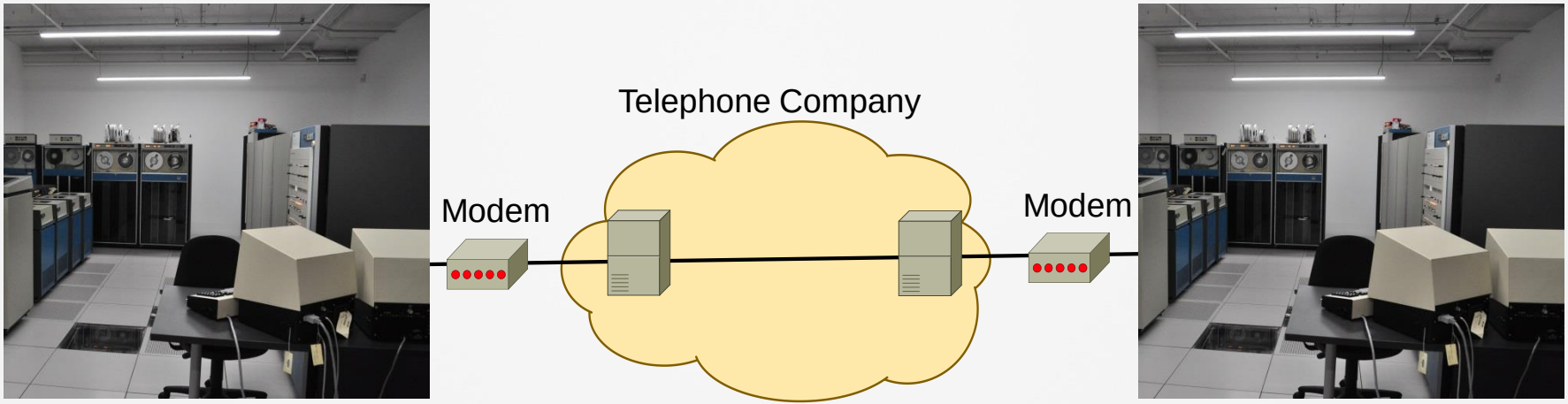
Intro to Networking

Early Days of Computing

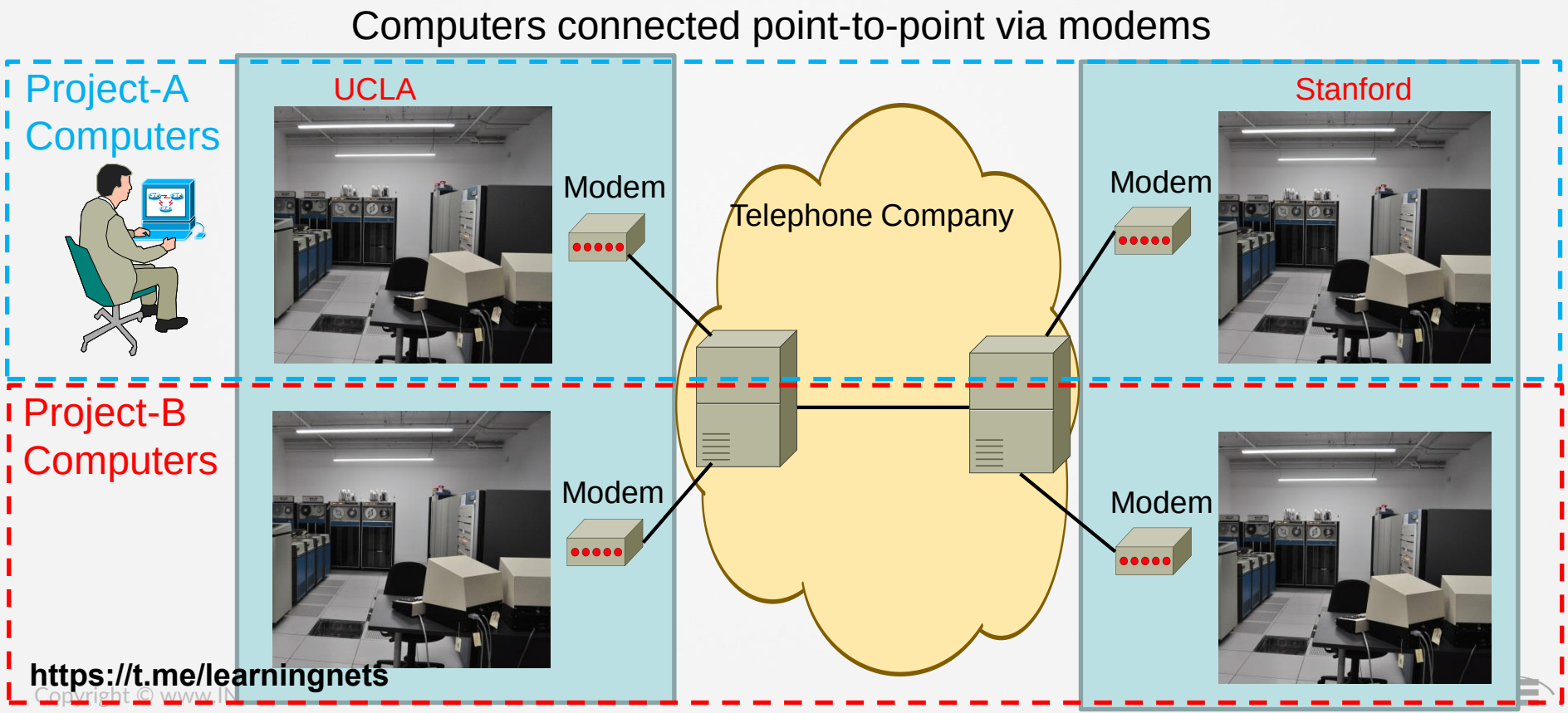
Terminals connected to computers (SDS Sigma-7 shown)

Early Days of Computing (2)

Computers connected point-to-point via modems

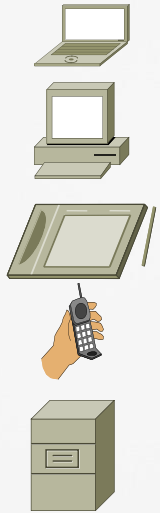


Early Days of Networking (1)

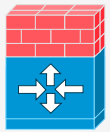


Physical Components of Networks - Hosts

- Hosts
  - Laptops
  - PCs
  - Tablets
  - Smartphones
  - Servers



Physical Components of Networks – Network Infrastructure

- Network Access Devices
  - Hubs 
  - Switches 
  - Access Points 
- Network Infrastructure Devices
  - Routers 
  - Firewalls 

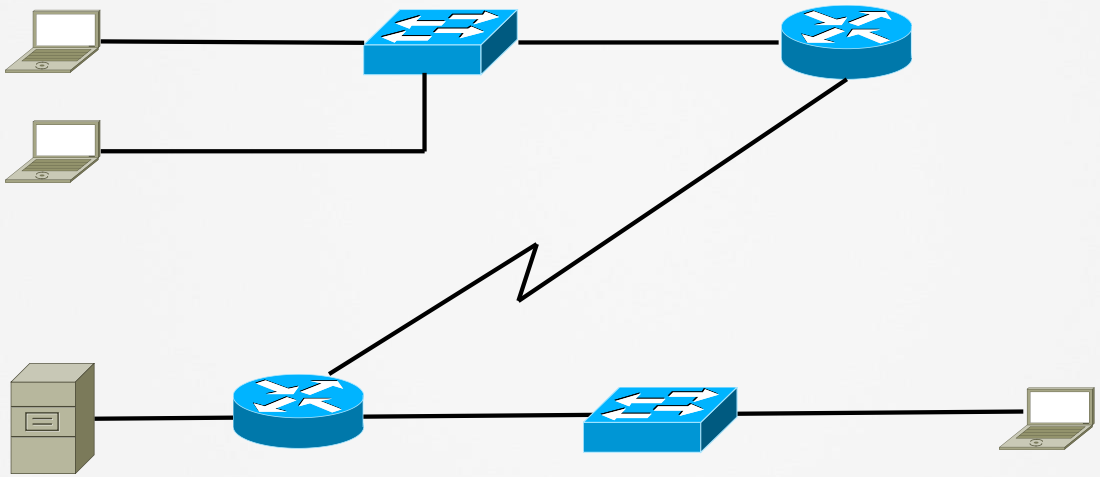
Cables

- **Copper Cables**
  - Co-axial
  - Twisted Pair
    - Shielded
    - Unshielded
- **Fiber Optic Cables**
  - MMF
  - SMF
  - GBICs

RJ-45

GBIC

Network/Topology Diagrams



Topologies – Logical and Physical

- **Logical Topology**
  - What the network looks like to the end-device
- **Physical Topology**
  - How the network is actually cabled
    - Bus
    - Star
    - Ring
- **Fully-Meshed vs. Partially-Meshed**

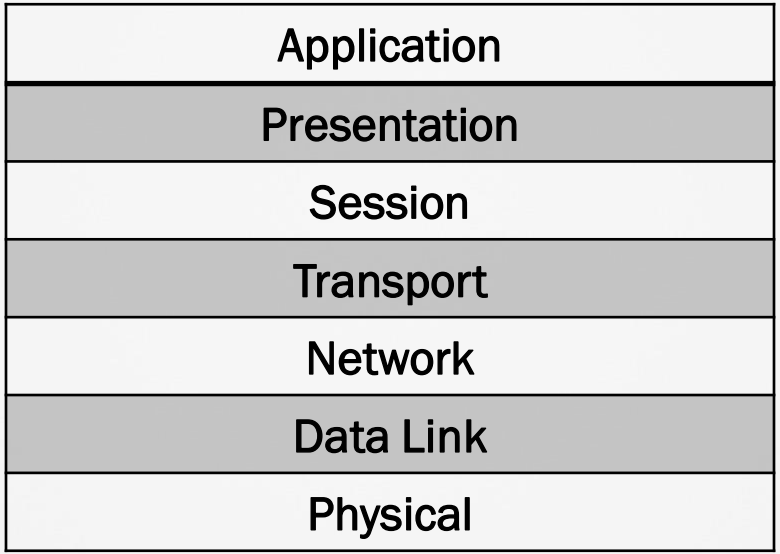
Any Questions?

OSI Reference Model
Part 1

OSI Layer

- Comprised of seven layers
- The benefits of using a layered approach are:
  - Provides easier troubleshooting
  - Standardizes the networking architecture
  - Allows vendor interoperability

OSI Layer



OSI Layer

- Upper layer
  - Application
  - Presentation
  - Session
- Lower layer
  - Network
  - Data link
  - Physical

Application Layer

Features

- Interacts with the user applications (Firefox, Outlook, etc.)
- Provides initial network connection for user applications
- Manages the application connections between hosts

Presentation Layer

- Performs encryption within an application
- Ensures that data is presented correctly to the application used
- Performs translation of cross-platform standards that may be understood by the local machine:
  - Pict. into .jpg file translation
  - .wav into .mp3

Session Layer

- Helps establish session with reserved port numbers
- Session identifier is assigned
- Tracks connections between hosts and remote computers/servers

Session Layer

- **Well-known ports**
  - Ranges from 0 to 1023
  - Port numbers used by well-known services
  - Examples: HTTP(80), HTTPS(443), DNS(53), FTP(20,21) ,TELNET(23), etc.
- **Registered ports**
  - Reserved for the applications
    - Ranges from 1024 to 65535
- **Ephemeral ports**

End of OSI Part-1

OSI Reference Model
Part 2

Transport Layer

- **Identifying services**
  - TCP
    - Connection oriented
    - Reliable
    - Protocol number 6
  - UDP
    - Connectionless
    - Unreliable
    - Protocol number 17

Transport Layer

- Multiplexing and de-multiplexing
- Fragmentation
- Sequencing and reassembling
- Windowing, buffering, congestion avoidance
- Error correction
- Examples: EIGRP, OSPF

Network Layer

- **Routed protocol**
  - Protocols that are used for identification
    - IP, IPX, AppleTalk
- **Routing protocol**
  - Protocols that are used to find the routed protocols
    - EIGRP, OSPF, etc.
- **Example**
  - Router

Data Link Layer

- MAC
  - 48-bit addressing system
    - Example: aaaa.aaaa.aaaa
    - First 24 bits are considered OUI
    - Remaining 24 bits are considered vendor assigned

Data Link Layer

- LLC
  - WAN protocols
    - PPP
    - HDLC
    - Frame Relay
- Example
  - Switch, bridge

Physical Layer

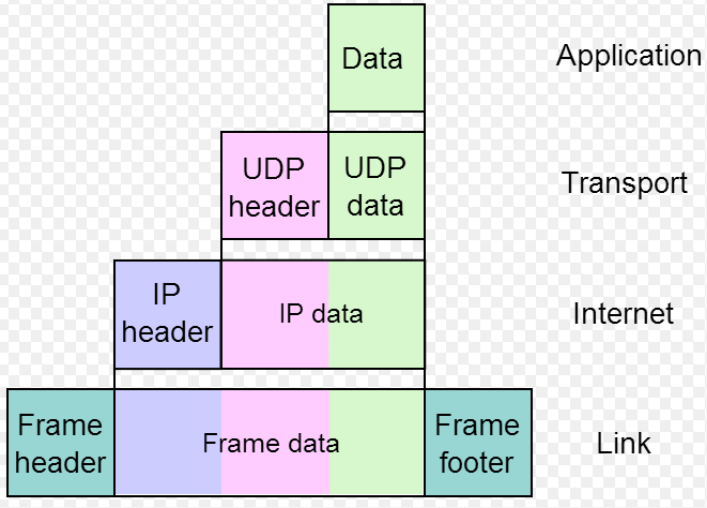
- Electrical signals carried over the physical layer
- Devices used at the physical layer
  - Hubs
  - Repeaters
  - Network interface cards (NICs)
  - Cables (Ethernet, fiber-optic, serial, etc.)

PDUs

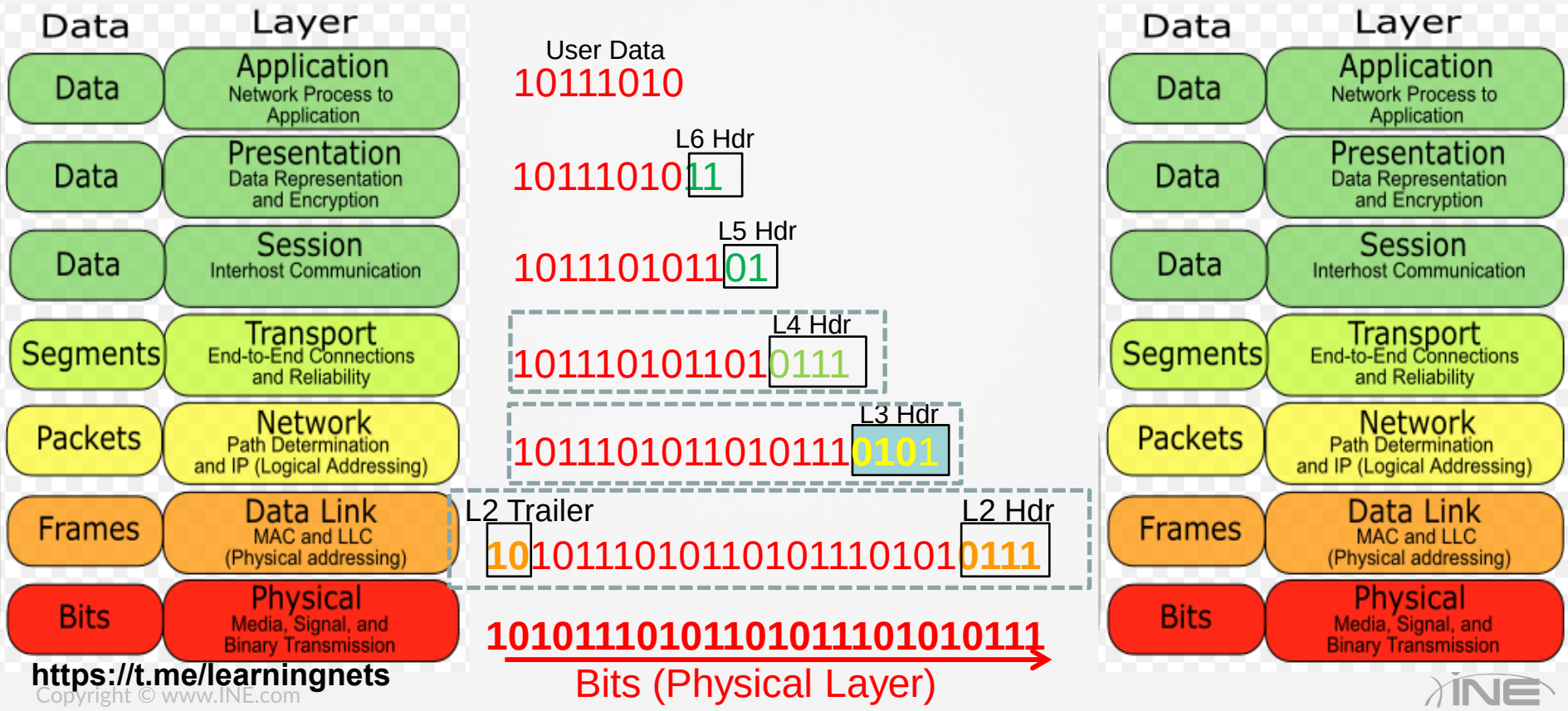
- PDU = Protocol Data Unit
  - The final, structured data unit created by an OSI Layer
- PDUs created at one layer are meant to be read by the same layer on receiving device

Encapsulation / Decapsulation

- Encapsulation: As each layer receives a PDU from the layer above it, headers are added.
- Decapsulation: As each layer receives a PDU from the layer below it, headers are inspected and then removed.



PDU Transportation & Recognition



Any Questions?

Ethernet Basics

Standards Organizations

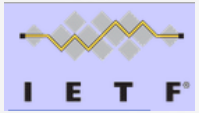
IEEE

IANA

ITU

FCC

IETF



Wi-Fi Alliance

INE is not affiliated with any of these organizations.

Broadcast & Collision Domains

- Broadcast domains
  - Everyone sees all frames
- Collision domains
  - Collection of devices that all access a shared medium
- How to send traffic on a wire that EVERYONE can access at the same time?
  - TDM
  - FDM
  - Other?

CSMA/CD

- Half Duplex vs. Full Duplex
- Carrier Sense
- Multiple Access
- Collision Detect

Distance, Cables, & Duplex

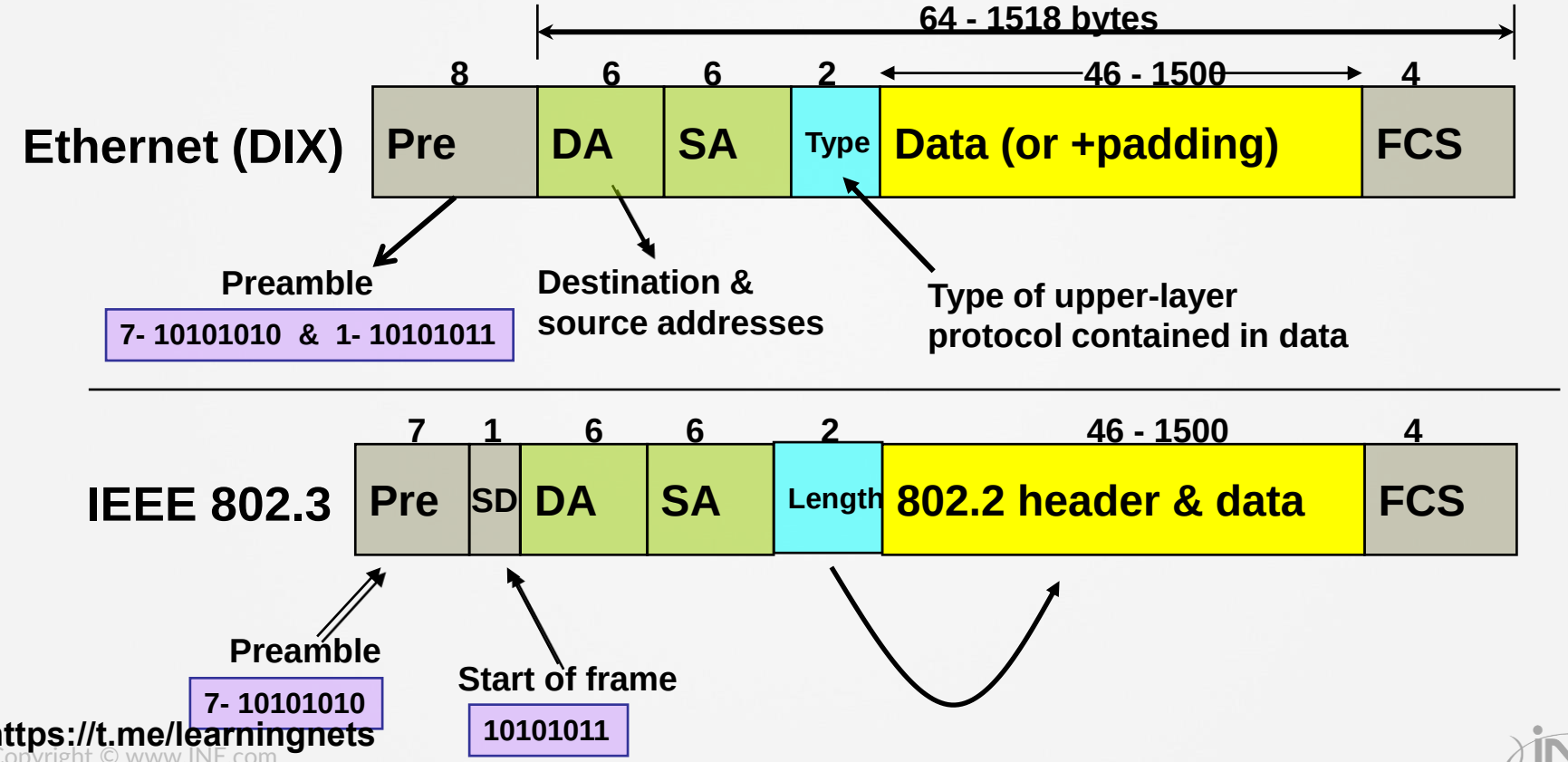
The developers of Ethernet had some additional decisions to make:

1. Maximum distance of transmission?
   - They decided on 100 meters.

2. What type of cable?
   - They decided on copper (coaxial) cable.

Understanding 10BaseX nomenclature

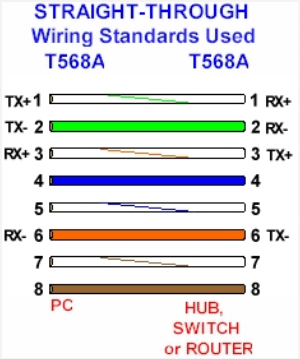
Ethernet Frame Structure



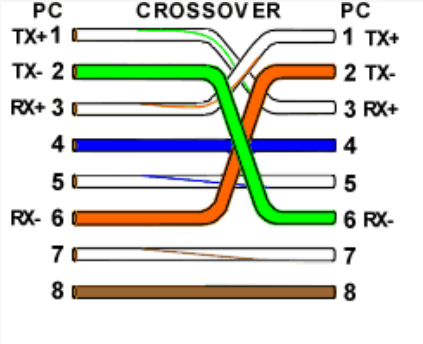
Ethernet Cabling Details

- Twisted-pair cabling comes in three varieties:

