Ability to avoid congestion
 - Delay less than **150 ms** from source to destination
- The entire network must be designed to support voice.



VLANs in a Multi-Switched Environment

Defining VLAN Trunks

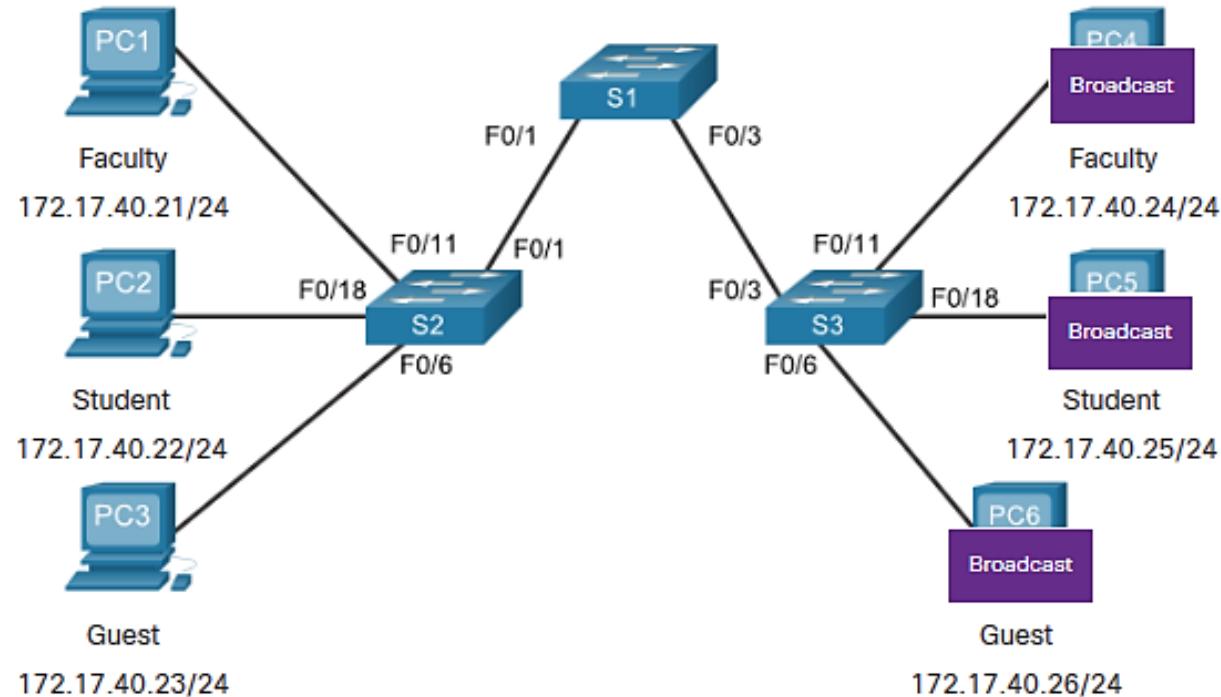
- A trunk is a **point-to-point link** between two network devices.
- Cisco trunk functions:
 - Allow more than one VLAN
 - Extend the VLAN across the entire network
 - **By default**, supports all VLANs
 - Supports 802.1Q trunking



VLANs in a Multi-Switched Environment

Networks without VLANs

- Without VLANs, all devices connected to the switches will receive all **unicast**, **multicast**, and **broadcast** traffic.

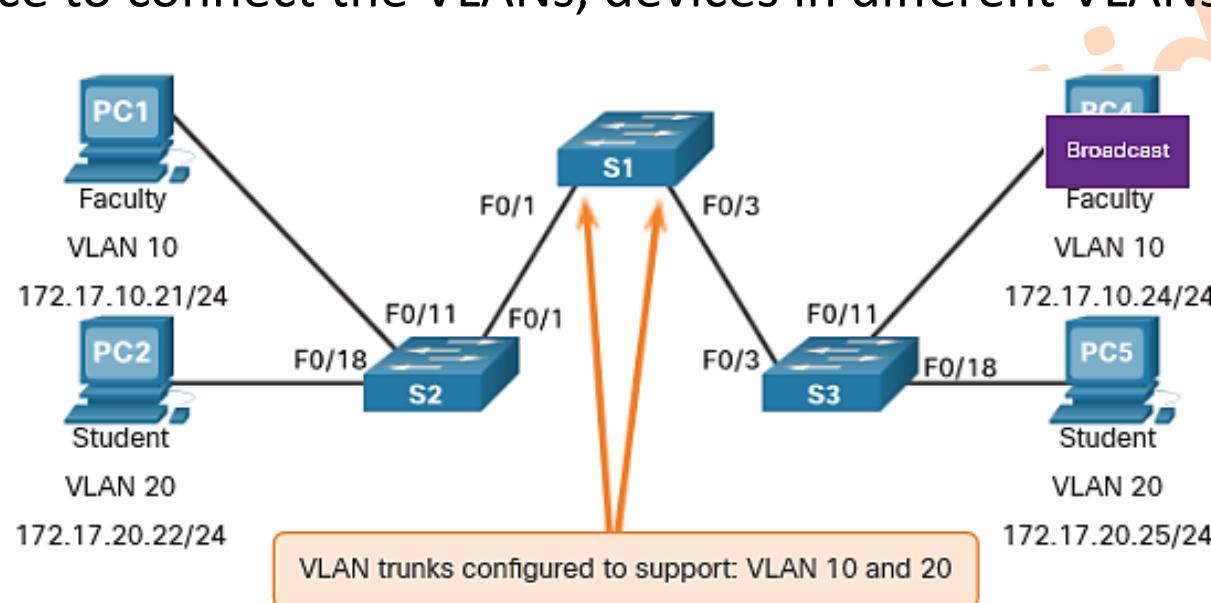


PC1 sends out a local L2 broadcast. The switches forward the broadcast frame out all available ports.

VLANs in a Multi-Switched Environment

Networks with VLANs

- With VLANs, unicast, multicast, and broadcast traffic is **confined** to a VLAN.
- Without a Layer 3 device to connect the VLANs, devices in different VLANs cannot communicate.

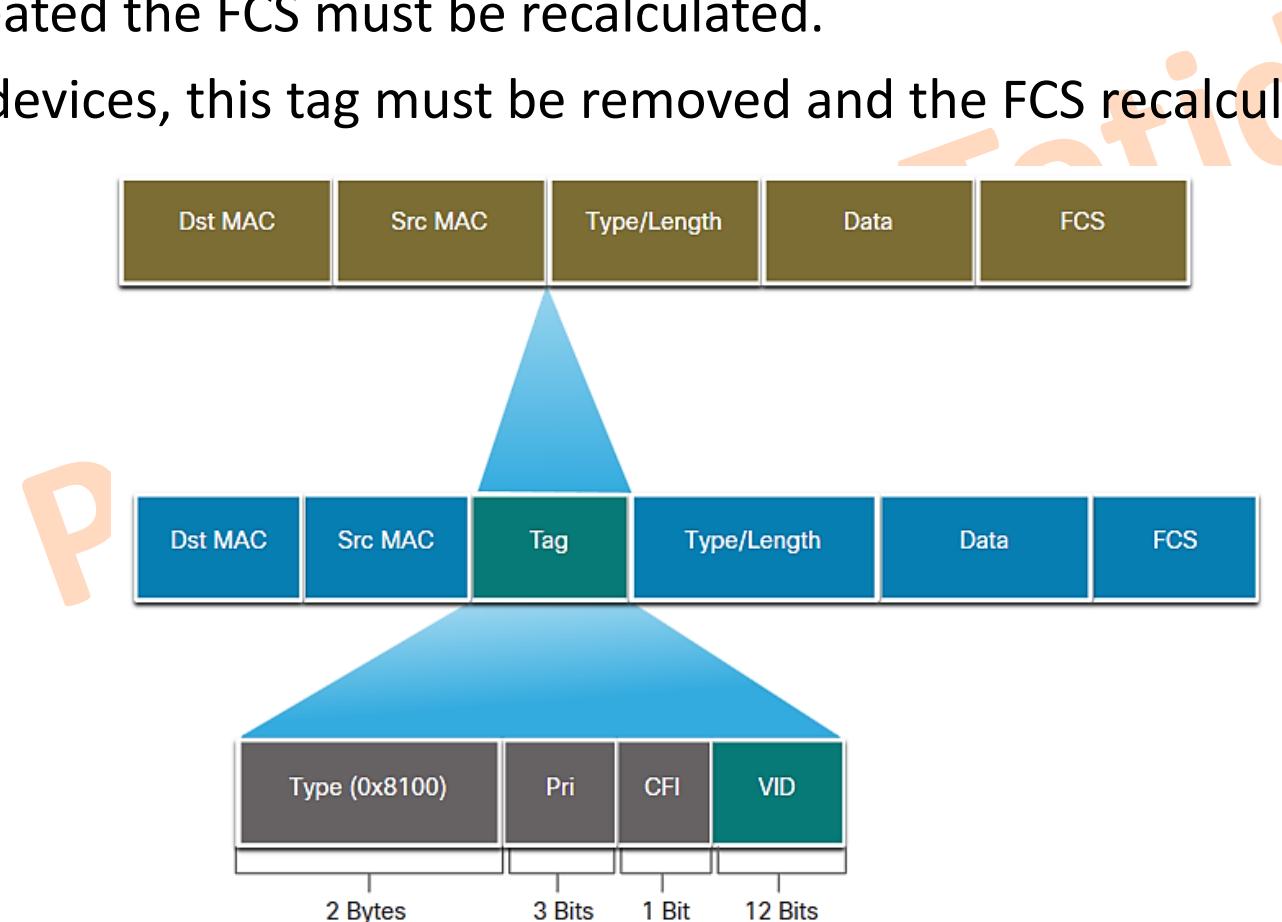


PC1 sends out a local L2 broadcast. The switches forward the broadcast frame only out ports configured for VLAN 10.

VLANs in a Multi-Switched Environment

VLAN Identification with a Tag

- The **IEEE 802.1Q** header is **4 Bytes**
- When the tag is created the FCS must be recalculated.
- When sent to end devices, this tag must be removed and the FCS recalculated back to its original number.

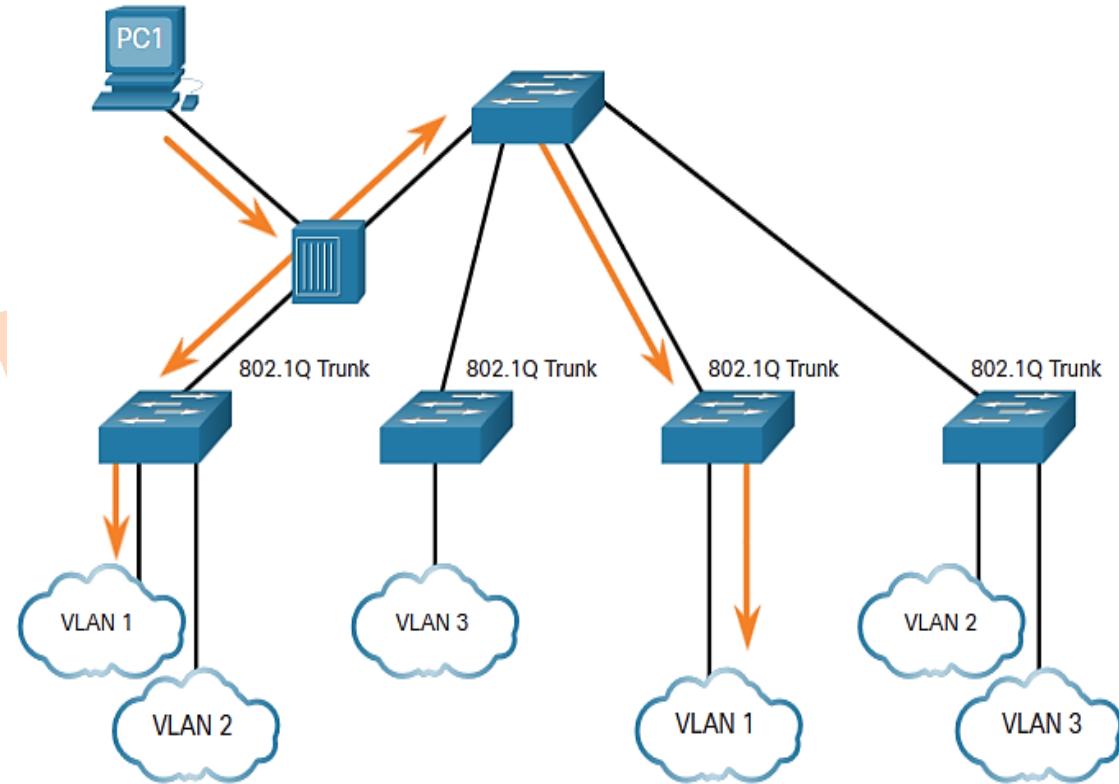


VLANs in a Multi-Switched Environment

Native VLANs and 802.1Q Tagging

- **802.1Q trunk basics:**

- Tagging is typically done on **all VLANs**.
- The use of a **native VLAN** was designed for legacy use, like the hub in the example.
 - Unless changed, **VLAN1** is the native VLAN.
- Both ends of a trunk link must be configured with **the same native VLAN**.
 - Each trunk is configured separately, so it is possible to have a different native VLANs on separate trunks.



VLAN Configuration

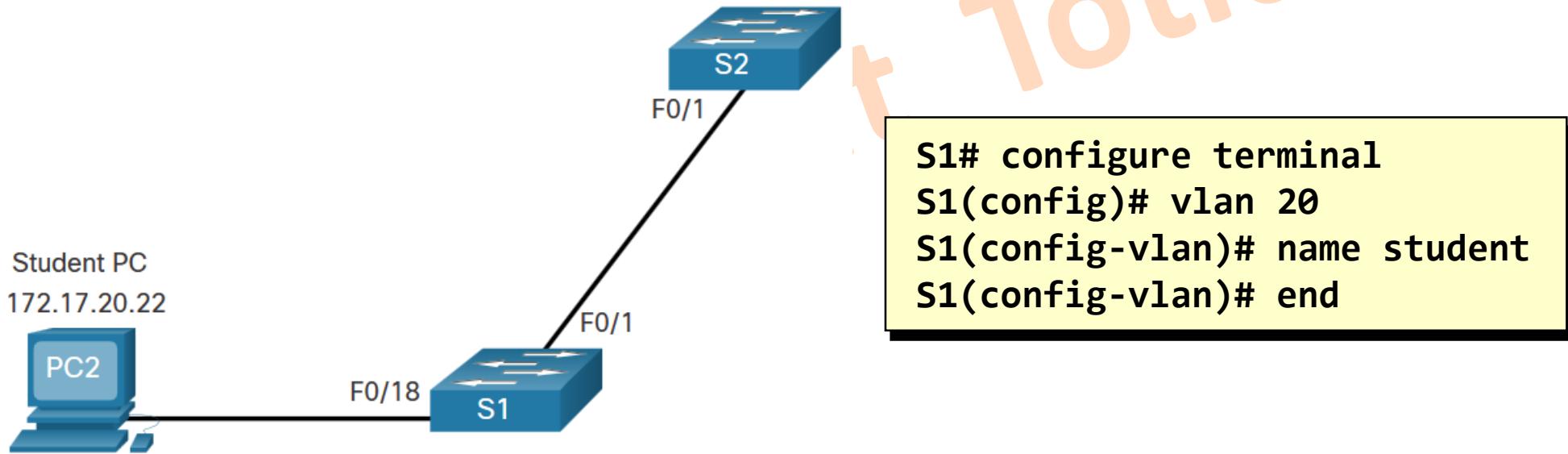
VLAN Ranges on Catalyst Switches

Normal Range VLAN 1 – 1005	Extended Range VLAN 1006 - 4095
Used in Small to Medium sized businesses	Used by Service Providers
1002 - 1005 are reserved for legacy VLANs	Are in Running-Config
1, 1002 - 1005 are auto created and cannot be deleted	Supports fewer VLAN features
Stored in the vlan.dat file in flash	Requires VTP configurations
VTP can synchronize between switches	

VLAN Configuration

VLAN Creation Example

- If the Student PC is going to be in VLAN 20, we will create the VLAN first and then name it.
- **If you do not name it**, the Cisco IOS will give it a default name of vlan and the four digit number of the VLAN. E.g. vlan0020 for VLAN 20.



VLAN Configuration

VLAN Port Assignment Commands

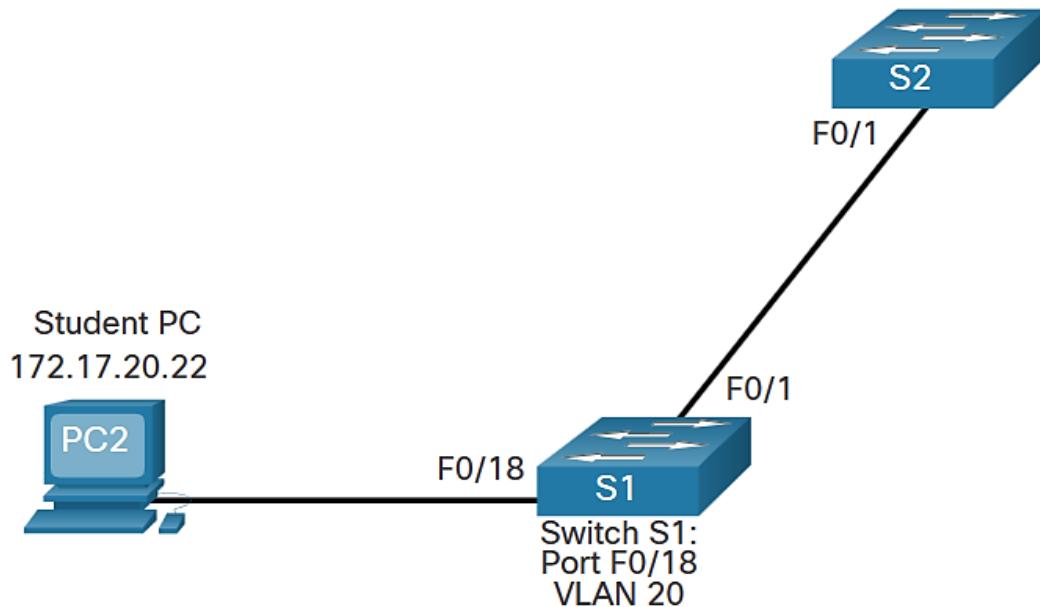
- Once the VLAN is created, we can then assign it to the correct interfaces.

Task	Command
Enter global configuration mode.	Switch# configure terminal
Enter interface configuration mode.	Switch(config)# interface <i>interface-id</i>
Set the port to access mode.	Switch(config-if)# switchport mode access
Assign the port to a VLAN.	Switch(config-if)# switchport access vlan <i>vlan-id</i>
Return to the privileged EXEC mode.	Switch(config-if)# end

VLAN Configuration

VLAN Port Assignment Example

- We can assign the VLAN to the port interface.
- Once the device is assigned the VLAN, then the end device will need the IP address information for that VLAN



```
S1# configure terminal  
S1(config)# interface fa0/18  
S1(config-if)# switchport mode access  
S1(config-if)# switchport access vlan 20  
S1(config-if)# end
```

VLAN Assignment

Changing VLAN Port Membership

- There are a number of ways to change VLAN membership:
 - re-enter **switchport access vlan *vlan-id*** command
 - use the **no switchport access vlan** to place interface back in VLAN 1
- Use the **show vlan brief** or the **show interface fa0/18 switchport** commands to verify the correct VLAN association.

```
S1(config)# interface fa0/18
S1(config-if)# no switchport access vlan
S1(config-if)# end
S1#
S1# show vlan brief
VLAN Name          Status      Ports
---- -----
1    default        active     Fa0/1, Fa0/2, 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, Fa0/24
                           Gi0/1, Gi0/2
20   student         active
1002 fddi-default   act/unsup
1003 token-ring-default act/unsup
1004 fddinet-default act/unsup
1005 trnet-default  act/unsup
```

```
S1# show interfaces fa0/18 switchport
Name: Fa0/18
Switchport: Enabled
Administrative Mode: static access
Operational Mode: static access
Administrative Trunking Encapsulation: negotiate
Operational Trunking Encapsulation: native
Negotiation of Trunking: Off
Access Mode VLAN: 1 (default)
Trunking Native Mode VLAN: 1 (default)
```

VLAN Assignment

Deleting VLANs

- Use the `no vlan vlan-id` global configuration mode command to remove VLAN.
- **Caution:** Before deleting a VLAN, reassign all member ports to a different VLAN.
 - Delete all VLANs with the `delete flash:vlan.dat` or `delete vlan.dat` commands.
 - Reload the switch when deleting all VLANs.

```
S1# conf t
S1(config)# no vlan 20
S1(config)# end
S1#
S1# sh vlan brief

VLAN Name          Status    Ports
---- ----
1    default        active   Fa0/1, Fa0/2, Fa0/3, Fa0/4
                           Fa0/5, Fa0/6, 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, Gi0/1
                           Gi0/2

1002 fddi-default   act/unsup
1003 token-ring-default   act/unsup
1004 fddinet-default   act/unsup
1005 trnet-default    act/unsup
S1#
```

Note: To restore to factory default – unplug all data cables, erase the startup-configuration and delete the `vlan.dat` file, then reload the device.

