

Table of Contents

1. NETWORKING DEVICES.....	3
2. INTERFACES AND CABLES	4
3. OSI MODEL & TCP/IP SUITE	10
4. INTRO TO THE CLI	15
5. ETHERNET LAN SWITCHING : PART 1	21
6. ETHERNET LAN SWITCHING : PART 2	26
7. IPv4 ADDRESSING : PART 1	30
8. IPv4 ADDRESSING : PART 2	38
9. SWITCH INTERFACES	41
10. THE IPv4 HEADER.....	47
11a. ROUTING FUNDAMENTALS : PART 1	50
11b. STATIC ROUTING : PART 2.....	53
12. LIFE OF A PACKET.....	59
13. SUBNETTING : PART 1.....	67
14. SUBNETTING : PART 2.....	71
15. SUBNETTING (VLSM) : PART 3.....	72
16. VLANS : PART 1	76
17. VLANS : PART 2	84
18. VLANS : PART 3	97
19. DTP / VTP (Not in Syllabus).....	105
20. SPANNING TREE PROTOCOL (STP) : PART 1	109
21. SPANNING TREE PROTOCOL (STP) : PART 2	120
22. RAPID SPANNING TREE PROTOCOL	127
23. ETHERCHANNEL.....	136
24. DYNAMIC ROUTING	149
25. RIP and EIGRP (IGP : DYNAMIC VECTOR)	160
26. OSPF : PART 1 (IGP : LINK STATE).....	169
27. OSPF : PART 2 (IGP : LINK STATE).....	175
28. OSPF : PART 3 (IGP: LINK STATE).....	183
29. FIRST HOP REDUNDANCY PROTOCOLS.....	190
30. TCP and UDP (LAYER 4 PROTOCOLS)	197
31. IPv6 : PART 1	205
32. IPv6 : PART 2.....	212
33. IPv6 : PART 3.....	219
34. STANDARD ACCESS CONTROL LISTS (ACL).....	226
35. EXTENDED ACCESS CONTROL LISTS (EACL)	232
36. CDP and LLDP (Layer 2 Discovery Protocol).....	240
37. NTP.....	248
38. DNS (Domain Name System)	258
39. DHCP (Dynamic Host Configuration Protocol)	265
40. SNMP (Simple Network Management Protocol)	277
41. SYSLOG	284
42. SSH (Secure Shell).....	288
43. FTP and TFTP.....	294
44. NAT (STATIC): PART 1.....	302
45. NAT (DYNAMIC): PART 2	307
46. QoS (Voice VLANs) : PART 1.....	315
47. QoS (Quality of Service) : PART 2	321
48. SECURITY FUNDAMENTALS	329
49. PORT SECURITY	335
50. DHCP SNOOPING (LAYER 2).....	343
51. DYNAMIC ARP INSPECTION.....	349
52. LAN ARCHITECTURES.....	355
53. WAN ARCHITECTURES.....	361
54a. VIRTUALIZATION AND CLOUD: PART 1	370
54b. VIRTUALIZATION (CONTAINERS): PART 2.....	376
54c. VIRTUALIZATION (VRF): PART 3.....	379
55. WIRELESS FUNDAMENTALS.....	382
56. WIRELESS ARCHITECTURES	391
57. WIRELESS SECURITY	401
58. WIRELESS CONFIGURATION.....	407
59. INTRODUCTION TO NETWORK AUTOMATION	428
60. JSON, XML, AND YAML	435
61. REST APIS.....	440
62. SOFTWARE DEFINED NETWORKING (SDN).....	444
63. ANSIBLE, PUPPET, AND CHEF	450

1. NETWORKING DEVICES

What is a network?

A computer network is a digital telecommunications network allows NODES to share RESOURCES.

A CLIENT is a device that accesses a service made available by a SERVER.

A SERVER is a device that provides functions or services for CLIENTS.

- Note : The same device can be a CLIENT in some situations and a SERVER in other situations.
Ex: A Peer-to-Peer network.

SWITCHES (Level 2):

- provide connectivity to hosts within the same LAN (Local Area Network)
- Have many network interfaces/ports for End Hosts to connect to.
- DO NOT provide connectivity between LANs/over the Internet.

ROUTERS (Level 3):

- have fewer network interfaces than switches.
- are used to provide connectivity BETWEEN LANs.
- are used to send data over the Internet.

FIREWALL (Can be Level 3,4, and 7):

- Firewalls are specialty hardware network security devices that control network traffic entering/exiting your network.
- Can be places "inside" or "outside" the network.
- Monitor and control network traffic based on configured rules.
- Are known as "Next-Generation Firewalls" when they include more modern and advanced filtering capabilities.
- Host-based firewalls are software applications that filter traffic entering and exiting a host machine, like a PC.

2. INTERFACES AND CABLES

SWITCHES provide many PORTS for connectivity (usually 24)
These PORTS tend to be RJ-45 (Registered Jack) ports.

WHAT IS ETHERNET?

- Ethernet is a collection of network protocols/standards.

Why do we need network protocols and standards?

- provide common communication standards over networks.
- provide common hardware standards to allow connectivity between devices.

Connections between devices operates at a set speed.

These speeds are measured in "bits per second" (bps)

A bit is a value of "0" or "1". A byte is 8 bits (0s and 1s)

Size	# of Bits
1 kilobit (Kb)	1,000
1 megabit (Mb)	1,000,000
1 gigabit (Gb)	1,000,000,000
1 terabit (Tb)	1,000,000,000,000

Ethernet standards are:

- Defined in the IEEE 802.3 standard in 1983
- IEEE = Institute of Electrical and Electronics Engineers

ETHERNET STANDARDS (COPPER)

Speed	Common Name	Standard	Cable Type	Max Transmission Distance
10 Mbps	Ethernet	802.3i	10BASE-T	100m Max
100 Mbps	Fast Ethernet	802.3u	100BASE-T	100m Max
1 Gbps	Gigabit Ethernet	802.3ab	1000BASE-T	100m Max
10 Gbps	10 Gigabit Ethernet	802.3an	10GBASE-T	100m Max

BASE = refers to Baseband Signaling

T = Twisted Pair

Most Ethernet uses copper cables.