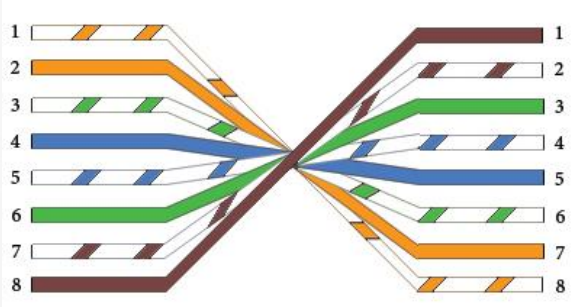
Straight-through



Crossover

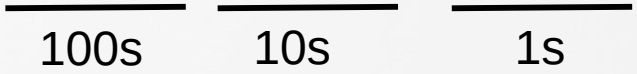


Rolled

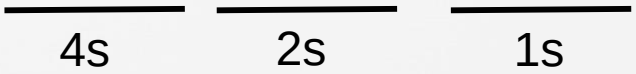


Binary and Decimal

- Decimal = Base10
  - ✓ In any given position one can select a single digit from 0 - 9



- Binary= Base2
  - ✓ In any given position one can select a single digit from 0 - 1



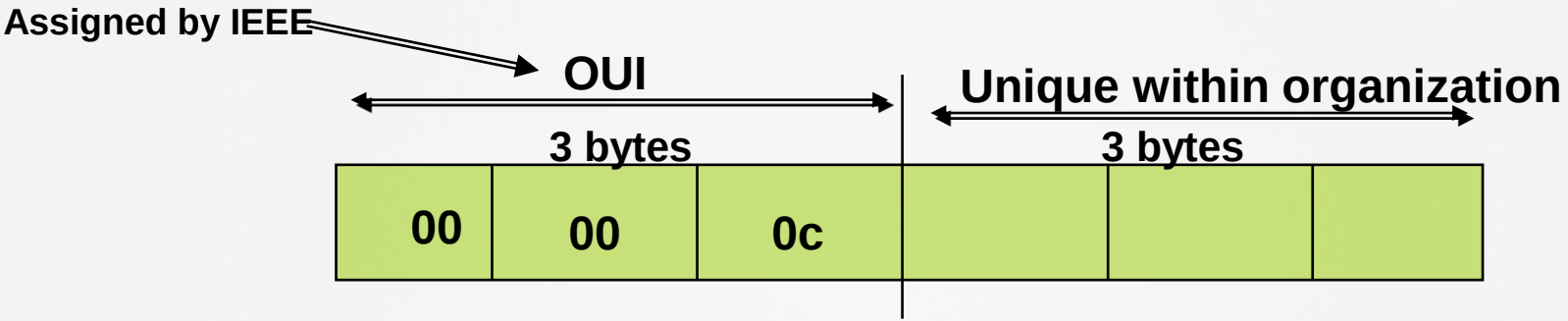
Hexadecimal

- Hexadecimal = Base16
  - ✓ In any given position one can select a single number from 0 - F

256s	16s	1s

0-9 are same as decimal
A = 10
B = 11
C = 12
D = 13
E = 14
F = 15

MAC Addresses

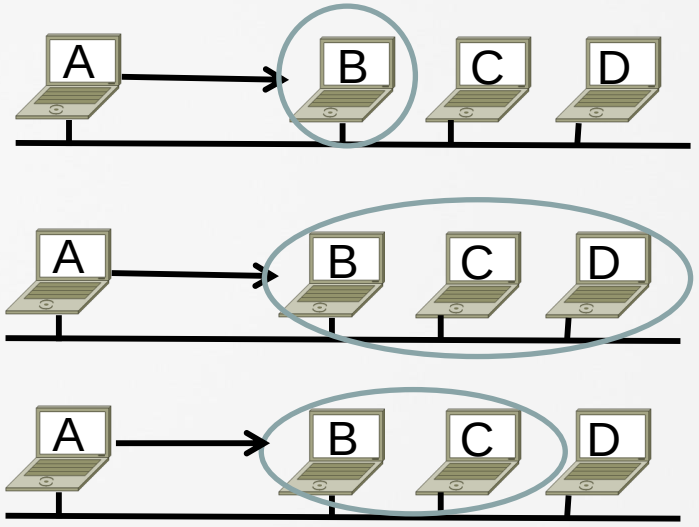


Two bits of 1st byte are very important

- **bit #1 (LSB) => Individual (0) or group (1) addresses- (only for DAs)**
- **bit #2 => Global (0) or Local (1)- LAA bit**

Data Transmission Types

- Unicast
  - One-to-one
- Broadcast
  - One-to-all
- Multicast
  - One-to-a-group

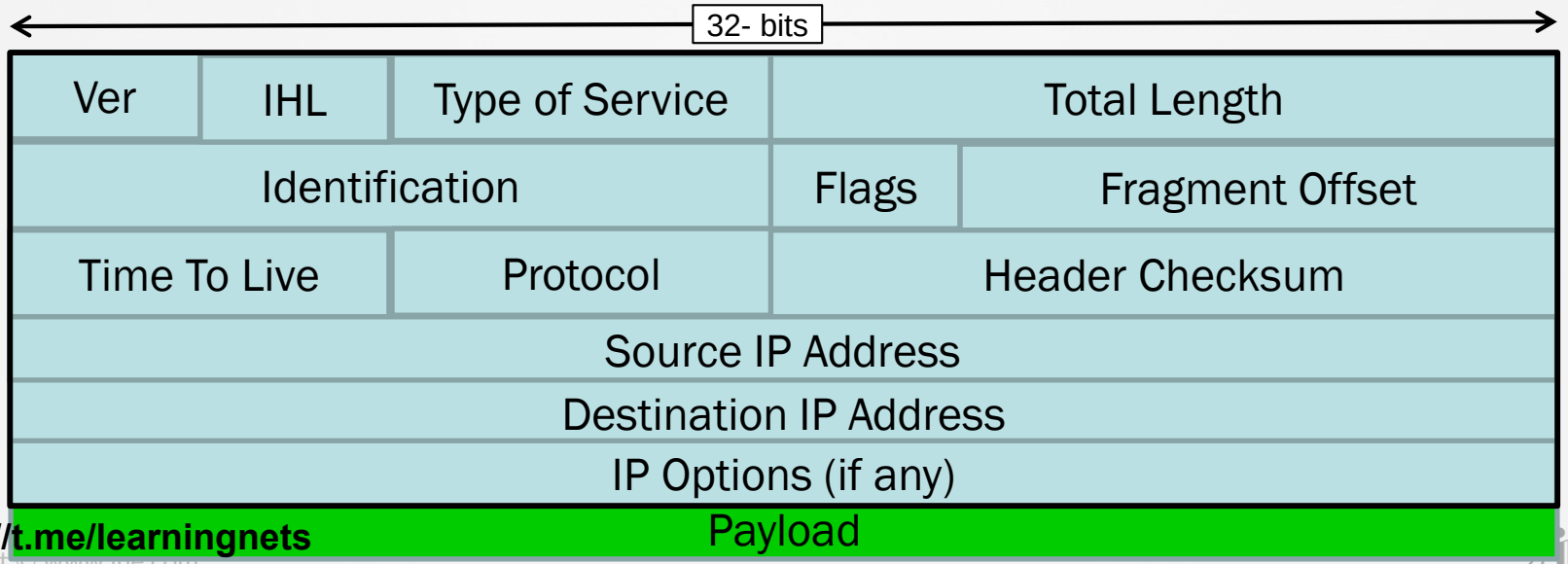


Any Questions?

IPv4, UDP and TCP

Introduction to IPv4

- Internet Protocol version 4
  - Resides at OSI Layer-3 (Network Layer)
  - Connectionless



Introduction to IPv4

- 32-bit addressing system
- Logical address for a network defined by IANA
- IPv4 addresses are comprised of 4 octets
- Dotted decimal notation is used to segment the octet

Communication Types

- **Unicast**
  - One-to-one communication
- **Multicast**
  - One-to-many communication
- **Broadcast**
  - One-to-all communication

DHCP

- **Dynamic Host Configuration Protocol**
  - Dynamic assignment of IP information
  - Based on older BootP protocol
  - Client / Server
  - Utilizes UDP (port 67 and 68)

ARP

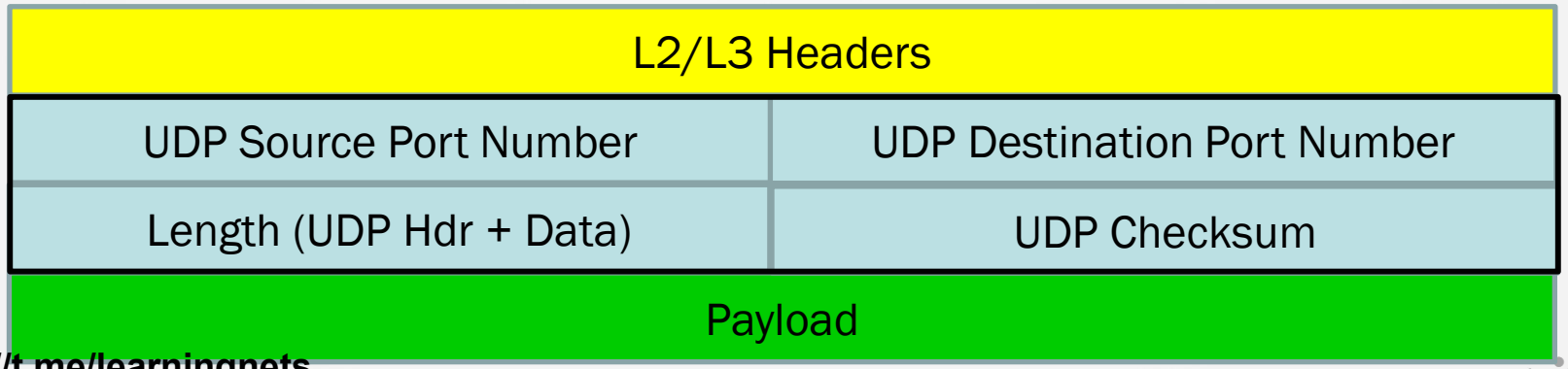
- **Address Resolution Protocol**
  - Used to resolve Layer-2 address of hosts on same LAN.
  - Broadcast-based
- **Proxy ARP**
  - Optional feature on routers and Wi-Fi access points
  - Router replies on behalf of hosts

DNS

- Domain Name Service
- Used by computers to resolve names to IP addresses.
- Typically uses UDP port 53.
- DNS server responds to DNS requests
  - Host sends DNS A-Record query
  - DNS server responds with A-Record query response.

OSI Transport Layer - UDP

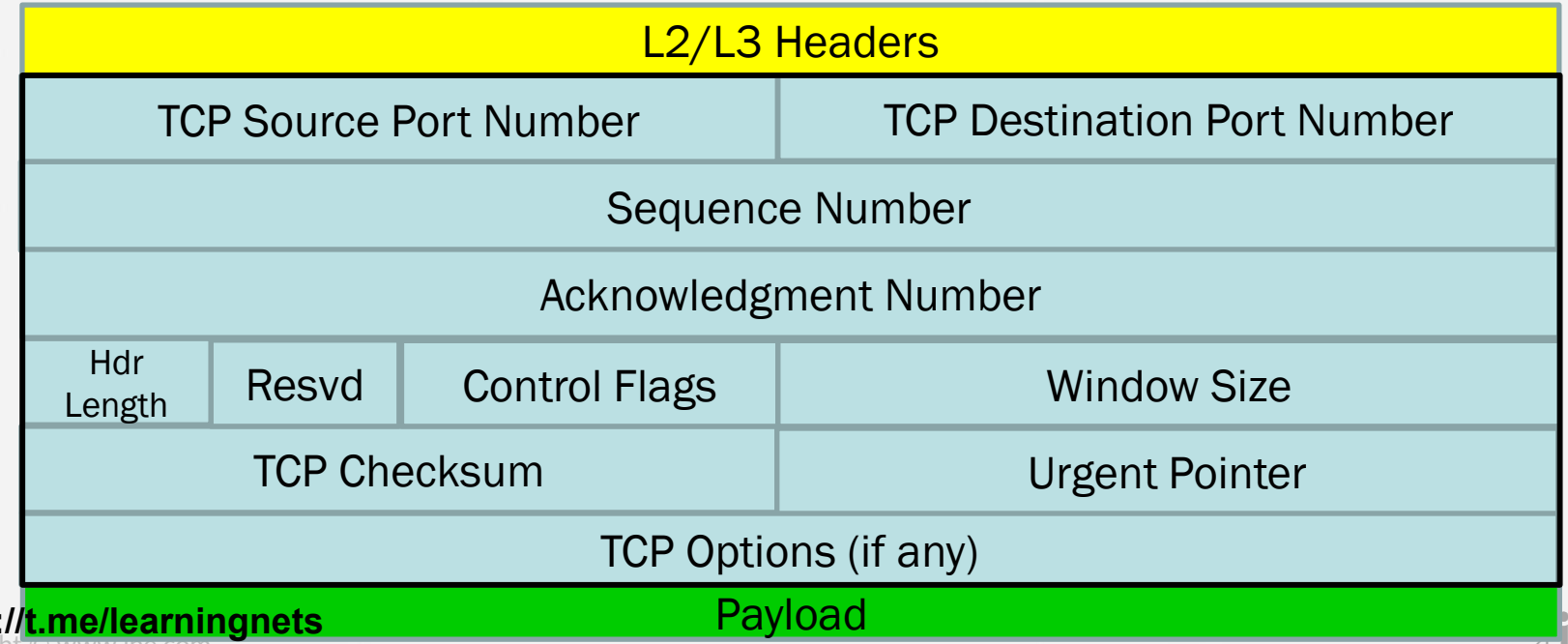
- Predominant protocols used at Layer-4
  - TCP (Transmission Control Protocol)
  - UDP (User Datagram Protocol)
- UDP
  - Connectionless



OSI Transport Layer - TCP

Transmission Control Protocol

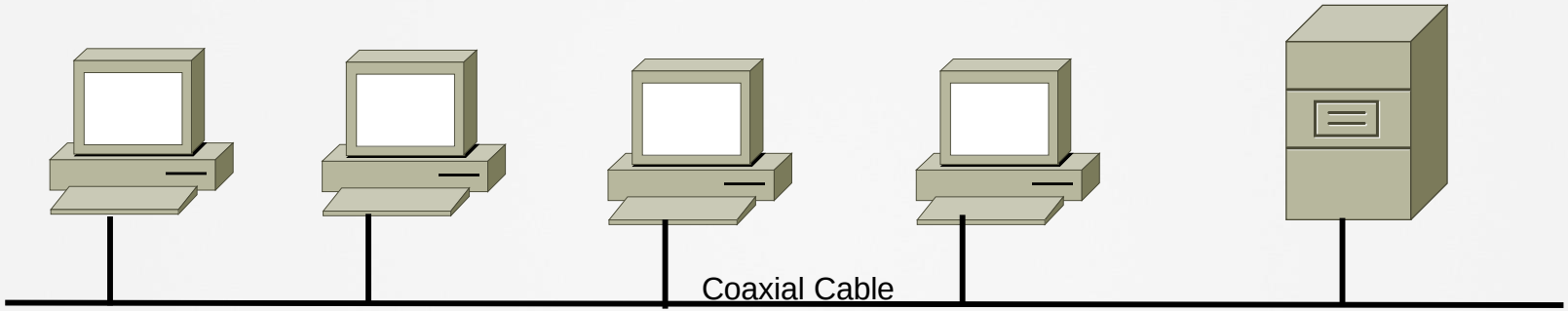
- Connection-oriented



Any Questions?

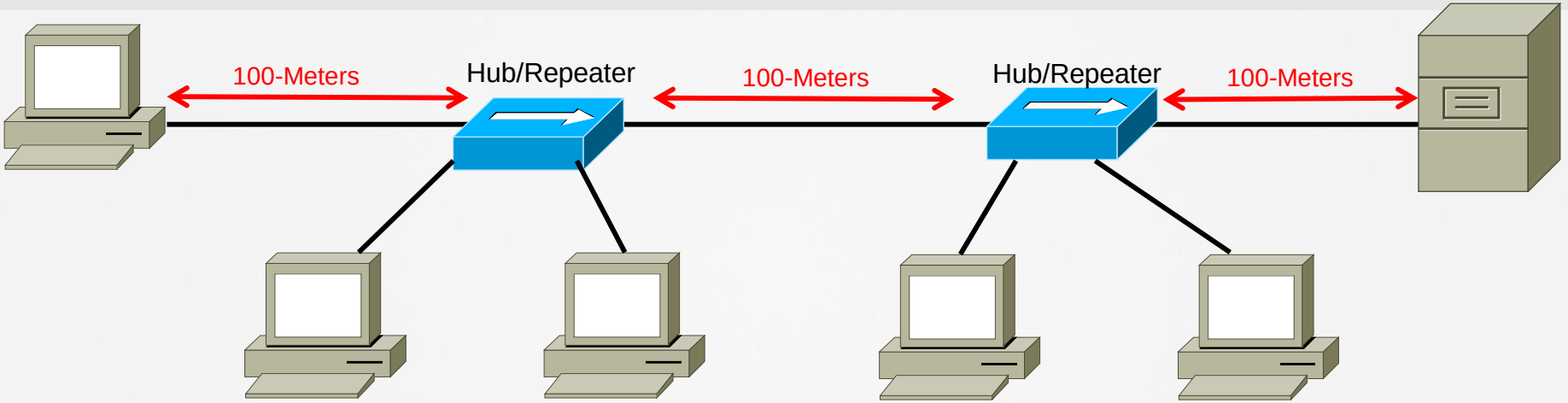
Switching

Evolution of Switching (1)



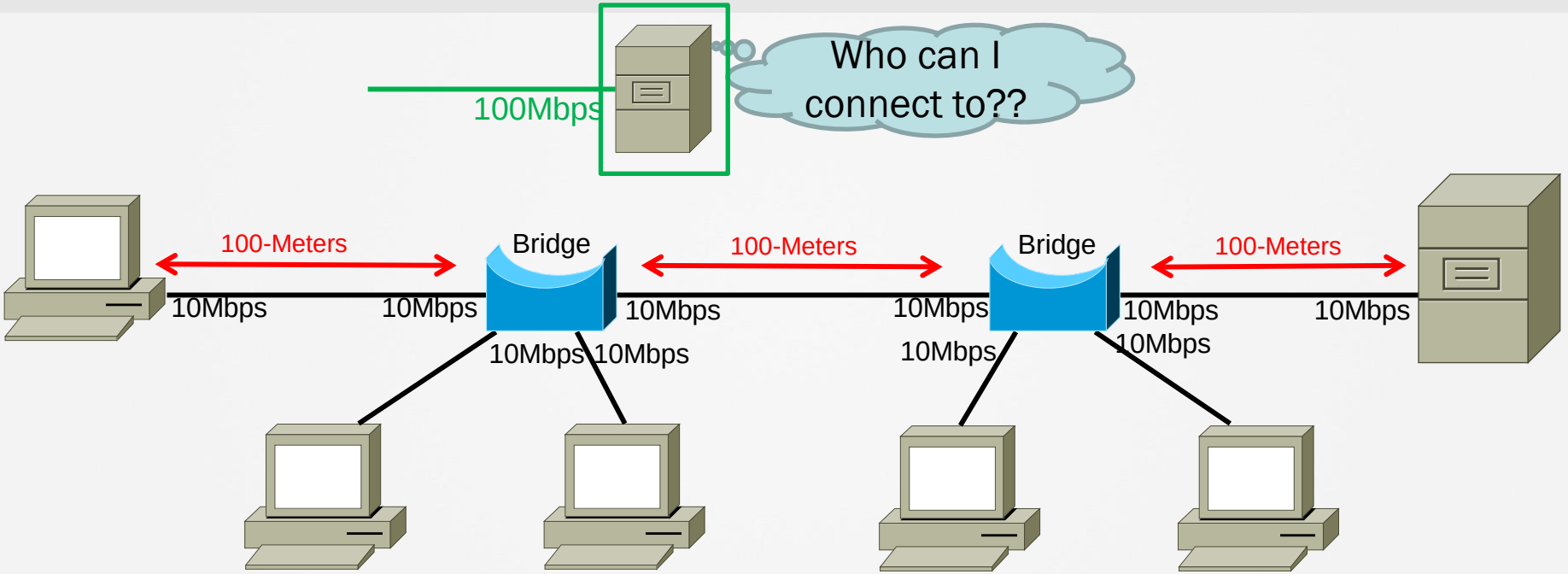
Vampire Tap

Evolution of Switching (2)

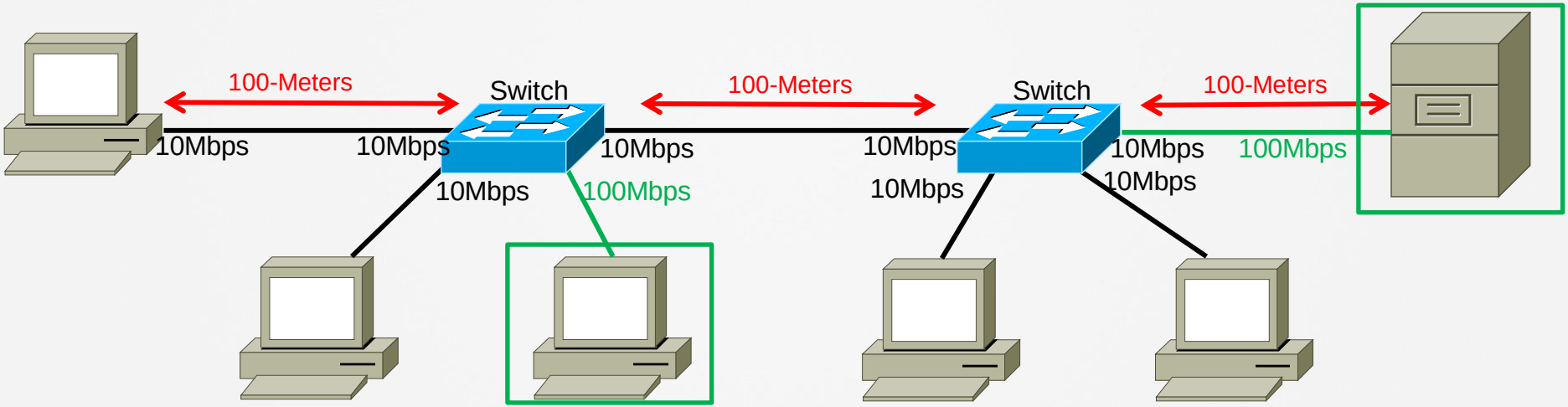


Ethernet Transceiver

Evolution of Switching (3)



Evolution of Switching (4)



Intro to Switching

- Switch is a multiport bridge
  - More ports than a bridge
  - Mixture of port speeds & types
- Forwards frames based on the MAC address table
- Separates collision domain
- Operates in data link layer

MAC-Address Table

- **Switch MAC Learning**
  - Based on Source MAC Address
  - Addresses age out after inactivity-timer
- **Switching forwarding**
  - Based on Destination MAC
  - Broadcast/Multicast/Unknown flooding
  - All ports initially in one, large, broadcast domain

Any Questions?