VLAN Assignment

Verifying VLAN Information

- VLAN configurations can be validated using the Cisco IOS `show vlan` and `show interfaces` command options.

```

S1# show vlan name student

VLAN Name                               Status    Ports
-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
20   student                            active   Fa0/11, Fa0/18

VLAN Type SAID MTU Parent RingNo BridgeNo Stp BrdgMode Trans1 Trans2
-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
20   enet 100020 1500 -      -      -      -      -      0      0

Remote SPAN VLAN
-----
Disabled

Primary Secondary Type          Ports
-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+

```

S1# show vlan summary

Number of existing VLANs	:	7
Number of existing VTP VLANs	:	7
Number of existing extended VLANs	:	0

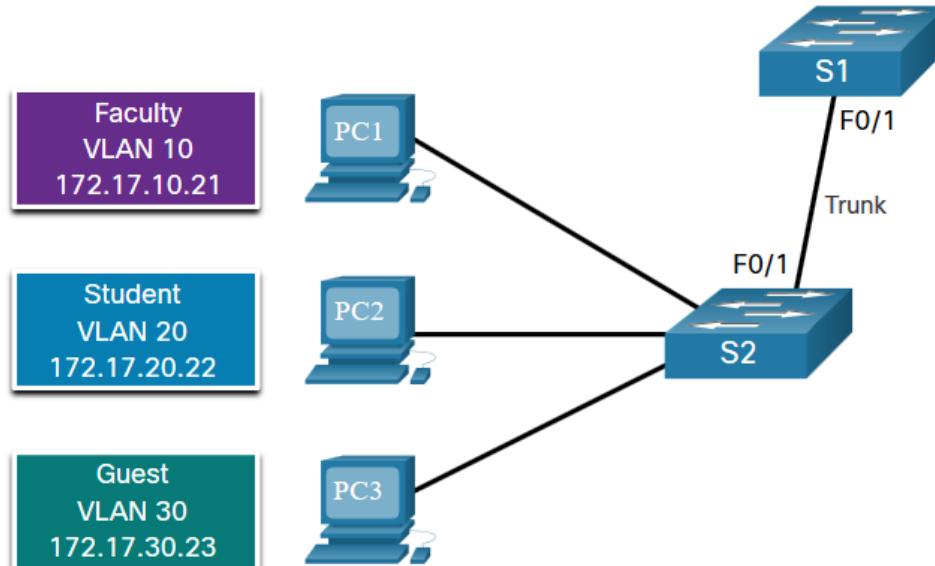
S1#

```
S1# show interfaces vlan 20
Vlan20 is up, line protocol is down
  Hardware is EtherSVI, address is 001c.57ec.0641 (bia 001c.57ec.0641)
  MTU 1500 bytes, BW 1000000 Kbit, DLY 10 usec,
    reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation ARPA, loopback not set
  ARP type: ARPA, ARP Timeout 04:00:00
  Last input never, output never, output hang never
  Last clearing of "show interface" counters never
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: fifo
  Output queue: 0/40 (size/max)
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
    0 packets input, 0 bytes, 0 no buffer
    Received 0 broadcasts (0 IP multicast)
    0 runts, 0 giants, 0 throttles
    0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored
    0 packets output, 0 bytes, 0 underruns
    0 output errors, 0 interface resets
    0 output buffer failures, 0 output buffers swapped out
```

VLAN Trunks

Trunk Configuration Example

- The subnets associated with each VLAN are:
 - VLAN 10** - Faculty/Staff - 172.17.10.0/24
 - VLAN 20** - Students - 172.17.20.0/24
 - VLAN 30** - Guests - 172.17.30.0/24
 - VLAN 99** - Native - 172.17.99.0/24
- F0/1 port on S1 is configured as a trunk port.
- Note:** This assumes a 2960 switch using 802.1q tagging. Layer 3 switches require the encapsulation to be configured before the trunk mode.



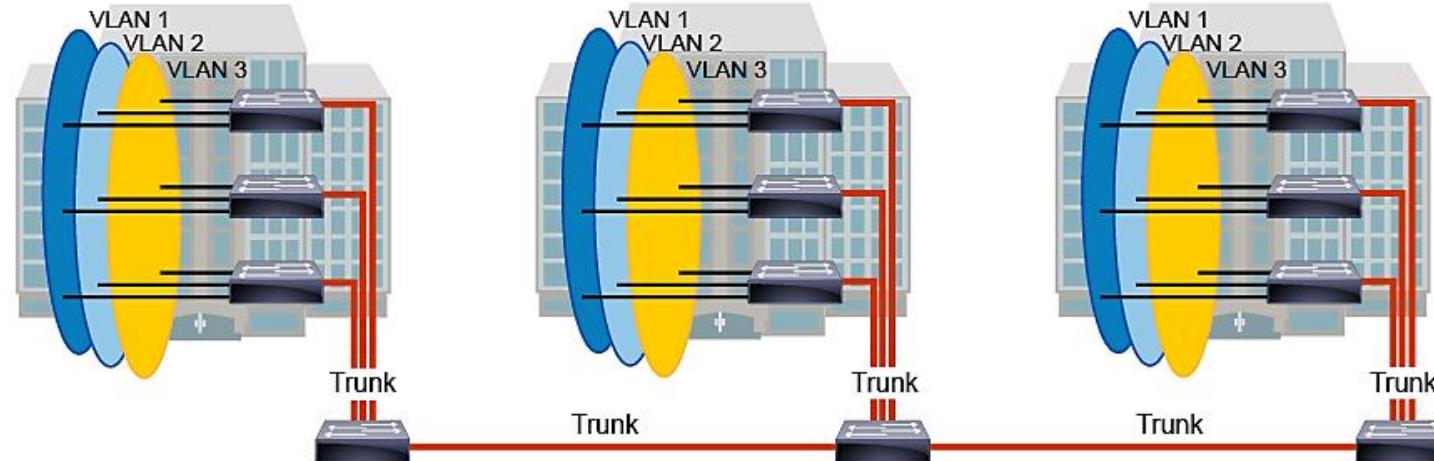
```
S1(config)# interface fa0/1
S1(config-if)# switchport mode trunk
S1(config-if)# switchport trunk allowed vlan 10,20,30,99
S1(config-if)# switchport trunk native vlan 99
S1(config-if)# end
```

Deploying VLANs

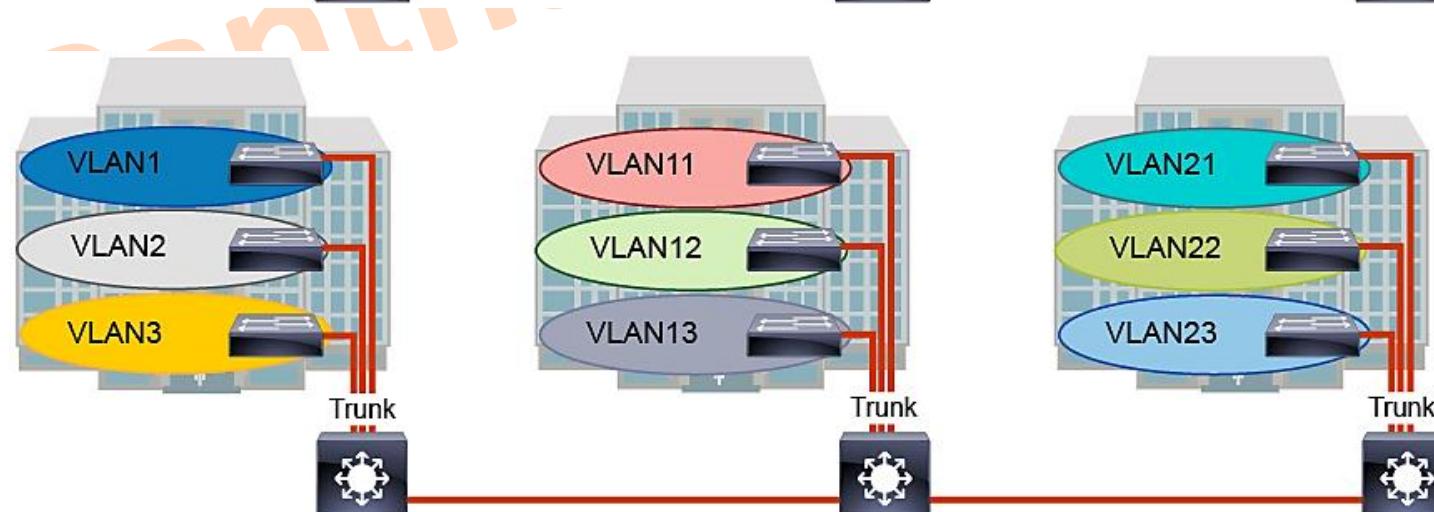
- Larger **flat networks** generally consist of many end devices in which **broadcasts** and **unknown unicast packets** are flooded on all ports in the network.
- One advantage of using VLANs is the capability to **segment the Layer 2 broadcast domain**.
 - All devices in a VLAN are members of the same broadcast domain.
 - If an end device transmits a Layer 2 broadcast, all other members of the VLAN receive the broadcast.
 - Switches filter the broadcast from all the ports or devices that are not part of the same VLAN.
- In a campus design, a network administrator can design a campus network with one of two models:
 - **End-to-End VLANs**
 - **Local VLANs**

Deploying VLANs

- End-to-End VLANs



- Local VLANs



End-to-End vs. Local VLANs

End-to-End VLANs	Local VLANs
<p>Pros:</p> <ul style="list-style-type: none">• Geographically dispersed users appear on the same segment.• The same policy (security, QoS) can be applied to the same group of users, regardless of their physical location.	<p>Pros:</p> <ul style="list-style-type: none">• Design is scalable.• Troubleshooting is easy.• Traffic flow is predictable.• Redundant paths can be built easily.
<p>Cons:</p> <ul style="list-style-type: none">• All switches need to know all VLANs.• Broadcast messages flood all switches.• Troubleshooting can be challenging.	<p>Cons:</p> <ul style="list-style-type: none">• More routing devices are required than in end-to-end models.• Users belong to the same broadcast domain when they are at the same location.

Panthakit Totid

Panthakit Totid



Basic Network For Trainee

Module 9

Routing Between VLANs

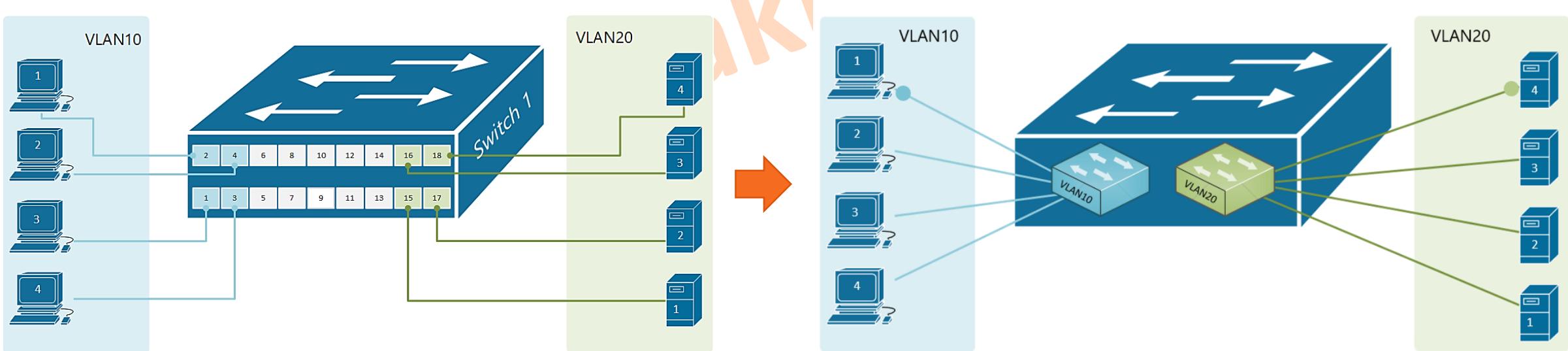
Panthakit Totid



Inter-VLAN Routing Operation

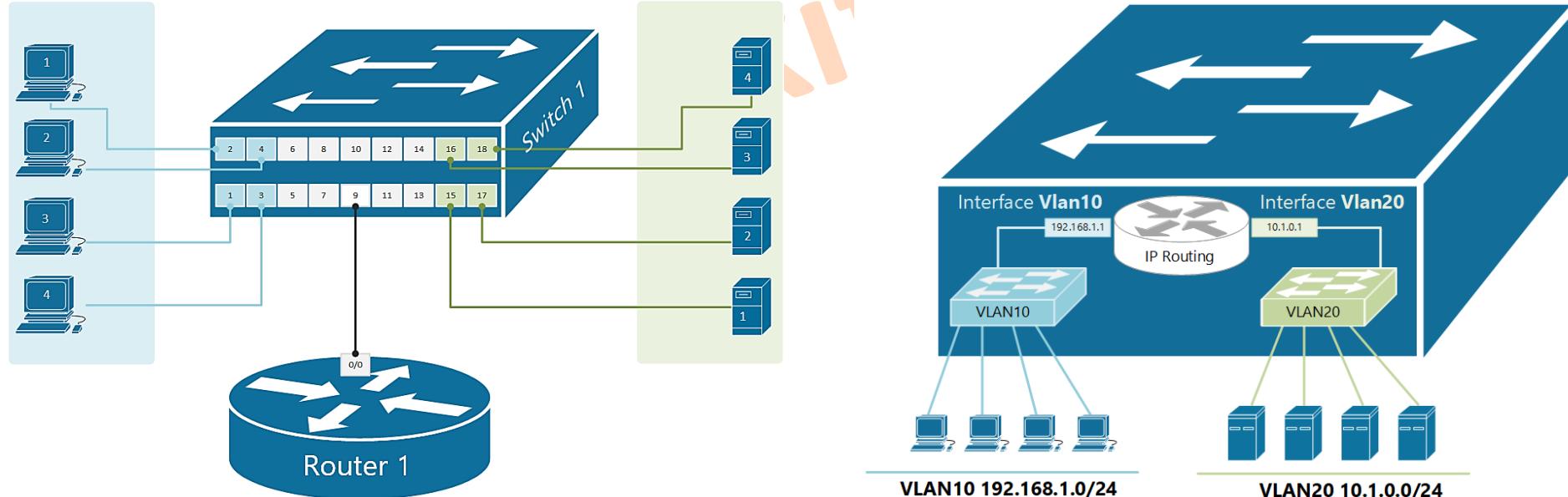
What is Inter-VLAN Routing?

- VLANs are used to segment switched Layer 2 networks for a variety of reasons.
 - Regardless of the reason, hosts in one VLAN cannot communicate with hosts in another VLAN unless there is a router or a Layer 3 switch to provide routing services.
- Inter-VLAN routing is a process for **forwarding network traffic from one VLAN to another, using a router.**



Options For Inter-VLAN Routing

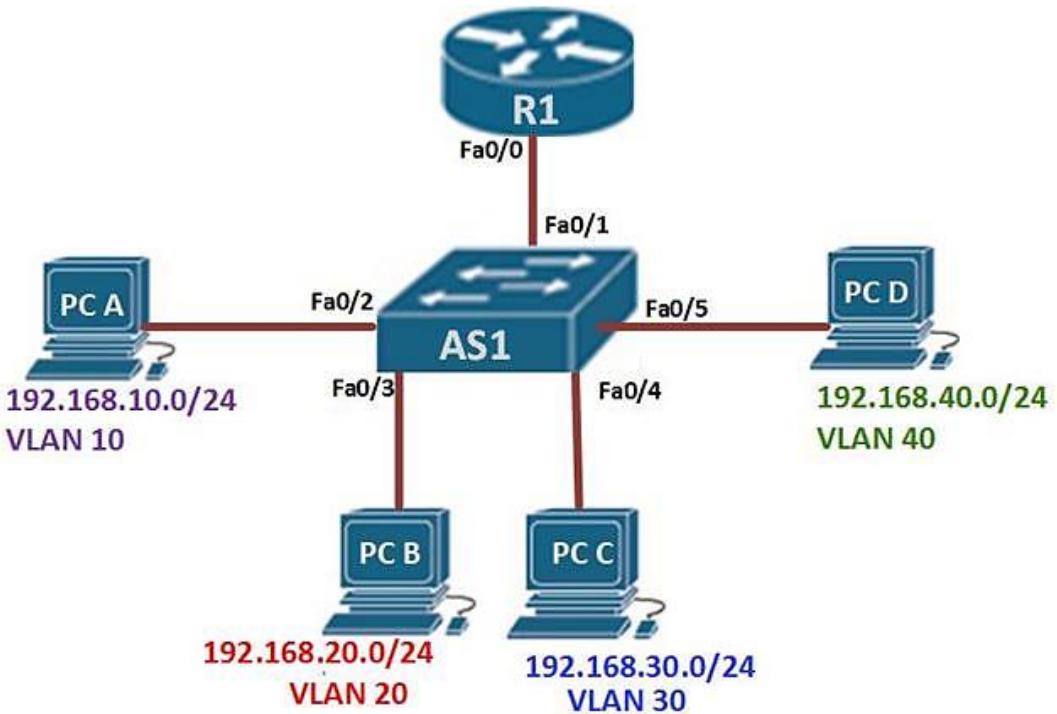
- There are three options for inter-VLAN routing:
 - **Legacy inter-VLAN routing** - This is a legacy solution. It does not scale well.
 - **Router-on-a-Stick** - This is an acceptable solution for a small to medium-sized network.
 - **Layer 3 switching using SVIs** - This is the most scalable solution for medium to large organizations.