UTP or Unshielded Twisted Pair (no metallic shield) Twist protects against EMI (Electromagnetic Interference)

Most use 8 wires (4 pairs) however ...

10/100BASE-T = 2 pairs (4 wires)

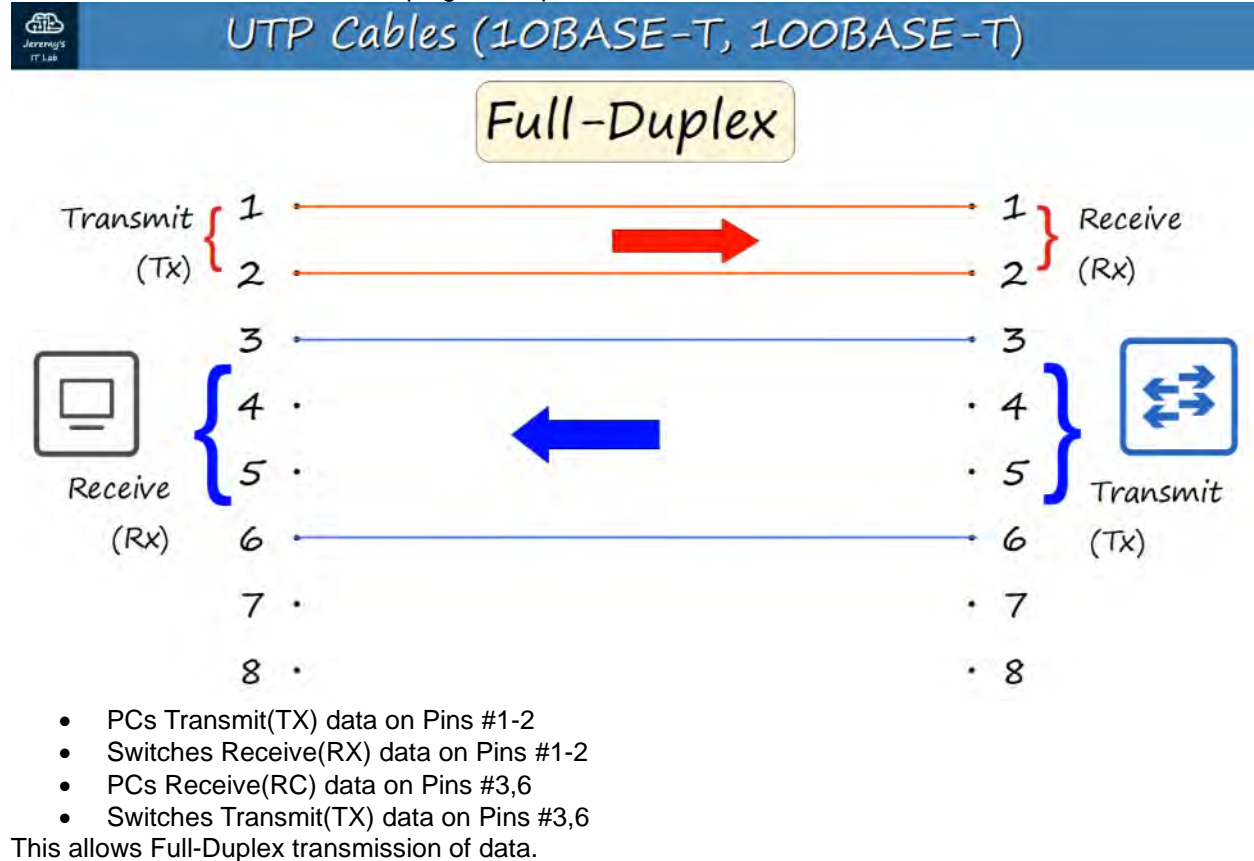
Explanation

Cat (Category) 5e is a kind of copper cable used for Ethernet connections. It supports speeds of up to 1 Gbps and should be a maximum of 100 meters in length to avoid signal attenuation.

Below are a few cable standards:

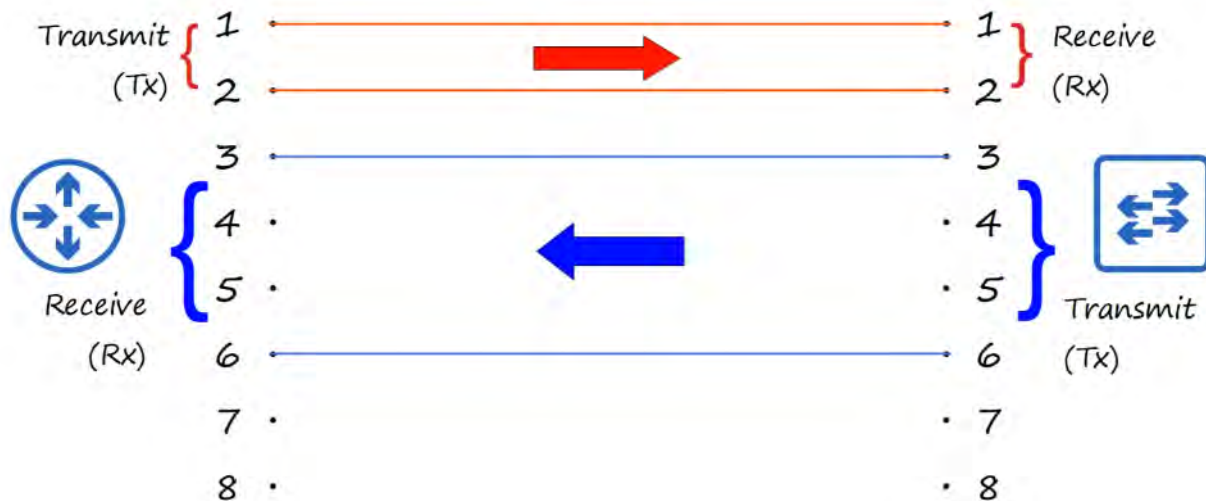
- **Cat 3**
 - 10 Mbps (10BASE-T)
- **Cat 5**
 - 100 Mbps (100BASE-T)
- **Cat 5e**
 - 1 Gbps (1000BASE-T)
- **Cat6a**
 - 10 Gbps (10GBASE-T)

How do devices communicate via their connections?
Each ethernet cable has a RJ-45 plug with 8 pins on the ends.



What if a Router / Switch connect?