Intro to Cisco IOS

Introduction to IOS

- Internetworking Operating System
- Native software for Cisco routers and switches
- Cisco develops different IOSs for different platforms
  - Example: Cisco 1841, Cisco 2821, etc.
- Usually operated through CLI

Device Startup Sequence

- Cisco routers and switches generally perform the same steps upon initial startup
  - Discover device hardware
  - Find and load IOS image
  - Find and load configuration file.
- Memory Types
  - Flash, NVRAM, and DRAM

Accessing Device via CLI

- Basically, two methods of configuring router/switch
  - CLI (command-line interface)
  - GUI (graphical user interface)
- Console port is used for initial configuration
- Prerequisites
  - Console cable
  - Terminal emulator

Accessing Device via CLI

- Connect console cable into the “console” port of a Cisco device
- Open terminal emulator software like Putty or SecureCRT
- Choose serial option with default baud rate, such as 9600

IOS Command Structure

- **IOS has a command hierarchy**
  - Router> - User (or EXEC) mode
  - Router# - Privileged EXEC (or Enable) mode
  - Configuration modes

    Router(config)# - Global Configuration Mode

    Router(config-if)# - Interface Configuration Mode

    Router(config-router)# - Router Configuration Mode
  - Usage of Exit, End, Ctrl-Z

Initial Configuration Commands

- Prevent syslog and event messages from interrupting CLI input
  - Router(config-line)# logging synchronous
- Prevent DNS resolution attempt for mis-typed commands
  - Router(config)# no ip domain-lookup
- Configure descriptive device name
  - Router(config)# hostname Lab-1-Rtr

Initial Configuration Commands

- Configure informative banner
  - Router(config)# banner motd
- Add IPv4 address to an interface
  - Router(config-if)#ip address <address><mask>
  - Router(config-if)# no shutdown

Monitoring Memory and Images

- Display current IOS version running
  - Router# show version
- Display all memory locations and file names
  - Router# dir all
- Display saved, startup configuration file
  - Router# show startup-config
- Display current running configuration
  - Router# show running-config

Saving and Deleting Configurations

- Save current Running Configuration

  Router# copy running-config startup-config

  Or…

  Router# write memory

- Setting a router back to factory defaults
  - Step-1: Delete startup configuration

    Router# erase startup-config

    Or…

    Router# write erase
  - Step-2: Reload the router

    Router# reload

Securing Device Access

- Configuring enable password
  - Switch(config)# enable password <password>

  OR
  - Switch(config)# enable secret <password>
- Configuring console password
  - Switch(config)# line console 0
  - Switch(config-line)# password <password>

Securing Cisco Devices

Configuring Telnet password

- Switch(config)# line vty 0 4
- Switch(config-line)# password <password>
- Switch(config-line)# login

OR

- Switch(config)# username <username> privilege 15 password <password>
- Switch(config-line)# login local

Any Questions?

Basic Switch Configuration

Initial Tasks

- Perform initial configuration on Switch
  - Hostname
  - Enable password
  - Console Password
  - Banner
  - “Convenience” commands
    - No ip domain-lookup
    - Logging synchronous
- Verify naming convention of ports on your switch
  - Show ip interface brief

Basic Switch Configuration

- Switchports primarily used for switching Layer-2 Ethernet Frames.
  - Don't natively support IP addressing
- Switch Management IP address configured on a logical interface.
  - Switched Virtual Interface (SVI)
  - Initially in same broadcast domain as all physical ports.
  - May be disabled by default.

Configuring Management Address

- **Configuration commands**
  - Switch(config)# interface vlan 1
  - Switch(config-if)# ip address <address> <subnet mask>
  - Switch(config-if)# no shutdown
  - Switch(config-if)# exit
  - Switch(config)# ip default-gateway <default-gateway>

Verification

- **Verification commands**
  - PING (Packet Internet Grouper)
  - Traceroute
- **Show commands**
  - Show ip interface brief
  - Show running-configuration
  - Show version
  - Show mac address-table

Configuration Example (Switch-to-Host)

Configuration on Sw1

- Switch> enable
- Switch# configure terminal
- Switch(config)# hostname Sw1
- Sw1(config)# interface GigabitEthernet1/0/5
- Sw1(config-if)# description **Connection to Bob Laptop**
- Sw1(config-if)# switchport mode access
- Sw1(config-if)# no shutdown

Configuration Example (Switch-to-Switch)

Configuration on Sw1

- Switch> enable
- Switch# configure terminal
- Switch(config)# hostname Sw1
- Sw1(config)# interface GigabitEthernet1/0/1
- Sw1(config-if)# description **Connection to Sw2**
- Sw1(config-if)# switchport mode dynamic desirable
- Sw1(config-if)# no shutdown

Configuration Example (Switch-to-Switch)

Configuration on Sw2

- Switch> enable
- Switch# configure terminal
- Switch(config)# hostname Sw2
- Sw2(config)# interface GigabitEthernet1/0/1
- Sw2(config-if)# description **Connection to Sw1**
- Sw2(config-if)# switchport mode dynamic desirable
- Sw2(config-if)# no shutdown

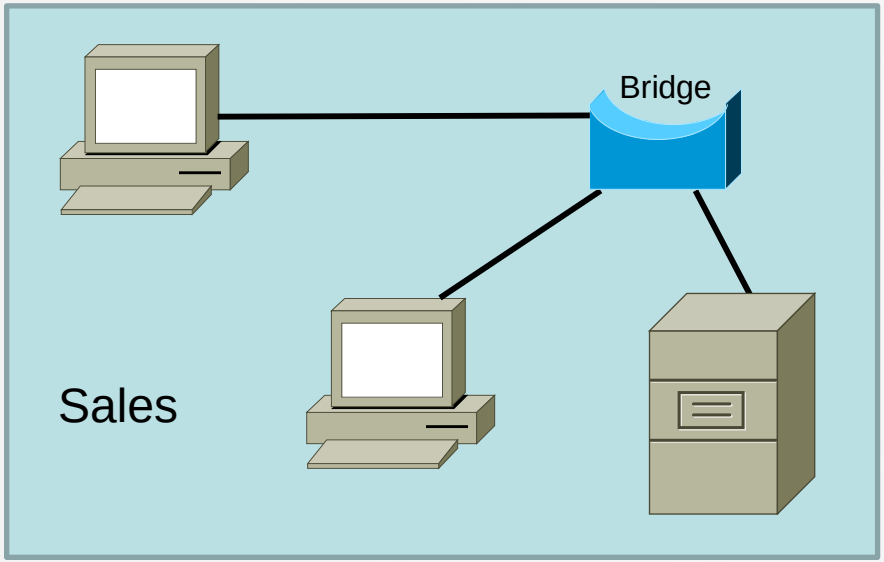
Basic Troubleshooting

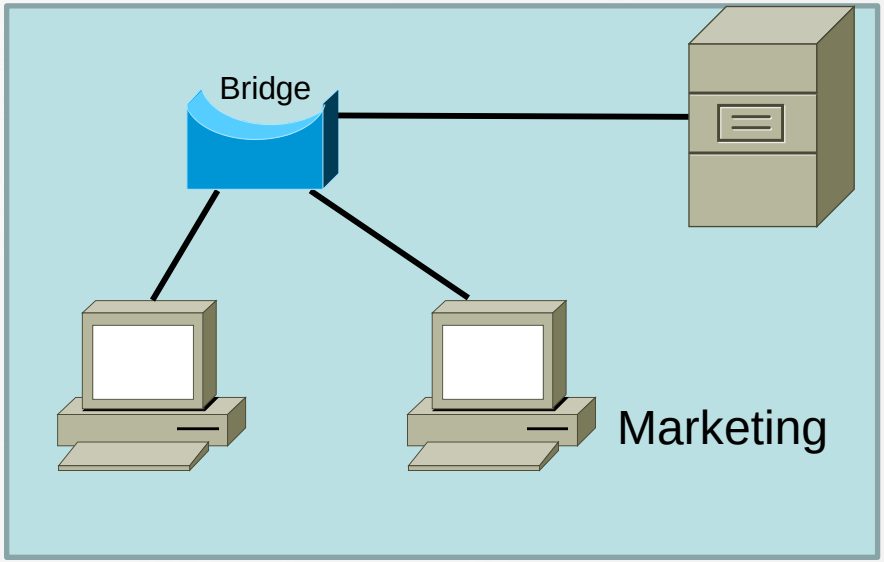
- Check for correct cable type
- Ensure `no shutdown` command in the interface (disabled by default)
- For interconnected Access Ports, check for same VLAN
- For interconnected Trunk, verify DTP compatibility modes

Virtual LAN (VLAN)
Part 1

Why VLANs?

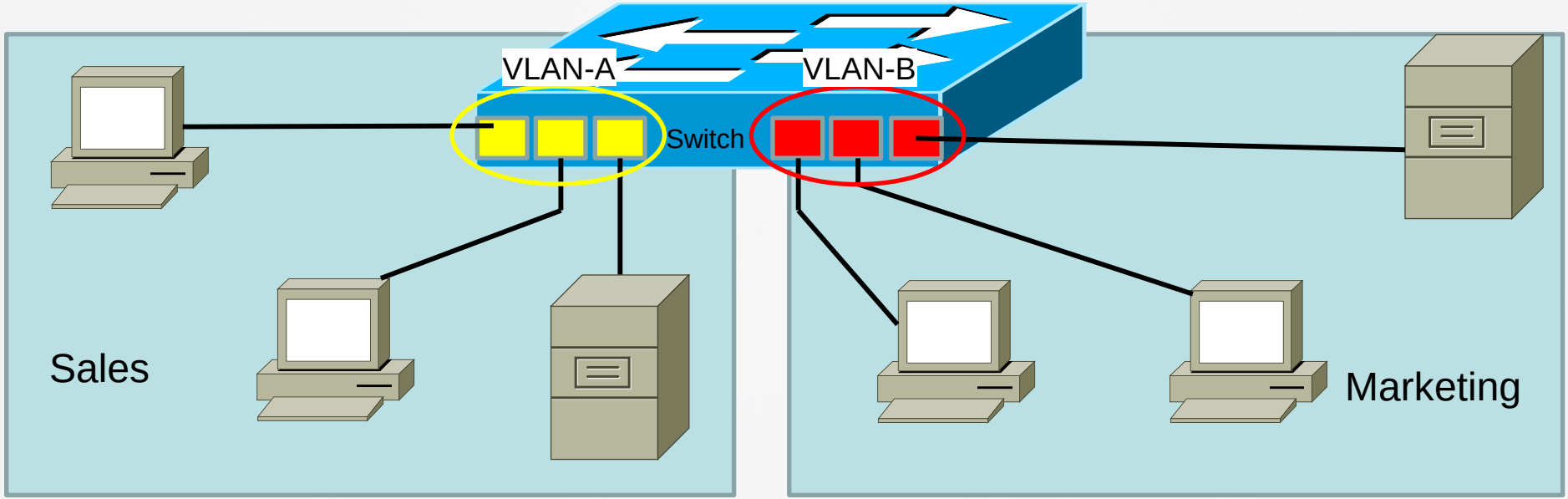
From this…





Why VLANs (2)?

To this…



Features

- Separates broadcast domain
- Provides better security
- Controls broadcast like ARP
- Provides hierarchical subnet usage

VLAN Ranges

- VLAN range is 1-4094
- 1-1001 are usable normal-range VLANs
- 1002-1005 are reserved for token ring
- 1006-4094 are extended-range VLANs

Configuring VLAN

- Legacy method with VLAN database
  - Sw1# vlan database
  - Sw1(vlan-database)# vlan <vlan-id>
  - Sw1(vlan-database)# end
- Modern method of configuring VLAN
  - Sw1(config)# vlan <vlan id>
  - Sw1(config-vlan)# name <vlan name>

Configuring Access Ports

- Access Port = Switchport configured for only a single broadcast domain (VLAN).

- Access port configuration
  - Switch(config)# interface <interface>
  - Switch(config-if)# switchport mode access
  - Switch(config-if)# switchport access vlan <vlan-id>

Verifying VLAN

- **Verification commands**
  - Sw1# show vlan <brief>
  - Sw1# show interface <type><number> switchport

Virtual LAN (VLAN)
Part 2

Port Types

- Trunk Port
  - Can have two or more VLANs configured
  - Can carry multiple VLAN information
  - By default, all the VLAN traffic is allowed from a trunk port

Trunking Encapsulation

- ISL (Inter-Switch Link)
  - Cisco proprietary
  - Traffic is encapsulated within 30-byte ISL frame
  - 26-byte header and 4-byte trailer
- 802.1Q
  - Open standard
  - All traffic except native VLAN is inserted with a 802.1q tag
  - Support concept of native VLAN

Native VLAN

- IEEE 802.1Q supported feature
- Frame without tag is considered native VLAN traffic
- Must match on both ends of the trunk
- By default, native VLAN is 1
- Can be changed using the **`switchport trunk native vlan <vlan-id>`** command

Configuring Trunking Encapsulation

- Static trunk configuration
  - Switch(config)# interface <interface>
  - Switch(config-if)#switchport trunk encapsulation dot1q
  - Switch(config-if)#switchport mode trunk
  - Switch(config-if)#end

Verifying Trunk

- Verifying VLAN and trunking
  - Switch# show vlan <brief>
  - Switch# show interface trunk
  - show interface status
  - show interface <interface> switchport

Any Questions?

Dynamic Trunking Protocol (DTP)

Dynamic Trunking Protocol

- Cisco proprietary feature that allows Cisco switches to negotiate trunk dynamically
- Three modes:
  - Auto
  - On
  - Desirable
- Desirable initiates the trunk, whereas Auto responds only

Implementing DTP

- Configuring DTP
  - Switch(config-if)# switchport mode dynamic [desirable|auto]
- Disabling DTP
  - Switch(config-if)# switchport nonegotiate

Verifying DTP

- Verification command
  - Switch# show interface trunk
  - Switch# show interface <interface> switchport

VLAN Trunking Protocol (VTP)

VLAN Trunking Protocol

- Used to advertise VLAN attributes
- Minimizes administrative overhead
- Uses revision number to determine recent update
- Higher revision number takes preference, default is 0
- Trunk should form and VTP domain should match on both ends

VTP Modes

- **Server**
  - Can add, remove, and modify VLAN information
  - VLAN information is stored in vlan.dat file located in the flash memory
  - VLAN 2-1001 are configurable
- **Client**
  - Cannot add, remove, or modify VLAN information
  - VLAN information is stored in vlan.dat file

VTP Modes

- Transparent
  - Can add, remove, and modify VLAN information
  - VLAN information is not stored, pass through only
  - Does support extended range VLANs
  - Changes on the server do not affect the VLAN database

Authentication

- VTP supports authentication
- All the switches should have the same domain name and VTP password
- MD5 hash is checked before accepting VLAN information
- Configured using the **`vtp password <password>`** command

Configuration & Verification

- **Configuring VTP**
  - Switch(config)# vtp mode server | client | transparent
  - Switch(config)# vtp domain <domain name>
  - Switch(config)# vtp password <password>
  - Switch(config)# vtp version 1 | 2 | 3
- **Verifying VTP**
  - Switch# show vtp status
  - Switch# show vtp password

VTP Pruning

- Reduces broadcast traffic
- Unnecessary VLANs are removed from the trunk
- Pruning eligible list is used to determine allowed VLANs
- Extended-range VLANs cannot be pruned
- Can be globally enabled using the **`vtp pruning`** command

Any Questions?

EtherChannel

Features

- Aggregates redundant links into a bundle
- Can provide aggregated bandwidth, avoiding congestion
- Can load balance using different algorithms
- Can bundle up to eight ports
- All the ports should have the same speed and duplex
- Provides loop-free Layer 2 network

PAgP

- Cisco proprietary
- Modes
  - On
    - No negotiation/forces the channel
  - Desirable
    - Sends PAgP initiation messages
  - Auto
    - Passively listens to the PAgP messages

LACP

- IEEE 802.3ad standard
- Modes
  - On
    - No negotiation/forces the channel
  - Active
    - Sends LACP initiation message
  - Passive
    - Passively listens to the LACP request

Configuring EtherChannel

- **Configuration commands**
  - Switch(config-if)# channel-group <group number> mode <mode>

Verifying EtherChannel

- **Verification commands**
  - Switch# show etherchannel summary

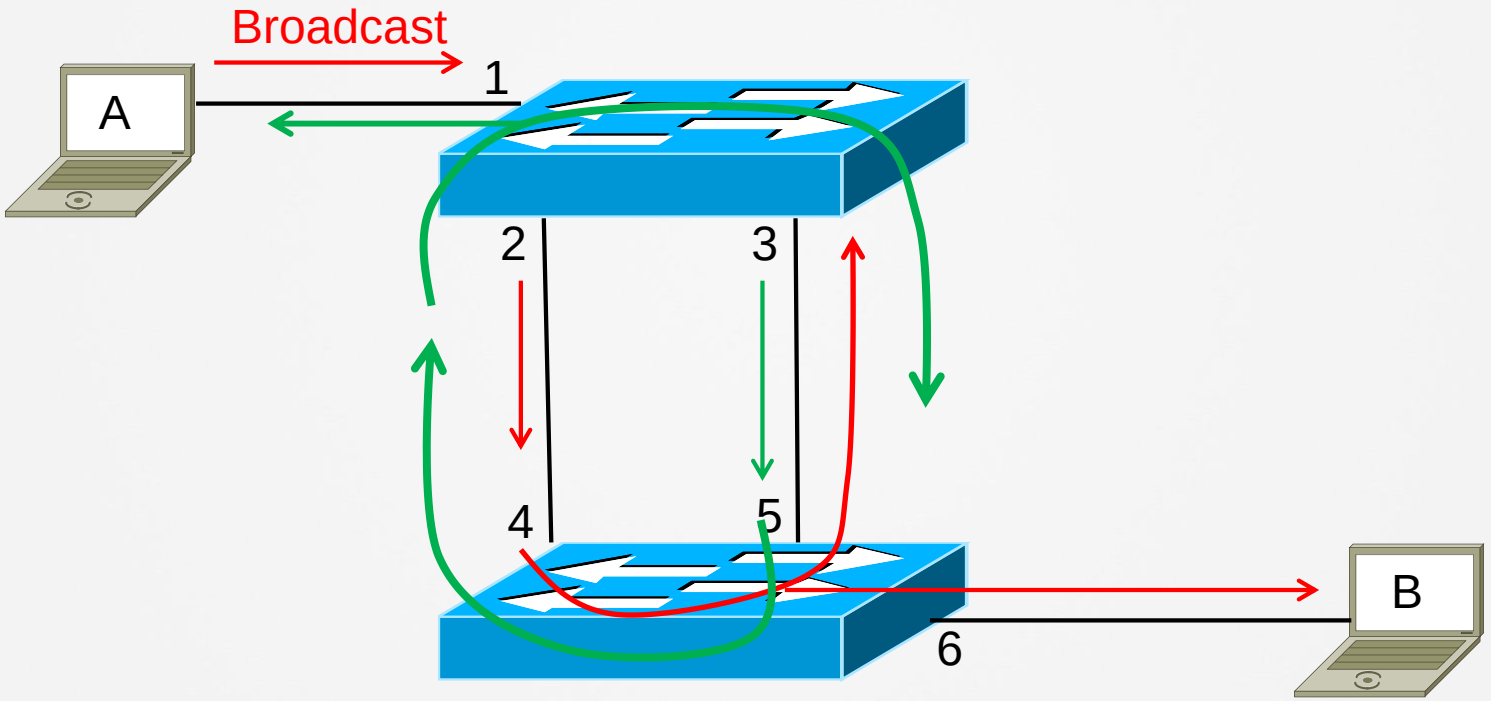
Any Questions?

Spanning-Tree Protocol (STP)

IEEE 802.1d

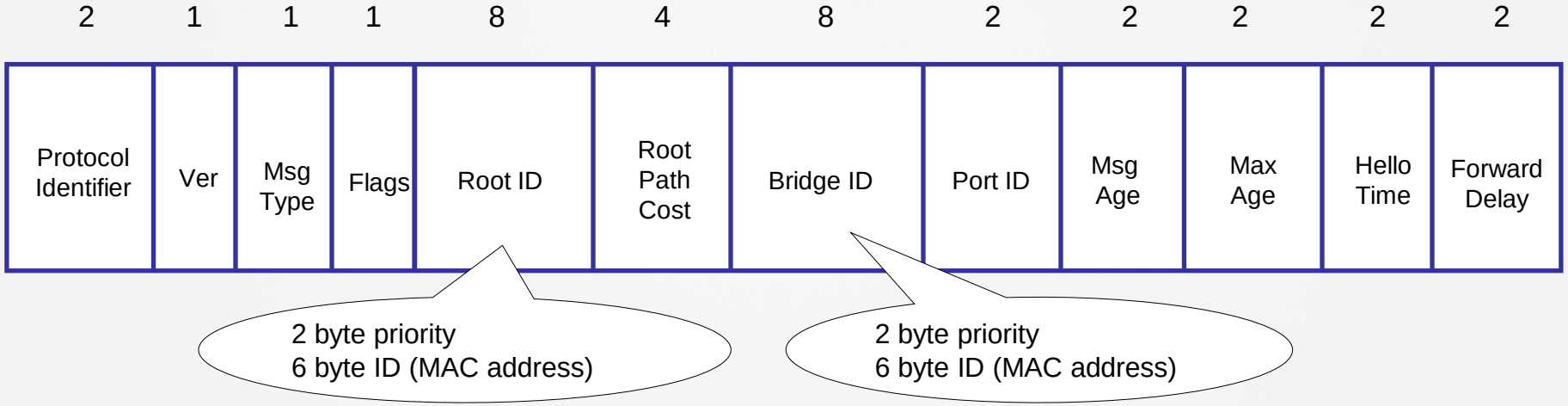
- Legacy protocol to prevent Layer 2 loop
- Usually called CST (Common Spanning Tree)
- No redundancy in traffic paths for frames
- Timers
  - Hello (2 seconds)
  - Max Age (20 seconds)
  - Forward Delay (30 seconds)

Bridging Loop



BPDUs

- BPDU = Bridge Protocol Data Unit
- Required to determine, and maintain, STP topology



STP Port Roles

- **Root port**
  - Port on a switch that is closest to the root bridge
- **Designated port**
  - Downstream port that is closest to the root
- **Blocking port**
  - Less-preferred port that is neither root nor designated
- **Forwarding port**
  - Port that is capable of forwarding data

STP Calculation

- Elects root bridge based on the lowest BID, where BID consists of priority and MAC
- Elects designated port, root port, and blocking ports based on STP cost or port priority
- Provides loop-free path and seamless convergence during failure
- Remember that with STP…LOWER is BETTER

STP Port States

STP Port States

- **Disabled**
  - Port that is in the down state, usually not part of STP topology
- **Blocking**
  - Port that is only allowed to receive the BPDU
  - Cannot send or receive data or add MAC addresses on its port

STP Port States

- **Listening**
  - Port that is allowed to send and receive BPDU
  - Can actively participate in the STP
  - Cannot send or receive data
- **Learning**
  - Allowed to send and receive BPDU
  - Can learn MAC addresses to add its address table
  - Cannot send or receive data

STP Port States

- **Forwarding**
  - Port that transitions to the forwarding state when the forwarding delay expires
  - Can send and receive data

STP Cost & Priority

- **Path cost**
  - Can be changed to influence the local switch to elect upstream root port
  - Affects all the downstream switches
- **Port priority**
  - Can be changed to influence downstream switch to elect root port
  - Locally significant

STP Cost & Priority

- **Bridge priority**
  - By default, STP bridge has priority of 32768
  - Can be configured in increments of 4096

Any Questions?

Implementing Spanning Tree

Per-VLAN Spanning Tree

- PVST = Cisco Default
- Number of STP instances depends on number of VLANs
- Effective where load sharing is required
- BPDU is sent for each VLAN
- Rapid convergence
- Both the encapsulations ISL and IEEE 802.1Q are supported
- Consumes resources because of control traffic overhead

Implementing PVST

- **Configuring priority per VLAN**
  - Switch(config)# spanning-tree vlan <vlan-id> priority <priority>
  - Switch(config)# spanning-tree vlan <vlan-id> root primary
  - Switch(config)# spanning-tree vlan <vlan-id> root secondary

Implementing PVST

- **Configuring port cost and port priority per VLAN**
  - Switch(config-if)# spanning-tree vlan <vlan-id> port-priority <priority>
  - Switch(config-if)# spanning-tree vlan <vlan-id> cost <cost>

Verifying PVST

- **Verification commands**
  - Switch# show spanning-tree
  - Switch# show spanning-tree vlan <vlan-id>
  - Switch# show spanning-tree root
  - Switch# show spanning-tree blocked ports

Any Questions?