Options For Inter-VLAN Routing

Router-on-a-Stick

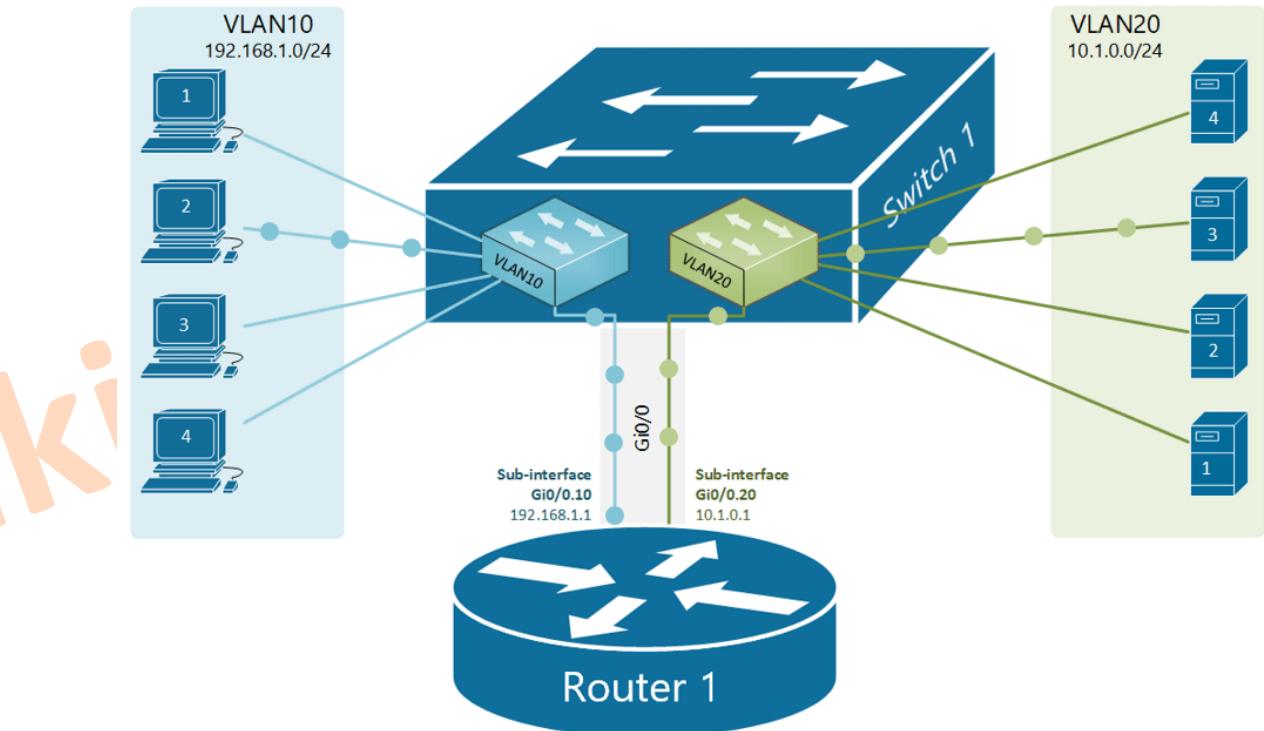
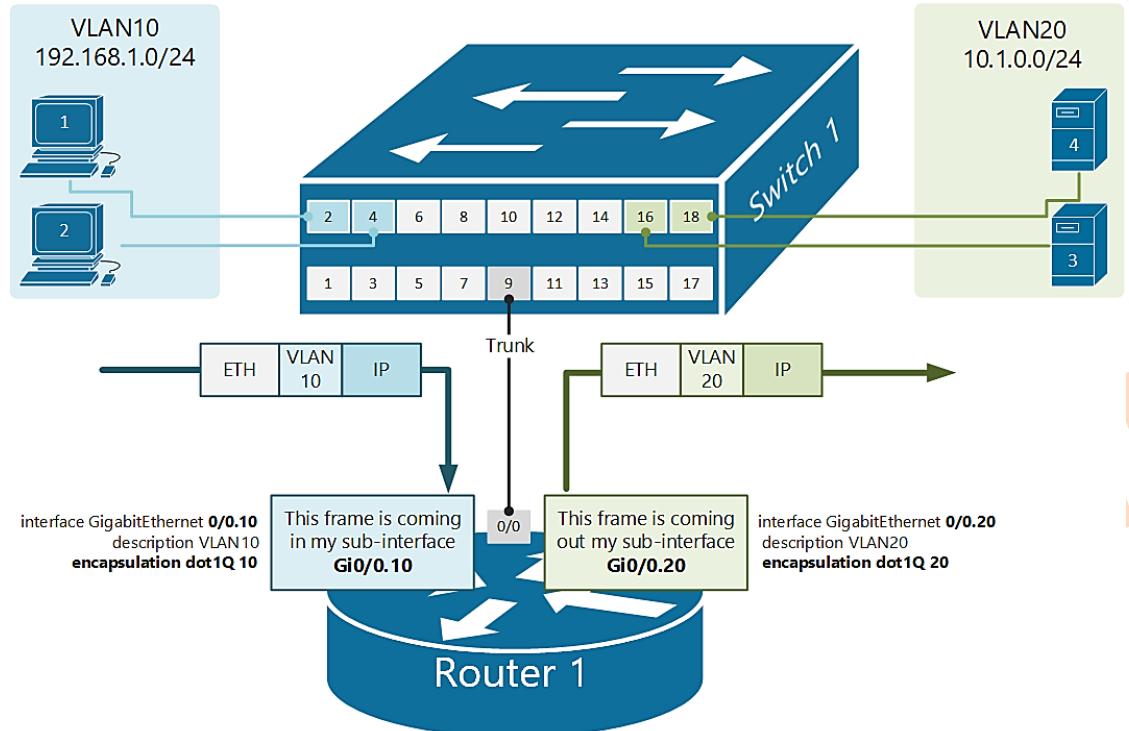
- The router-on-a-stick approach uses **only one of the router's physical interface**.
 - One of the router's physical interfaces is configured as a 802.1Q trunk port so it can understand VLAN tags.
 - Logical subinterfaces are created; **one subinterface per VLAN**.
 - Each subinterface is configured with an IP address from the VLAN it represents.
 - VLAN members (hosts) are configured to use the **subinterface address as a default gateway**.



In this example, the R1 interface is configured as a trunk link and connects to the trunk F0/1 port on AS1.

Options For Inter-VLAN Routing (Cont.)

Router-on-a-Stick

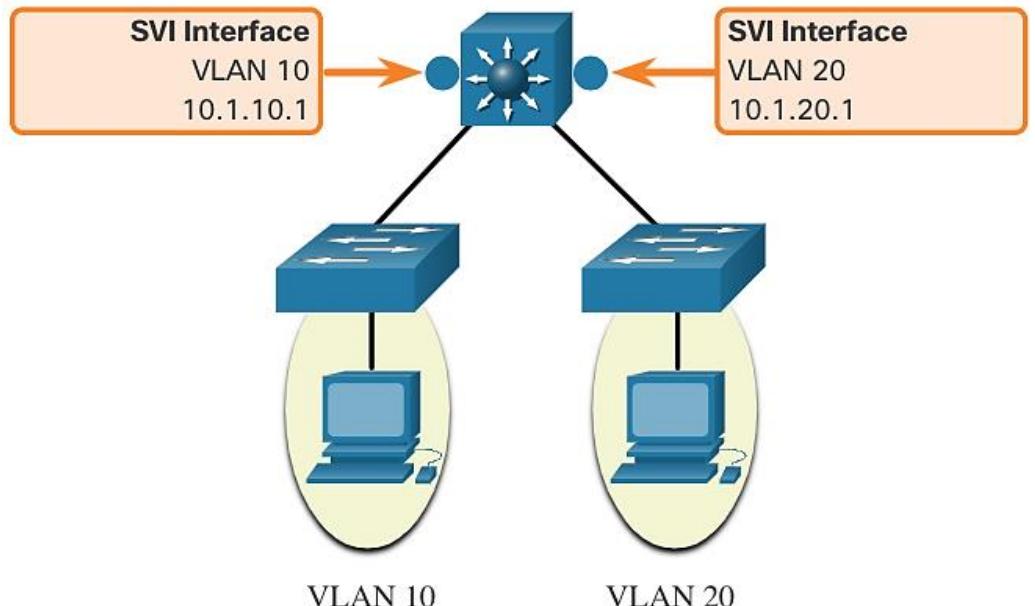


```
R1(config)# interface GigabitEthernet 0/0.10
R1(config-subif)# encapsulation dot1q 10
R1(config-subif)# ip address 192.168.1.1 255.255.255.0
R1(config-subif)# interface GigabitEthernet 0/0.20
R1(config-subif)# encapsulation dot1q 20
R1(config-subif)# ip address 10.1.0.1 255.255.255.0
```

Options For Inter-VLAN Routing (Cont.)

Layer 3 Switch

- The modern method of performing inter-VLAN routing is to use Layer 3 switches and **switched virtual interfaces (SVI)**.
- An SVI is a virtual interface that is configured on a Layer 3 switch, as shown in the figure.

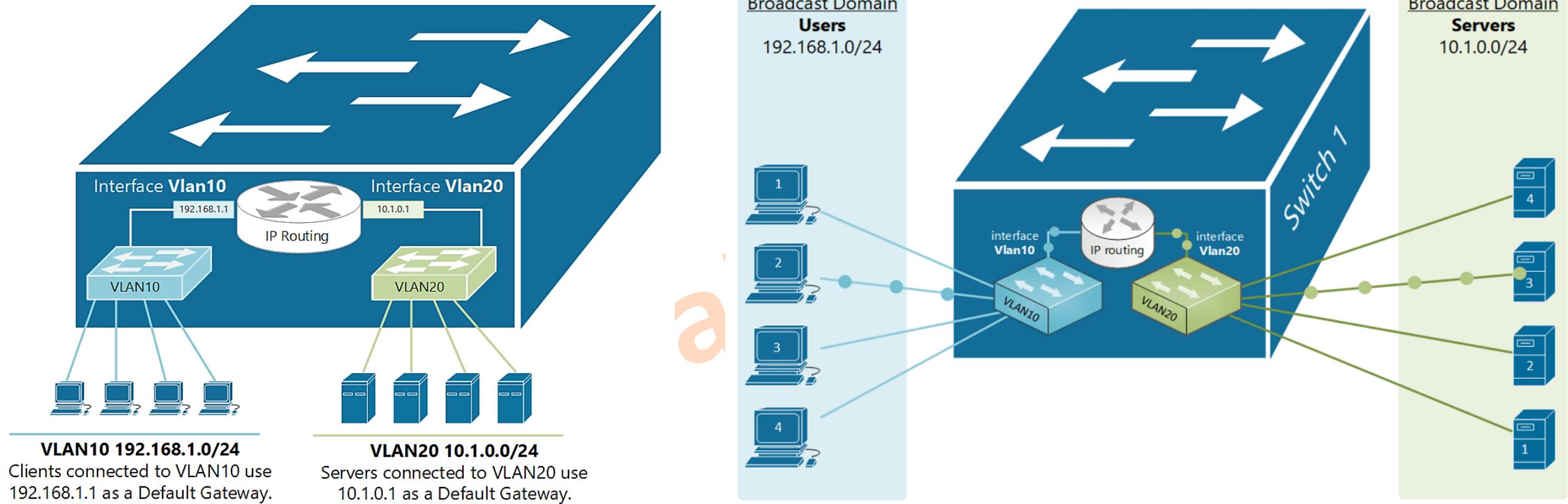


```
ip routing
!
interface Vlan10
 ip address 10.1.10.1 255.255.255.0
 no shutdown
!
interface Vlan20
 ip address 10.1.20.1 255.255.255.0
 no shutdown
```

A Layer 3 switch is also called a ***multilayer switch*** as it operates at Layer 2 and Layer 3. However, in this course we use the term Layer 3 switch.

Options For Inter-VLAN Routing (Cont.)

Layer 3 Switch



```
SW1# show ip route
<--- output omitted --->
C        10.1.0.0/24 is directly connected, Vlan20
L        10.1.0.1/32 is directly connected, Vlan20
C        192.168.1.0/24 is directly connected, Vlan10
L        192.168.1.1/32 is directly connected, Vlan10
```

Options For Inter-VLAN Routing (Cont.)

Layer 3 Switch

- Inter-VLAN SVIs are created the same way that the management VLAN interface is configured. The SVI is created for a VLAN that exists on the switch.
- The following are **advantages** of using Layer 3 switches for inter-VLAN routing:
 - They are much faster than router-on-a-stick because everything is hardware switched and routed.
 - There is no need for external links from the switch to the router for routing.
 - They are not limited to one link because Layer 2 EtherChannels can be used as trunk links between the switches to increase bandwidth.
 - Latency is much lower because data does not need to leave the switch in order to be routed to a different network.
 - They are more commonly deployed in a campus LAN than routers.
 - The only disadvantage is that Layer 3 switches are more expensive.



Basic Network For Trainee

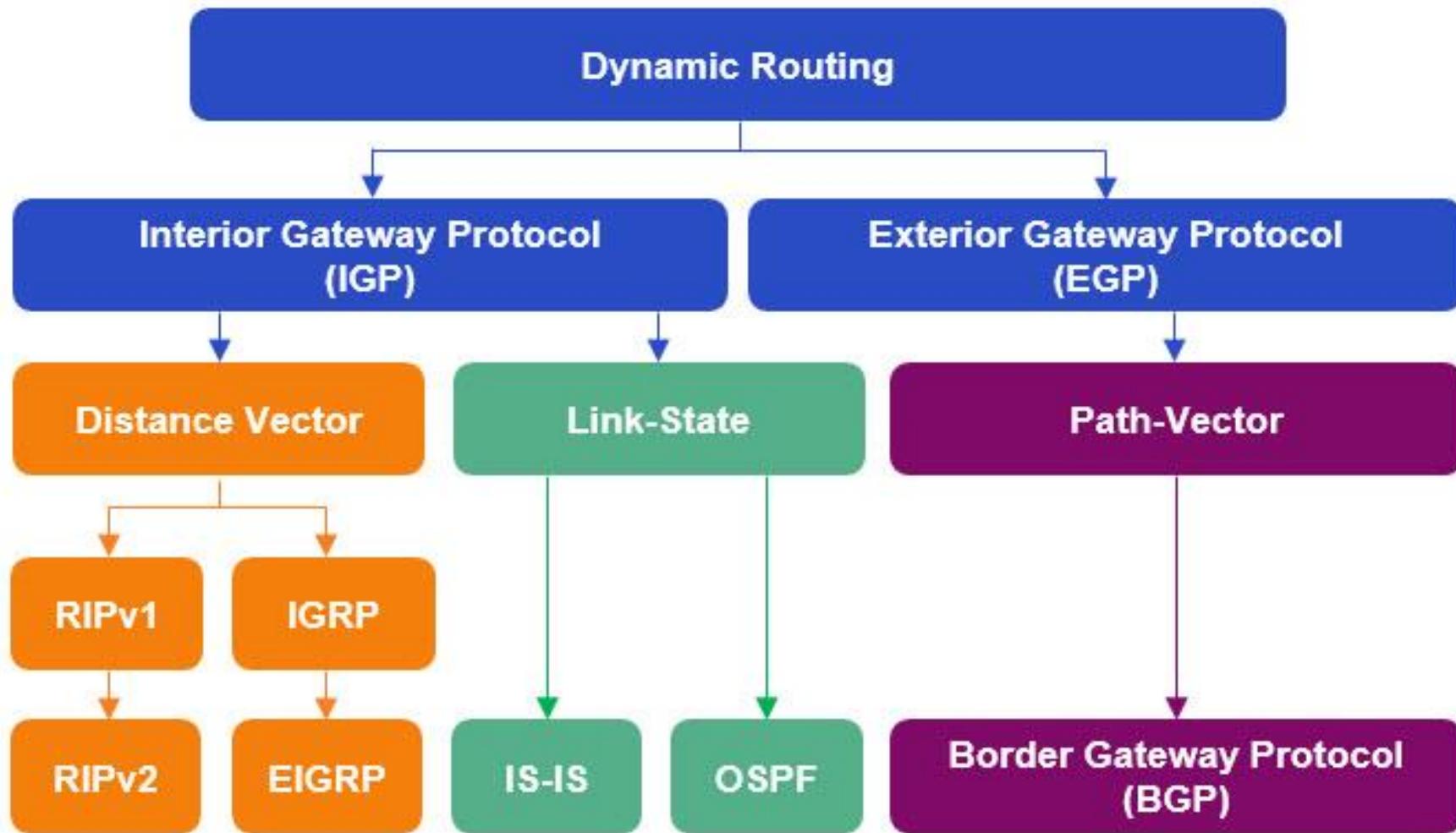
Module 10

Dynamic Routing

Panthakit Totid

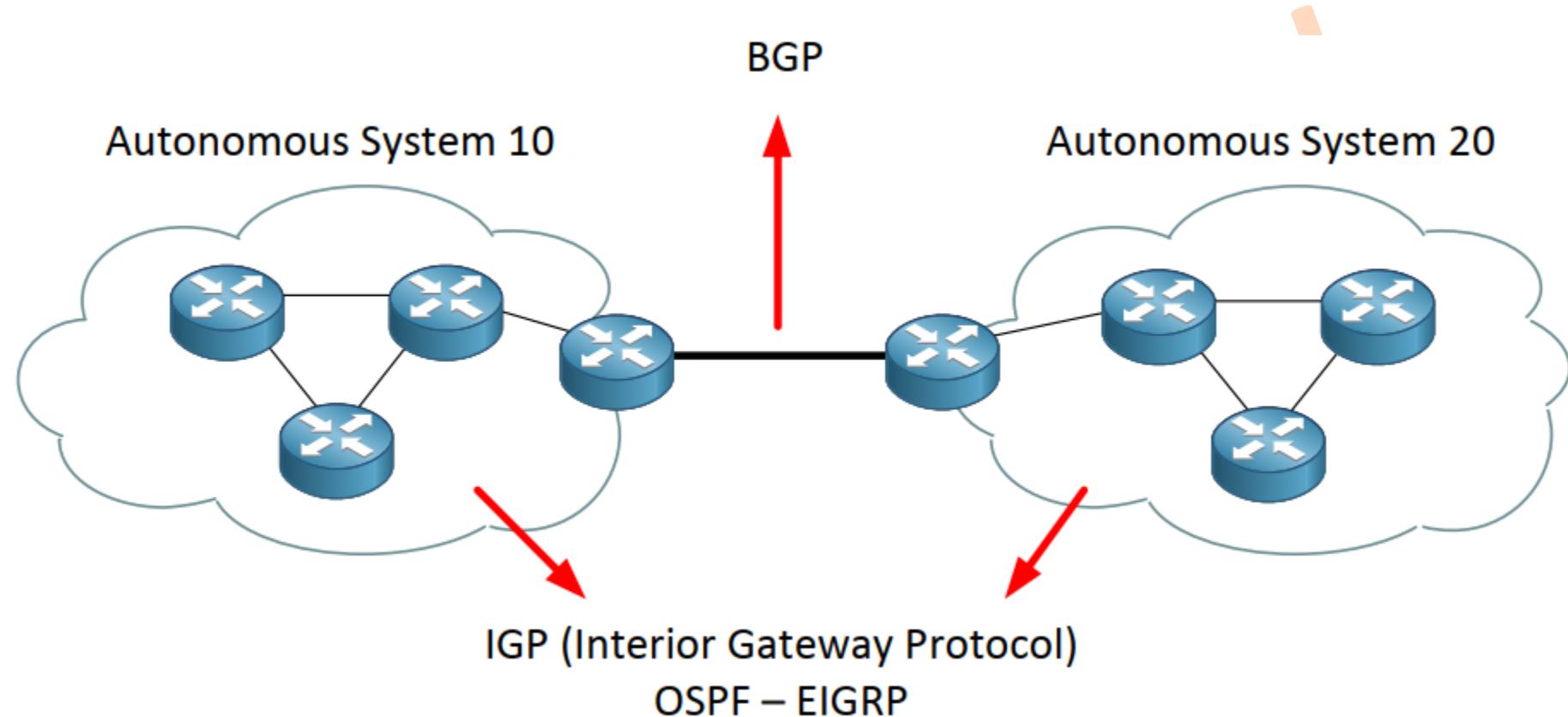


Dynamic Routing Protocol Concepts



Dynamic Routing Protocol Concepts (Cont.)

- **Autonomous System** is a collection of routers/networks that belongs to a single administrative domain.



Dynamic Routing Protocol Concepts (Cont.)

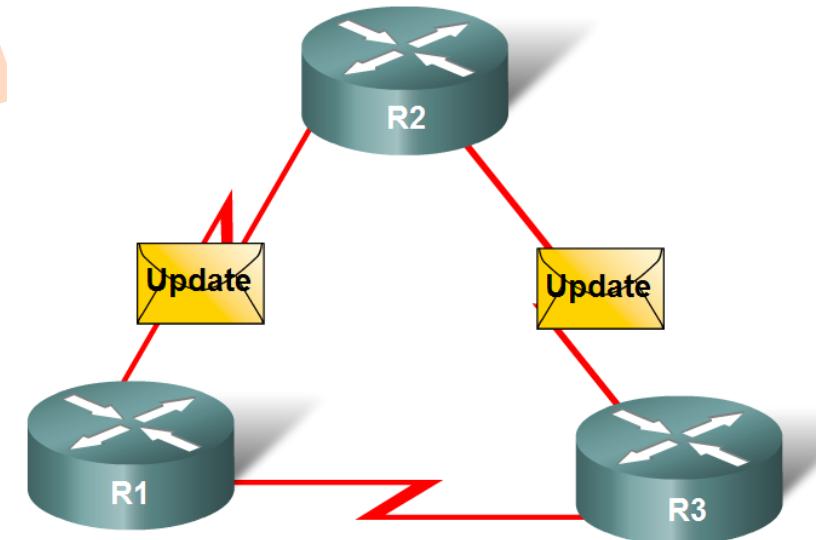
- You can divide routing protocols that are based on which type of information about network reachability is exchanged between the routers.