UTP Cables (10BASE-T, 100BASE-T)



- Routers Transmit(TX) data on Pins #1-2
- Routers Receive(RX) data on Pins #3,6
- Switches Transmit(TX) data on Pins #3,6
- Switches Receive(RX) data on Pins #1-2

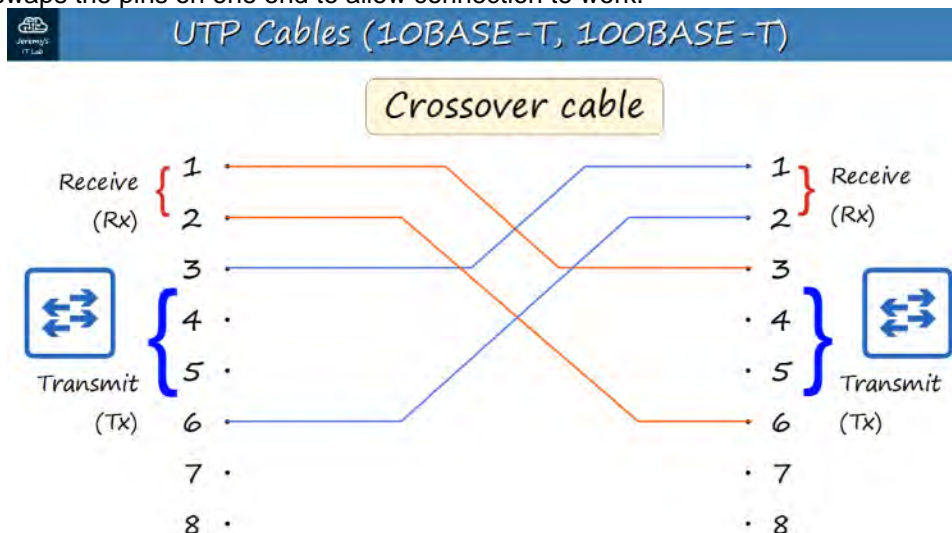
Routers and PCs connect the same way with Switches.

The cable used to connect is called a "Straight-Through" cable.

What if we want to connect similar devices to each other?

We CANNOT use a "Straight-Through" cable. We MUST use a "Crossover" cable.

This cable swaps the pins on one end to allow connection to work.



PIN#1 -----> PIN#3 PIN#2 -----> PIN#6
PIN#3 -----> PIN#1 PIN#6 -----> PIN#2

DEVICE TYPE	TRANSMIT (TX) PINS	RECEIVE (RX) PINS
ROUTER	1 and 2	3 and 6
FIREWALL	1 and 2	3 and 6
PC	1 and 2	3 and 6
SWITCH	3 and 6	1 and 2

Most modern equipment now has AUTO MDI-X which **automatically detects** which pins their neighbour is transmitting on and adjust the pins they receive data on.

1000BASE-T/10GBASE-T = 4 pairs (8 wires)

Each wire pair is **bidirectional** so can transmit/receive much faster than 10/100BASE-T.

UTP Cables (1000BASE-T, 10GBASE-T)



Fiber-Optic Connections:

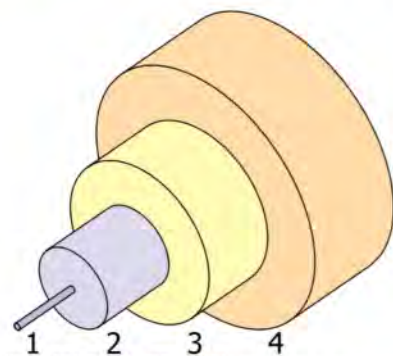
- Defined in the IEEE 802.3ae standard

SFP Transceiver (Small Form-Factor Pluggable) allows fiber-optic cables to connect to switches/routers.

- Have separate cables to transmit / receive.

4 parts to a fiber-optic cable.

Fiber-Optic Connections



- 1: the fiberglass core itself
- 2: cladding that reflects light
- 3: a protective buffer
- 4: the outer jacket of the cable

single-mode

multimode

Original by Bob Mellish, SVG derivative by Benzhill
(https://commons.wikimedia.org/wiki/File:Singlemode_fibre_structure.svg), "Singlemode fibre structure", <https://creativecommons.org/licenses/by-sa/3.0/legalcode>

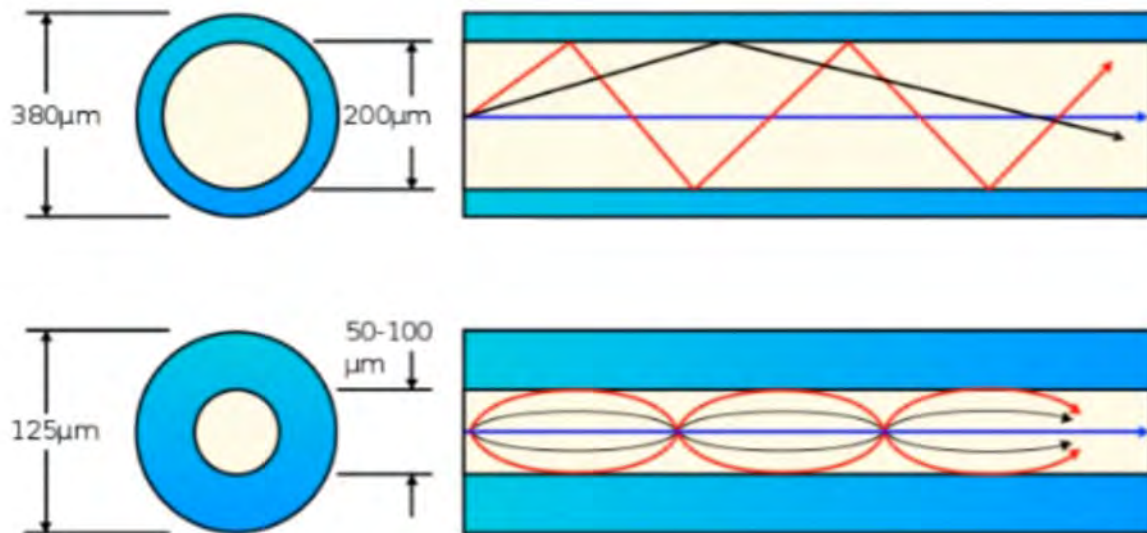
There are TWO types of fiberoptic cable.