Rapid Spanning Tree Protocol (RSTP)

RSTP (802.11w)

- Enhancement to legacy 802.1d STP
- Designed to speed up convergence
- Link type is derived from duplex mode
- Full duplex link is considered as a P2P
- Half duplex link is assumed to be shared

RSTP Port Roles

- **Root port**
  - Port that has best root path cost to the root
- **Designated port**
  - Downstream port that has best root path cost to the root
- **Alternate port**
  - Port that has alternate path to the root
  - Can only listen to the BPDUs
- **Backup port**
  - Considered as a backup designated port

Blocking

RSTP Port States

- **Discarding**
  - Combines the 802.1d disabled, blocking, and listening states
  - No MAC addresses are learned and incoming frames are dropped
- **Learning**
  - Cannot send or receive data
  - MAC addresses are learned

RSTP Port States

- Forwarding
  - Can send and receive data

Configuring & Verifying RSTP

- **Configuring Rapid Mode**
  - Sw1(config)# spanning-tree mode rapid-pvst
- **Verifying RSTP**
  - Sw1# show spanning-tree summary

BPDU Protection Mechanisms

Portfast

- Access Ports typically connect to hosts
  - Laptops/PCs
  - Servers
- End users don't want to wait up to a minute to gain network connectivity
- Portfast designed to speed up this process

Portfast Operation & Restrictions

- When enabled on a port, Portfast places port immediately into Forwarding state upon initial connection
- Not to be used on VLAN Trunk ports unless there is certainty about lack-of-loops

Portfast Configuration

- Configuration

```
(config-if)#spanning-tree portfast
   or…
(config)#spanning-tree portfast default
```

- Verification

```
Switch#show spanning-tree interface
<type/number> portfast
```

BPDU Guard

- Usually configured on access ports that lead to hosts
- If any BPDU is seen, port goes into err-disabled state
- Configuration
  - (config-if)#**spanning-tree bpduguard enable**
  - (config)#**spanning-tree portfast bpduguard default**

BPDU Filter

- Configured in access ports
- Does not send or receive BPDU
- Does not go into err-disabled when it receives unauthorized BPDU
- Configured with the **`spanning-tree bpdufilter enable`** interface-specific command

Verifying BPDU Guard/Filter

- Switch# show spanning-tree interface <interface> detail

```
Sw1#sho spanning-tree int fast 0/1 detail
 Port 3 (FastEthernet0/1) of VLAN0001 is designated forwarding
   Port path cost 19, Port priority 128, Port Identifier 128.3.
   <output omitted>
   The port is in the portfast mode
   Link type is point-to-point by default
   Bpdu guard is enabled
   Bpdu filter is enabled
   BPDU: sent 0, received 0
Sw1#
```

Any Questions?

Root Guard/Loop Guard

Root Guard

- Usually enabled on downstream interface
- Interface is placed into “root inconsistent” mode if superior BPDU is detected
- Configured with **`spanning-tree guard root`**

Loop Guard

- Used to prevent loops caused by unidirectional link
- Uses BPDU keepalive to detect unidirectional link
  - BPDUs should always be received on a Blocked port
  - If BPDUs don't arrive, normally convert to a Designated port
- Configured with **`spanning-tree guard loop`** interface-specific command

Port Security

Port Security

- Used to limit access to a port based on MAC address or quantity of connected devices
- Can be configured on static access and trunk ports (but not “dynamic” ports)
- A secure port cannot be:
  - Destination port for SPAN
  - Port-channel
  - Private VLAN port

Port Security Violation Modes

- **Shutdown**
  - Disables the port by placing it in err-disable state
  - Generates an SNMP trap and syslog message
- **Protect**
  - Does not accept traffic from new device after violation occurs
- **Restrict**
  - Works just like protect mode and generates SNMP and syslog

Implementing Port Security

- **Enabling port security**
  - Switch(config-if)# switchport port-security
- **Limiting number of MAC addresses**
  - Switch(config-if)# switchport port-security maximum <number>
  - Switch(config-if)# switchport port-security mac-address <MAC> <sticky>

Implementing Port Security Violation Mode

- **Setting violation mode**
  - Switch(config-if)# switchport port-security violation <protect | restrict | shutdown>

Implementing Port Security

- Configuring recovery interval
  - Switch(config)# errdisable recovery psecure-violation
  - Switch(config)# errdisable recovery interval <interval in sec>

Verifying Port Security

- Switch# Show port-security
- Switch# Show port-security interface <intf-type> <intf-number>

Any Questions?

Cisco Discovery Protocol (CDP)

Cisco Discovery Protocol (CDP)

- Cisco proprietary
- Layer 2 protocol for neighbor discovery
- Provides information of platform, interface, IP address, and OS version
- Equivalent to the open standard LLDP
- Helps with preparing network diagram

Configuration

Enabling CDP

- Router(config)# cdp run
- Router(config)# cdp timer <seconds>

Disabling CDP

- Router(config)# no cdp run
- Router(config-if)#no cdp enable

Verifying CDP

- **Verification commands**
  - Router# show cdp neighbor
  - Router# show cdp neighbor < interface>
  - Router# show cdp neighbor <interface> detail

Any Questions?

IPv4 Addressing

Classes of IPv4

- Classes:
  - Class A: 0.0.0.0 through 127.255.255.255
  - Class B: 128.0.0.0 through 191.255.255.255
  - Class C: 192.0.0.0 through 223.255.255.255
  - Class D: 224.0.0.0 through 239.255.255.255
  - Class E: 240.0.0.0 through 255.255.255.255
    - Note: 127 ranges are considered as loopbacks
    - Note: 169.254 ranges are considered as APIPA

Subnet Mask

- Helps identify network and host portion of network
- Default subnet masks:
  - Class A: 255.0.0.0 or /8
  - Class B: 255.255.0.0 or /16
  - Class C: 255.255.255.0 or /24
- Typically called classful address

IPv4 Addresses: Public & Private

- IP addresses “leased” to a corporation are known as ***public IP addresses***.
- IP addresses that are unregistered and may overlap from one company to the next, are known as ***private IP addresses***.

IPv4 Addresses: Private

- **Private IPv4 address:**
  - Defined in RFC 1918
  - For internal use only
- **Range of private address**
  - Class A : 10.0.0.0 through 10.255.255.255
  - Class B : 172.16.0.0 through 172.31.255.255
  - Class C : 192.168.0.0 through 192.168.255.255

IPv4 Addresses: Public

- **Public IPv4 addresses**
  - Globally unique
  - Should be purchased
  - Usually used in Internet edge
- **Range of public addresses**
  - Beyond the RFC 1918 space, all addresses are public

IP addressing – Router Configuration

- Router(config-if)# ip address <address><mask>
- Verification
  - Show interface <type><number>
  - Show ip interface brief

Any Questions?

IPv4 Subnetting

IPv4 Subnetting

- Each IP network that is purchased is only good for a single broadcast domain (VLAN).
- Often unused/unallocated host space within a given network.
- Subnetting = Dividing a single, allocated network into multiple sub-networks.
- Minor loss of available hosts addresses.

IPv4 Subnetting

- Every device running IPv4 uses the same process to determine its local network:
  - Identify local IP address in binary
  - Identify local subnet mask in binary
  - Extract network portion of address by comparing current address and subnet mask
- Subnet mask is referenced instead of classfull network address

IPv4 Subnetting

- **Example:**
  - Class C
    - Network : 192.168.10.0/24
    - Default subnet mask: 255.255.255.0
    - Host requirement: 10
  - Bits required
    - 11111111.11111111.11111111.1111**0000**
    - Only **4 bits** are required in the hosts field instead of default 8 bits

IPv4 Subnetting – How many hosts?

- New subnet mask
  - 255.255.255.240 [11110000=240 in the last octet]
- Number of usable hosts per subnet
  - 255.255.255.1111**0000** (bits in green are “host” bits)
  - 2^4-2 =14 hosts per subnet

    2^4=16

    All zeroes host = subnet address

    All one’s host = broadcast address

IPv4 Subnetting – How many subnets?

- New subnet mask
  - 255.255.255.240 [11110000=240 in the last octet]
- Range of available subnetworks
  - 255.255.255.11110000
  - Four subnet bits: $2^4 = 16$
    - Original network was 192.168.10.0 /24
    - Each subnet will be a multiple of 16.

      192.168.10.0 /28

      192.168.10.16 /28

      192.168.10.32 /28

VLSM & CIDR

- **Same Length Subnet Masking**
  - Previous example, each network utilized the same mask.
- **VLSM**
  - Variable length subnet masking
  - Provides ability to allocate IPv4 as per the host requirements
  - Subnet mask can be variable
    - Ex: /25 , /26, /27 from /24 block

VLSM & CIDR

- CIDR
  - Classless Interdomain Routing
  - Beyond the classful behavior
  - Class A address can be treated as Class B & C or vice versa
  - Ex: 10.0.0.0/24 [/24 is prefix-length from Class C]

Any Questions?

IPv4 Summarization

IPv4 Summarization

- Process of combining multiple subnetworks into a single network advertisement.
- Network ID and subnet mask are referenced
- Usually called *aggregation*
- Efficient in large networks, provides addressing hierarchy

IPv4 Summarization - Example

- Example
  - Network : 10.10.32.0 /20
  - Network: 10.10.48.0 /20
  - Subnet mask: 255.255.240.0
- Conversion of network-id into bits
  - 10.10.0010hhhh.hhhhhhhh /20
  - 10.10.0011hhhh.hhhhhhhh /20
  - AND operation result : 10.10.00hhhhhh.hhhhhhhh

"h" = Host Bit

10.10.0.0 /18 (summarized network)

Summarization and Supernetting

IPv4 Summarization

- Aggregating multiple subnets into a single network advertisement.
- That advertisement does not break classfull boundaries.

IPv4 Supernetting

- Aggregating multiple networks (could be subnets or classfull networks) into a single network advertisement.
- That advertisement breaks classfull boundaries.

Example: 10.0.0.0 /7 is a Supernet

IPv4 Supernetting

- Example
  - Network : 192.168.1.0/24
  - Network: 192.168.2.0/24
- Conversion of network-id into bits
  - 192.168.**00000001**.hhhhhhhh
  - 192.168.**00000010**.hhhhhhhh
  - AND operation result : 192.168.**000000**hh.hhhhhhhh

    192.168.0.0 /22 (Supernet)

IPv4 Summarization and Supernetting

- Some routers perform summarization by default.
- Supernetting can only be done manually.

- When performing summarization or supernetting ask yourself, "*what bits…from left-to-right…do all of these networks have in common?*"
  - Answer to the above question will determine new mask.

Any Questions?

Inter-VLAN Routing

Inter-VLAN Routing

- Two ways to configure inter-VLAN routing
  - Router-on-a-stick model
  - Routing with SVI
- A router is usually configured using sub-interface
- Single point of failure

Implementing Inter-VLAN Routing

- Configuring a trunk interface that is connected to the router
  - Switch(config-if)# switchport mode trunk
- Configuring sub-interface for respective VLANs
  - Router(config-sub-if)# encapsulation dot1q <vlan-id>
  - Router(config-sub-if)# ip address <address> <subnet mask>

Implementing Inter-VLAN Routing (SVIs)

- Multilayer Switches can route between VLANs
- Requires a separate SVI for each VLAN
  - Each SVI needs a physical port (Access or Trunk) in that VLAN.
- Hosts point to IP address on SVI as their default gateway.

Implementing Inter-VLAN Routing (SVIs)

- Configuration Example

```
Switch(config)# interface vlan 2
Switch(config-if)#ip add 2.2.2.2 255.0.0.0
Switch(config-if)#no shutdown

Switch(config)# interface vlan 3
Switch(config-if)#ip add 3.3.3.3 255.0.0.0
Switch(config-if)#no shutdown
```

Verifying Inter-VLAN Routing

- Verification commands
  - Switch# show mac address-table
  - Router# show ip route connected
  - Optionally, “ping” is the best way to test inter-VLAN routing

Any Questions?

IP Routing

What is “Routing”?

- Process of forwarding packets between networks.
- Basic components needed to route:
  - Routable Packet (IPv4, IPv6, etc)
  - Network address
  - Subnet mask
  - Next Hop
  - Metric

Types of Routes

- Connected
- Static
- Dynamic

General Rules of Routing

- Router will only use routes with reachable “next hops”
- Routers will only use the “best” routes
- Routes must be “believable” (how do I know this route is still good?)
- Router will only accept routes that match its own, active protocols
  - No IPv6 routes accepted if router not an IPv6 host

Routing Components

- Autonomous System (AS) number
  - 16-bit numbering system
  - Group of devices under a single technical administration
  - Usually an IGP is considered an AS
  - Ranges from 1 through 65535

Routing Components

- **Administrative Distance (AD)**
  - Defines trustworthiness of a routing protocol
  - 8-bit numbering system
  - Ranges from 0 through 255

Administrative Distance Values

Protocols	AD Value
Connected	0
Static	1
EIGRP	90
OSPF	110
IS-IS	115
RIP	120
iBGP/eBGP	200/20
Unreachable	255

Routing Metric

- Used for best path selection process
- IGPs use metric for shortest path calculation
- Lower value is preferred
- Depends on the routing protocol architecture
  - EIGRP metric = composite formula utilizing link bandwidth + delay
  - RIP metric = hop count
  - OSPF metric = link bandwidth

Routing Updates

- **Incremental update**
  - Only changes are sent in the routing update
- **Full update**
  - All of the routing table is sent in the update
- **Periodic update**
  - Sent in the specified time interval
- **Triggered update**
  - Sent whenever change is detected

Any Questions?

Dynamic Routing

Dynamic Routing

- Usually configured in large/ISP networks
- Can be categorized into two sections
  - IGP (Interior Gateway Protocol)
    - Protocol that works within the Autonomous System Number
  - EGP (Exterior Gateway Protocol)
    - Protocol that interconnects multiple Autonomous System Numbers
- Dynamic failover

Interior Gateway Protocol

- Typically works within the Autonomous System
- Can be categorized into three sections
  - Distance vector
    - Elects shortest path based on the total metric of a route
    - Visibility of network topology limited
    - Ex: IGRP, RIPv1/v2
  - Link state
    - Elects shortest path based on the link cost
    - Complete visibility to network topology
    - Ex: OSPF, IS-IS

Interior Gateway Protocol

- **Hybrid**
  - Combines features of distance vector and link state
  - Ex: EIGRP

Exterior Gateway Protocol

- Connects multiple Autonomous System Numbers
- Can carry more routing tables than IGPs
  - Example: BGP

Any Questions?

Static Routing

Static Routing

- Typically used in the small network
- Information on destination prefix and gateway are required
- Can increase administrative overhead
- No dynamic fail-over
- Preferred over dynamic routing protocols because of lower administrative distance

Implementing Static Routing

- **Configuration (next-hop)**
  - Router(config)#ip route <destination-network> <destination subnet mask> <next-hop>
- **Configuration (outgoing interface)**
  - Router(config)#ip route <destination-network> <destination subnet mask> <outgoing interface>

Verification

- **Verification commands**
  - Router# show ip route
  - Router# show ip route static
  - Router# show running-config | include ip route
  - Router# show ip protocol
  - Router# show ip route <prefix> <mask>

Default Routing

Default Routing

- Eliminates requirements of destination prefix and mask
- Usually configured for Internet-specific routing with ISP
- Can cause routing loop if not configured properly

Implementing Default Route

- **Configuration (next-hop)**
  - Router(config)#ip route 0.0.0.0 0.0.0.0 <next-hop>
- **Configuration (outgoing interface)**
  - Router(config)#ip route 0.0.0.0 0.0.0.0 <outgoing-interface>

Verification

- **Verification commands**
  - Router# show ip route
  - Router# show running-config | include ip route
  - Router# show ip route <prefix> <mask>

Troubleshooting

- Troubleshooting commands
  - Router# debug ip packet
  - Router# defaut ip routing

Floating Static Route

Floating Static Route

- Can be configured as a backup route
- Administrative distance can be increased to make a route backup
- Provide redundancy between two statically defined routes

Implementing Static Route Floating

Configuration

- Router(config)#ip route <destination-network> <destination subnet mask> <next-hop> <AD value>

Verification

- **Verification commands**
  - Router# show ip route
  - Router# show running-config | include ip route
  - Router# show ip route <prefix> <mask>

Troubleshoot

- Troubleshooting commands
  - Router# debug ip packet
  - Router# defaut ip routing

Enhanced Interior Gateway Routing Protocol (EIGRP)

Introduction to EIGRP

- Open standard
- Hybrid IGP
  - Characteristics of both Link State and Distance Vector
- Metric based from link bandwidth & delay
- Supports manual and automatic summarization
- Supports MD5 authentication
- Supports unequal cost load-balancing

EIGRP Packets

- Most packets sent to 224.0.0.10
- Neighbor relationships
  - Hello packets
- Routing Updates
  - Update (unicast initially, then multicast)
  - Acknowledgments (always unicast)
  - Query
  - Reply

EIGRP Metric Formula

- Metric = $256 * \left[\left[(K1 \times BW) + \frac{(K2 \times BW)}{(256 - Load)} + (K3 \times Delay) \right] \times \left[\frac{K5}{(Reliability + K4)} \right] \right]$

This part is not used in the default formula

- By Default: K1 = 1, K2 = 0, K3 = 1, K4 = K5 = 0
- Delay is sum of all the delays of the link along the paths
- Bandwidth is the lowest bandwidth of the link along the paths
- Default Metric is Bandwidth + Delay

Forming Neighbors

- Hello sent to 224.0.0.10
  - Required matching parameters
    - ✓Source IP Subnet
    - ✓K-Values
    - ✓Autonomous System Value
  - Hello and Hold time don't need to match
- Passive Interface and its effect on EIGRP

EIGRP: Categories of Routes

- **EIGRP Internal**
  - Route that was originated within Autonomous System with the “network” command.
  - Admin Distance = 90
- **EIGRP External**
  - Route that was previously learned via some non-EIGRP method and injected into EIGRP with “redistribute” command
  - Admin Distance = 170

DUAL Terminology

- **Successor**
  - Best route having lowest total metric (distance)
- **Feasible successor**
  - Backup routes with higher metrics

DUAL Terminology

- **Feasible distance**
  - Best (lowest) total distance between local router and destination prefix.
- **Reported distance**
  - Distance from neighbor to destination
- **Advertised Distance**
  - Distance as reported by upstream neighbor

EIGRP Tables

- Neighbor table
  - Neighbor information is recorded
- Topology table
  - Backup routes are recorded
- Routing table
  - Best routes are recorded

EIGRP Variance

- Variance allows unequal cost load-balancing

```
Router(config)# router eigrp 100
Router(config-rtr)#variance X
```

- The "X" above is simply a multiplier
  - Multiply FD of all routes in topology table by "X" = Result "YY" for each route.
  - Compare result "YY" against all Feasible Successors
  - If distance of any FS routes ≤ YY, install route in table

EIGRP Authentication

EIGRP Authentication

- Supports MD5 authentication
- Uses a combination of key-chain and key-string with authentication password
- More secure than plain-text authentication
- Can be used with multiple time-based key-chains

Configuring EIGRP Authentication

- **Global configuration commands**
  - Router(config)# key-chain <name>
  - Router(config-keychain)# key <key-id>
  - Router(config-keychain-key)# key-string <password>
  - Router(config-keychain-key)# send-lifetime <duration>
  - Router(config-keychain-key)# accept-lifetime <duration>
  - Router(config-keychain-key)# end

Configuring EIGRP Authentication

- **Interface configuration commands**
  - Router(config-if)# ip authentication eigrp <AS-Number> mode md5
  - Router(config-if)# ip authentication key-chain eigrp <AS-Number> <key-chain name>

Verifying EIGRP Authentication

- **Verification commands**
  - Router# debug eigrp packets

Implementing EIGRP

Basic EIGRP Configuration

- Configuration commands
  - Router(config)# router eigrp <AS-number>
  - Router(config-router)# no auto-summary
  - Router(config-router)# network <network-id>
  - Router(config-router)# end
- AS number should match between EIGRP routers

Manipulating EIGRP Routes

- Enable Unequal Cost Load-Balancing
  - Router(config)# router eigrp <AS-number>
  - Router(config-router)# variance <multiplier>

- Make routes more, or less, preferable to EIGRP
  - Router(config-if)# bandwidth <1-10000000>
    Bandwidth in kilobits
  - Router(config-if)# delay <1-16777215>
    Throughput delay (tens of microseconds)

Verification

- Verification commands
  - Router# show ip eigrp neighbor
  - Router# show ip eigrp topology
  - Router# show ip route eigrp
- Above commands display neighbor table, topology table, and routing table, respectively

Troubleshooting EIGRP

- Troubleshooting commands
  - Router# debug ip eigrp
  - Router# debug eigrp packet
  - Router# defaut ip routing
  - Router# show ip eigrp traffic

Open Shortest Path First (OSPF)

Introduction to OSPF

- Open standard
- SPF (shortest path first) algorithm
- Hello used for neighbor relationship
  - Hello timer = 10 seconds
  - Hold timer = 40 seconds
- Works based on area hierarchy, minimizes LSA flooding
- Supports clear-text and MD5 authentication

OSPF Packet Types

- Multiple OSPF Packet Types
  - Hello
  - Database Descriptor
  - Link State Request
  - Link State Update
  - Link State Acknowledgment
- Most packets sent as multicasts

OSPF Network Types (1)

- OSPF classifies links (network type) based on Layer-2 encapsulation.
- Network Type determines things such as:
  - Can neighbors be discovered dynamically?
  - What subnet mask used to identify link?
  - Is there a need for DR/BDR election?
- Type can be manually changed on some interfaces.