Distance Vector Protocols	Link-State Protocols	Path Vector Protocols
<p>Exchanges routes as vectors of distance and direction</p> <ul style="list-style-type: none">Distance = metric of link or hop countDirection = next-hop neighbor	<ul style="list-style-type: none">Exchanges information about the whole topologyEach router determines the best paths on its own, using SPF	<p>Exchanges routes as the vector of path and direction</p> <ul style="list-style-type: none">Path = list of different path attributesDirection = next-hop neighbor
EIGRP and RIPv2	OSPF and IS-IS	BGP

Dynamic Routing Protocol Concepts (Cont.)

- A routing protocol is a set of processes, algorithms, and messages that are used to exchange routing information and populate the routing table with the choice of best paths.
- The purpose of dynamic routing protocols includes the following:
 - **Discovery** of remote networks
 - Maintaining **up-to-date** routing information
 - **Choosing the best path** to destination networks
 - Ability to **find a new best path** if the current path is no longer available



Routers dynamically pass updates

Dynamic Routing Advantages and Disadvantages

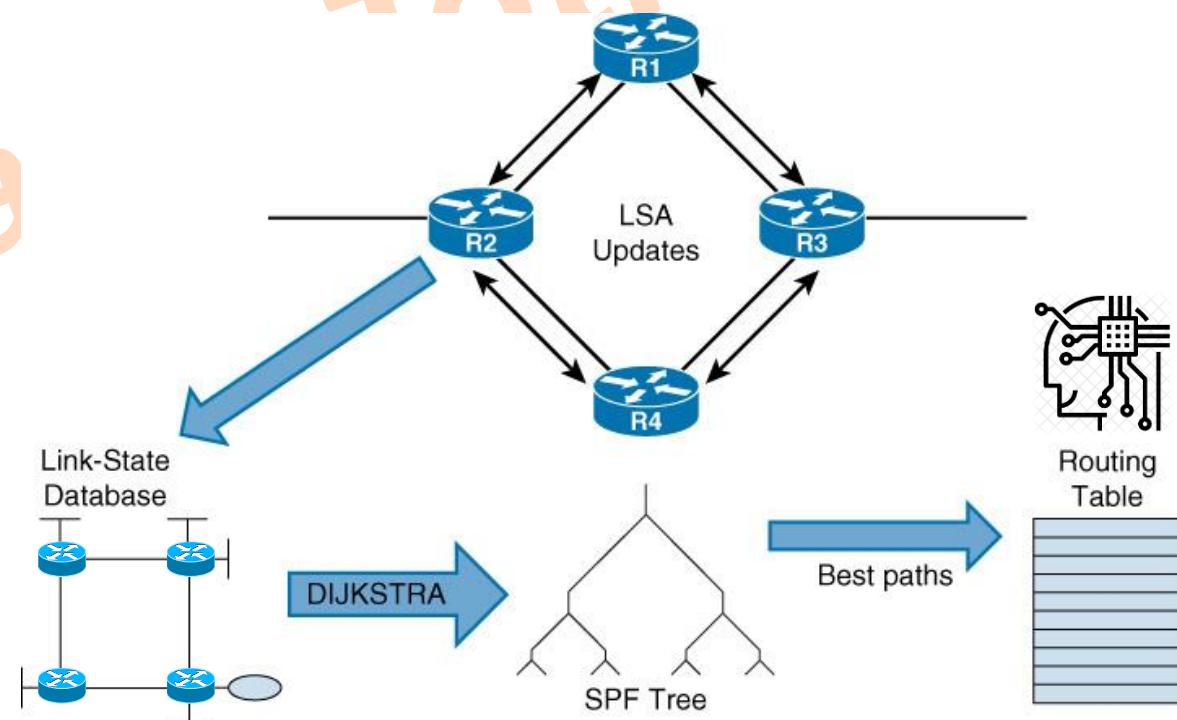
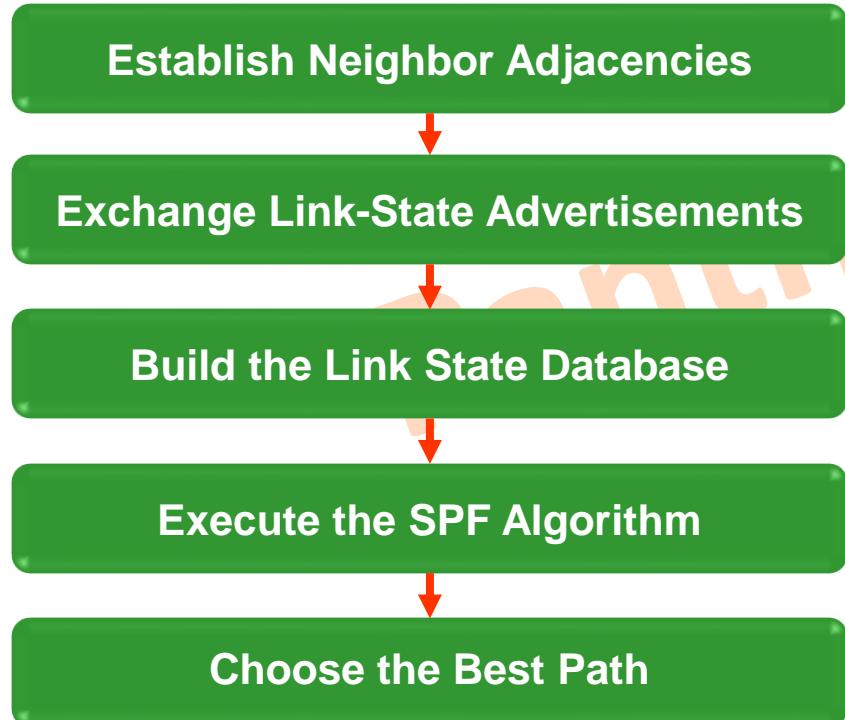
Advantages	Disadvantages
Suitable in all topologies where multiple routers are required	Can be more complex to implement
Generally independent of the network size	Less secure, additional configuration settings are required to secure
Automatically adapts topology to reroute traffic if possible	Route depends on the current topology
	Requires additional CPU, RAM, and link bandwidth

Introduction to OSPF

- OSPF (Open Shortest Path First) เป็น Routing Protocol แบบ Link State
- ใช้พื้นฐานการคำนวณจากอัลกอริทึม Shortest Path First (SPF) หรือ Dijkstra's Algorithm
- ถูกออกแบบมาเพื่อเอาชนะข้อจำกัดที่มีใน Routing Protocol แบบ Distance Vector (เช่น RIP) โดยมีความสามารถในการตอบสนองได้อย่างรวดเร็ว (Fast Convergence) ต่อการเปลี่ยนแปลงที่เกิดขึ้น
- มีการส่งข้อมูลตรวจสอบสถานะระหว่าง Router ข้างเคียงด้วยกันตลอดเวลา
- มีการปรับปรุงตารางเส้นทาง (Update Routing Table) ก็ต่อเมื่อในกรณีที่พบว่า Network มีการเปลี่ยนแปลงเท่านั้น (Event Triggered Updates)
- นำเอาค่า Bandwidth (Interface Speed) มาใช้ในการคำนวณแต่ละเส้นทาง (ยึดหยุ่นกว่าการใช้ Hop Count ของ RIP) เพื่อหาเส้นทางที่ดีที่สุดไปยังเครือข่ายต่าง ๆ
- รองรับการเชื่อมต่อเป็นเครือข่ายขนาดใหญ่ ด้วยการแยกเป็นลำดับชั้น (Hierarchy)

OSPF Operation Overview

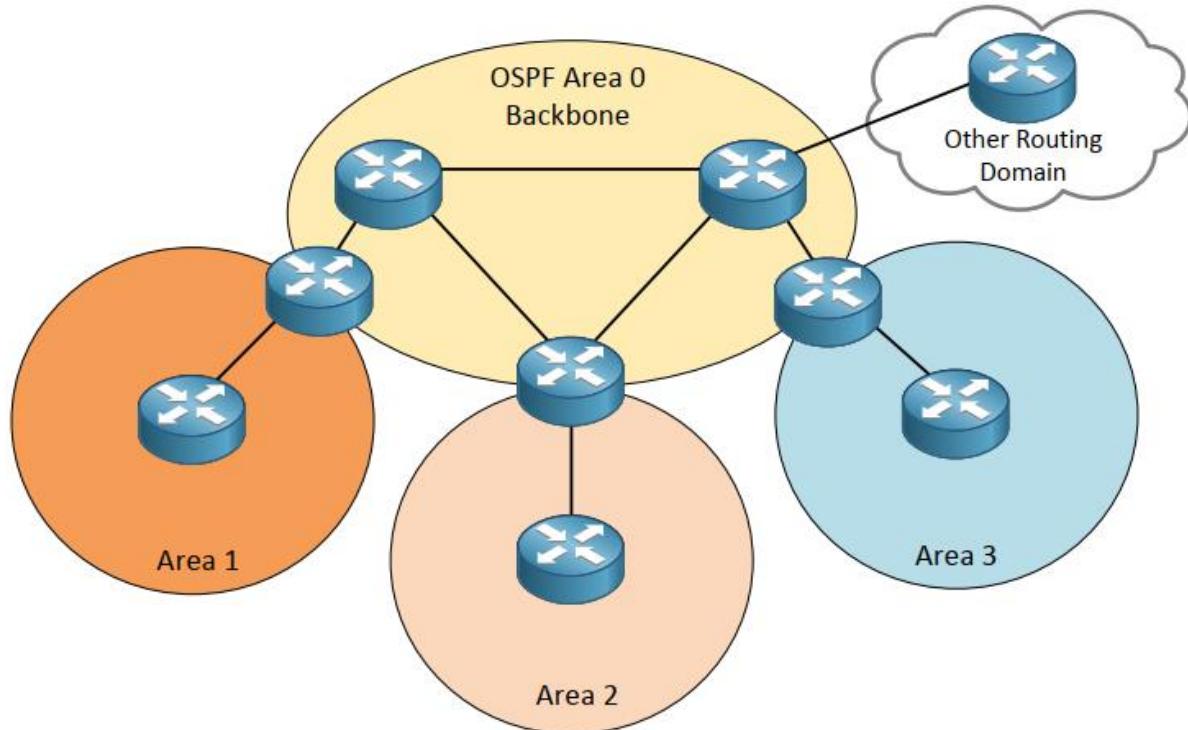
- Router จะส่งข้อมูลสถานะของ Interface ไปให้ Router เพื่อนบ้าน (Neighbor Router)
- Router จะมี Topology Map ของเครือข่ายทั้งหมดเพื่อคำนวณเส้นทางที่ดีที่สุดด้วยตัวเอง
- เส้นทางที่คำนวณแล้วว่าดีที่สุดเพื่อไปยังเครือข่ายปลายทาง จะถูกนำไปติดตั้งใน Routing Table



OSPF Database Tables

Database	Table	Description
Adjacency	Neighbor	<ul style="list-style-type: none">Lists all neighbor routers to which a router has established bidirectional communicationUnique for each routerView using the show ip ospf neighbor command
Link-state (LSDB)	Topology	<ul style="list-style-type: none">Lists information about all other routersRepresents the network topologyContains the same LSDB as all other routers in the same areaView using the show ip ospf database command
Forwarding	Routing	<ul style="list-style-type: none">Lists routes generated when the SPF algorithm is run on the link-state database.Unique to each router and contains information on how and where to send packets destined for remote networksView using the show ip route command

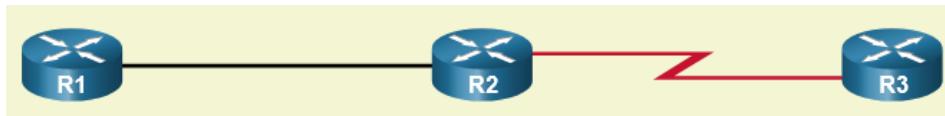
Hierarchical Structure of OSPF



- OSPF ทำงานบนเครือข่ายในลักษณะที่เป็นรูปแบบ Hierarchy ซึ่งจะแบ่งเป็นหลาย ๆ Area โดยที่แต่ละ Area จะเชื่อมโยงด้วย **Area 0 (Backbone Area)** เสมอ
- การแบ่งเป็นหลาย ๆ Area นั้นมีประโยชน์ดังนี้
 - ลด Routing Overhead (ลดการ Flooding ของ Link State Packet)
 - จำกัดขอบเขตความไม่เสถียรของเครือข่ายให้อยู่ภายใน Area เดียว
 - เพิ่มความเร็วในการ Convergence

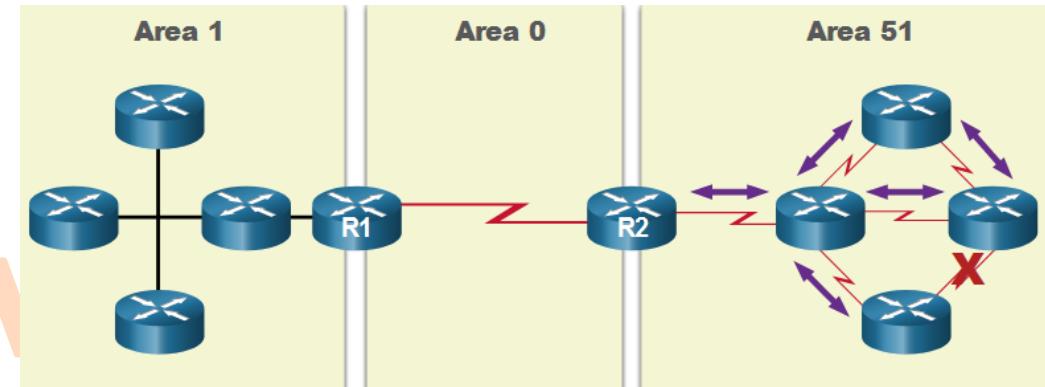
Hierarchical Structure of OSPF (Cont.)

Single-Area OSPF



- All routers contained in one area
- Called the **backbone area**
- Known as Area 0
- **Used in smaller networks with few routers**

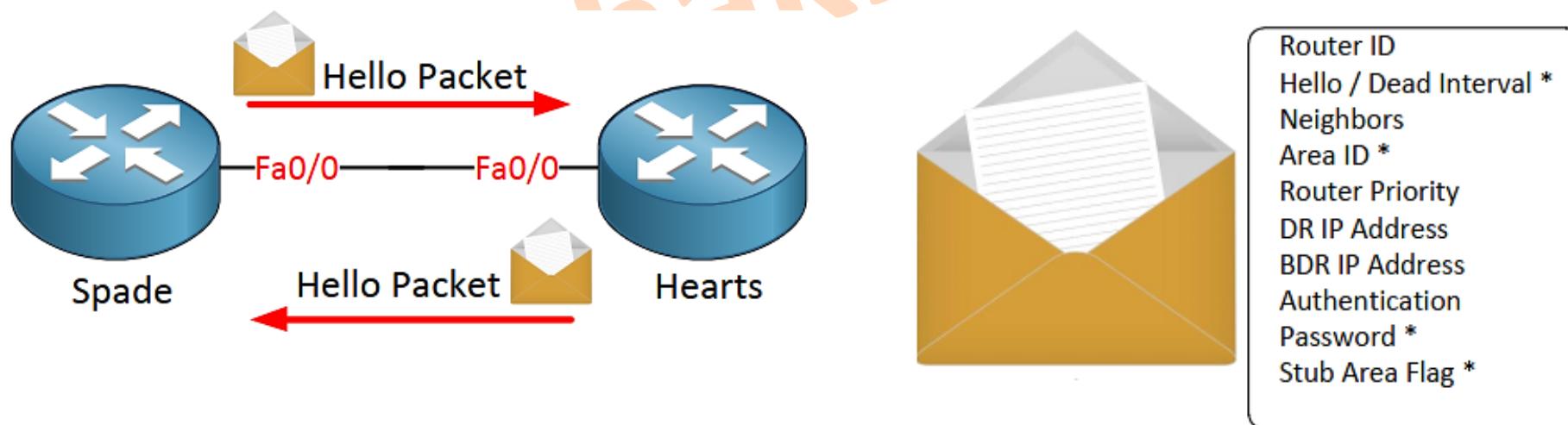
Multiarea OSPF



- Designed using a hierarchical scheme
- **All areas connect to area 0**
- More commonly seen with numerous areas around area 0 (like a daisy or aster)
- Routers that connect area 0 to another area is known as an **Area Border Router (ABR)**
- Used in large networks
- Multiple areas reduces processing and memory overhead
- A failure in one area does not affect other areas

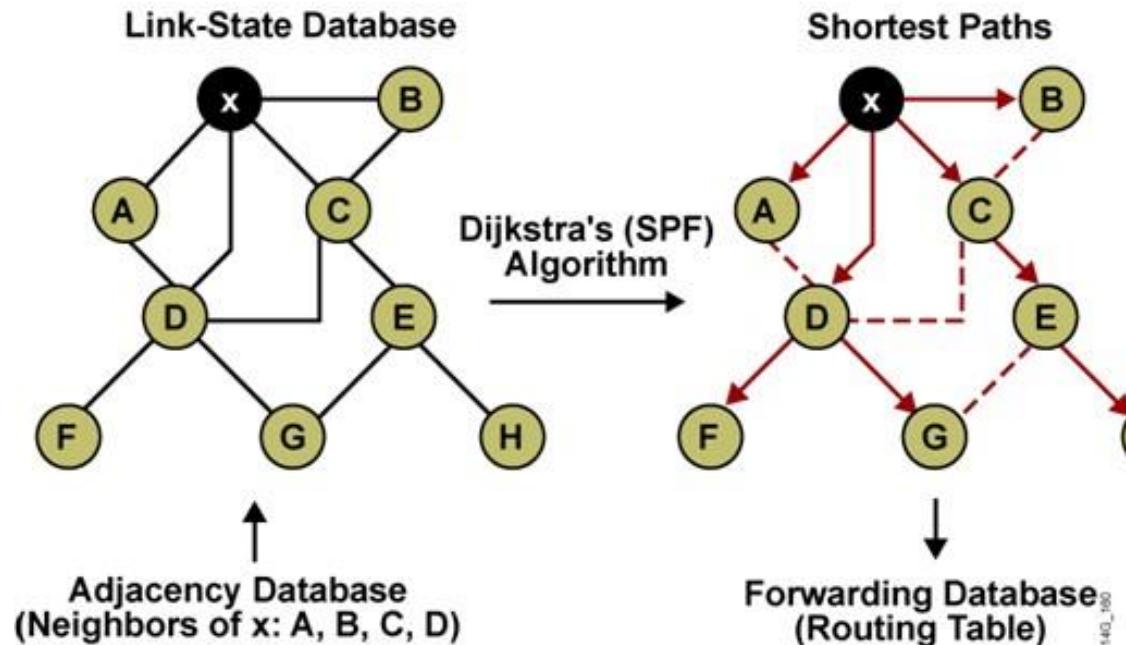
OSPF Neighbor Relationship (OSPF Adjacency)

- Router แต่ละตัวจะแลกเปลี่ยน Hello Packet เพื่อติดต่อกับ Router เพื่อนบ้าน
 - หนึ่งในข้อกำหนดที่สำคัญของการฟอร์มความสัมพันธ์ระหว่าง Router เพื่อนบ้าน คือ จะต้องมีช่วงเวลา Hello Interval และ Dead Interval เท่ากัน
 - Default Hello Interval = 10 วินาที และ Dead Interval = 40 วินาที
 - เมื่อเกิดสถานะ Adjacency ระหว่างกันได้แล้ว จึงเริ่ม Synchronize LSDB ให้ตรงกัน



OSPF Best Path Selection

- OSPF เลือกเส้นทางที่ดีที่สุด โดยใช้หลักการ SPF ซึ่งจะเริ่มต้นโดย Router ที่เป็นเจ้าของ LSDB จะจัดให้ตนเองอยู่ในตำแหน่งของ Root ใน Tree Topology และเลือกเส้นทางที่มีผลรวมของ Metric ที่น้อยที่สุดไปยัง Router ตัวอื่น ๆ



Assume all links are Ethernet, with an OSPF cost of 10.