Single-Mode:



- Narrower than multimode
- Lighter enters at a single angle (mode) from a laser-based transmitter.
- Allows longer cables than both UTP and multimode fiber.
- More expensive than multimode fiber (due to more expensive laser-based SFP transmitters)

Multimode:



- Core is wider than Single-mode
- Allows multiple angles (modes) of light waves to enter core
- Allows longer cables than UTP but shorter than single-mode
- Cheaper than single-mode fiber (due to cheaper LED-based SFP transmitter)

Fiber Optic Standards:

Speed	Standard	Connection Speed	Mode Support	Max Transmission Distance
1000BASE-LX	802.3z	1 Gbps	Multimode / Single	550 meters (Multi) / 5km (Single)
10GBASE-SR	802.3ae	10 Gbps	Multimode	400 meters
10GBASE-LR	802.3ae	10 Gbps	Single	10 kilometers
10GBASE-ER	802.3ae	10 Gbps	Single	30 kilometers

UTP vs Fiber-Optic Cabling:

UTP are:

- Lower cost than fiber-optic.
- Shorter maximum distance than fiber-optic (~100m).
- Can be vulnerable to EMI (Electromagnetic Interference).
- RJ45 ports used with UTP are cheaper than SFP ports.
- Emit (leak) a faint signal outside of cable, which can be copied (security risk).

Fiber-Optic:

- Higher cost than UTP.
- Longer maximum distance than UTP.
- No vulnerability to EMI.
- SFP ports are more expensive than RJ45 ports (single-mode is more expensive than multimode).
- Does not emit any signal outside of the cable (no security risk).

3. OSI MODEL & TCP/IP SUITE

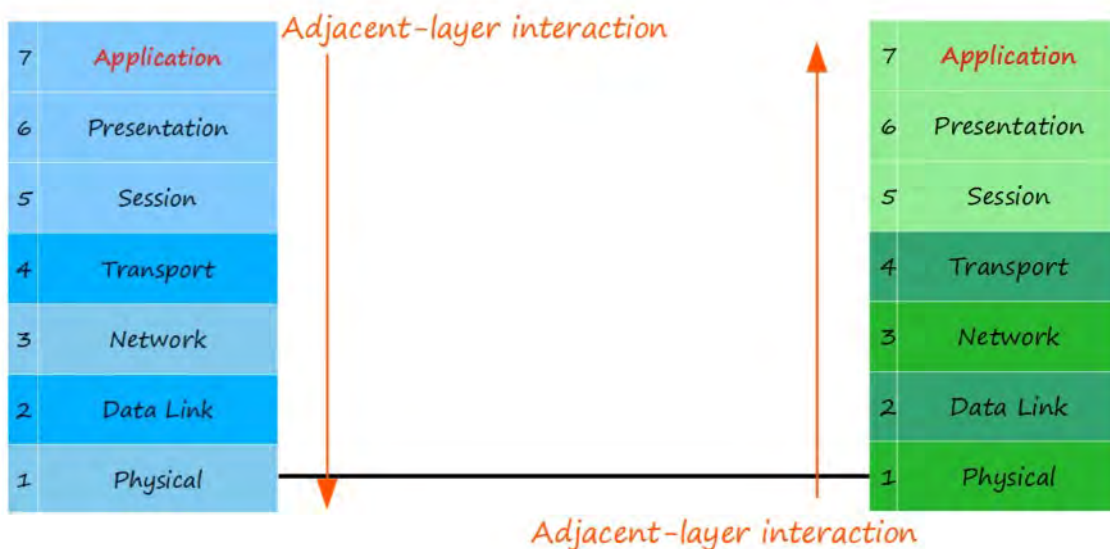
What is a networking model?

Networking models categorize and provide a structure for networking protocols and standards.
(Protocols are a set of logical rules defining how network devices and software should work)

OSI MODEL

- Open Systems Interconnection Model
- Conceptual model that categorizes and standardizes the different functions in a network.
- Created by the "International Organization for Standardization" (ISO)
- Functions are divided into 7 "Layers"
- These layers work together to make the network work.

OSI Model – Application Layer



As data moves from the top layer, downward, the process is called "encapsulation"

As data moves from the bottom layer, upward, the process is called "de-encapsulation"

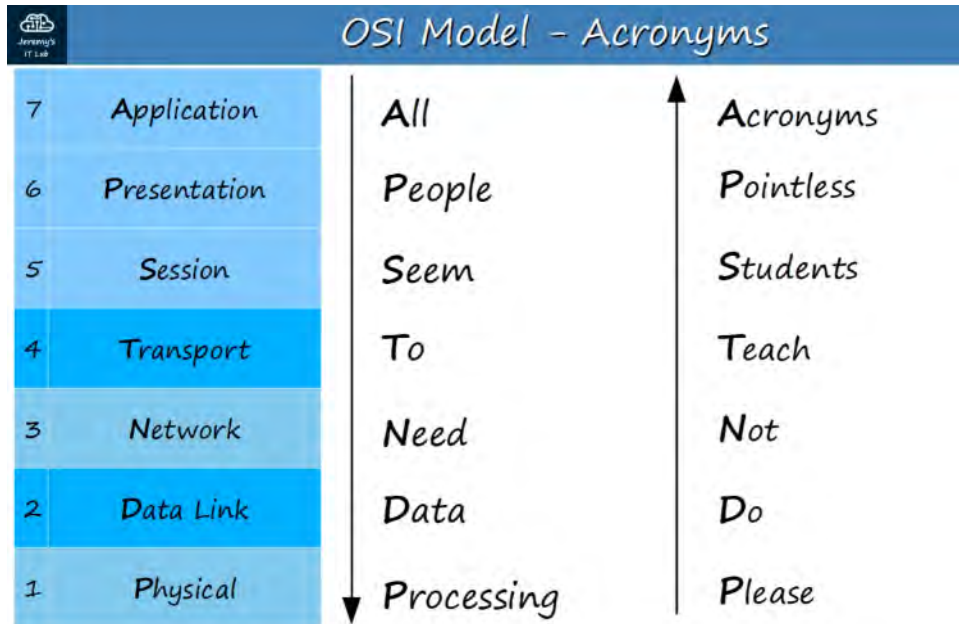
When interactions occur on the same layer, it's called "same-layer interaction"

OSI Model – The Upper Layers

7	Application
6	Presentation
5	Session
4	Transport
3	Network
2	Data Link
1	Physical

- Network engineers don't usually work with the top 3 layers.
- Application developers work with the top layers of the OSI model to connect their applications over networks.