OSPF Network Types (2)

- Broadcast
  - Ethernet, Token Ring, FDDI
- Point-to-Point
  - Loopback interfaces, PPP, HDLC
- Non Broadcast Multi-access
  - Frame-Relay
- Point-to-Multipoint

OSPF Neighborship

- Hello packet contains some important information, which should match for a proper neighborship
  - Hello and dead interval
  - Authentication
  - Area – ID
  - Prefix-length
  - Stub area flag

Building Neighbor Relationships

- Exchange of Hello Packets
  - May need to manually specify neighbor address depending on network type.
- Exchange of Database Descriptors
- Exchange of Link State Requests and Updates
- Fully Loaded

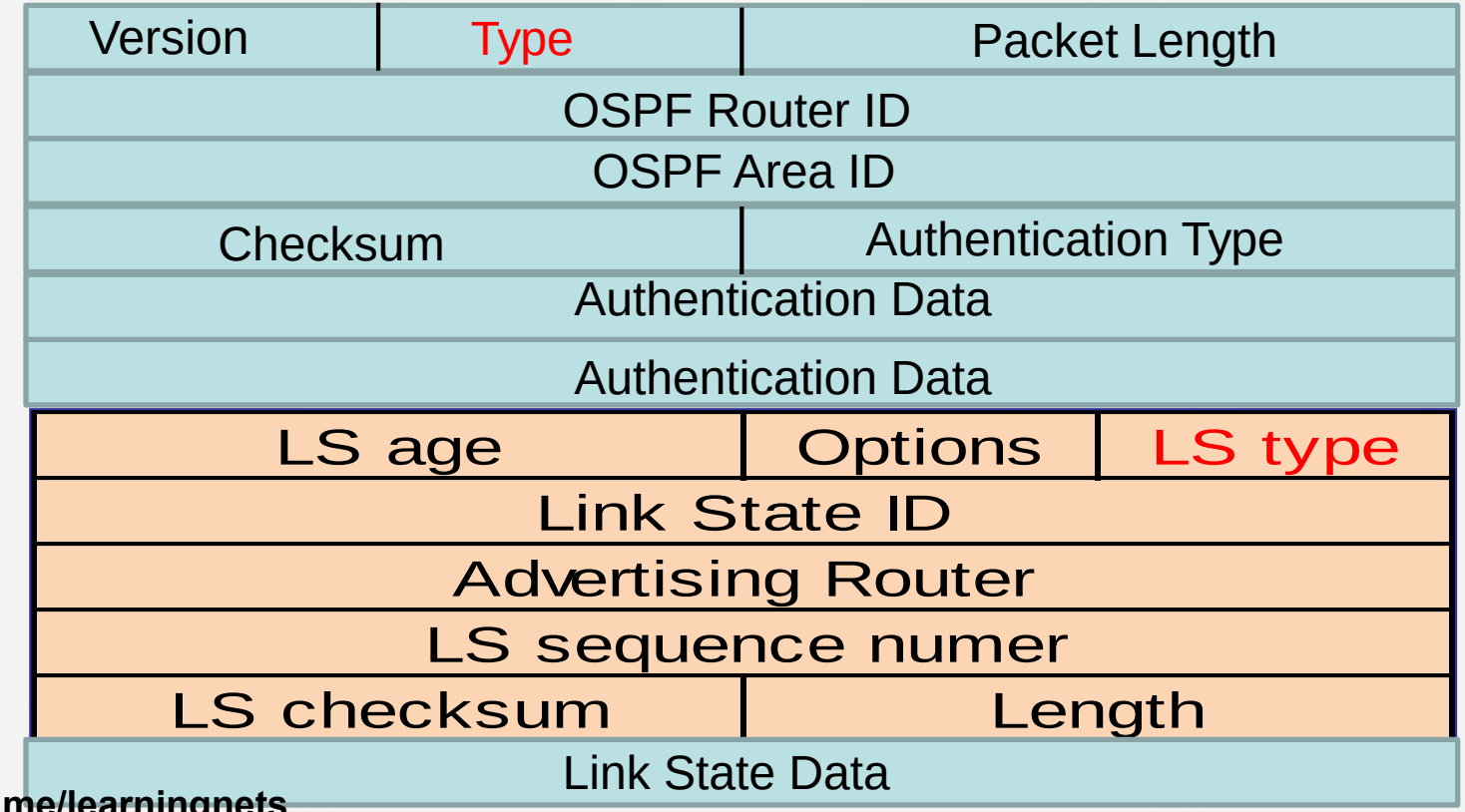
OSPF Router Roles

- Synchronizes LSDB every 30 minutes
- Area 0 is called a backbone; other areas are called non-backbone areas or regular areas
- Any router that connects multiple areas is called an ABR (area border router)
- Any router that connects multiple AS's is called an ASBR (autonomous system border router)

OSPF LSAs

- LSA = Link State Advertisement
- Carried within an OSPF Link State Update Packet
- Different types carry different data
- Age out after 1-hour…refreshed every 30-minutes.

OSPF Packet and LSA Header



Router LSA

- LSA Type-1 (Router LSA)
- Describes the state of connected links
- Bits to indicate special capabilities of router.
  - ABR
  - ASBR
- Confined to local area only

Network LSA

- Type-2 (Network LSA)
- Only created by Designated Routers
- Describes:
  - All adjacent neighbors of DR
  - Subnet mask of link
- Confined to local area only

Summary LSA

- Type-3 (Summary LSA)
- Describes summarized info of links from one area into an adjacent area
- Created by ABRs
- Confined to local area only, but other ABRs may modify and continue to forward.

ASBR-Summary LSA

- Type-4 (ASBR-Summary LSA)
- Advertises the ASBR into remote areas
- Created by ABRs
- Allows routers that are not in same area as ASBR to forward traffic to it.

External LSAs

- Type-5 (External LSA)
- Advertises non-OSPF routes into OSPF
- Created by ASBRs
- Propagated throughout entire OSPF domain.

OSPF: Other Info

- Supports VLSM and CIDR
- Manual summarization on the boundary/border routers such as ABR and ASBR
- Routing update is sent using multicast address
  - Multicast address : 224.0.0.5 (or 224.0.0.6 when DR/BDR present)
- Supports special area types such as stub, totally stub, and NSSA

OSPF Tables

- Neighbor table
  - Contains neighbor information
- Database description
  - Contains database information
- Routing table
  - Contains best route on the basis of link cost

OSPF Router-id & Priority

Router-Identifier

- OSPF elects a router-id when the process comes up
- Router-id is elected on the basis of the given hierarchy.
  - Router-id command under the OSPF instance
  - Highest IP address of a loopback interface
  - Highest IP address of a physically up/up interface
  - Can be configured using the `router-id <router-id>` command under the routing instance

OSPF Priority

- All OSPF routers have default priority of 1
- 8-bit value, 255 being more preferred and 0 being less preferred
- Router with priority 0 cannot participate in DR/BDR election
- Can be configured with the `ip ospf priority < priority>` interface-specific command

Configuration & Verification

- **Configuring OSPF priority**
  - Router(config-if)# ip ospf priority <priority>
- **Verifying OSPF priority**
  - Router# show ip ospf neighbor
  - Router# show ip ospf interface

Configuration & Verification

- Configuring OSPF router-id
  - Router(config-router)# router-id <router-id>
- Verifying OSPF router-id
  - Router# show ip ospf interface
- Both the router-id and priority need OSPF instance reset to take effect after they are changed

OSPF DR & BDR Election

DR & BDR Election

- OSPF elects a DR (designated router) and a BDR (backup designated router) in broadcast and non-broadcast multi-access networks
- DR is responsible for sending an update to the neighbors that are received from other neighbors
- Special multicast address used for sending routing updates to DR/BDR: 224.0.0.6

DR & BDR Election

- OSPF router with the highest priority becomes the DR
- OSPF router with lower priority than DR becomes BDR
- A BDR takes DR's if DR fails
- Other OSPF routers are known as DROTHERS

DR & BDR Election

- If OSPF priority has not been configured, highest router-id is referenced
- OSPF router with highest router-id becomes DR
- OSPF router with lower router-id than DR's becomes the BDR

Configuration & Verification

- **Configuring priority**
  - Router(config-interface)# ip ospf priority <priority>
- **Verifying DR and BDR**
  - Router# show ip ospf neighbor
  - Router# show ip ospf interface

OSPF Authentication

OSPF Authentication

- Two kinds of authentication supported
  - Simple password
  - Message digest 5
- Can be configured under the OSPF instance or in the interface
- MD5 is preferred method of authentication because the simple-password authentication is less secure
- Password should match for neighbor establishment

Configuring OSPF Authentication

- Simple password authentication
  - Router(config-if)# ip ospf authentication
  - Router(config-if)# ip ospf authentication key <key-id> <key>
- MD5 authentication
  - Router(config-if)# ip ospf authentication message-digest
  - Router(config-if)# ip ospf message-digest-key <key-id> md5 <key>

Verifying OSPF Authentication

- **Troubleshooting commands**
  - Router# debug ip ospf adjacency
  - Router# show ip ospf interface

Implementing OSPF

Configuration

Configuration commands

- Router(config)# router ospf <process-id>
- Router(config-router)# network <network-id> <WC mask> area < area-id>
- Router(config-router)# end

Verification

- **Verification commands**
  - Router# show ip ospf neighbor
  - Router# show ip ospf database
  - Router# show ip route ospf

Troubleshooting OSPF

- Troubleshooting commands
  - Router# debug ip ospf adjacency
  - Router# debug ip packets
- Area-id, prefix mask, and authentication should match each other

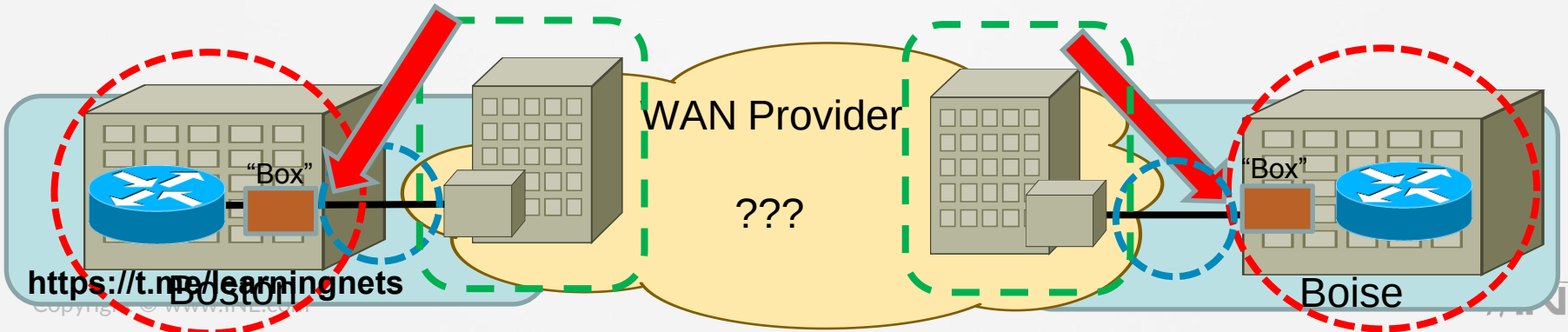
Introduction to WANs

What is a WAN?

- WAN = Wide Area Network
- Covers more geographical distance than LAN
- Lease the services of another network (WAN Service Provider)
- Many different WAN topologies
  - Point-to-Point
  - Hub-and-Spoke
  - Full Mesh

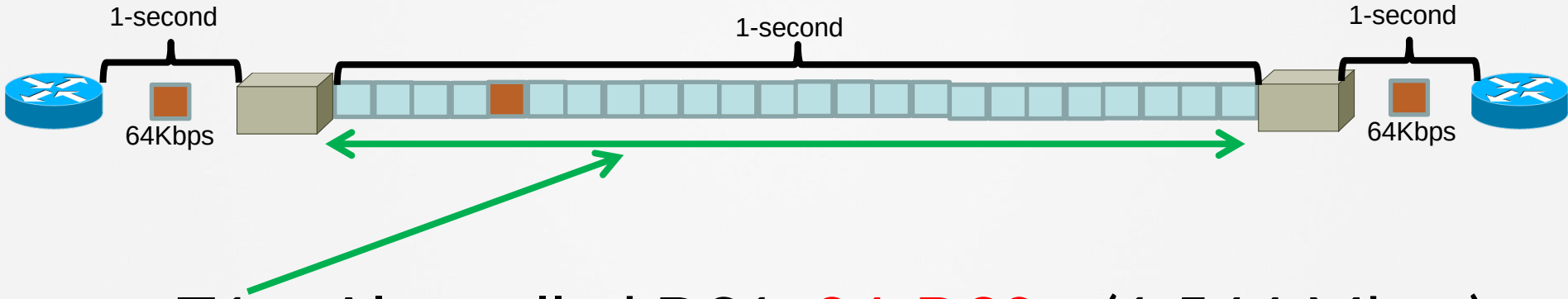
WAN Terminology (1)

- CPE
- Demarcation Point
- Local Loop
- Central Office / POP



DS0s, DS3s, and OCs (1)

- DSO (Digital Signal 0)



- T1 – Also called DS1 24-DS0s (1.544 Mbps)

DS0s, DS3s, and OCs (2)

- E1 – European 30-DS0s (2.048 Mbps)
- T3 – Also called DS3
  - 28 DS1s (T1s) bundled together
  - 44.736Mbps
- OC-3 – Optical Carrier
  - 3 bundled DS3s (T3s)
  - 155.52Mbps

Leased Lines

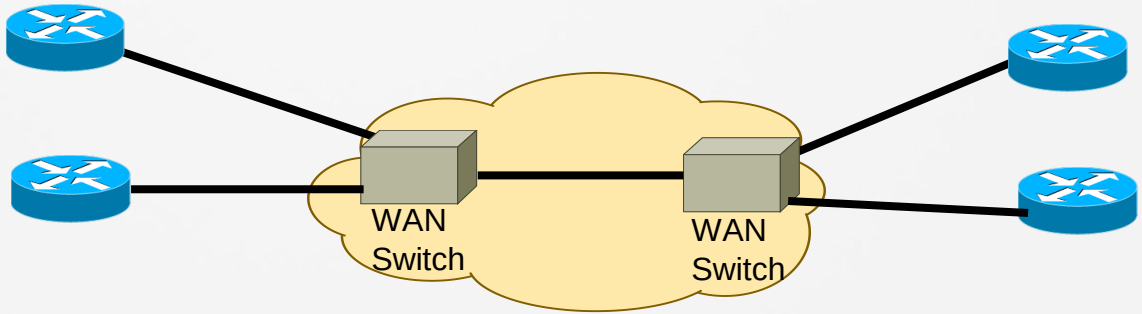
- **Dedicated Leased Lines**
  - Point-to-Point
  - Always up
  - Up to 45Mbps
  - Uses PPP or HDLC

Circuit Switching

- Like a Phone Call
- POTS (Plain old telephone service)
- ISDN (Integrated Services Digital Network)

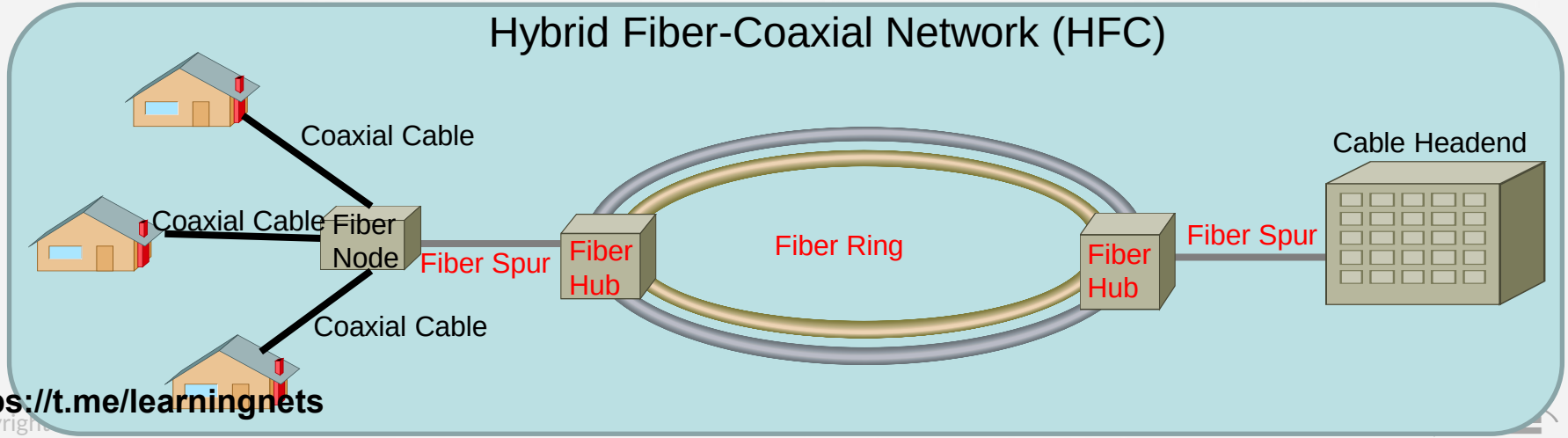
Packet Switched

- Utilizes a line that is “always up” like dedicated Leased Line.
- Bandwidth is shared with other customers
- Payment can be based on three factors:
  - Access Rate
  - CIR (Committed Information Rate)
  - Burst



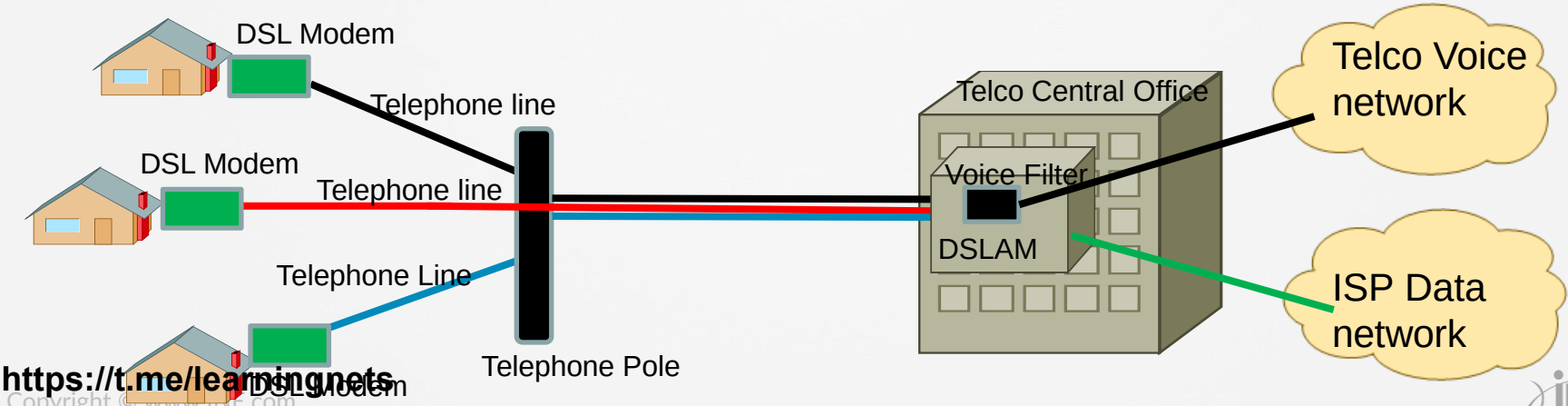
Types of Packet Switched Networks (Cable)

- Cable
  - Uses Co-Axial cable from subscriber (home) to Fiber Node
  - Fiber Optic used from that point on
  - Cable subscribers share bandwidth up to Fiber Node



Types of Packet Switched Networks (DSL)

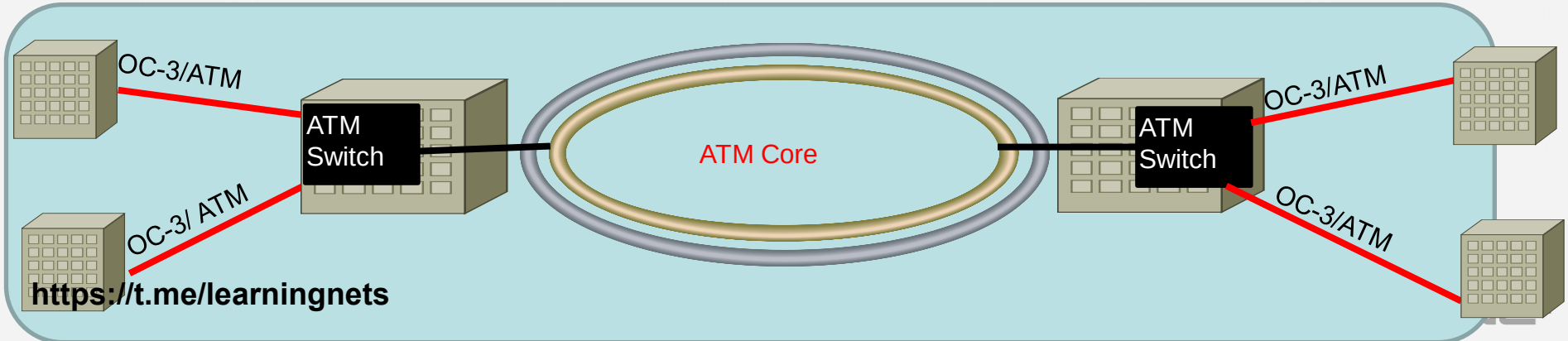
- DSL
  - Digital Subscriber Line
  - Uses same pair of telephone wires to deliver voice and data
  - Slower than cable,
  - Uses different frequencies to separate voice & data.



Types of Packet Switched Networks (ATM)

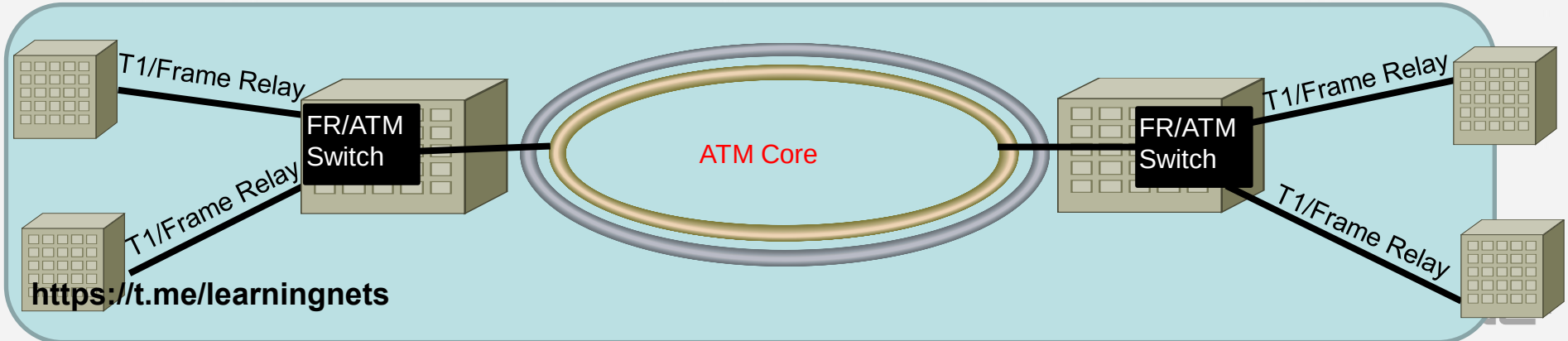
Asynchronous Transfer Mode

- Fiber used in ATM core network
- Last mile could be copper or fiber
- ATM cells always same, fixed size (53-Bytes)
- Primarily designed for latency sensitive applications



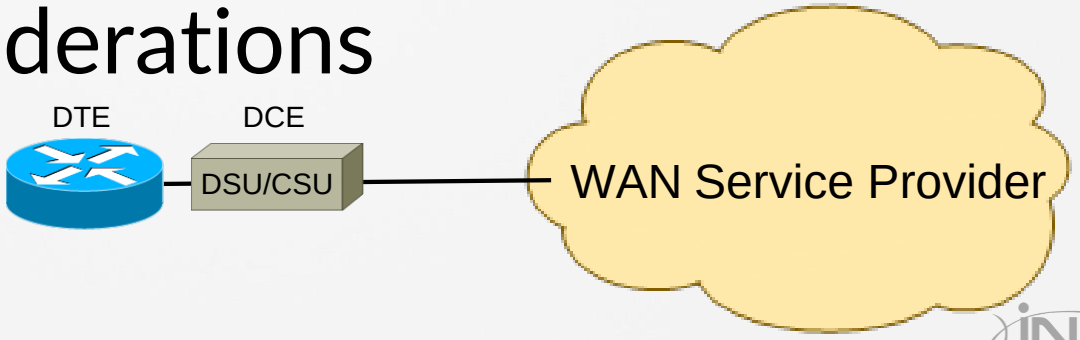
Types of Packet Switched Networks (Frame-Relay)

- **Frame Relay (legacy technology)**
  - Layer-2 Specification
  - Uses same pair of telephone wires for last mile.
  - Data placed into Frame-Relay Headers
  - Utilizes PVCs (Permanent Virtual Circuits)
  - ATM typically used in the Core



WAN Characteristics (1)

- Typically utilize different Layer-2 encapsulations than LAN
  - Frame Relay
  - ATM
  - PPP
- Timing Considerations
  - CSU/DSU
  - DCE and DTE

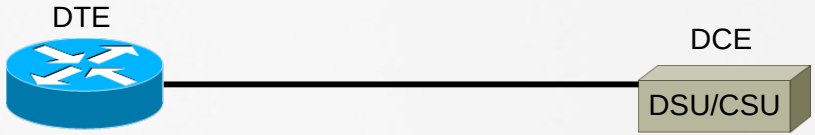


WAN Characteristics (2)

- **Choices to be made when selecting a WAN service.**
  - How many sites do I need to reach?
  - Does it need to be “always on”?
  - How much bandwidth do I need?
  - Do I want to pay for a fixed amount of bandwidth…or pay-as-I-go?
  - Is Layer-2 authentication important to me?