Routing Table: Router X		
Target	Metric	Neighbor Router
A	10	-
B	10	-
C	10	-
D	10	-
E	20	C
F	20	D
G	20	G
H	30	C

OSPF Cost

การกำหนดค่า Cost ของ Routing Protocol OSPF สามารถทำได้ 2 วิธีคือ

- ตั้งค่า **Cost** โดยตรงบน Interface ที่เปิดการใช้งาน OSPF เอาไว้

```
Router(config-if)# ip ospf cost 10
```

- ตั้งค่า **Bandwidth** โดยตรงบน Interface ที่เปิดการใช้งาน OSPF เอาไว้

```
Router(config-if)# bandwidth 256
```

โดยค่า **Cost** จะเป็นส่วนผกผันกับค่า **Bandwidth** นั่นคือ ถ้า Interface นั้นมี Bandwidth สูง ค่า Cost จะมีค่าน้อย และถ้า Interface นั้นมี Bandwidth ต่ำ ค่า Cost จะมีค่ามากแทน ซึ่งสูตรในการคำนวณคือ

$$\text{Cost} = \frac{\text{Reference Bandwidth (Default} = 10^8 \text{ bps)}}{\text{Interface Bandwidth (bps)}}$$

OSPF Cost (Cont.)

- ตัวอย่างการหาค่า Cost ของ Interface แบบ Ethernet

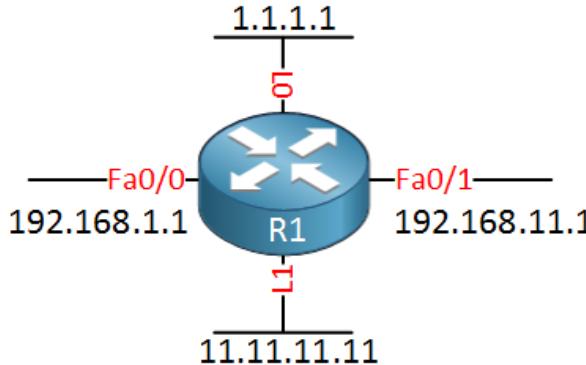
Interface แบบ Ethernet มี Bandwidth = 10Mbps หรือ 10,000,000 bps

ดังนั้น จะคำนวณได้ค่า Cost คือ $\frac{100,000,000 \text{ bps}}{10,000,000 \text{ bps}} = 10$

Interface Type	Cost
10 Gbps Ethernet	1
1 Gbps Ethernet	1
100 Mbps Ethernet	1
10 Mbps Ethernet	10
1.544 Mbps Serial	64
128 kbps Serial	781
64 kbps Serial	1562



OSPF Router ID



R1# show ip interface brief

Interface	IP-Address	OK?	Method	Status	Protocol
FastEthernet0/0	192.168.1.1	YES	manual	up	up
FastEthernet0/1	192.168.11.1	YES	manual	up	up
Loopback0	1.1.1.1	YES	manual	up	up
Loopback1	11.11.11.11	YES	manual	up	up

- แต่ละ Router จะต้องมีหมายเลขอ้างอิง (Router ID) เป็นของตัวเอง เพื่อให้สามารถแลกเปลี่ยนข้อมูลกับ Router อื่น ๆ ได้อย่างถูกต้อง
- Router ID ของแต่ละ Router ถูกกำหนดจาก Priority (จากมากน้อย) ดังนี้
 - ตั้งค่าโดยตรง (Manual Configure)
 - IP ของ Loopback Interface ที่มีค่ามากที่สุด (up)
 - IP ของ Physical Interface ที่มีค่ามากที่สุด (up)

OSPF Single Area

- ตั้งค่าเปิดใช้ OSPF เป็น Routing Protocol ของ Router

```
Router(config)#router ospf {process-id}
```

- ตั้งค่า Interface ที่อนุญาตให้ทำงาน

```
Router(config-router)#network {network-id} {wildcard mask} area 0
```

หรือสามารถตั้งค่า OSPF โดยการเปิดการทำงานภายใต้ Interface ได้เช่นกัน

- ตั้งค่าเปิดใช้ OSPF เป็น Routing Protocol ของ Router

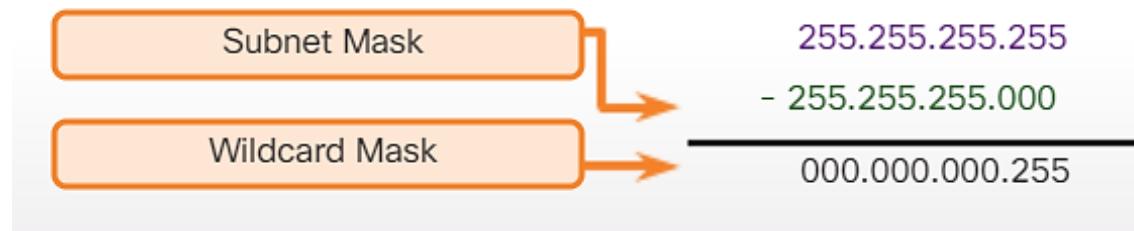
```
Router(config)#router ospf {process-id}
```

- เข้าไปที่ Interface ที่จะอนุญาตให้ทำงาน

```
Router(config-if)#ip ospf {process-id} area 0
```

OSPF Wildcard Mask

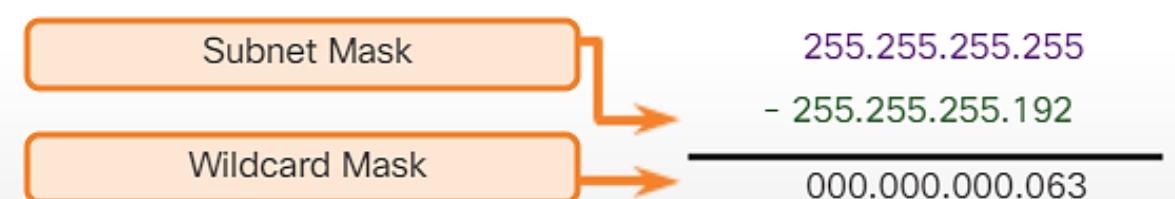
- Wildcard Mask คือ ข้อมูลตัวกรองขนาด 32 บิต สำหรับนำมาใช้ระบุ “กลุ่มของ IP Address” ที่สนใจ โดยที่
 - บิตที่เป็น 0 หมายถึง ตำแหน่ง IP ที่สนใจ (Match)
 - บิตที่เป็น 1 หมายถึง ตำแหน่ง IP ที่ไม่สนใจ (Ignore)



/24 mask

เช่น 192.168.90.0 0.0.0.255 → สนใจช่วง IP ที่อยู่ระหว่าง 192.168.90.1 – 192.168.90.254 นั่นเอง

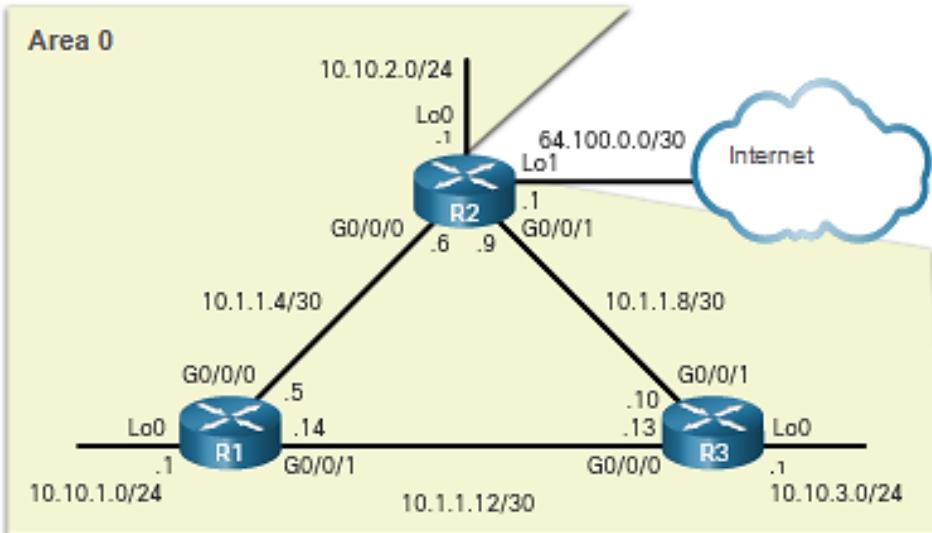
tantid



/26 mask

เช่น 10.10.10.64 0.0.0.63 → สนใจช่วง IP ที่อยู่ระหว่าง 10.10.10.65 – 10.10.10.126 นั่นเอง

Basic OSPF Configuration (Single Area) - Option 1

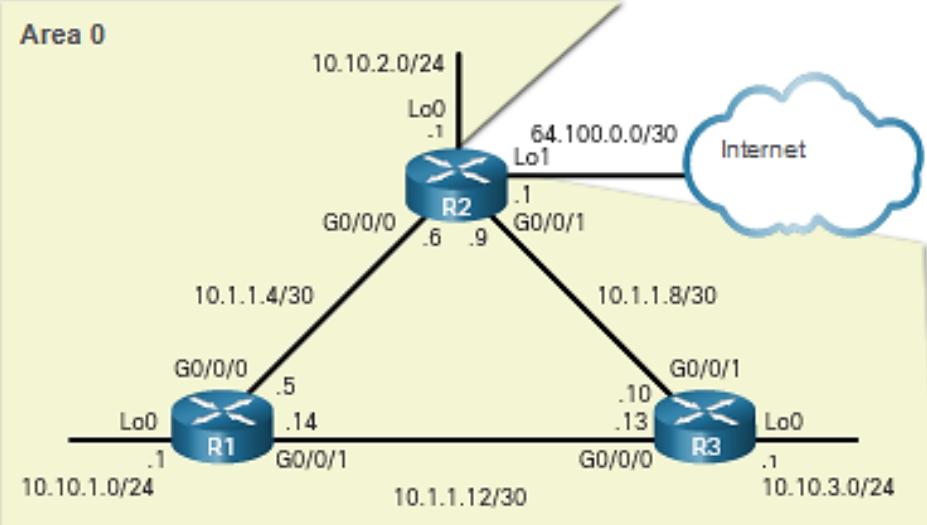


```
R1(config)# router ospf 10
R1(config-router)# network 10.10.1.0 0.0.0.255 area 0
R1(config-router)# network 10.1.1.4 0.0.0.3 area 0
R1(config-router)# network 10.1.1.12 0.0.0.3 area 0
```

```
R2(config)# router ospf 20
R2(config-router)# network 10.10.2.0 0.0.0.255 area 0
R2(config-router)# network 10.1.1.4 0.0.0.3 area 0
R2(config-router)# network 10.1.1.8 0.0.0.3 area 0
```

```
R3(config)# router ospf 30
R3(config-router)# network 10.10.3.0 0.0.0.255 area 0
R3(config-router)# network 10.1.1.8 0.0.0.3 area 0
R3(config-router)# network 10.1.1.12 0.0.0.3 area 0
```

Verifying Routing Table



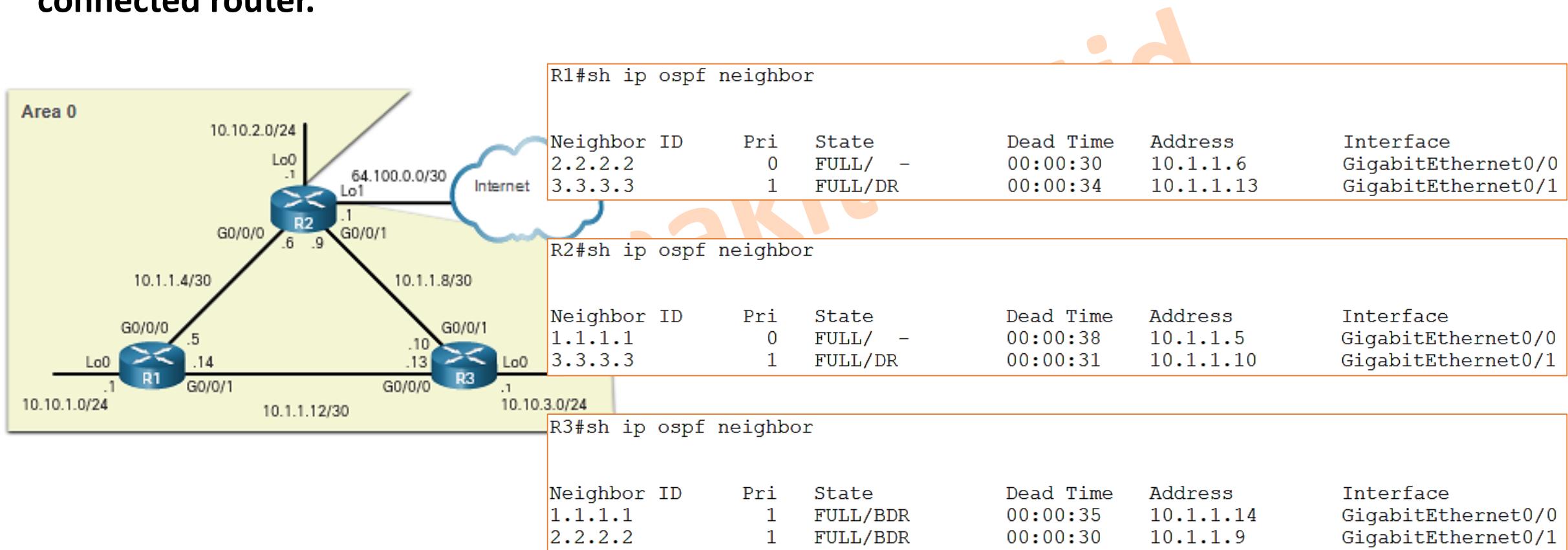
```
R1# show ip route ospf
0      10.1.1.8/30 [110/2] via 10.1.1.6, 00:05:33, G0/0
                  [110/2] via 10.1.1.13, 00:05:33, G0/1
0      10.10.2.1/32 [110/2] via 10.1.1.6, 00:10:14, G0/0
0      10.10.3.1/32 [110/2] via 10.1.1.13, 00:06:35, G0/1
```

```
R2# show ip route ospf
0      10.1.1.12/30 [110/2] via 10.1.1.5, 00:05:59, G0/0
                  [110/2] via 10.1.1.10, 00:05:59, G0/1
0      10.10.1.1/32 [110/2] via 10.1.1.5, 00:10:43, G0/0
0      10.10.3.1/32 [110/2] via 10.1.1.10, 00:05:59, G0/1
```

```
R3# show ip route ospf
0      10.1.1.4/30 [110/2] via 10.1.1.14, 00:06:25, G0/0
                  [110/2] via 10.1.1.9, 00:06:25, G0/1
0      10.10.1.1/32 [110/2] via 10.1.1.14, 00:07:27, G0/0
0      10.10.2.1/32 [110/2] via 10.1.1.9, 00:06:25, G0/1
```

Verify OSPF Neighbors

- Use the **show ip ospf neighbor** to verify the router has formed an adjacency with a directly-connected router.



Default Route Propagation

- To propagate a default route, the edge router must be configured with the following:
 - A default static route using the **ip route 0.0.0.0 0.0.0.0 [next-hop-address | exit-intf]** command.
 - The **default-information originate** router configuration command. This instructs R2 to be the source of the default route information and propagate the default static route in OSPF updates.
- In the example, R2 is configured with a loopback to simulate a connection to the internet. A default route is configured and propagated to all other OSPF routers in the routing domain.
 - **Note:** When configuring static routes, **best practice is to use the next-hop IP address.**
 - However, when simulating a connection to the internet, there is no next-hop IP address. Therefore, we use the **exit-intf** argument.

Default Route Propagation (Cont.)

```
R2(config)# interface lo1
R2(config-if)# ip address 64.100.0.1 255.255.255.252
R2(config-if)# exit
R2(config)# ip route 0.0.0.0 0.0.0.0 loopback 1
%Default route without gateway, if not a point-to-point interface,
may impact performance
R2(config)# router ospf 10
R2(config-router)# default-information originate
R2(config-router)# end
```

Verify the Propagated Default Route

- You can verify the default route settings on R2 using the **show ip route** command. You can also verify that R1 and R3 received a default route.
- Notice that the route source on R1 is **O*E2**, signifying that it was learned using OSPFv2. The asterisk identifies this as a good candidate for the default route. The E2 designation identifies that it is an external route. The meaning of E1 and E2 is beyond the scope of this module.

```
R2# sh ip route | include /0
S*  0.0.0.0/0 is directly connected, Loopback1
```

```
R1# sh ip route | in 0
0      10.1.1.8/30 [110/20] via 10.1.1.6, 00:01:47, GigabitEthernet0/0
0      10.10.2.0/24 [110/11] via 10.1.1.6, 00:01:47, GigabitEthernet0/0
0      10.10.3.0/24 [110/21] via 10.1.1.6, 00:01:47, GigabitEthernet0/0
O*E2 0.0.0.0/0 [110/1] via 10.1.1.6, 00:01:47, GigabitEthernet0/0
```