Mnemonic to help remember the Data Layer Names / Order



The layers are :

7 - APPLICATION

- This Layer is closest to end user.
- Interacts with software applications.
- HTTP and HTTPS are Layer 7 protocols

Functions of Layer 7 include:

- Identifying communication partners
- Synchronizing communication

6 - PRESENTATION

- Translates data to the appropriate format (between Application and Network formats) to be sent over the network.

5 - SESSION

- Controls dialogues (sessions) between communicating hosts.
- Establishes, manages, and terminates connections between local application and the remote application.

Network engineers don't usually work with the top 3 layers. Application developers work with the top layers of the OSI model to connect their applications over networks.

4 - TRANSPORT

- Segments and reassembles data for communication between end hosts.
- Breaks large pieces of data into smaller segments which can be more easily sent over the network and are less likely to cause transmission problems if errors occur.
- Provides HOST-TO-HOST (end to end) communication

When Data from Layer 7-5 arrives, it receives a Layer 4 Header in the Transport layer.

<< DATA + L4 Header >>

This is called a SEGMENT.

3 - NETWORK

- Provides connectivity between end hosts on different networks (ie: outside of the LAN).
- Provides logical addressing (IP Addresses).
- Provides path selection between source and destination

- **ROUTERS** operate at Layer 3.

When Data and the Layer 4 Header arrive in the Network Layer, it receives a Layer 3 Header.

<< DATA + L4 Header + L3 Header >>

This is called a **PACKET**.

2 - DATA LINK

- Provides NODE-TO-NODE connectivity and data transfer (for example, PC to Switch, Switch to Router, Router to Router)
- Defines how data is formatted for transmission over physical medium (for example, copper UTP cables)
- Detects and (possibly) corrects Physical (Layer 1) errors.
- Uses Layer 2 addressing, separate from Layer 3 addressing.
- **SWITCHES** operate at Layer 2

When the Layer 3 Packet arrives, a Layer 2 Trailer and Header are added.

<< L2 Trailer + DATA + L4 Header + L3 Header + L2 Header >>

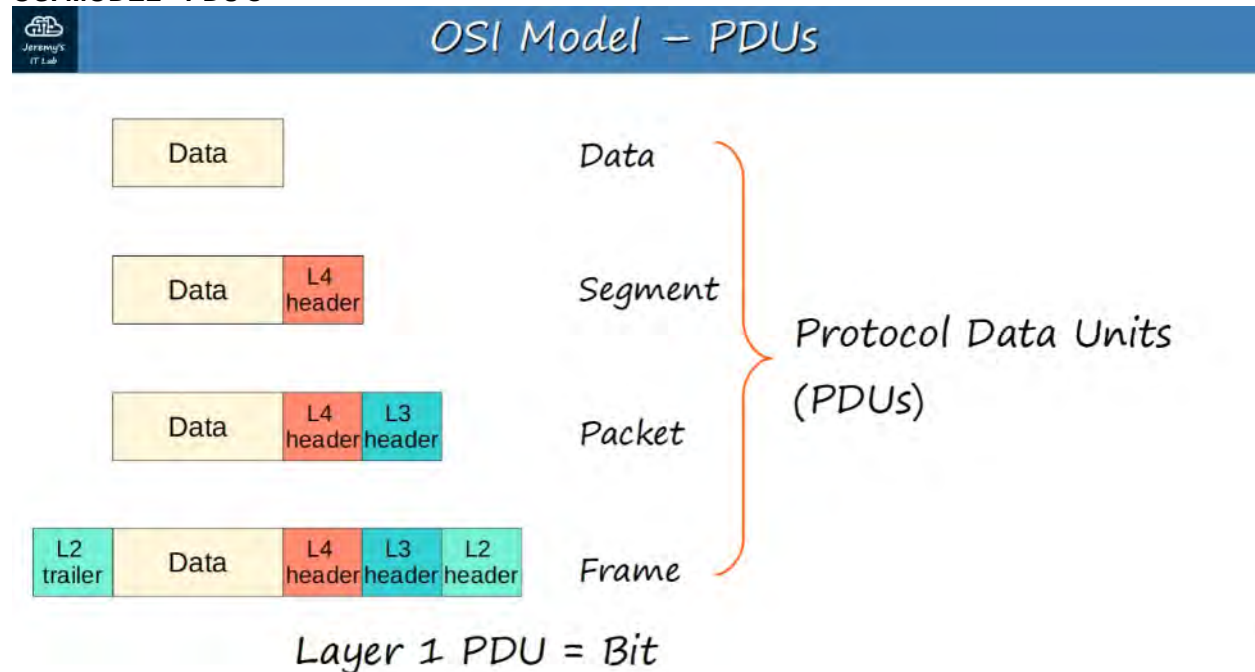
This is called a **FRAME**.

All the steps leading up to transmission is called **ENCAPSULATION**. When the frame is sent to the receiver, it then goes through the reverse process, **DE-ENCAPSULATION**, stripping off layers while travelling from OSI Layer 1 to Layer 7.