WAN Cabling

- Connection from Router to WAN Equipment
  - Serial Cables
  - Fiber Optic
  - Twisted Pair



- Multiple Connector Types
  - EIA/TIA-232
  - X.21
  - V.35
  - HSSI

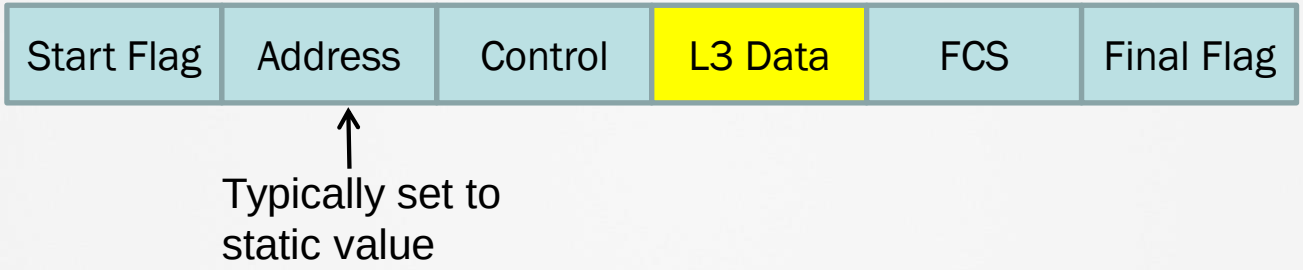
HDLC

HDLC

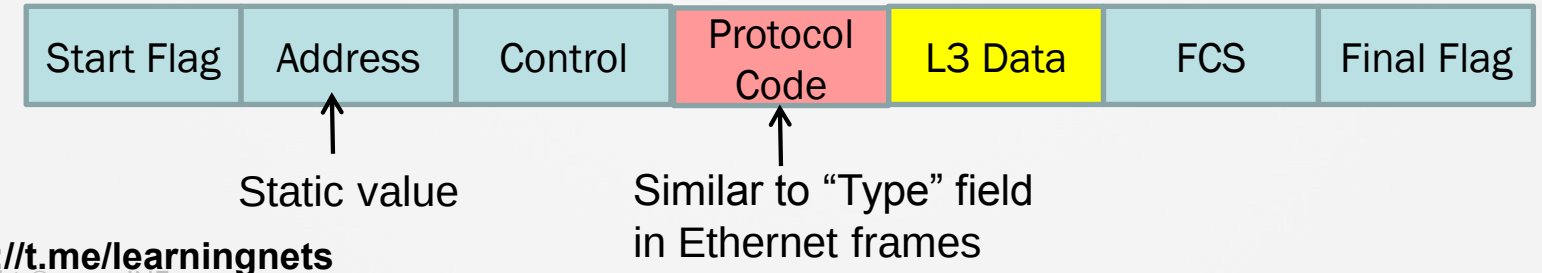
- High-Level Data Link Control
- Doesn't work in cross-vendor environment
- No authentication support
- Default encapsulation type for Cisco routers

HDLC Frame Formats

ISO HDLC Frame Format

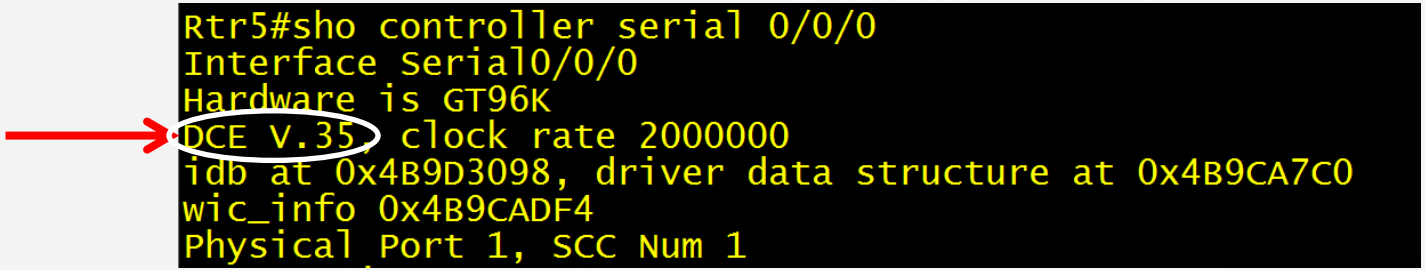


Cisco HDLC Frame Format



Configuring HDLC

- Change encapsulation
  - Router(config-if)# encapsulation hdlc
- Determine which side is DCE
  - Show controller serial x/y



- Set clock rate for DCE (if back-to-back)
  - Router(config-if)# clock rate <rate>

Verifying HDLC

- **Verification command**
  - Router# show interface <interface>

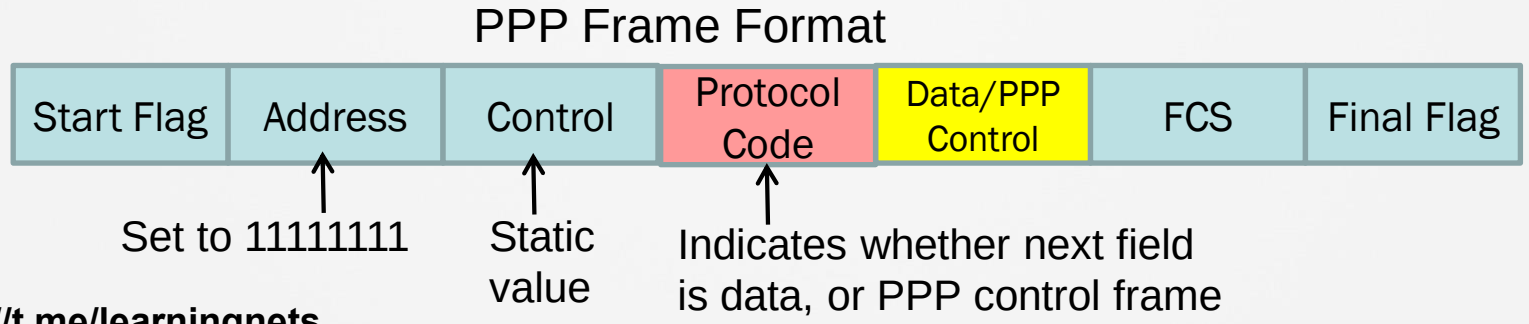
Point-to-Point Protocol

PPP

- Point-to-Point Protocol
- Open standard
- Operates in the LLC sub-layer of data link layer in OSI
- Originally designed for dial-up connections (modems, ISDN, etc.)
- Only one possible destination

Point-to-Point Technologies

- No Layer 3 to Layer 2 resolution required
- Useful for wide area network, where leased lines exist or other P2P networks
- Supports authentication



LCP and NCP

- PPP must negotiate a connection
- Moves through a series of required steps prior to transport of user data
  - LCP – Link Control Protocol
  - Authentication (optional)
  - NCP – Network Control Protocol
- State events and transitions can be monitored in real-time with “debug ppp negotiations”.

PPP Authentication
Part 1

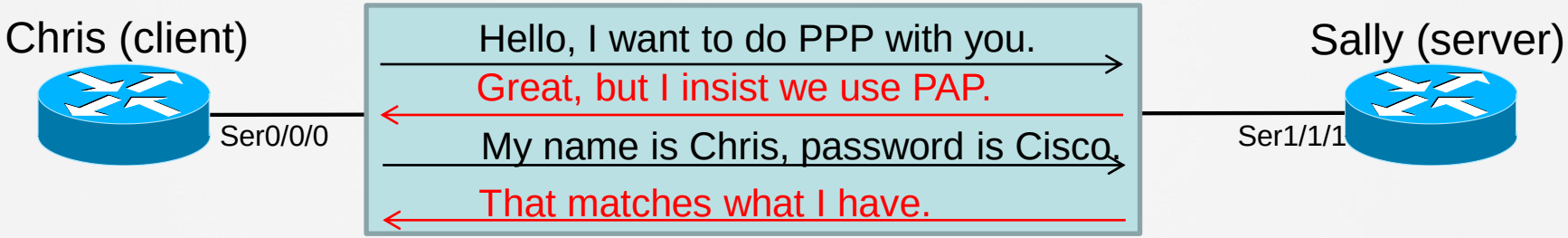
PAP

- Password Authentication Protocol
- Sends clear text username and password for authentication
- Two-way handshake
- Less secure than CHAP
- By default, hostname is sent as the username

PAP Authentication – One-way

» PPP PAP authentication options

- One way (client authenticates against server)




```
Hostname Chris
!
interface serial 0/0/0
        ip address 1.1.1.1 255.255.0.0
        encapsulation ppp
        ppp pap sent-username Chris password Cisco
```

```
Hostname Sally
Username Chris password Cisco
!
interface serial 1/1/1
        ip address 1.1.1.2 255.255.0.0
        encapsulation ppp
        ppp authentication pap
```

PAP Authentication – Two-way

- Two way (both peers authenticate each other)


```
Hostname Chris
Username Sally password Server
!
interface serial 0/0/0
        ip address 1.1.1.1 255.255.0.0
        encapsulation ppp
        ppp authentication pap
        ppp pap sent-username Chris password Cisco
```

```
Hostname Sally
Username Chris password Cisco
!
interface serial 1/1/1
        ip address 1.1.1.2 255.255.0.0
        encapsulation ppp
        ppp authentication pap
        ppp pap sent-username Sally password Server
```

Verifying PAP Authentication

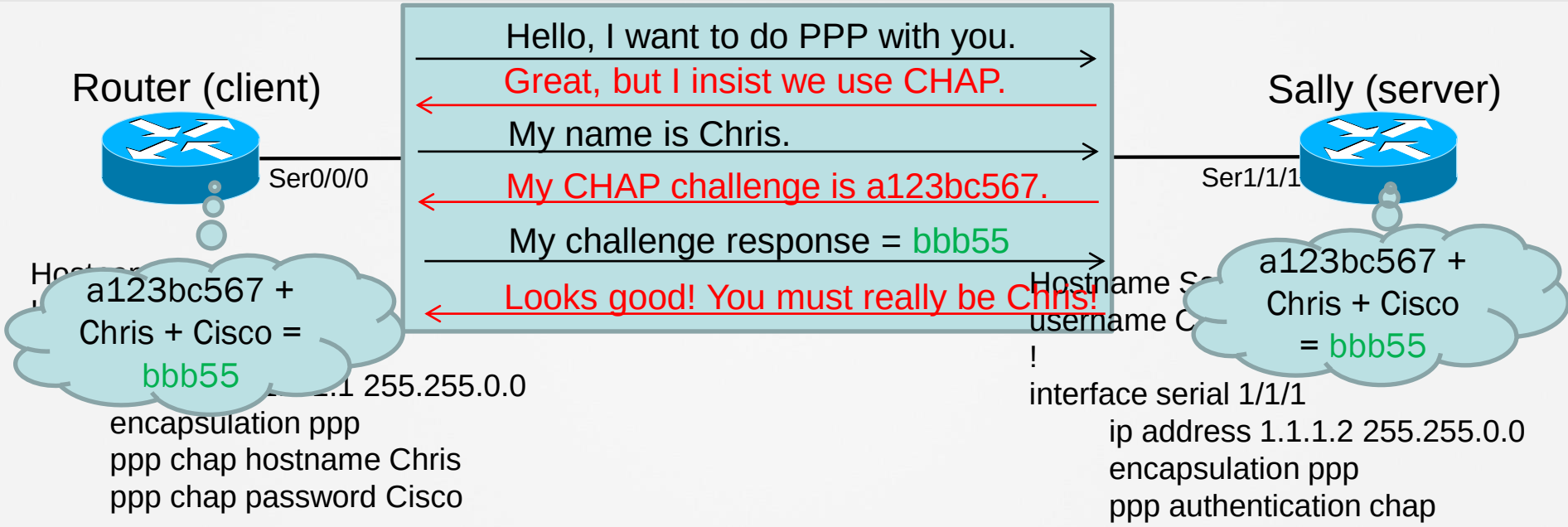
- Verification command
  - Router# debug ppp negotiations
  - Router# show interface serial <number>
  - Router# show users
- Note: Upon successful authentication, a PAP server should show the users with IP addresses who are authenticated

PPP Authentication
Part 2

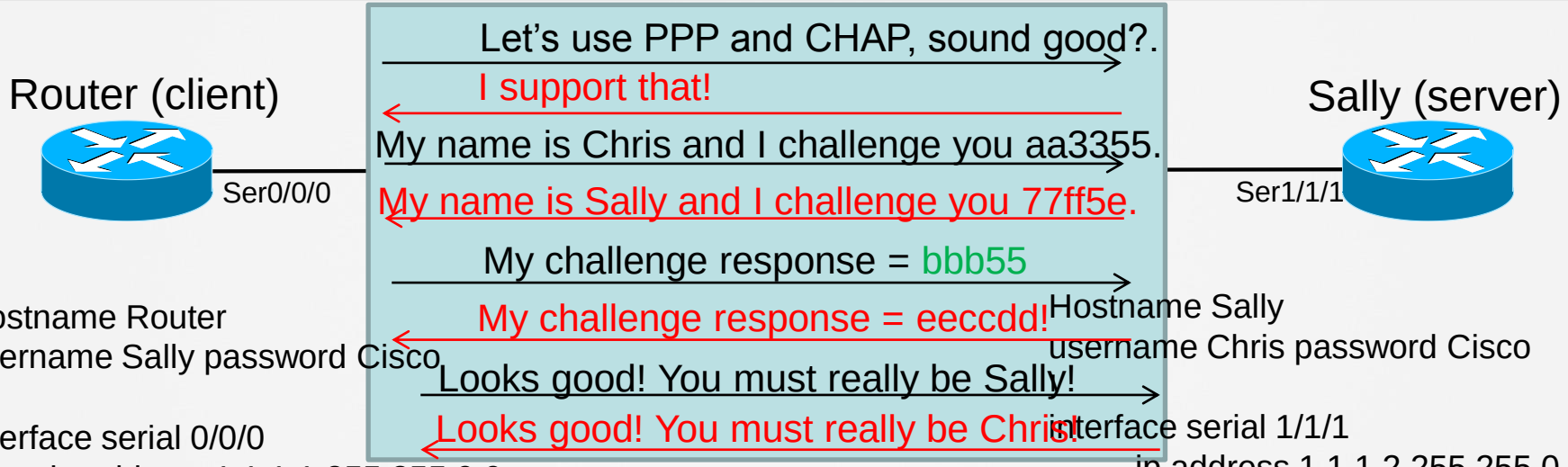
CHAP

- Challenge Handshake Authentication Protocol
- Three-way handshake
- More secure than PAP
- By default, hostname is sent as the username; username can be explicitly configured

CHAP Authentication – One-way



CHAP Authentication – Two-way




```
Hostname Router
username Sally password Cisco
!
interface serial 0/0/0
        ip address 1.1.1.1 255.255.0.0
        encapsulation ppp
        ppp authentication chap
        ppp chap hostname Chris
        ppp chap password Cisco
```

```
Hostname Sally
username Chris password Cisco

interface serial 1/1/1
ip address 1.1.1.2 255.255.0.0
encapsulation ppp
ppp authentication chap
ppp chap hostname Sally
ppp chap password Cisco
```

https://t.me/learningnets

Configuring CHAP Authentication (Server)

- Change encapsulation
  - Router(config-if)# encapsulation ppp
- Create local user database
  - Router(config)# username <username> password < password>
- Configure CHAP server
  - Router(config-if)# ppp authentication chap

Configuring CHAP Authentication (Client)

- Change encapsulation
  - Router(config-if)# encapsulation ppp
- Configure to send username and password
  - Router(config-if)# ppp chap password <password>
  - Router(config-if)# ppp chap hostname <username>

Verifying CHAP Authentication

- Verification command
  - Router# show users
  - Router# debug ppp negotiations

- Note: Upon successful authentication, a CHAP server should show the users with IP addresses who are authenticated

Frame Relay

Introduction to Frame Relay

- Multipoint technology (NBMA)
- Requires Layer 3 to Layer 2 resolution
- Addresses limitations of point-to-point technologies
- Legacy technology typically used in service provider end

DLCI

- Data-Link Connection Identifier
- Works as a Layer 2 address in Frame Relay
- Identifies Layer-2 path for data transmission
- 10-bit value that ranges from 0 through 1023, where 0-15 and 1007-1023 are reserved

LMI

- Local Management Interface
- Works as a keepalive between Frame Relay switch and end device
- Propagates DLCI or circuit information to the hub and spoke
- Standards of LMI: Cisco, Q933a, ANSI

Inverse ARP

- Dynamically maps the destination IP with corresponding local DLCI
- Only one DLCI can be mapped with a Layer 3 address
  - Each DLCI is a point-to-point connection to a remote router.
- Does not function on sub-interfaces

CIR

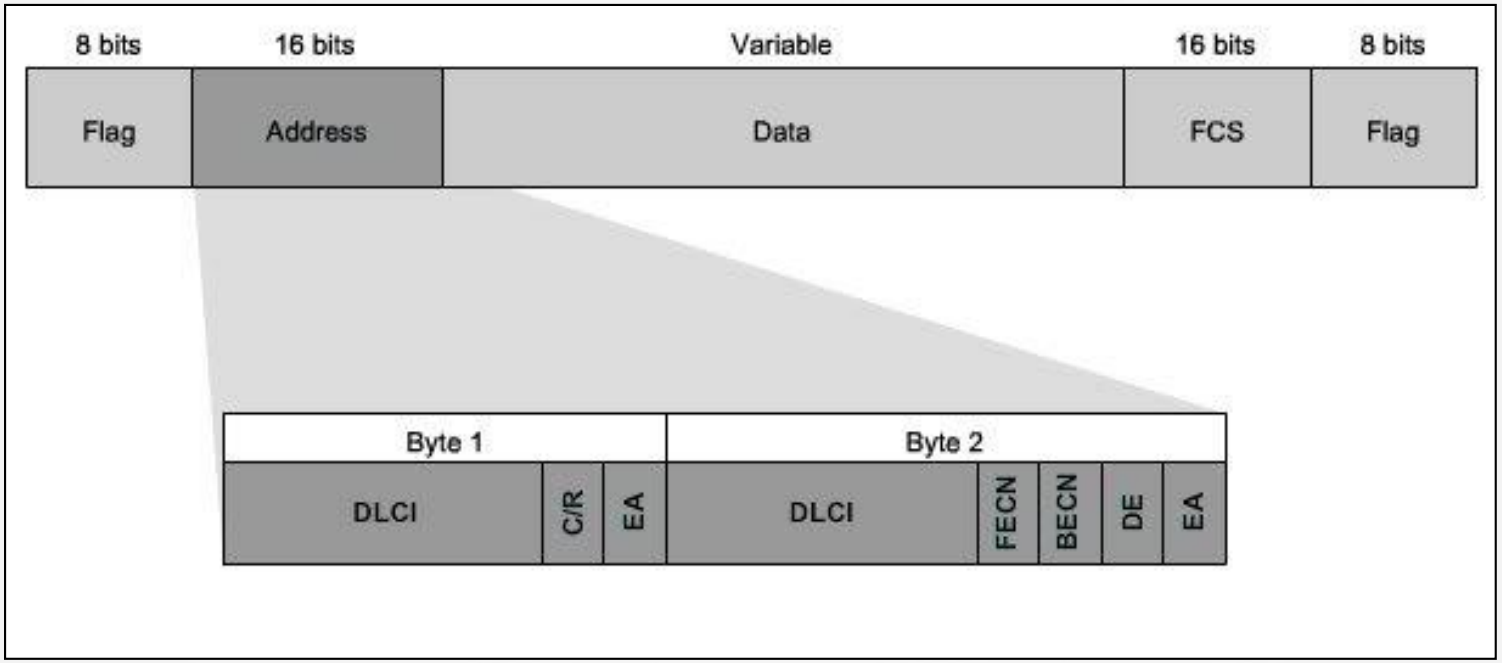
- Committed Information Rate
- Amount of data rate that a service provider guarantees
- Anything above the CIR is considered as burst

DE, FECNs and BECNs

- Frame Relay Switches monitor data usage.