Basic Network For Trainee

Module 11

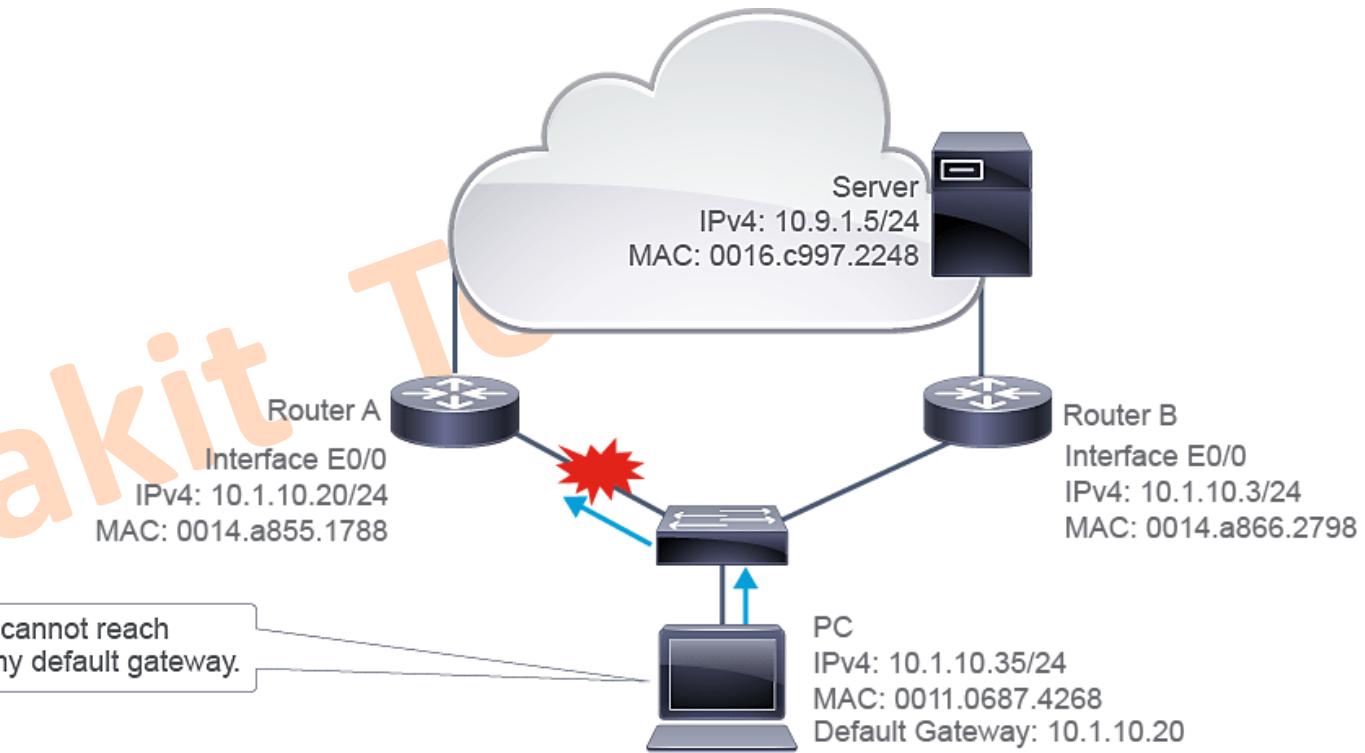
Layer 3 Redundancy

Panthakit Totid



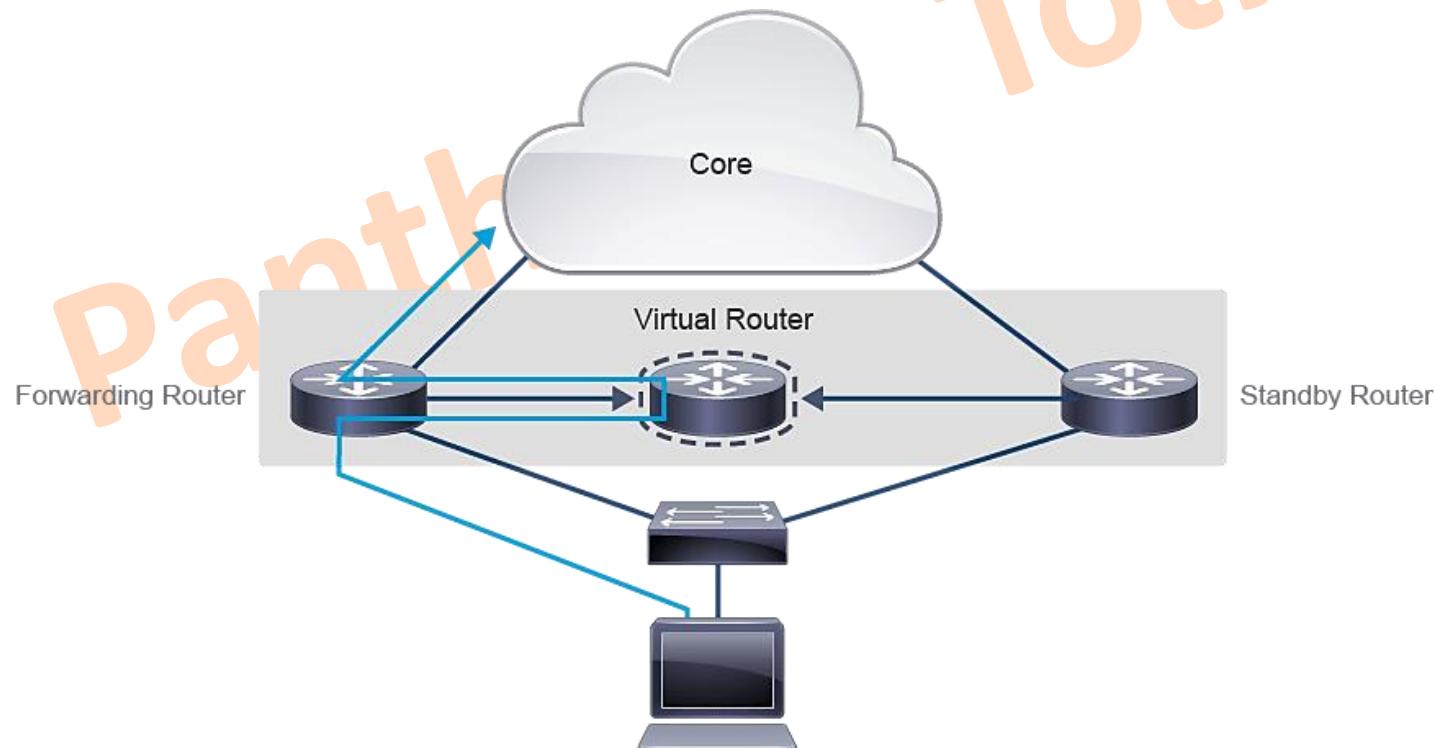
Need for Default Gateway Redundancy

- Redundant equipment alone does not guarantee failover.
 - Most IPv4 hosts do not run a dynamic routing protocol to build a list of reachable networks.
 - Instead, they rely on a *manually configured* or *dynamically learned default gateway* to route all packets.
 - The end device is configured with a single default gateway IPv4 address, which does *not dynamically update* when the network topology changes.



Understanding FHRP

- **First Hop Redundancy Protocols (FHRPs)** are a group of protocols with similar functionality that enable a set of routers or Layer 3 switches to present an illusion of a "**virtual**" router.
 - The virtual router is assigned a **virtual IP address** and **virtual MAC address**, which is shared by two routers.
 - All the routers in the **same HSRP group** respond to the virtual pair of IP, MAC



Understanding FHRP (Cont.)

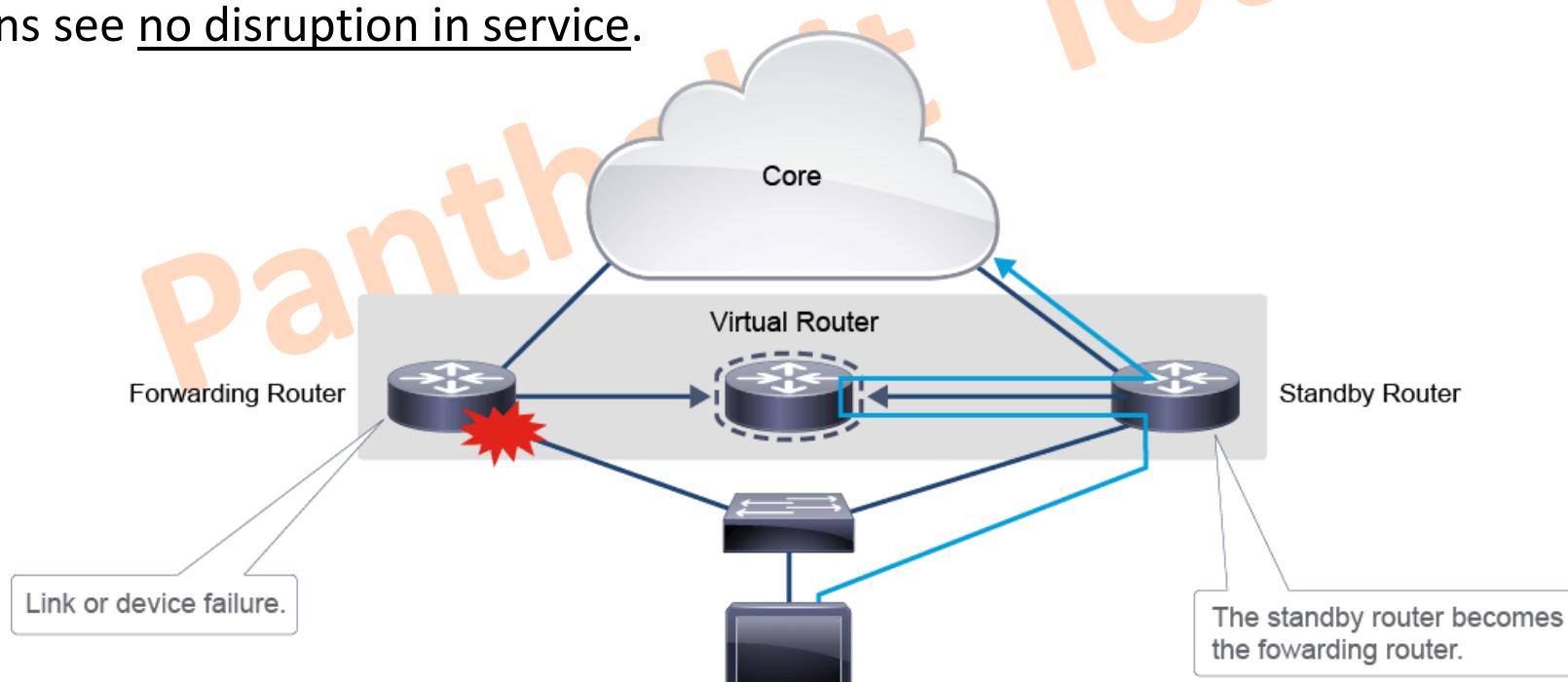
A common feature of FHRP is to provide a default gateway failover that is transparent to hosts. Cisco routers and switches typically support the use of three FHRPs:

- **Hot Standby Router Protocol (HSRP)**
 - HSRP is an FHRP that *Cisco designed* to create a redundancy framework between network routers or Layer 3 switches to achieve default gateway failover capabilities.
 - Only one router per subnet forwards traffic.
- **Virtual Router Redundancy Protocol (VRRP)**
 - VRRP is an open FHRP *standard* that offers the ability to add more than two routers for additional redundancy.
 - Only one router per subnet forwards traffic.
- **Gateway Load Balancing Protocol (GLBP)**
 - GLBP is an FHRP that *Cisco designed* to allow multiple active forwarders to load-balance outgoing traffic on a per host basis rather than a per subnet basis like HSRP.

Understanding FHRP (Cont.)

When the forwarding router or the link, where FHRP is configured, **fails**, these steps take place:

1. The standby router **stops seeing** hello messages from the forwarding router.
2. The standby router **assumes the role** of the forwarding router.
3. Because the new forwarding router assumes both the IP and MAC addresses of the virtual router, the end stations see no disruption in service.



Understanding HSRP

HSRP Overview

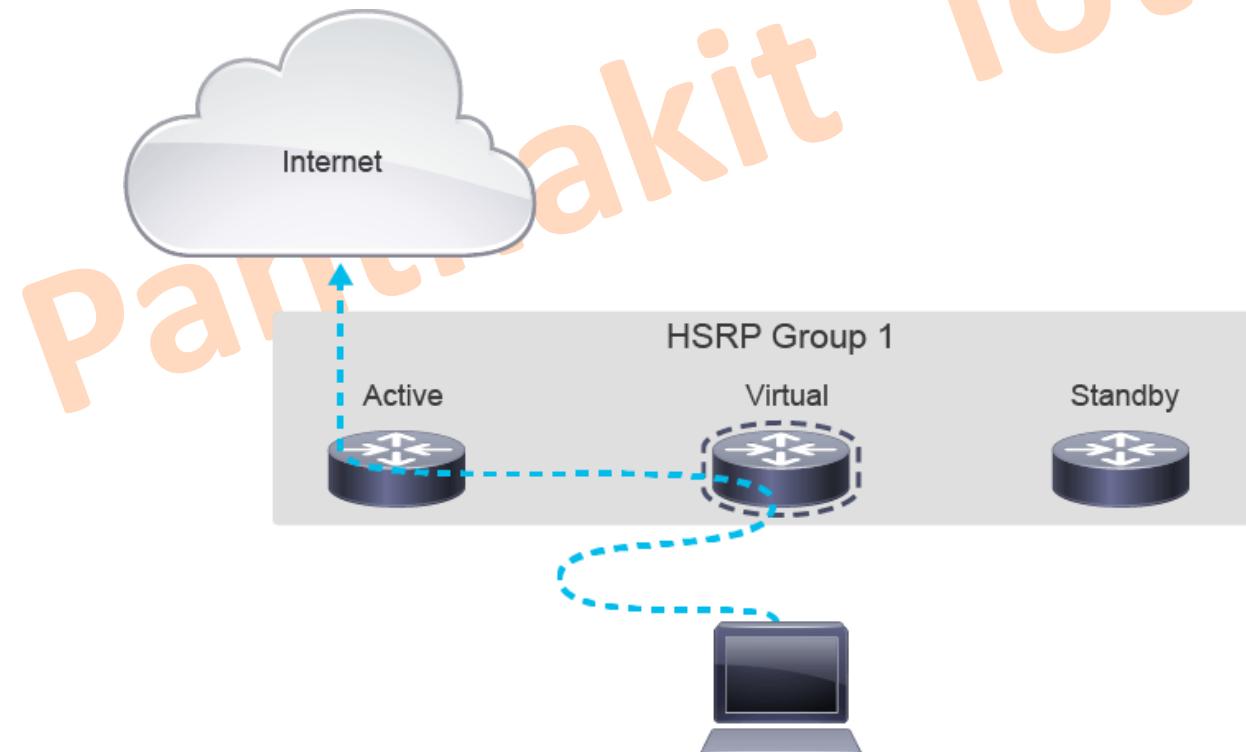
- HSRP defines a **standby group of routers**, while one router is designated as the **active router**.
- HSRP provides gateway redundancy by sharing IP and MAC addresses between redundant gateways.
- The protocol consists of virtual IP and MAC addresses that the two routers that belong to the same HSRP group share between each other.



Understanding HSRP (Cont.)

HSRP Overview

- When IPv4 hosts use **ARP** to resolve the MAC address of the default gateway IPv4 address, the **active HSRP router responds** with the shared virtual MAC address.
- The packets that are received on the virtual IPv4 address are forwarded to the active router.



Understanding HSRP (Cont.)

The HSRP active and the standby router perform the following functions:

- **Active router:**

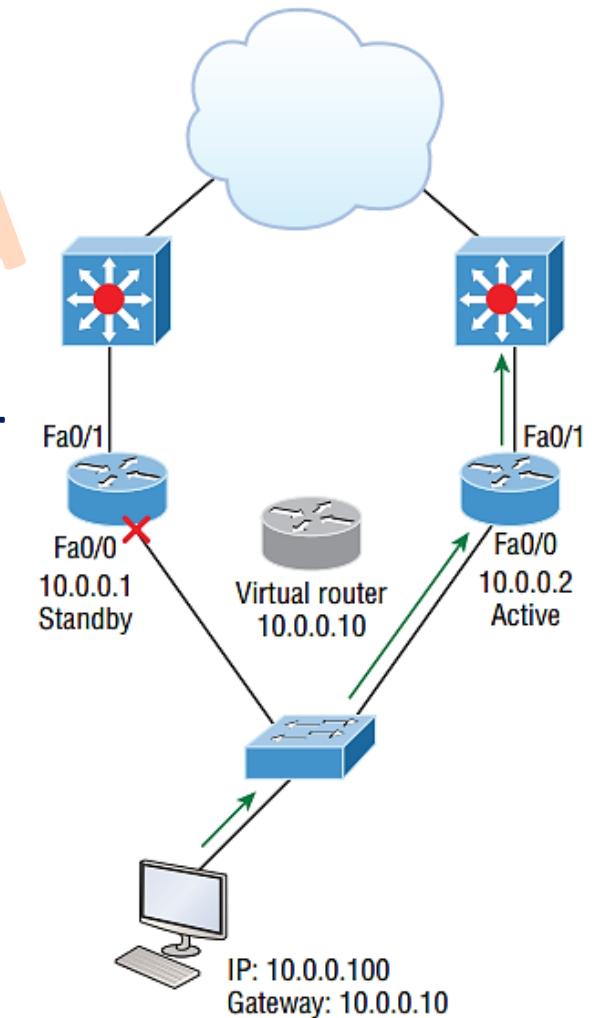
- Responds to default gateway ARP requests with the virtual router MAC address.
- Assumes active forwarding of packets for the virtual router.
- Sends hello messages between the active and standby routers.
- Knows the virtual router IPv4 address.

- **Standby router:**

- Sends hello messages.
- Listens for periodic hello messages.
- Assumes active forwarding of packets if it does not hear from active router.
- Sends Gratuitous ARP message when standby becomes active.

Understanding HSRP (Cont.)

- The function of the HSRP standby router is to monitor the operational status of the HSRP group and to quickly assume the packet-forwarding responsibility if the active router becomes inoperable. When the primary HSRP router comes back online, it will *not regain the active role by default (non-preemptive)*.
- To transfer the active role to the primary router, you have to **configure pre-emption**.
- The **standby preempt** command enables the HSRP router with the *highest priority* to immediately become the active router.
- Priority is determined first by the configured priority value and then by the IPv4 address.



Understanding HSRP (Cont.)

HSRP for IPv4 has two versions: Version 1 and Version 2.

- The **default HSRP version is 1**. Because the two versions are not compatible, you must use the same version on your HSRP enabled routers.
- Routers with **HSRP Version 1** send hello packets to the multicast address of **224.0.0.2** (reserved multicast address used to communicate to all routers) on **UDP port 1985**
- **HSRP Version 2** uses the **224.0.0.102** multicast address on **UDP port 1985**.

Virtual MAC address

- **HSRP Version 1** uses a MAC address in the form **0000.0C07.ACXX**
- **HSRP Version 2** use a MAC address in the form **0000.0C9F.FXXX**
- Where **XX** or **XXX** stand for the group number
 - For example, the virtual MAC address for an HSRP Version 2 virtual router in **group 10** would be **0000.0C9F.F00A**. The A in 00A is the hexadecimal value for 10.

Configuring HSRP

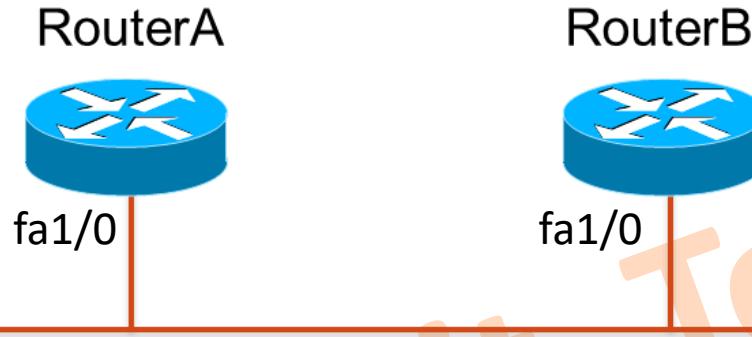
- Configure HSRP on the interface.

```
Router(config-if)# standby group-number ip ip-address
```

- The group number is optional and indicates the HSRP group to which this interface belongs.
- Specifying a unique group number in the standby commands enables the creation of multiple HSRP groups. **The default group is 0.**
- The IP address is that of the virtual router IP address for the HSRP group.

```
Router(config-if)# standby group-number priority priority-value
```

Configuring HSRP Example



```
RouterA(config)# interface fa1/0
RouterA(config-if)# ip address 10.1.1.1 255.255.255.0
RouterA(config-if)# standby 10 ip 10.1.1.254
RouterA(config-if)# standby 10 priority 110
RouterA(config-if)# standby 10 preempt
```

```
RouterB(config)# interface fa1/0
RouterB(config-if)# ip address 10.1.1.2 255.255.255.0
RouterB(config-if)# standby 10 ip 10.1.1.254
RouterB(config-if)# standby 10 preempt
```



Basic Network For Trainee

Module 12

ACL Concepts

Panthakit Totid



ACL Overview

The following ACL features are important to remember:

- Ordered series of statements
- ACL statements specify the following:
 - Action
 - Permit
 - Deny
 - Matching rule
- ACLs are applied for the following:
 - Traffic filtering
 - Selecting traffic
 - To be analyzed, forward, or processed in other ways, for example, classify traffic to enable priority processing.