1 - PHYSICAL

- Defines physical characteristics of the medium used to transfer data between devices. For example : voltage levels, maximum transmission distances, physical connectors, cable specs.
- Digital bits are converted into electrical (for wired connections) or radio (for wireless connections) signals.
- All of the information in SECTION 2 (NETWORKING DEVICES) is related to the Physical Layer

OSI MODEL - PDU's



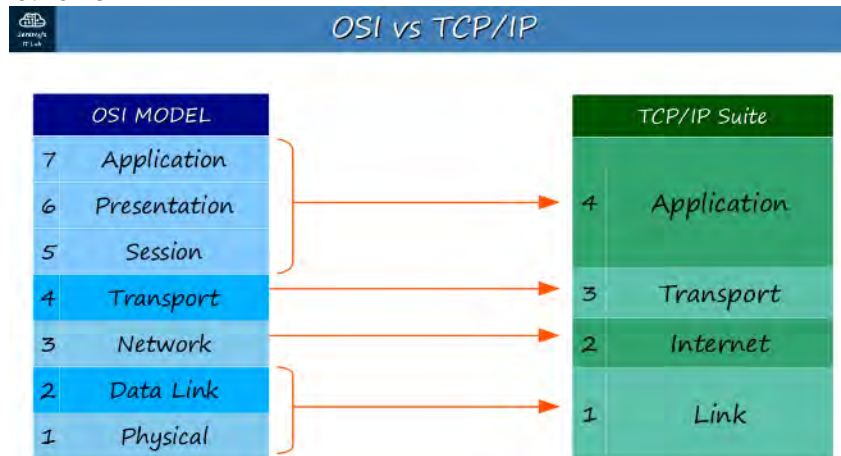
A PDU is a Protocol Data Unit. Each step of the process is a PDU.

OSI LAYER #	PDU NAME	PROTOCOL DATA ADDED
7-5	DATA	Data

OSI LAYER #	PDU NAME	PROTOCOL DATA ADDED
4	SEGMENT	Layer 4 Header Added
3	PACKET	Layer 3 Header Added
2	FRAME	Layer 2 Trailer and Header Added
1	BIT	0s and 1s Transmission
<< L2 Trailer + DATA + L4 Header + L3 Header + L2 Header >>		

TCP/IP Suite

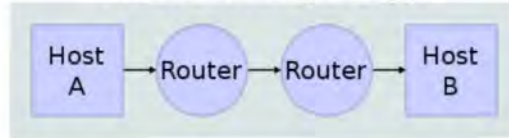
- Conceptual model and set of communications protocols used in the Internet and other networks.
- Known as TCP/IP because those are two of the foundational protocols in the suite.
- Developed by the US Dept. of Defense through DARPA (Defense Advanced Research Projects Agency).
- Similar structure to the OSI Model, but fewer layers.
- THIS is the model actually in use in modern networks.
- - Note : The OSI Model still influences how network engineers think and talk about networks.



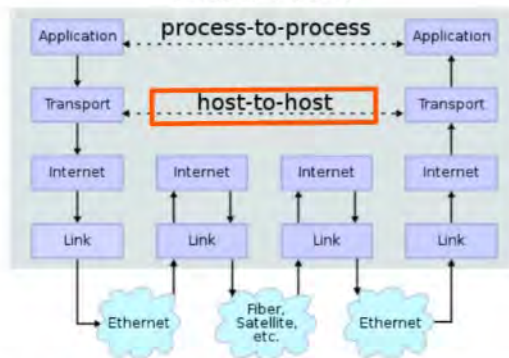
Layer Interactions

TCP/IP Suite

Network Topology



Data Flow



en:User:Kibrose (https://commons.wikimedia.org/wiki/File:IP_stack_connections.svg), „IP stack connections“, <https://creativecommons.org/licenses/by-sa/3.0/legalcode>

Adjacent-Layer Interactions:

- Interactions between different layers of the OSI Model on same host.

Example:

Layers 5-7 sending Data to Layer 4, which then adds a Layer 4 header (creating a SEGMENT).

Same-Layer Interactions:

- Interactions between the same Layer on different hosts.
- The concept of Same-Layer interaction allows you to ignore the other layers involved and focus on the interactions between a single layer on different devices.

Example:

The Application Layer of YouTube's web server and your PC's browser.

4. INTRO TO THE CLI

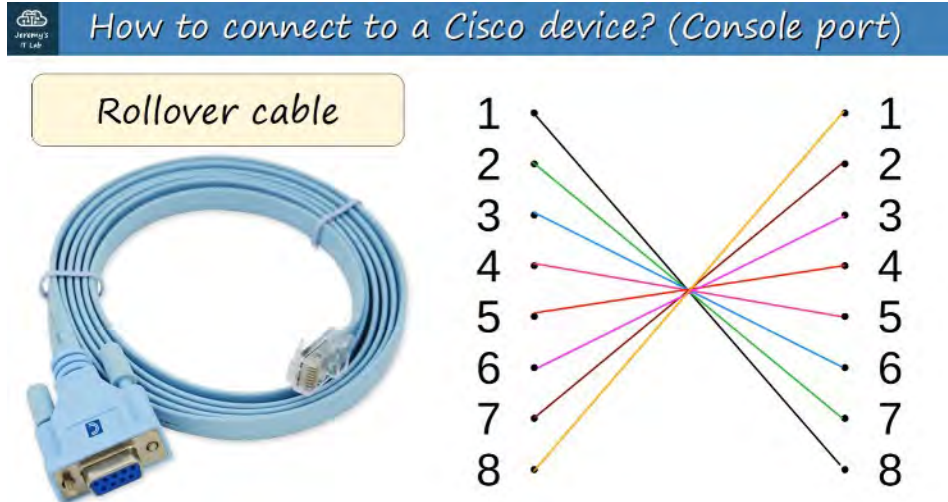
What is a CLI?

- A "Command-line Interface"
- The interface you use to configure Cisco devices

A GUI is a "Graphical User Interface"

How do you connect to a Cisco Device?

- Console Port : When you first configure a device, you have to connect via the Console Port. You can use a "Rollover cable" : DB9 serial connector to RJ45 OR a DB9 Serial to USB



How do you actually access the CLI?

- You need to use a TERMINAL EMULATOR (Example: PuTTY is a popular choice) and connect via "Serial" (default settings)

Cisco Default Settings are:

Speed (baud) : 9600 bits/second Data bits: 8 data bits Stop bits: 1 stop bit (sent after 8 data bits are sent)
Parity: None Flow Control: None

When you first enter the CLI you will DEFAULT be in what is called 'User EXEC' mode.

USER EXEC MODE:

(Hostname) > // Prompt looks like THIS //

- User EXEC mode is very limited.
- User can look at some things but can't make ANY changes to the configuration.
- AKA 'User Mode'

Using the 'enable' command, in User EXEC mode, switches you to 'Privileged EXEC' mode.

PRIVILEGED EXEC MODE:

- Provides complete access to view the device's configuration, restart the device, etc.
- Cannot change the configuration, but can change the time on the device, save the configuration file, etc.