- If congestion occurs, various bits in Frame-Relay header can be set to notify end-devices, or drop frames in the WAN cloud.
  - DE – Discard Eligible bit
  - FECN = Forward Explicit Congestion Notification
  - BECN – Backwards Explicit Congestion Notification

Frame Relay Header



Implementing Frame Relay

Multipoint Configuration

- Configuration commands
  - Router(config-if)# encapsulation frame-relay
  - Router(config-if)# ip address <address> <subnet mask>
  - Router(config-if)# end
- Configuring static mapping
  - Router(config-if)# frame-relay map ip <destination address> <dlci> broadcast

Point-to-Point Configuration (sub-interface)

Configuration commands

- Router(config-if)# no ip address
- Router(config-if)# encapsulation frame-relay
- Router(config-if)# no shut
- Router(config-if)# exit
- Router(config)# interface serial x/y.<sub-interface number>
- Router(config-sub-if)# ip address <address> <subnet mask>
- Router(config-sub-if)# frame-relay interface-dlci <dlci>

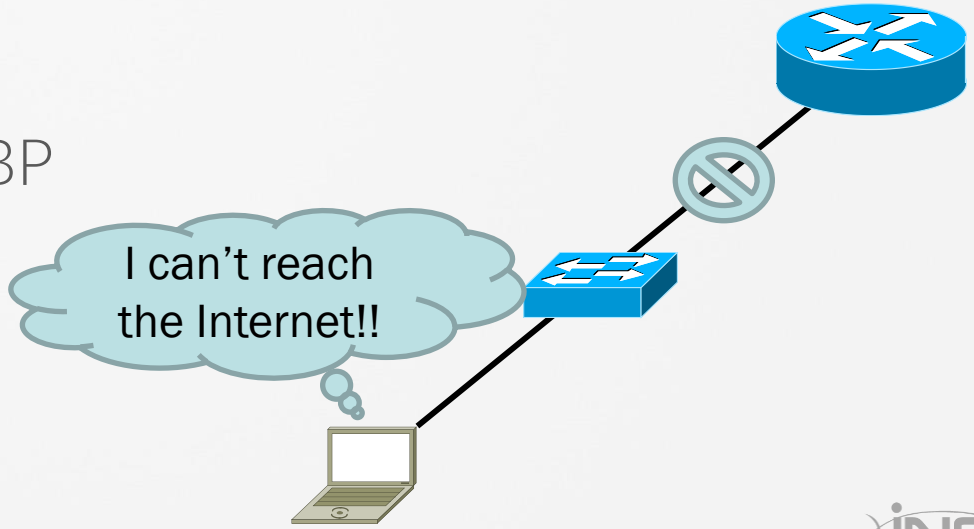
Verifying Frame Relay

- Verification commands
  - Router# show frame-relay pvc
  - Router# show frame-relay map

Introduction to FHRP

FHRP

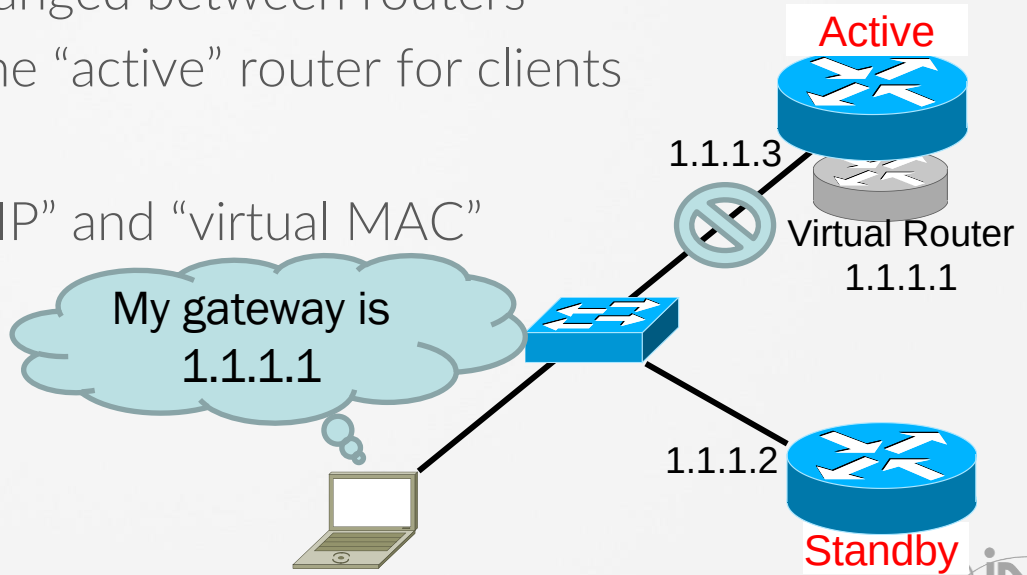
- First Hop Redundancy Protocols
  - Generic term for any protocol that provides redundancy for client default gateway
- Cisco Proprietary
  - HSRP / HSRPv2 / GLBP
- Standardized (IETF)
  - VRRP



General Characteristics

All FHRPs have some common characteristics:

- Require two-or-more routers
- Hello's/Keepalives exchanged between routers
- One router elected as the "active" router for clients
- Sub-second failover
- Routers share a "virtual IP" and "virtual MAC"



HSRP

- Hot Standby Router Protocol
- Cisco proprietary
- Uses UDP port 1985 and multicast address 224.0.0.2
- Two roles: Active and Standby
- HSRP router with highest priority is considered "Active"
  - Default priority = 100

HSRP

- MAC address: 0000.0c07.acxx
  - *xx* refers to the group number in hexadecimal
- Preemption disabled by default
- HSRP/VRRP = No load-sharing feature
- MHSRP can be used for load sharing
- Requires multiple VLANs

HSRP Authentication

- **Authentication supported**
  - Plain text
  - MD5
- **Plain-text configuration**
  - Sw1(config-if)#standby <group-id> authentication <password>

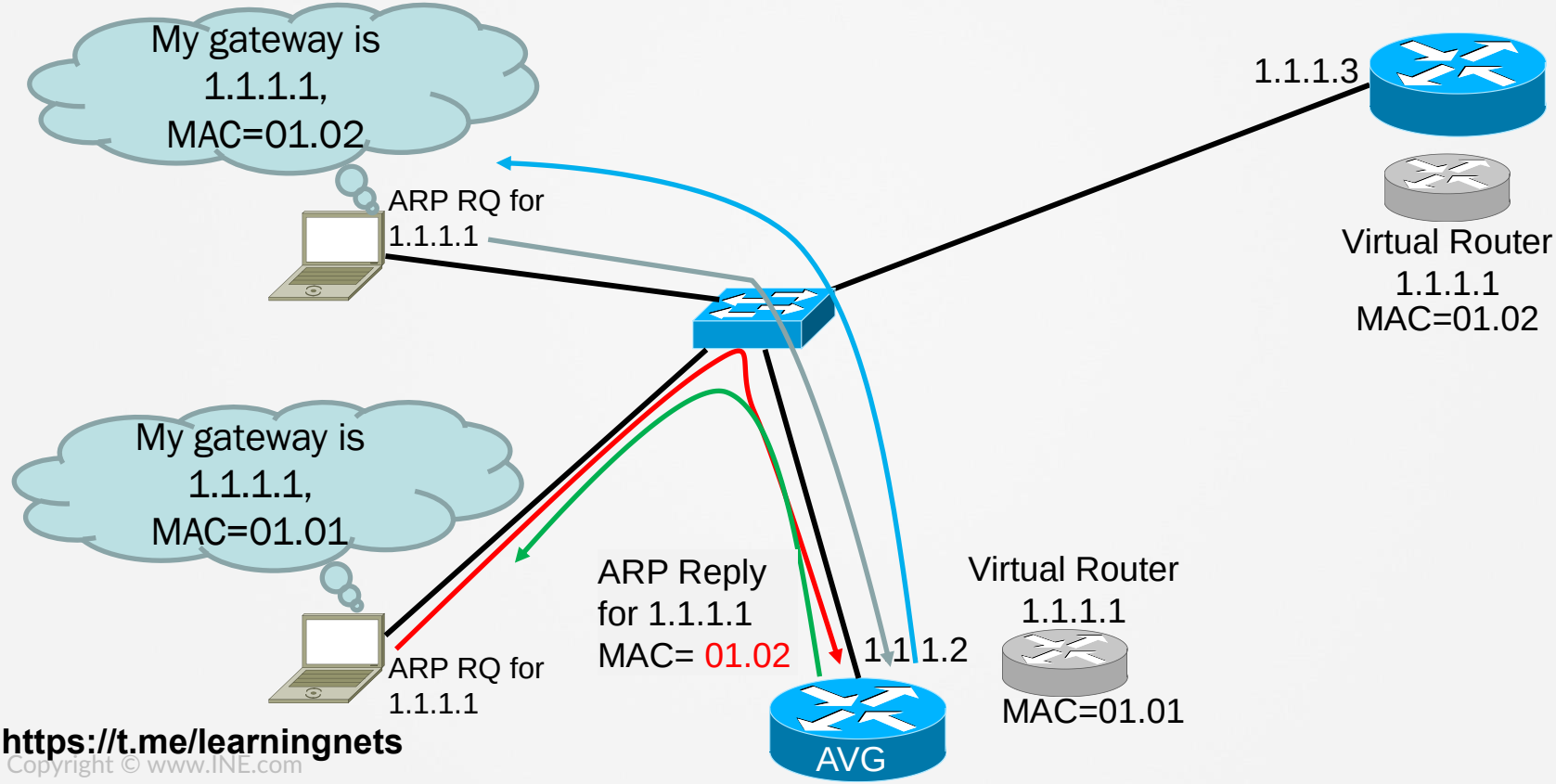
HSRP Timers

- Hello timer – 3 sec
- Hold timer – 3 sec

GLBP

- Gateway Load-Balancing Protocol
- Cisco Proprietary
- Provides gateway redundancy AND per-host load-balancing.
- AVG (Active Virtual Gateway) in charge of determining host-to-gateway allocations.

GLBP



Implementing HSRP

Implementing HSRP

- Enabling HSRP in the interface
  - Switch(config-if)# standby <group-id> <virtual-ip>
- Configuring priority
  - Switch(config-if)# standby <group-id> priority <priority>
- Enabling preemption
  - Switch(config-if)# standby <group-id> preempt

Verifying HSRP

- **Verification commands**
  - Switch# show standby
  - Switch# show standby brief

IOS DHCP Server

DHCP

- Dynamic Host Configuration Protocol
- Operates in application layer
- Provides dynamic method of leasing an IP address to a host
- Uses UDP ports 67 and 68
- Can be useful for a small networks where a dedicated DHCP server is not available

Configuration

Configuring DHCP

- Router(config)# service dhcp
- Router(config)# ip dhcp pool <pool-name>
- Router(config-dhcp)# network < network-id>
- Router(config-dhcp)# default-router <default gateway>
- Router(config-dhcp)# dns-server <dns-server address>
- Router(config-dhcp)# lease <duration>
- Router(config-dhcp)# end

Configuration

Configuring exclude list

- Router(config)# ip dhcp excluded-address <start-ip> <end-ip>

Verifying DHCP

- Verification commands
  - Router# show ip dhcp binding

Access Control List

Introduction to ACL

- Packet filtering mechanism
- Can filter packets on the basis of Layer 3 and Layer 4 header
- Should have at least one permit statement
- Works in sequential order; statement with lower sequence is preferred and checked

Introduction to ACL

- Only one ACL can be applied per interface, per direction
- Can be applied inbound and outbound
  - Inbound – before routing
  - Outbound – after routing
- Implicit deny rule applied at the end of the sequence if nothing has been defined

Types of ACL

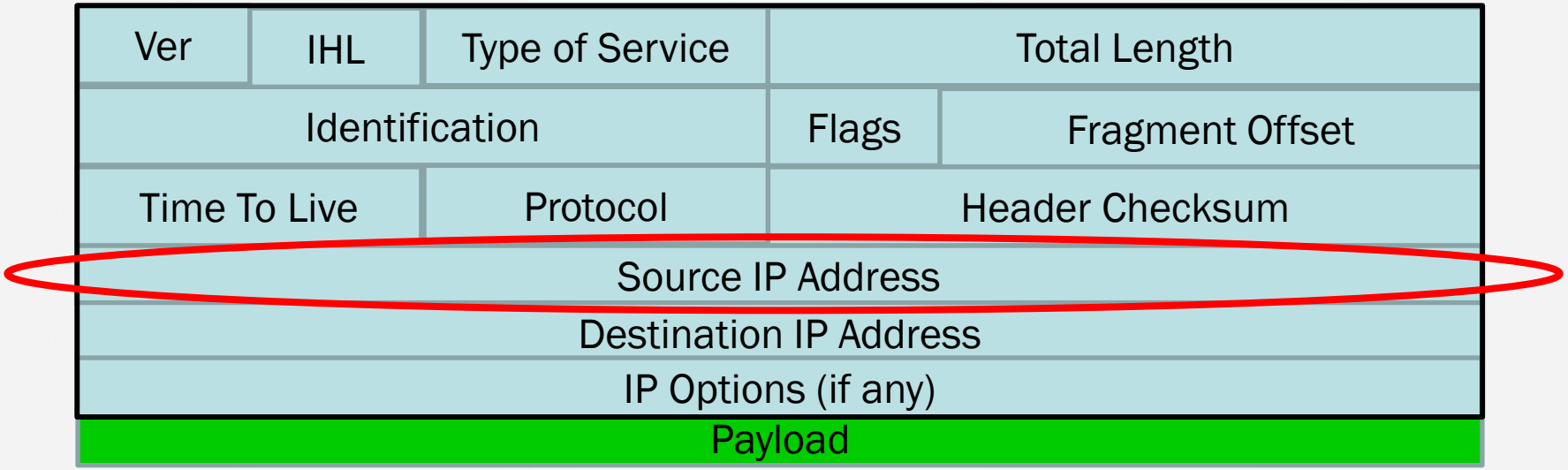
- Standard ACL
- Extended ACL
- Named ACL
  - Standard Named
  - Extended Named

Standard ACL

Standard ACL

- Filters traffic based on Layer 3 header
- Source IP address is checked
- ACL numbers range from 1 through 99
- Should be applied nearest to destination
- No intelligence of checking destination address and port numbers

Where can a Standard ACL Look?



Configuring Standard ACL

- **Configuration command**
  - Router(config)# access-list <ACL no> <permit | deny> <source address> <wild card mask>
- **Applying configuration**
  - Router(config-if)# ip access-group < ACL no> < in | out >

Verifying Standard ACL

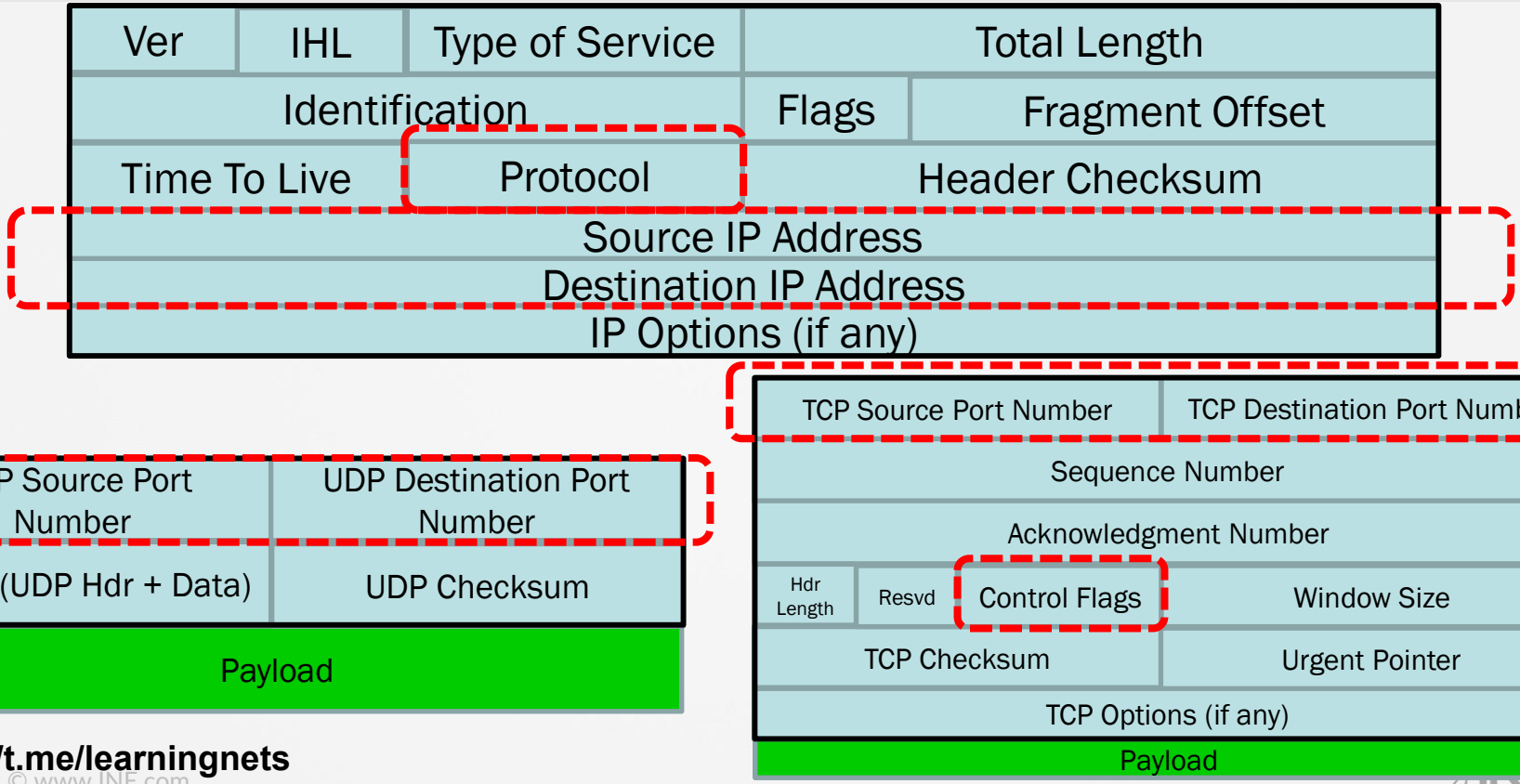
- **Verification commands**
  - Router# show ip access-list
  - Router# show ip interface
  - Router# show run | inc access-list

Extended ACL

Extended ACL

- Filters traffic based on layer 3 and 4 header
- Source and destination IP and port numbers are checked
- ACL numbers range from 100 through 199
- Should be applied nearest to source
- Capable of transport header inspection

Where can an Extended ACL Look?



Configuring Extended ACL

- **Configuration command**
  - Router(config)# access-list <ACL no> <permit | deny> < protocol> <source address> <wildcard mask> < destination address> <wildcard mask> < port numbers>
- **Applying configuration**
  - Router(config-if)# ip access-group < ACL no> < in | out >

Verifying Extended ACL

Verification commands

- Router# show ip access-list
- Router# show ip interface
- Router# show run | inc access-list

Named ACL

Named ACL

- Individual statements can be edited, unlike numbered ACLs
- Can be used with naming convention
- Use of name instead of number makes management easier
- More flexible than numbered ACLs

Configuring Standard Named ACL

- **Configuration command**
  - Router(config)# ip access-list standard <name>
  - Router(config-std-acl)# <permit | deny> <source address>
- **Applying configuration**
  - Router(config-if)# ip access-group <name> <in | out>

Configuring Extended Named ACL

- **Configuration commands**
  - Router(config)# ip access-list extended <name>
  - Router(config-ext-acl)# <permit | deny> <protocol> <source-address> <wildcard mask> <destination-address> <wildcard mask>
- **Applying configuration**
  - Router(config-if)# ip access-group <name> <in | out>

Network Address Translation

Introduction to NAT

- Separates LAN from WAN and provides accessibility to the outside world
- Translates RFC1918 space addresses into public addresses
- Provides security
- Helps reduce public IP address consumption
- Hides private addresses from outsiders

Public & Private Addresses

- Private addresses cannot go outside the network
- At least one public address is required for a private host to communicate to the Internet
- Public IP addresses are globally routable

Types of NAT

- Static NAT
- Dynamic NAT
- PAT (Port Address Translation)

Static NAT

Static NAT

- One to one mapping
- One private host requires a public IP address
- Usually deployed at server end

Configuring Static NAT

Configuration commands

- Router(config-if)# ip nat inside
- Router(config-if)# ip nat outside
- Router(config)# ip nat inside source static <private address> < public address>

Verifying Static NAT

- **Verification commands**
  - Router# show ip nat translation

Dynamic NAT

Dynamic NAT

- Many to many mapping
- One private host requires a public IP address obtained from a pool of available addresses.
- Usually deployed at server end
- Easier from the perspective of scalability

Configuring Dynamic NAT

Configuration commands

- Router(config-if)# ip nat inside
- Router(config-if)# ip nat outside
- Router(config# access-list < acl no> <permit | deny > <source-address> <wildcard mask>
- Router(config)# ip nat pool <name> <start-ip> <end-ip> netmask <subnet mask>
- Router(config)# ip nat inside source list < acl no> pool <name>

Verifying Dynamic NAT

- **Verification commands**
  - Router# show ip nat translation

Port Address Translation (PAT)

PAT

- Port Address Translation
- One to many mapping
- One public address can provide multiple host connections
- Usually deployed at client end
- Easier from the perspective of scalability

Configuring PAT

Configuration commands

- Router(config-if)# ip nat inside
- Router(config-if)# ip nat outside
- Router(config# access-list < acl no> <permit | deny > <source-address> <wildcard mask>
- Router(config)# ip nat pool <name> <start-address> <end-address> netmask < subnet mask>
- Router(config)# ip nat inside source list < acl no> pool <name> overload

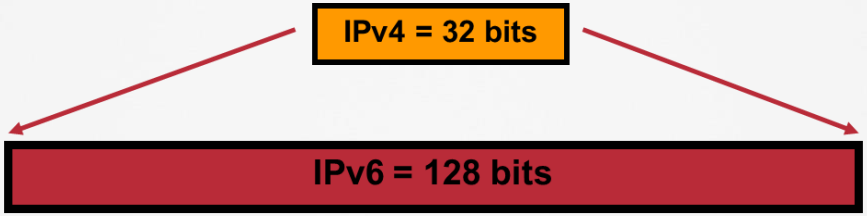
Verifying PAT

- Verification commands
  - Router# show ip nat translation

Introduction to IPv6

Introduction to IPv6

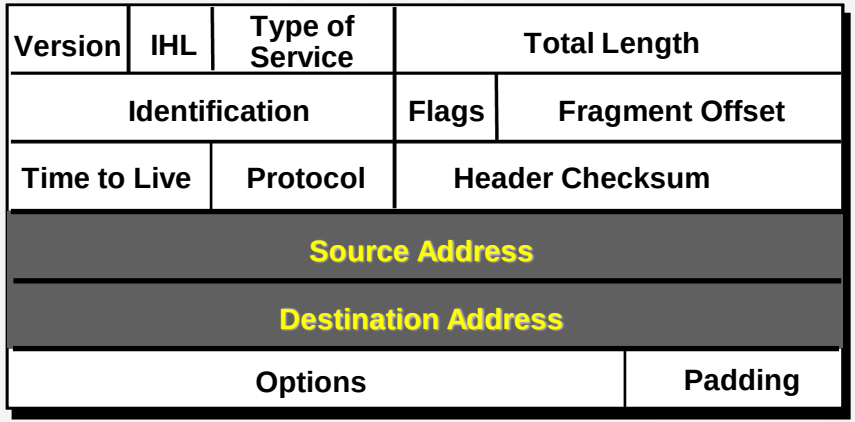
- 128-bit addressing system



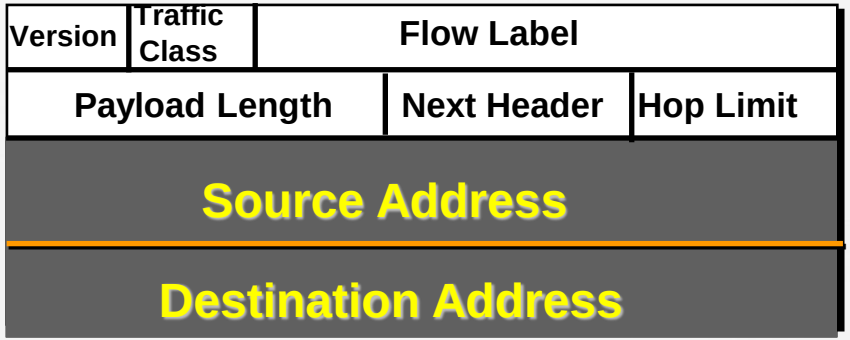
- Expressed in hexadecimal instead of decimal
- Colon ":" used to separate group of four-hex characters (a "word")
- 4 bits = 1 hex character
- Example: 2001:0000:0000:0000:0000:0001:1230:000A

IPv4 and IPv6 Header Comparison

IPv4 Header:



IPv6 Header:



Introduction to IPv6

- IPv6 addresses are complex and not easy to write as IPv4
- Can be suppressed to minimize overhead
  - Leading zeros in each word can be omitted
  - A single set of consecutive zeros can be replaced with double colon "::"
  - Multiple double colons are not allowed in a single address
  - Example: 2001:0000:0000:0000:0000:0001:0000:000A/64
  - Suppress: 2001::1:0:A/64

Types of IPv6 Addresses

- **Link-local addresses**
  - Assigned automatically as an IPv6 host comes online
  - Similar to the 169.254.x.x address of IPv4
  - Always begin with "FE80::/10"
  - Last 64 bits is the 48-bit MAC address with "FFFE" inserted in the middle
- **Global Unicast Addresses**
  - Have their high-level 3 bits set to 001; ex: 2000::/3
  - Global routing prefix is 48 bit or less

Types of IPv6 Addresses

- **Unique Local Addresses**
  - Not globally routable
  - Similar to the private addresses of IPv4
  - Always begin with "FC00::/7"
- **Solicited Node Multicast Address**
  - FF02:0:0:0:0:1:FF/104
  - Last 24-bits taken from interface-ID (next slide)
  - Automatically provisioned like Link-Local addresses
  - Used for L3-to-L2 address resolution

No more broadcasts!!