Command Action
Permit = Include into Selection
Deny = Do not Include into Selection

Standard IP access list 15

```
10 permit host 10.0.0.10
20 deny 10.0.0.0 0.0.0.255
30 permit host 172.16.1.1
40 deny 172.16.0.0 0.0.255.255
50 permit any
```

Extended IP access list PING_BLOCK

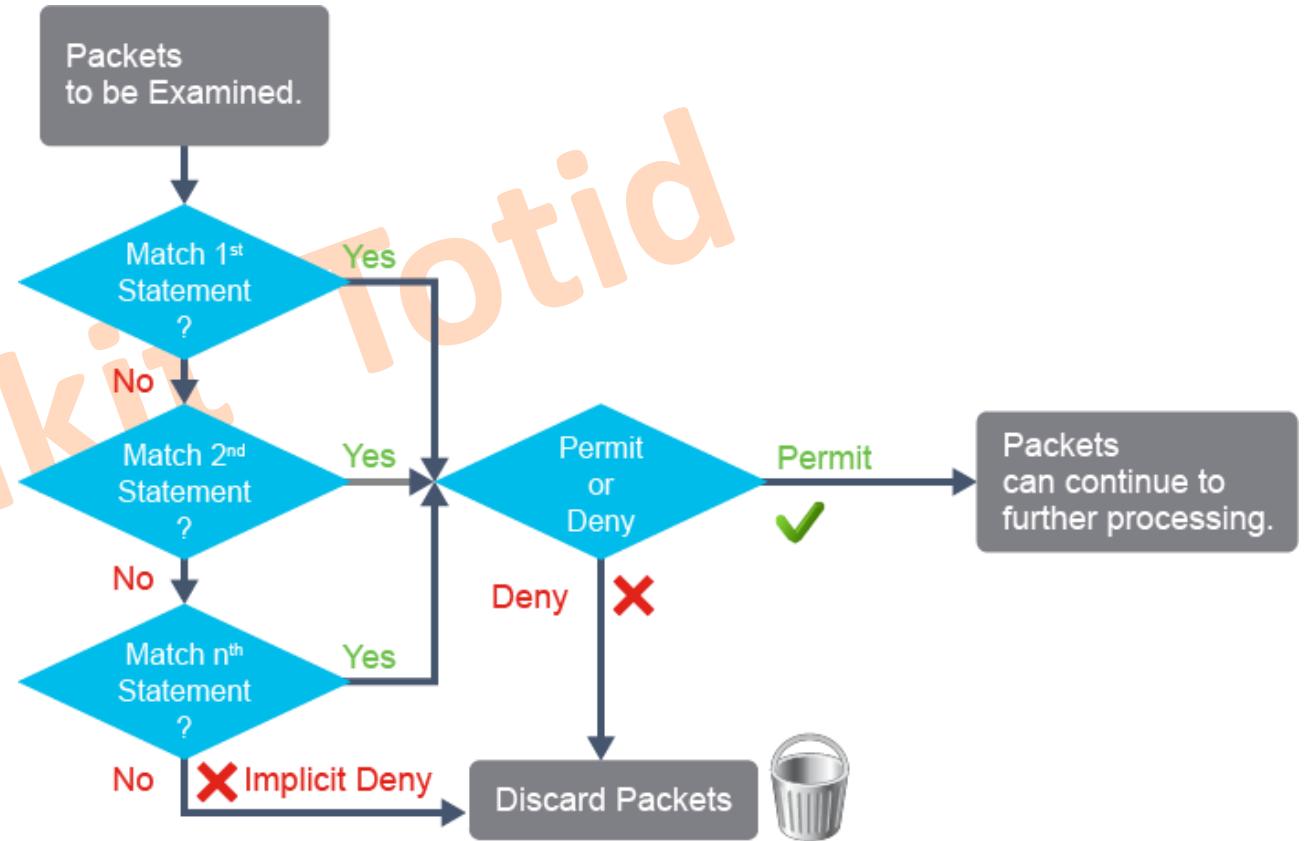
```
10 permit icmp host 10.0.0.10 any
20 deny icmp 10.0.0.0 0.0.0.255 any
30 permit ip any any
```

Matching Rules

ACL Operation

ACL tests include the following:

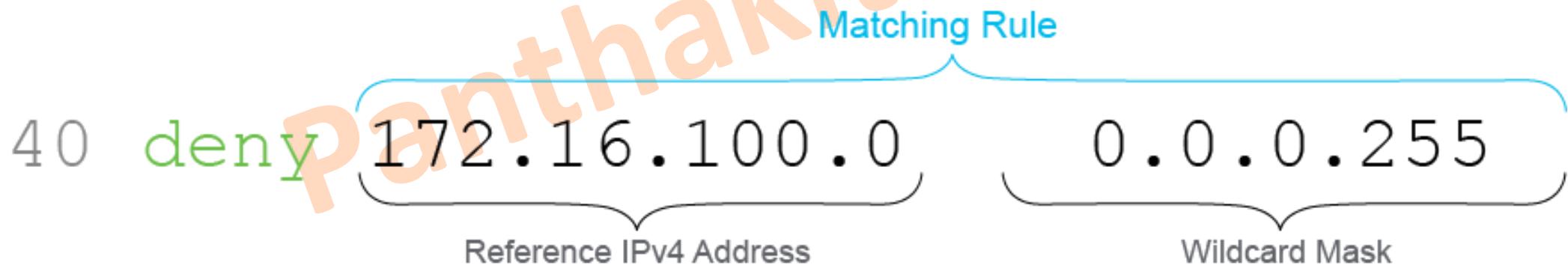
- An ACL consists of a series of permit and deny statements.
 - **Access control entries (ACEs) or ACL statements**
- An ACL is evaluated one by one in *top-down order*.
- The first match executes the permit or deny action and stops further ACL matching.
- There is an *implicit deny all statement* at the end of each ACL.



ACL Wildcard Masking

Matching criteria/matching rule has two elements:

- IPv4 address provides a **reference** against which IPv4 packet information is evaluated.
- **Wildcard mask** provides evaluation criteria:
 - **0** = this bit must match the value in the reference IPv4 address
 - **1** = this bit can have whatever value (ignored)



ACL Wildcard Masking

Reference IP address: 172.16.100.1

Matching Requirement Interpretation	Wildcard Mask 3rd Octet	Reference IPv4 3rd Octet (100 Decimal)	Resulting Match Pattern	Examples of Matching Octet Decimal Values
Match all bits of the reference IP.	0 0 0 0 0 0 0 0	0 1 1 0 0 1 0 0	0 1 1 0 0 1 0 0	100
Match the first seven bits of the reference IP. The last bit can be any value.	0 0 0 0 0 0 0 1	0 1 1 0 0 1 0 0	0 1 1 0 0 1 0 ×	100, 101
Match the first six bits of the reference IP. The last two bits can be any value.	0 0 0 0 0 0 1 1	0 1 1 0 0 1 0 0	0 1 1 0 0 1 ××	100, 101, 102, 103
Match the first 5 bits of the reference IP. The last three bits can be any value.	0 0 0 0 0 1 1 1	0 1 1 0 0 1 0 0	0 1 1 0 0 ×××	96 - 103
Match just the first four bits.	0 0 0 0 1 1 1 1	0 1 1 0 0 1 0 0	0 1 1 0 ××××	96 - 111
Match just the first three bits.	0 0 0 1 1 1 1 1	0 1 1 0 0 1 0 0	0 1 1 ××××	96 - 127
Match just the first two bits.	0 0 1 1 1 1 1 1	0 1 1 0 0 1 0 0	0 1 ××××	64 - 127
Match just the first bit.	0 1 1 1 1 1 1 1	0 1 1 0 0 1 0 0	0 ××××	0 - 127
Match for any value.	1 1 1 1 1 1 1 1	0 1 1 0 0 1 0 0	×××××	0 - 255

ACL Wildcard Masking

- The flexibility available with wildcard masks is to identify subsets, such as **odd or even matches**.

Matching Rule: 172.16.100.0 0.0.0.255	
Reference IP Address: 172.16.100.0	10101100.00010000.01100100.00000000
Wildcard Mask: 0.0.0.255	00000000.00000000.00000000.11111111
Resulting Pattern: 172.16.100.X	10101100.00010000.01100100.XXXXXXXX

Matching Rule: 172.16.100.1 0.0.0.255	
Reference IP Address: 172.16.100.1	10101100.00010000.01100100.00000001
Wildcard Mask: 0.0.0.255	00000000.00000000.00000000.11111111
Resulting Pattern: 172.16.100.X	10101100.00010000.01100100.XXXXXXXX

Matching Rule: 192.168.5.1 0.0.254.255	
Reference IP Address: 192.168.5.1	11000000.10101000.0000101.00000001
Wildcard Mask: 0.0.254.255	00000000.00000000.11111110.11111111
Resulting Pattern: 192.168.odd#.X	11000000.10101000.XXXXXXX1.XXXXXXXX

ACL Wildcard Masking

- **Partial match** requirement results in a range of addresses matching the criteria, such as IPv4 addresses of many subnets. By carefully setting wildcard masks, with one ACE you can select a single IPv4 address or multiple IPv4 addresses.
- The example illustrate two usages of wildcard masks: to *match one* or a *range of subnets*.

IPv4 address	192.168.16.0	11000000.10101000.00010000.00000000
Wildcard Mask	0.0.15.255	00000000.00000000.0000 1111.11111111
Permitted IPv4 Address	192.168.16.0/24 to 192.168.31.0/24	11000000.10101000.00010000.00000000 11000000.10101000.00011111.00000000

Wildcard Mask Abbreviations

ACL statement keywords

- **host**
 - Wildcard mask **0.0.0.0**
- **any**
 - Wildcard mask **255.255.255.255**

Matching Rule Written with a Wildcard Mask	Matching Rule Written using Keywords
172.30.16.5 0.0.0.0	host 172.30.16.5
Examples: permit 172.30.16.5 0.0.0.0 deny 172.30.16.5 0.0.0.0	permit host 172.30.16.5 deny host 172.30.16.5

Examples:
permit 172.30.16.5 **255.255.255.255**
deny 172.30.16.5 **255.255.255.255**
permit 0.0.0.0 **255.255.255.255**

If a *wildcard mask is omitted* in the matching criteria in a standard IPv4 ACL, a wildcard mask of **0.0.0.0** is assumed.

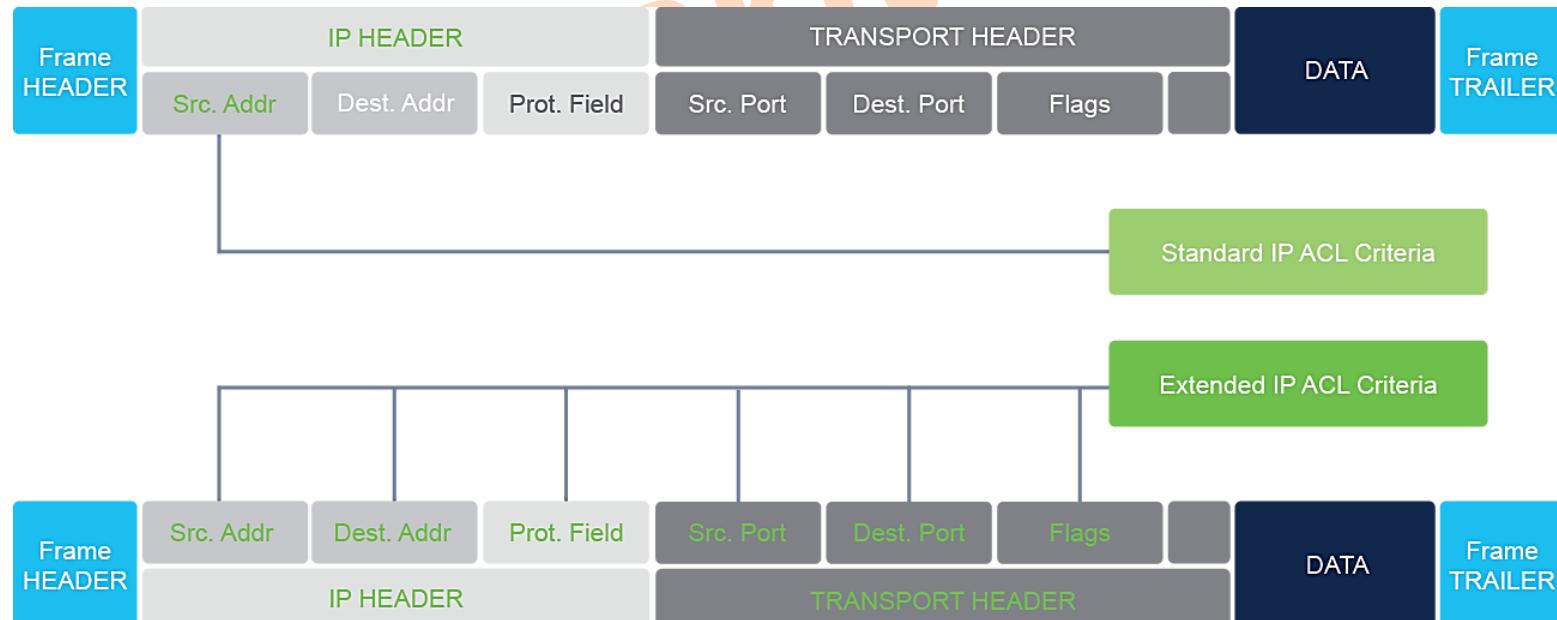
Guidelines for ACL Creation - ACL Best Practices

- Using ACLs requires attention to detail and great care.
- Mistakes can be costly in terms of downtime, troubleshooting efforts, and poor network service.
- Basic planning is required before configuring an ACL.

Guideline	Benefit
Base ACLs on the organizational security policies.	This will ensure you implement organizational security guidelines.
Write out what you want the ACL to do.	This will help you avoid inadvertently creating potential access problems.
Use a text editor to create, edit, and save all of your ACLs.	This will help you create a library of reusable ACLs.
Document the ACLs using the remark command.	This will help you (and others) understand the purpose of an ACE.
Test the ACLs on a development network before implementing them on a production network.	This will help you avoid costly errors.

Types of Basic ACLs

- **Standard IP ACLs**
 - Specify matching rules for **source addresses** of packets only
 - Filter traffic based in the **IP layer**, which means that they do not distinguish between TCP, UDP, or HTTPS traffic
- **Extended IP ACLs**
 - Examine both the **source and destination IP addresses**.
 - Also can check for **specific protocols, port numbers, and other parameters**



Types of Basic ACLs

The following are **configuration methods** for creating ACLs:

- **Numbered ACLs**
 - Use a number for identification
 - Configured in the global configuration mode
- **Named ACLs**
 - Use a descriptive name or number for identification
 - ACL statements are configured in Named Access List configuration mode

IPv4 ACL Type	Number Range or Identifier
Numbered Standard	1 – 99, 1300 – 1999
Numbered Extended	100 – 199, 2000 – 2699
Named (Standard and Extended)	Name

Configuring Standard IPv4 ACLs

```
access-list access-list-number {permit | deny} {source [source-wildcard] | host {address | name} | any}
```

Numbered ACL Configuration Command
Number Indicating ACL Type
Action to Perform on Matching Packet
Matching Criteria for Source IPv4 Address:

- Option 1: Reference IPv4 Address and a Wildcard Mask
- Option 2: Keyword **host** and a Reference IPv4 Address or Keyword **host** and Host Name
- Option 3: Keyword **any**

Examples of Different Configurations of the Same Standard IPv4 Access List:

- Numbered Configuration Method

```
RouterX(config)# access-list 1 deny host 172.16.3.3
RouterX(config)# access-list 1 permit 172.16.0.0 0.0.255.255
```

Numbered ACL Command and Standard ACL Number
Action
Source Matching Criteria
1st Statement: Keyword **host** with Reference IPv4 Address
2nd Statement: Reference IPv4 Address with Wildcard Mask

- Named Configuration Method

```
RouterX(config)# ip access-list standard acl2
RouterX(config-std-nacl)# deny host 172.16.3.3
RouterX(config-std-nacl)# permit 172.16.0.0 0.0.255.255
```

acl2 Specified as ACL Name

Configuring Extended IPv4 ACLs

[sequence-number] {permit | deny} protocol {source matching criteria} {destination matching criteria}

Sequence Number of the ACL Statement Action to Perform on Matching Packet Keyword Indicating Protocol Suite: ip, icmp, tcp or udp Source Matching Criteria for:

- Source IPv4 Address
- Source Port

 Destination Matching Criteria for:

- Destination IPv4 Address
- Destination Port

Example of an Extended ACL Statement:

- Allows TCP Connections from Ports 56000 to 60000 on the Host 172.16.3.3 to Port 80 on Host 203.0.113.30:

```
Router(config)# ip access-list extended 101
Router(config-ext-nacl)# permit tcp host 172.16.3.3 range 56000 60000 host 203.0.113.30 eq 80
```

Action Protocol Source Matching Criteria Destination Matching Criteria
Source IPv4 Address Source Port Destination IPv4 Address and Port

Configuring Extended IPv4 ACLs (Cont.)

- An example of a *numbered extended ACL* configuration on RouterX, **denying remote access via Telnet or SSH** from the devices in the **172.16.3.0/24** subnet and **permitting other traffic**, is below:

```
RouterX(config)# access-list 101 deny tcp 172.16.3.0 0.0.0.255 any eq 22
RouterX(config)# access-list 101 deny tcp 172.16.3.0 0.0.0.255 any eq telnet
RouterX(config)# access-list 101 permit ip 172.16.3.0 0.0.0.255 any
```

- An example of a *named configuration* of the same extended ACL from the previous example is:

```
RouterX(config)# ip access-list extended 101
RouterX(config-ext-nacl)# deny tcp 172.16.3.0 0.0.0.255 any eq 22
RouterX(config-ext-nacl)# deny tcp 172.16.3.0 0.0.0.255 any eq 23
RouterX(config-ext-nacl)# permit ip 172.16.3.0 0.0.0.255 any
```

Verifying and Modifying IPv4 ACLs

To verify the configured access-list, you can use the following commands:

- The **show access-lists** command displays the content of all configured ACLs. The output can be narrowed to a specific list by providing its number or its name
- The **show ip access-lists** command displays the content of all IPv4 access list. The output can be narrowed by specifying a specific ACL number or name.

```
RouterX# show access-lists 1
Standard IP access list 1
    10 deny host 172.16.3.3
    20 permit 172.16.0.0 0.0.255.255
```

```
RouterX# show ip access-lists 101
Extended IP access list 101
    10 deny tcp 172.16.3.0 0.0.0.255 any eq 22
    20 deny tcp 172.16.3.0 0.0.0.255 any eq telnet
    30 permit ip 172.16.3.0 0.0.0.255 any
```

Verifying and Modifying IPv4 ACLs (Cont.)

Delete an entire ACL

```
Router(config)# no access-list access-list-number
```

```
Router(config)# no access-list standard|extended access-list-name
```

Modify individual ACL statements

- Not possible using the numbered configuration method:
 - The **no** version *deletes entire ACL*:

```
Router(config)# no access-list 15 permit host 192.168.1.1
```

- You would have to **first copy the entire access list**, modify it in the text editor, delete it from the configuration and enter the modified ACL statements.

Verifying and Modifying IPv4 ACLs (Cont.)

Modify individual ACL statements

- Specify number as names to modify individual ACEs

```
Router(config)# ip access-list standard 1
```

- Add a specific statement: *sequence number* followed by the command

- When you choose the sequence number, you choose where the new entry will be placed in the ACL.
- You can use any number that is not currently assigned, even if it is not a multiple of 10.

```
Router(config-std-nacl)# 15 deny host 172.16.4.4
```

- Existing statements *cannot be modified directly*.