(Hostname)# // Prompt looks like THIS //

USE a Question Mark (?) to view the available commands in ANY mode. Combining ? with a letter or partial command will list all the commands with those letters.

User EXEC Mode

```
Router>?
Exec commands:
<1-99> Session number to resume
connect Open a terminal connection
disable Turn off privileged commands
disconnect Disconnect an existing network connection
enable Turn on privileged commands
exit Exit from the EXEC
logout Exit from the EXEC
ping Send echo messages
resume Resume an active network connection
show Show running system information
ssh Open a secure shell client connection
telnet Open a telnet connection
terminal Set terminal line parameters
traceroute Trace route to destination
Router>
```

Privileged EXEC Mode

```
Router#?
Exec commands:
<1-99> Session number to resume
auto Exec level Automation
clear Reset functions
clock Manage the system clock
configure Enter configuration mode
connect Open a terminal connection
copy Copy from one file to another
debug Debugging functions (see also 'undebg')
delete Delete a file
dir List files on a filesystem
disable Turn off privileged commands
disconnect Disconnect an existing network connection
enable Turn on privileged commands
erase Erase a filesystem
exit Exit from the EXEC
logout Exit from the EXEC
mkdir Create new directory
more Display the contents of a file
no Disable debugging informations
ping Send echo messages
reload Halt and perform a cold restart
resume Resume an active network connection
rmdir Remove existing directory
send Send a message to other tty lines
setup Run the SETUP command facility
show Show running system information
ssh Open a secure shell client connection
telnet Open a telnet connection
terminal Set terminal line parameters
traceroute Trace route to destination
undebg Disable debugging functions (see also 'debug')
vlan Configure VLAN parameters
write Write running configuration to memory, network, or terminal
Router#
```

USE the TAB key to complete partially entered commands IF the command exists.

GLOBAL CONFIGURATION MODE:

To enter Global Configuration Mode, enter the command, within Privileged EXEC mode

'configure terminal' (or 'conf t')

Router# configure terminal Router(config) #

Router(config) # run

Router(config) # no

Type 'exit' to drop back into 'Privileged EXEC' mode.

To Enable Password for User EXEC mode:

Router(config)# enable password (password)

- Passwords ARE case-sensitive.

// This command encrypts plain-text passwords, visible in the config files, using simple encryption.

Router(config)# service password-encryption

If you enable 'service password-encryption'

- Current passwords WILL be encrypted.
- Future passwords WILL be encrypted.
- The 'enable secret' WILL NOT be effected.

If you disable 'service password-encryption'

- Current passwords WILL NOT be decrypted.
- Future passwords WILL NOT be encrypted.
- The 'enable secret' WILL NOT be effected.

// This command enables passwords for the Privileged EXEC mode.

Router(config)# enable secret (password)

// enable secret will ALWAYS be encrypted (at level 5)

There are TWO separate configuration files kept on the device at once.

Running-config :

- The current, ACTIVE configuration file on the device. As you enter commands in the CLI, you edit the active configuration.

Startup-config :

- The configuration file that will be loaded upon RESTART of the device.

To see the configuration files, inside 'Privileged EXEC' mode:

Router# show running-config // for running config //

OR

Router# show startup-config // for startup config //

To SAVE the Running configuration file, you can:

Router# write Building configuration... [OK]

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

Router# copy running-config startup-config

Destination filename [startup-config]?

Building configuration... [OK]

To encrypt passwords:

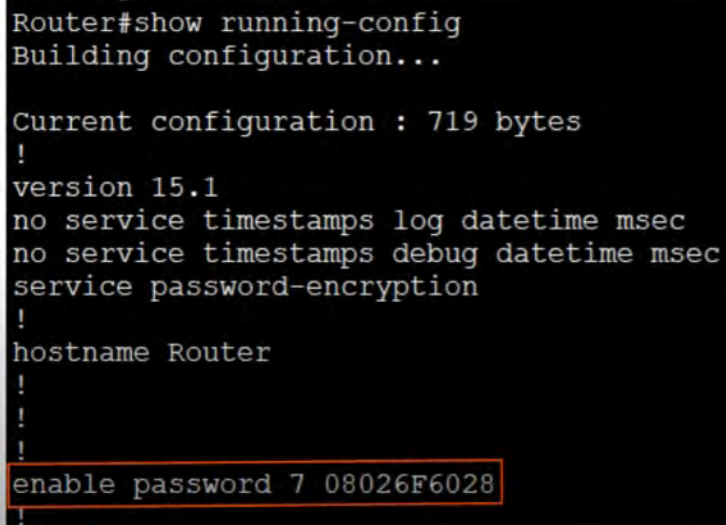
Router# conf t

Router(config)# service password-encryption

This makes all current passwords *encrypted*

Future passwords will ALSO be *encrypted*

"Enable secret" will not be effected (it's ALWAYS encrypted)



```
Router#show running-config
Building configuration...

Current configuration : 719 bytes
!
version 15.1
no service timestamps log datetime msec
no service timestamps debug datetime msec
service password-encryption
!
hostname Router
!
!
!
enable password 7 08026F6028
!
```

Now you will see that the password is no longer in plaintext.

"7" refers to the type of encryption used to encrypt the password. In this case, "7" uses Cisco's proprietary encryption.

"7" is fairly easy to crack since the encryption is weak.

For BETTER / STRONGER encryption, use "enable secret"

```
Router(config)#enable secret Cisco
Router(config)#do sh run
Building configuration...

Current configuration : 766 bytes
!
version 15.1
no service timestamps log datetime msec
no service timestamps debug datetime msec
service password-encryption
!
hostname Router
!
!
!
enable secret 5 $1$mERr$YlCkLMcTYWwkF1Cndt1l.
enable password 7 08026F6028
```

“5” refers to MD5 encryption.

Can still be cracked but it's much much stronger.