IPv4/IPv6 “Host” bits

- **IPv4 host address** — 20.20.20.3 /24
  - Network/subnet portion (prefix) — 20.20.20.x
  - Host portion (host bits) — X.X.X.3 /24

- **IPv6 host addresses** — 2001:aabb:cc11::3a/64
  - Network/subnet portion (prefix) — 2001:aabb:cc11::/64
  - Interface Identifier — :0000:0000:0000:003a/64

ICMPv6 and NDP

- NDP = Neighbor Discovery Protocol

- Makes use of new ICMPv6 message types
  - Neighbor Solicitation
  - Neighbor Advertisement
  - Router Solicitation
  - Router Advertisement

Implementing IPv6

Configuration – Enabling IPv6

Router(config-if)# ipv6 address <address> / <prefix-length> <EUI-64>

Router(config-if)# ipv6 address autoconfig <default>

Router(config-if)# no shutdown

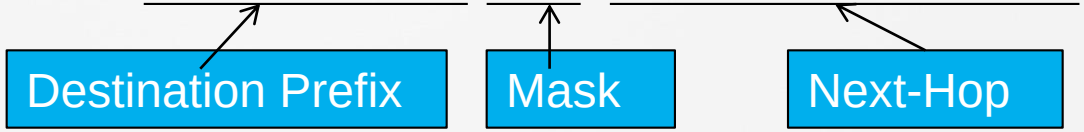
Router(config)# ipv6 unicast-routing

- Permits router to send ICMPv6 Router Advertisements
- Allows IPv6 routing between connected interfaces
- Allows operation of IPv6 routing protocols

IPv6 Static Routes

- Purpose is the same as IPv4 Static Routes
- Must supply destination prefix, mask and next-hop

Router(config)#ipv6 route 2001:2:3:4::/64 2001:aa:bb:cc::77



Router(config)# ipv6 route ::/0 2001:aa:bb:cc::77

Verifying IPv6

- **Verification commands**
  - Router# show ipv6 interface brief
  - Router# show ipv6 routers
  - Router# show ipv6 route
  - Router# debug ipv6 nd

Implementing RIPng

RIPng

- RIP Next Generation
- Similarities with RIPv2 for IPv4:
  - Distance-vector, radius of 15 hops, split-horizon and etc.
- RIPng
  - IPv6 prefix, next-hop IPv6 address
  - Uses the multicast group FF02::9, the all-rip-routers multicast group, as the destination address for RIP updates
  - Can create multiple, named, RIP processes within a single router.

Implementing RIPng

- Enablgin RIPng
  - Router(config)# ipv6 unicast routing
  - Router(config-if)# ipv6 rip <name> enable
- Options for RIPng
  - Router(config)# ipv6 router rip <name>
    - Redistribution
    - Route Filtering
    - Changing of default values
    - Etc

Verifying RIPng

- Verification commands
  - Router# show ipv6 rip database
  - Router# show ipv6 route rip
  - Router# show ipv6 protocols

Implementing EIGRPv6

IPv6 EIGRP

- **Similarities to IPv4 EIGRP**
  - Most packets transmitted via multicast
  - Same metric formula
  - Utilizes same message types (hello, update, ack, etc)
- **IPv6 Uniqueness**
  - Packets sent to FF02::A
  - Peers with Link-Local address of neighbors
  - Next-Hop address is Link-Local of peer

EIGRPv6

- Enabling EIGRPv6
  - Router(config)# ipv6 unicast routing
  - Router(config)# ipv6 router eigrp <number>
  - Router(config)# no shutdown
  - Router(config)# router-id x.x.x.x
  - Router(config-router)# exit
- Applying EIGRPv6
  - Router(config-if)# ipv6 eigrp <number>

Verifying EIGRPv6

- **Verification commands**
  - Router# show ipv6 eigrp neighbor
  - Router# show ipv6 eigrp topology
  - Router# show ipv6 route eigrp
  - Router# show ipv6 protocols

Implementing OSPFv3

OSPFv3 Theory

» Similarities to IPv4 OSPF (OSPF v2)

- Same OSPF message types (Hello, LS-Update, etc)
- Uses same SPF algorithm, LSDB, and Metric
- Requires a Router-ID (32-bit)
- Same process for neighbor discovery and establishment

» Differences with OSPF v3

- Neighbors formed with Link Local addresses
- Neighbors don't need same global prefixes on same link.
- Changes to name/functionality of some LSAs

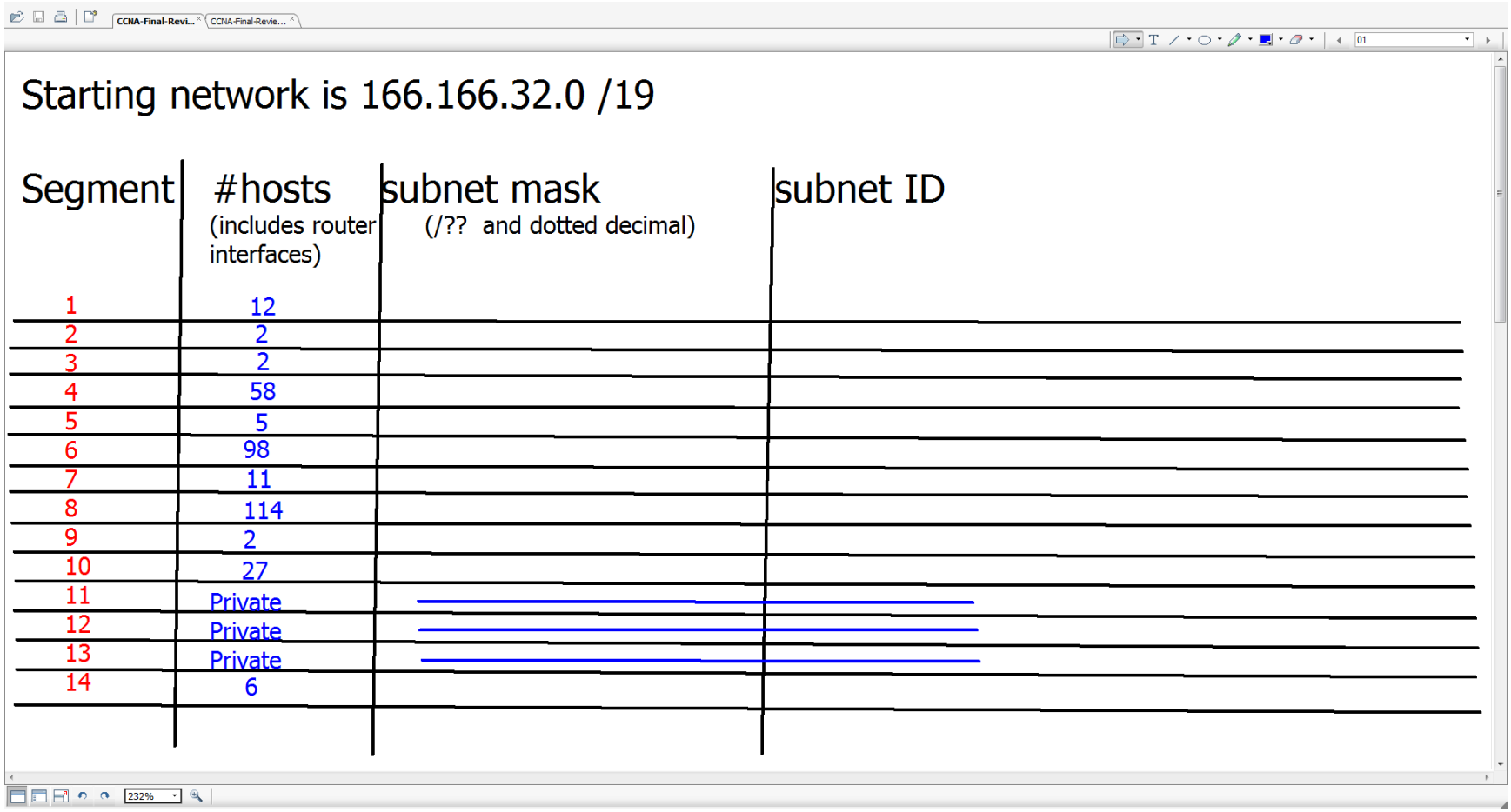
OSPFv3

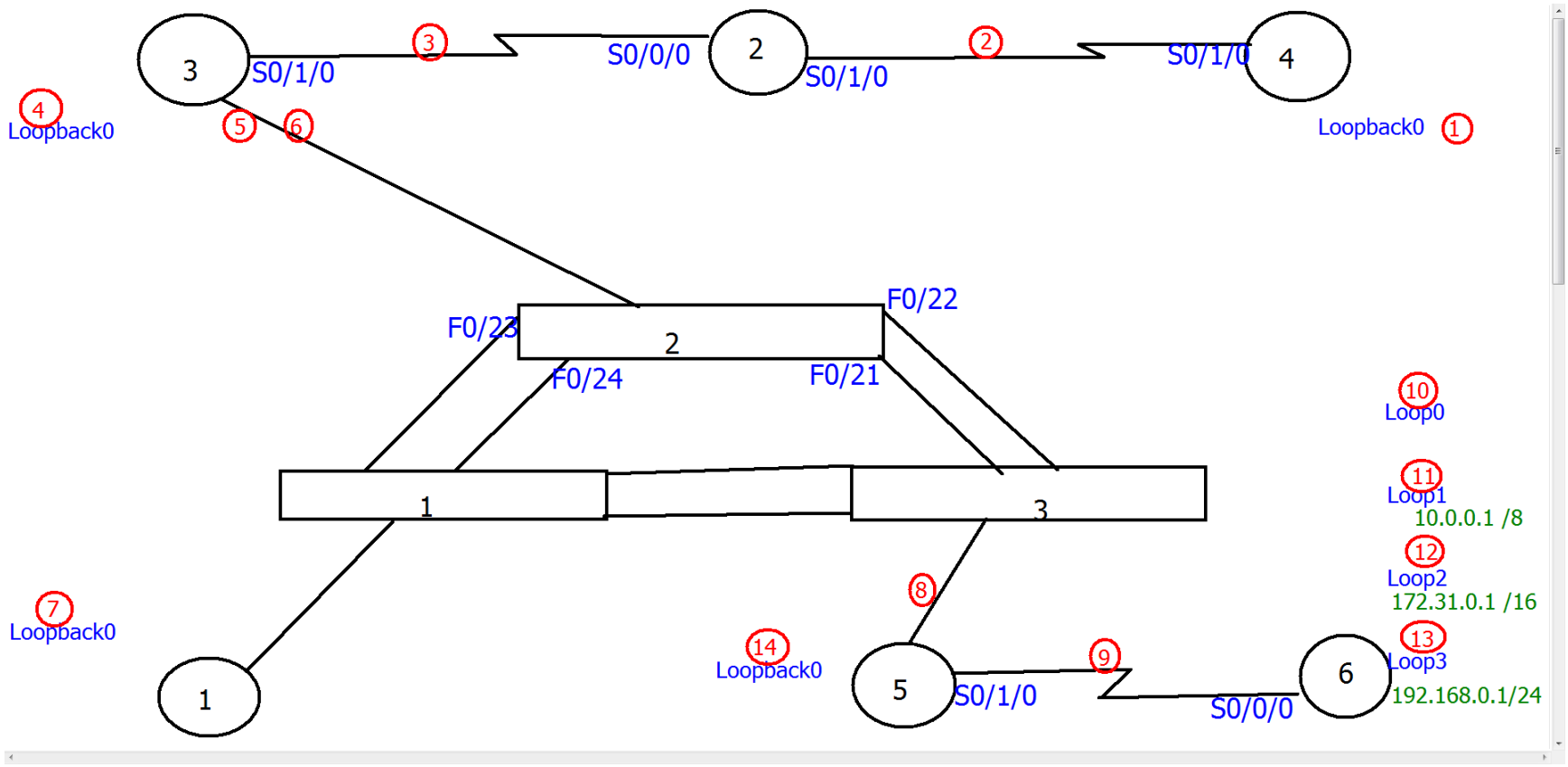
- Enabling OSPFv3
  - Router(config)# ipv6 unicast routing
  - Router(config)# ipv6 router ospf <number>
  - Router(config-router)# router-id <address>
  - Router(config-router)# exit
- Applying OSPFv3
  - Router(config-if)# ipv6 ospf <number> area <area-id>

Verifying OSPFv3

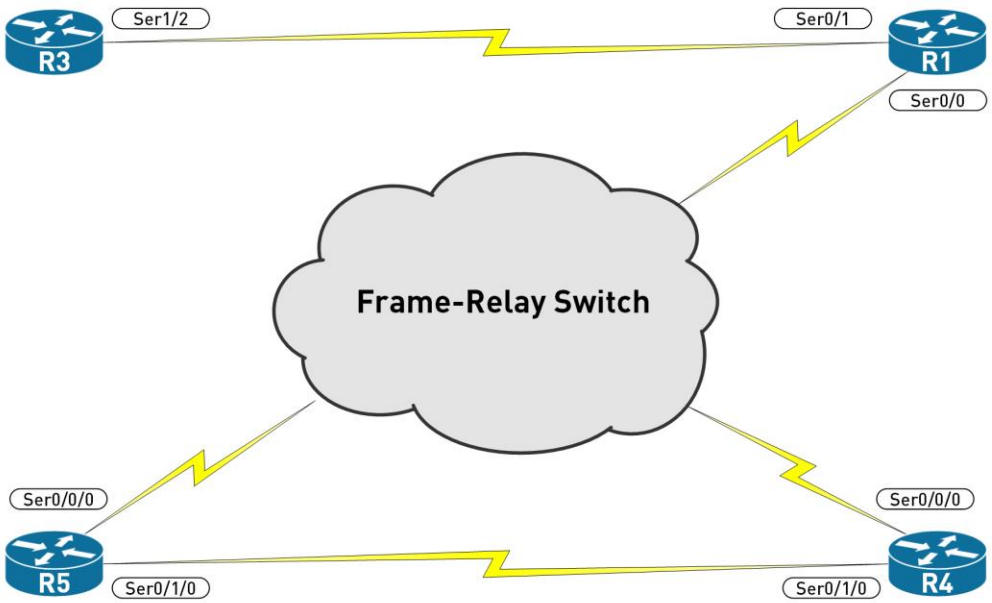
Verification commands

- Router# show ipv6 ospf neighbor
- Router# show ipv6 ospf database
- Router# show ipv6 route ospf
- Router# show ipv6 protocols

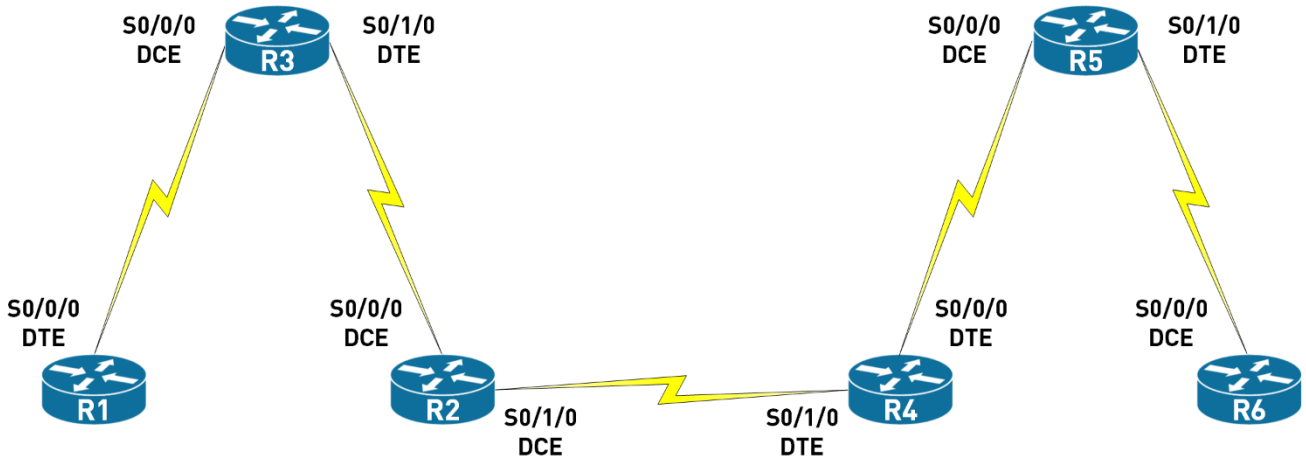




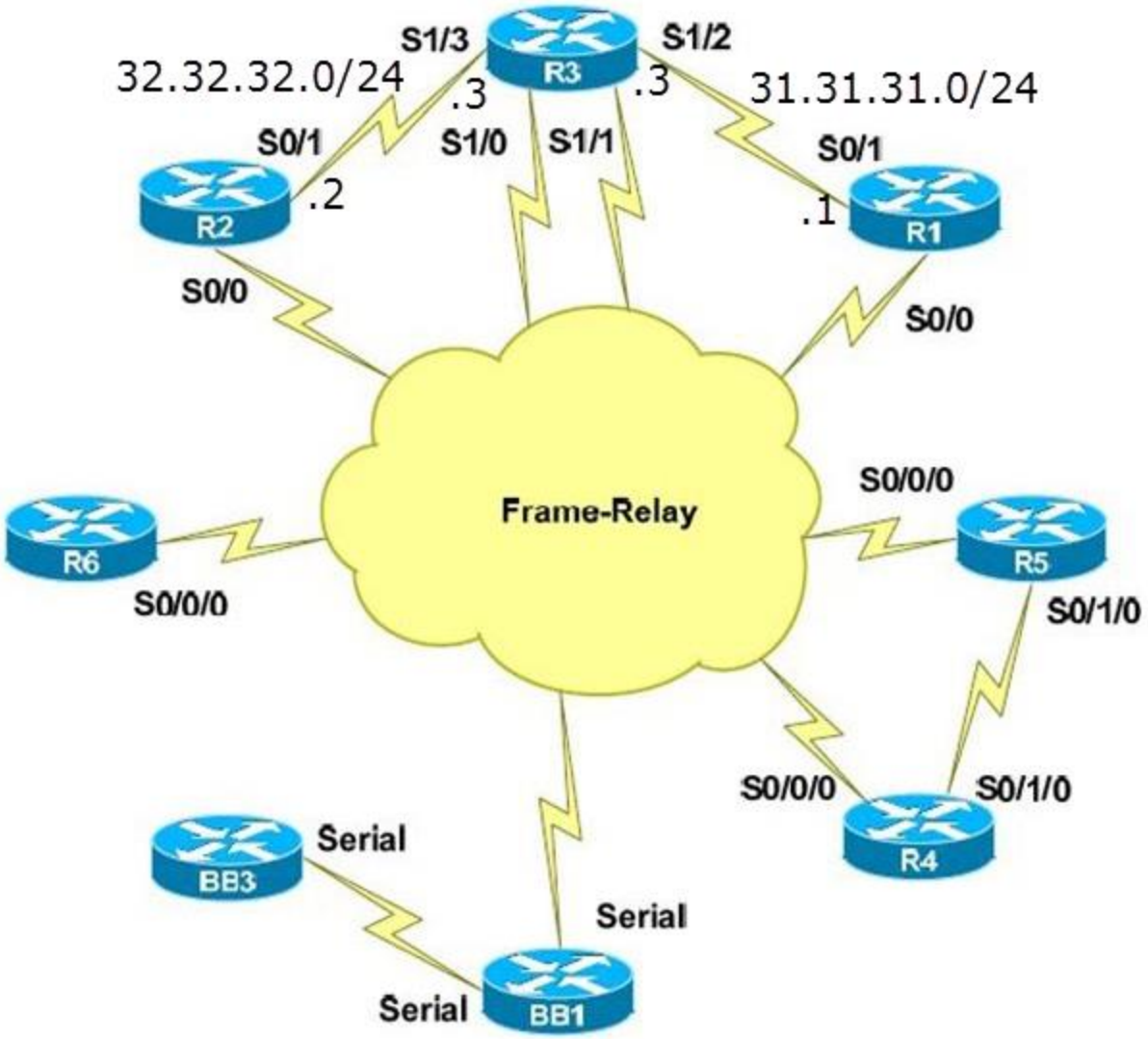
Topology #3



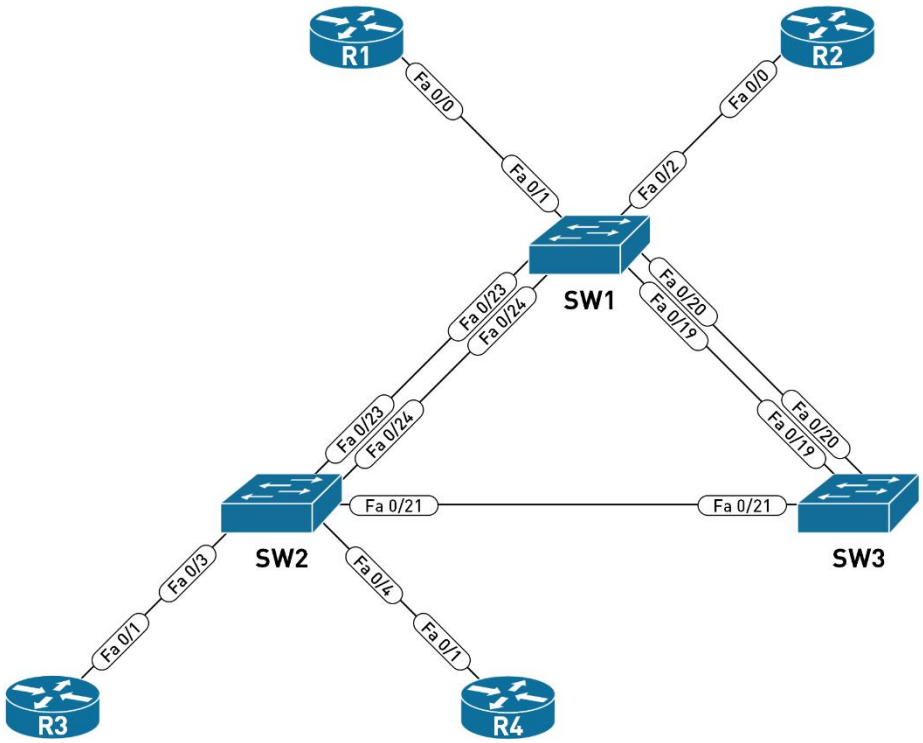
CCNA Routing & Switching
Technologies Class



CCNA Routing & Switching
Technologies Class

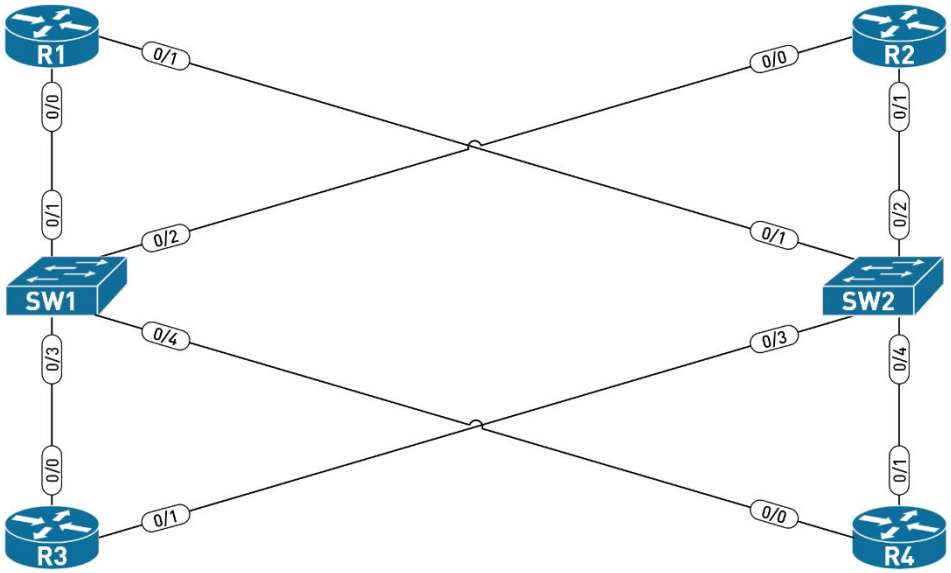


Topology #1



CCNA Routing & Switching
Technologies Class

Topology #2



CCNA Routing & Switching
Technologies Class