- To modify an existing statement, **delete it** and type in a new statement.

```
Router(config-std-nacl)# no [sequence-number]
```

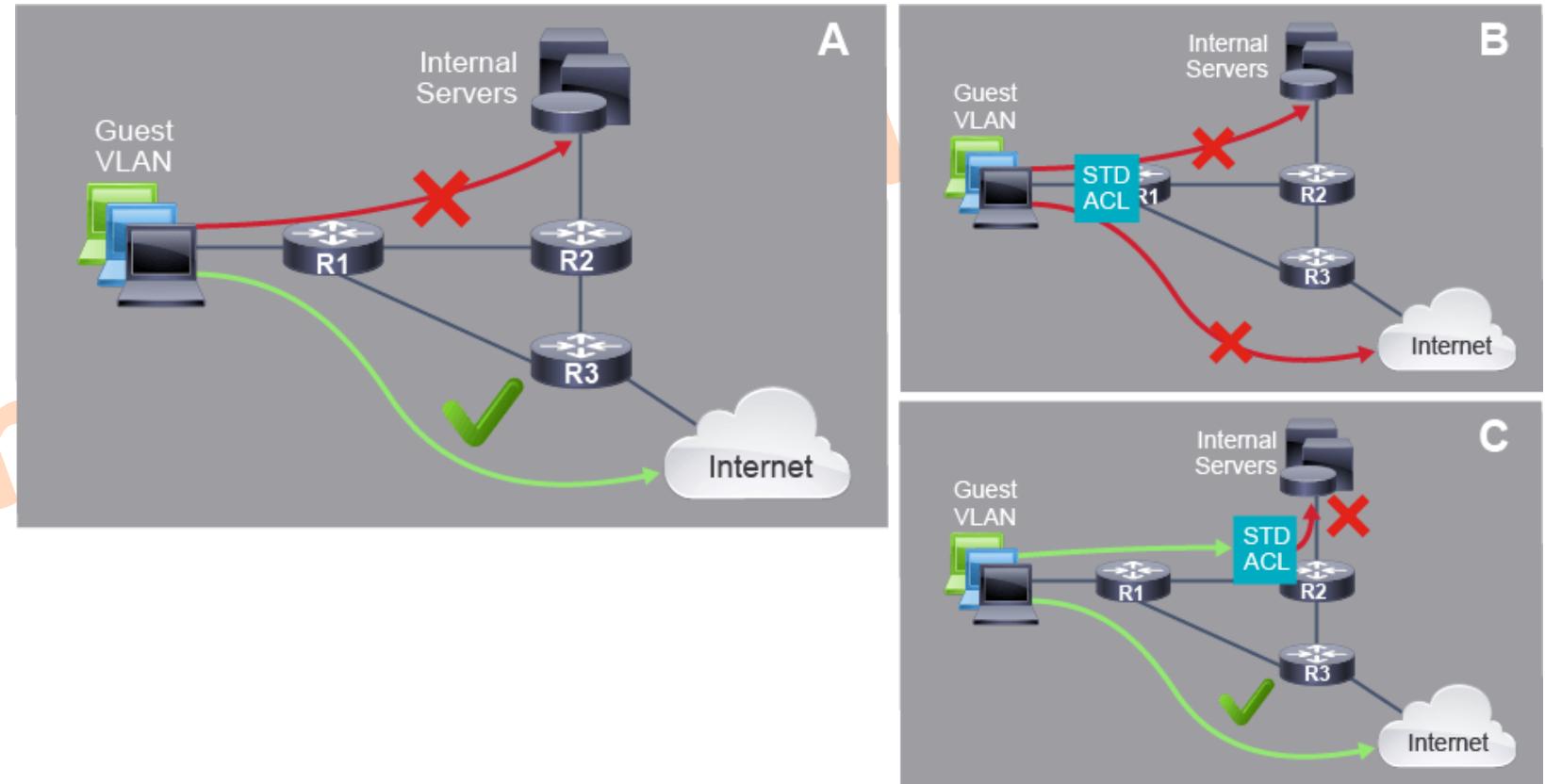
```
RouterX# show access-lists 1
Standard IP access list 1
  10 deny host 172.16.3.3
  15 deny host 172.16.4.4
  20 permit 172.16.0.0 0.0.255.255
```

Applying IPv4 ACLs to Filter Network Traffic

Standard ACLs

- Placed **close to the destination** of traffic
- If placed too close to source, may cause filtering out too much traffic
- Filter entire TCP/IP Protocol suite

Part

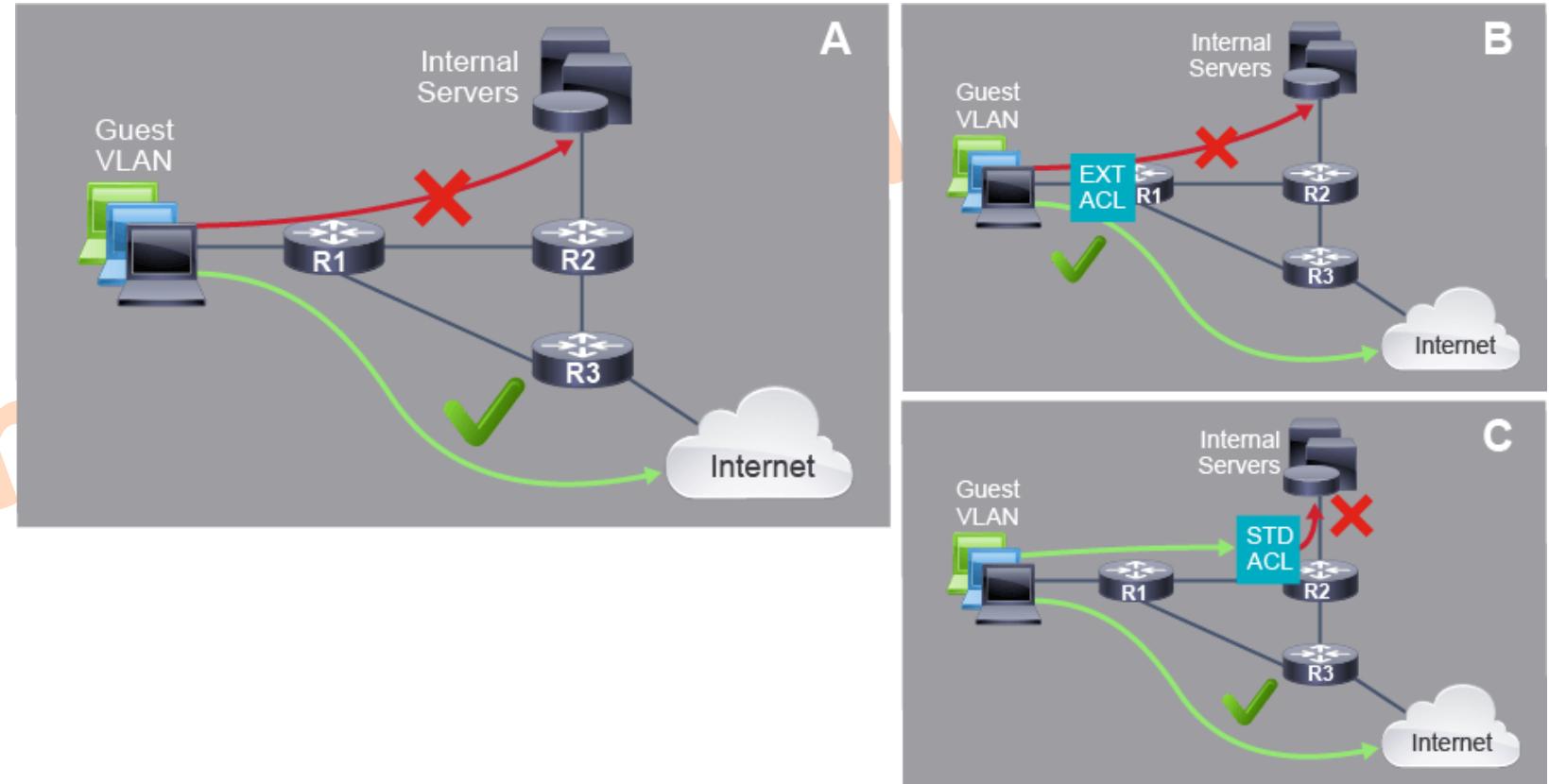


Applying IPv4 ACLs to Filter Network Traffic (Cont.)

Extended ACLs

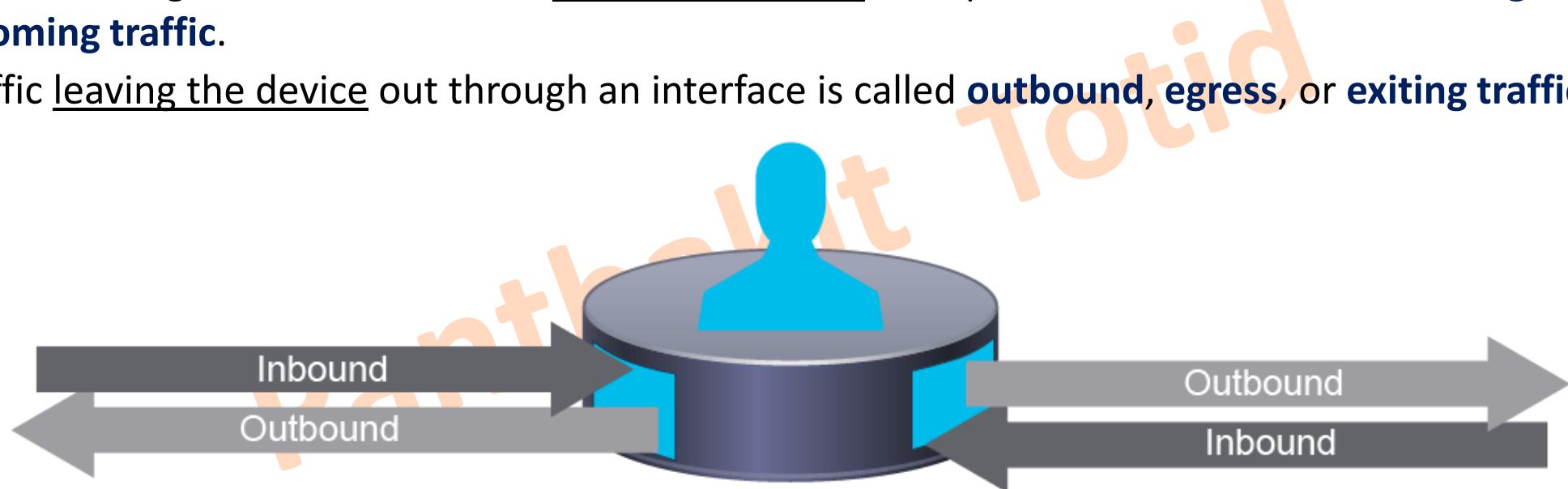
- Placed **close to the source** of denied traffic
- Provide granular control
- Prevent traffic from crossing network infrastructure unnecessarily

Part



Applying IPv4 ACLs to Filter Network Traffic (Cont.)

- There are two possible traffic directions:
 - Traffic arriving on an interface that enters the device to be processed is called **inbound, ingress, or incoming traffic**.
 - Traffic leaving the device out through an interface is called **outbound, egress, or exiting traffic**.

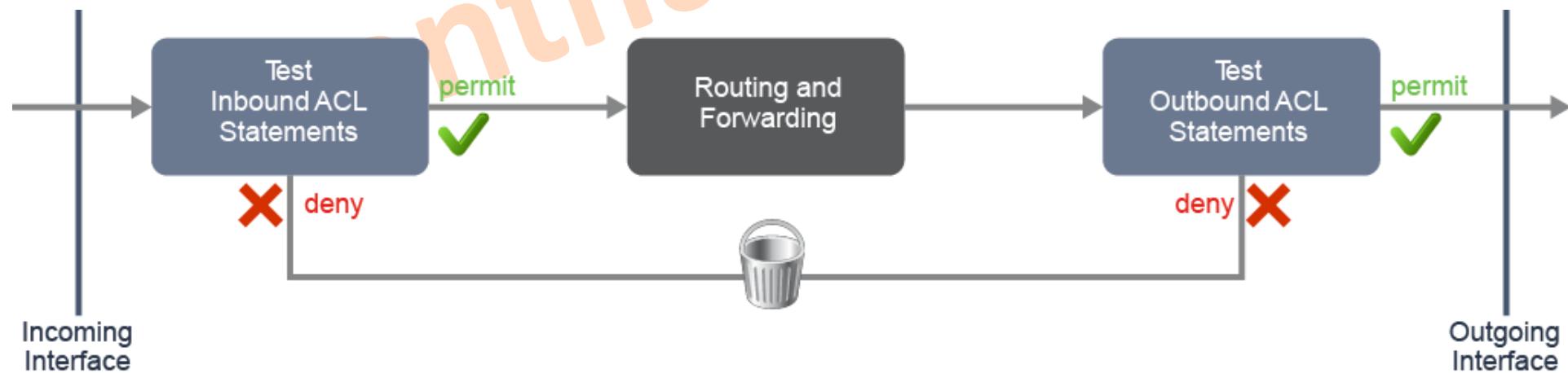


ACLs for traffic filtering do not act on packets that originate from the router itself.

Applying IPv4 ACLs to Filter Network Traffic (Cont.)

How packet processing occurs when ACLs are applied:

- **Inbound ACLs** process incoming packets as they enter the interface, *before they are routed* to the outbound interface.
 - Efficient, it saves the overhead of routing lookups if the packet is discarded.
 - If the packet is permitted by the ACL, it is then processed for routing.
- **Outbound ACLs** process packets that are routed to the outbound interface.
 - They are processed before they exit the interfaces.



Applying IPv4 ACLs to Filter Network Traffic (Cont.)

- After you have configured an ACL, you link the ACL to an interface using the **ip access-group** command.
- To **remove an ACL from an interface**, first enter the **no ip access-group** command on the interface, then enter the global **no access-list** command to remove the entire ACL if needed.
- You can configure **one ACL per protocol, per direction, per interface**:

```
Router(config-if)# ip access-group {access-list-number | access-list-name} {in | out}
```

Router (config-if) # ip access-group {access-list-number | access-list-name} {in | out}

Interface Configuration Mode

Command for Associating an ACL to an Interface

The Number or the Name of the ACL You Wish to Link to the Interface

Direction of Traffic that is to be Processed by the ACL
In=Inbound
Out=Outbound

```
Branch(config-if)# ip access-group 101 in
```

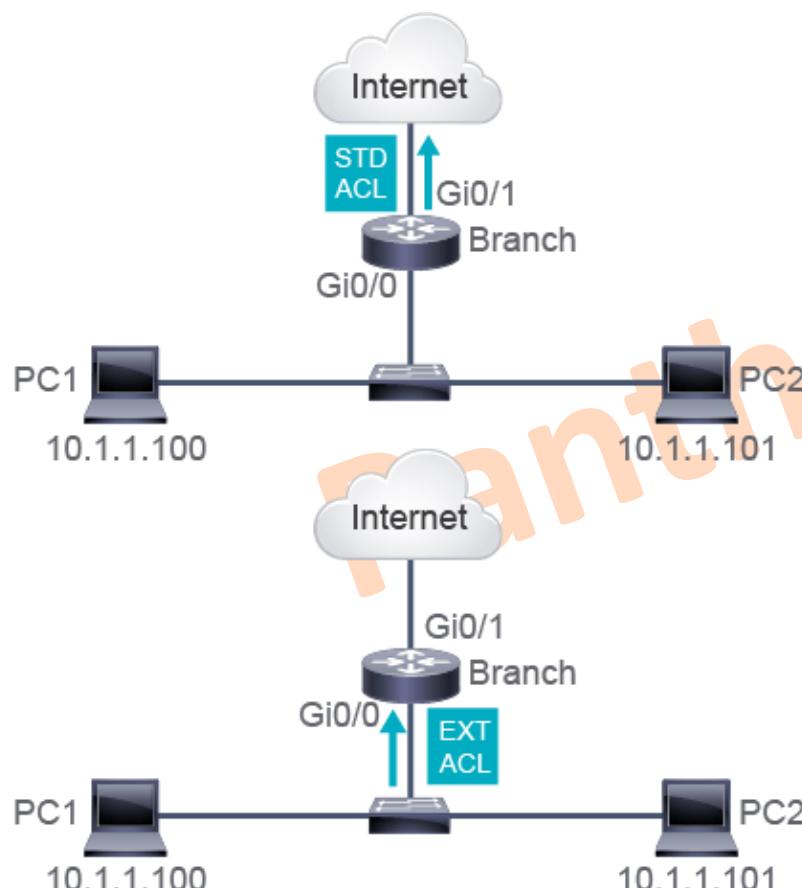
Applying Extended ACL 101 on the Interface as an Inbound Filter

```
Branch(config-if)# ip access-group PERMIT_ICMP out
```

Appling extended ACL PERMIT_ICMP on the interface as an oubound filter.

Applying IPv4 ACLs to Filter Network Traffic (Cont.)

- The figure shows a scenario in which ACL is used to **deny access to the internet only** to the host IPv4 address **10.1.1.101**. Traffic from other hosts within 10.1.1.0/24 is allowed.



Security Policy
• Deny Internet Access for PC2

```
BRANCH# show access-lists
Standard IP access list 15
    10 deny 10.1.1.101
    20 permit 10.1.1.0 0.0.0.255
BRANCH(config)# interface GigabitEthernet 0/1
BRANCH(config-if)# ip access-group 15 out
```

```
BRANCH# show access-lists
Extended IP access list NOINTERNET_PC2
    10 deny ip host 10.1.1.101 any
    20 permit ip 10.1.1.0 0.0.0.255 any
BRANCH(config)#interface GigabitEthernet 0/0
BRANCH(config-if) ip access-group
NOINTERNET_PC2 in
```

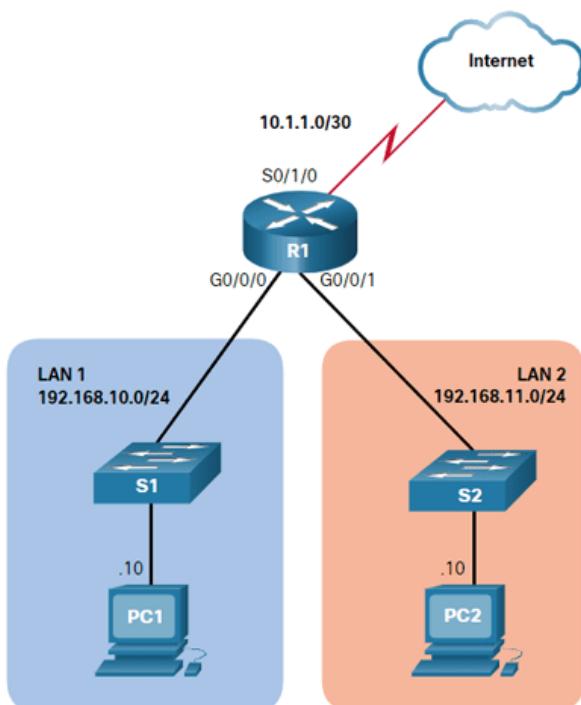
Secure VTY Ports with a Standard IPv4 ACL

- A standard ACL can *secure remote administrative access* to a device using the **vty lines** by implementing the following two steps:
 - Create an ACL to identify which administrative hosts should be allowed remote access.
 - Apply the ACL to incoming traffic on the vty lines.

```
Router(config-lines)# access-class {access-list-number|access-list-name} {in|out}
```

Secure VTY Ports with a Standard IPv4 ACL (Cont.)

- This example demonstrates how to configure an ACL to filter vty traffic.
 - First, a local database entry for a user ADMIN and password class is configured.
 - The vty lines on R1 are configured to use the local database for authentication, permit SSH traffic, and use the ADMIN-HOST ACL to restrict traffic.



```
R1(config)# username ADMIN secret class
R1(config)# ip access-list standard ADMIN-HOST
R1(config-std-nacl)# remark This ACL secures incoming vty lines
R1(config-std-nacl)# permit 192.168.10.10
R1(config-std-nacl)# deny any
R1(config-std-nacl)# exit
R1(config)# line vty 0 4
R1(config-line)# login local
R1(config-line)# transport input telnet
R1(config-line)# access-class ADMIN-HOST in
```

Secure VTY Ports with a Standard IPv4 ACL (Cont.)

- After an ACL to restrict access to the vty lines is configured, it is important to verify it works as expected.
- To verify the ACL statistics, issue the `show access-lists` command.
 - The match in the permit line of the output is a result of a successful SSH connection by host with IP address 192.168.10.10.
 - The match in the deny statement is due to the failed attempt to create a SSH connection from a device on another network.

```
R1#  
Oct  9 15:11:19.544: %SEC_LOGIN-5-LOGIN_SUCCESS: Login Success [user: admin] [Source: 192.168.10.10]  
[localport: 23] at 15:11:19 UTC Wed Oct 9 2019  
R1# show access-lists  
Standard IP access list ADMIN-HOST  
    10 permit 192.168.10.10  (2 matches)  
    20 deny   any   (2 matches)  
R1#
```