Once you use “enable secret” command, this will override “enable password”

To CANCEL or delete a command you entered, use the “no” keyword

```
Router(config)#no service password-encryption
Router(config)#do sh run
Building configuration...

Current configuration : 769 bytes
!
version 15.1
no service timestamps log datetime msec
no service timestamps debug datetime msec
no service password-encryption
!
hostname Router
!
!
!
enable secret 5 $1$mERr$YlCkLMcTYWwkF1Cndt1l.
enable password 7 08026F6028
!
```

In this instance, disabling “service password-encryption”:

- current passwords will NOT be decrypted (unchanged)
- future passwords will NOT be encrypted
- the “enable secret” will not be effected



Modes Review

Router> = user EXEC mode

Router# = privileged EXEC mode

Router(config)# = global configuration mode



Command Review

Router>**enable**
##used to enter privileged EXEC mode

Router#**configure terminal**
##used to enter global configuration mode

Router(config)#**enable password** password
##configures a password to protect privileged exec mode



Command Review

Router(config)#**service password-encryption**
##encrypts the enable password (and other passwords)

Router(config)#**enable secret** password
##configures a more secure, always-encrypted enable password

Router(config)#**run** privileged-exec-level-command
##executes a privileged-exec level command from global configuration mode



Command Review

Router(config)#**no** command
##removes the command

Router(config)#**show running-config**
##displays the current, active configuration file

Router(config)#**show startup-config**
##displays the saved configuration file which will be loaded if the device is restarted



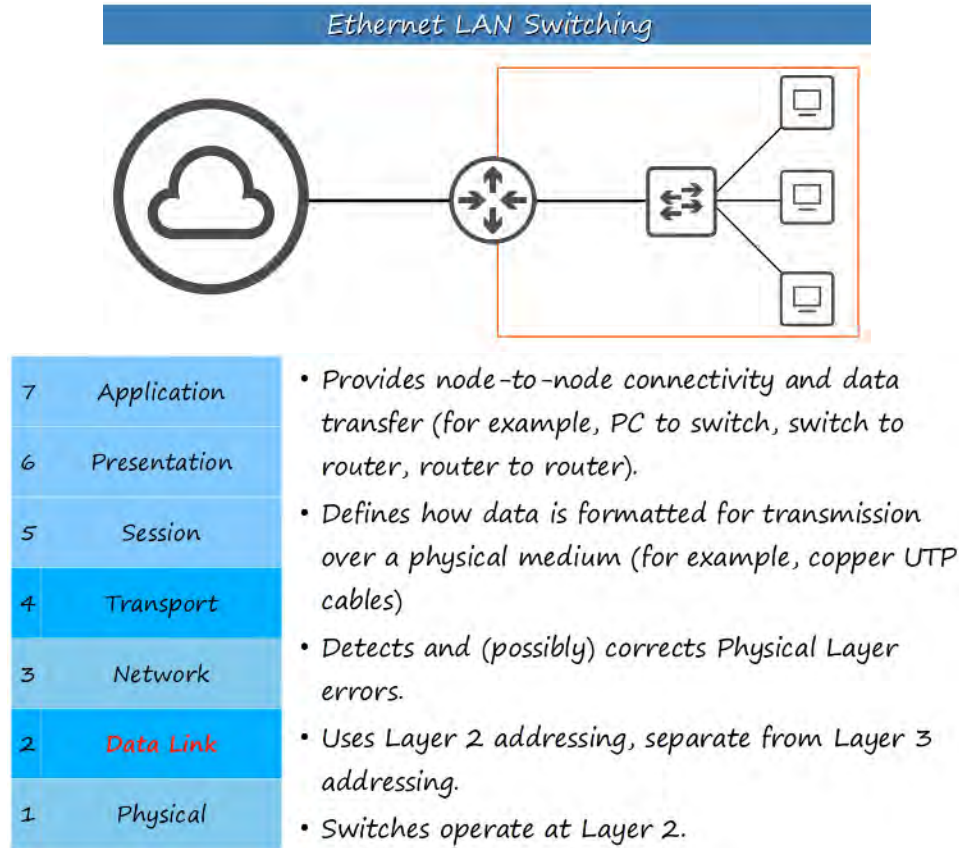
Command Review

```
Router(config)#write  
##saves the configuration
```

```
Router(config)#write memory  
##saves the configuration
```

```
Router(config)#copy running-config startup-config  
##saves the configuration
```

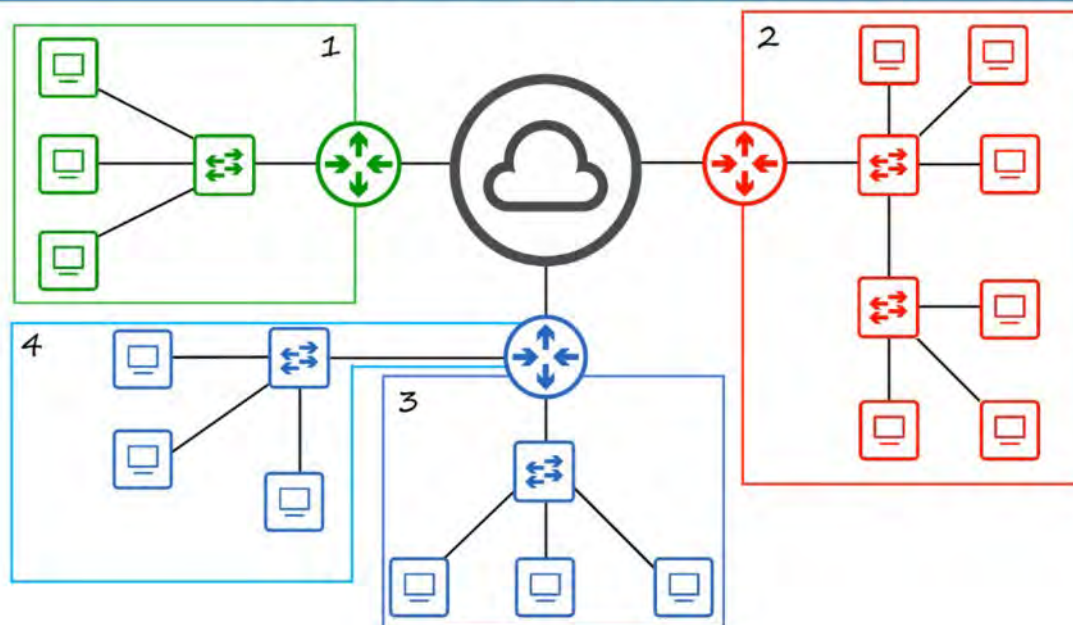
5. ETHERNET LAN SWITCHING : PART 1



LAN's

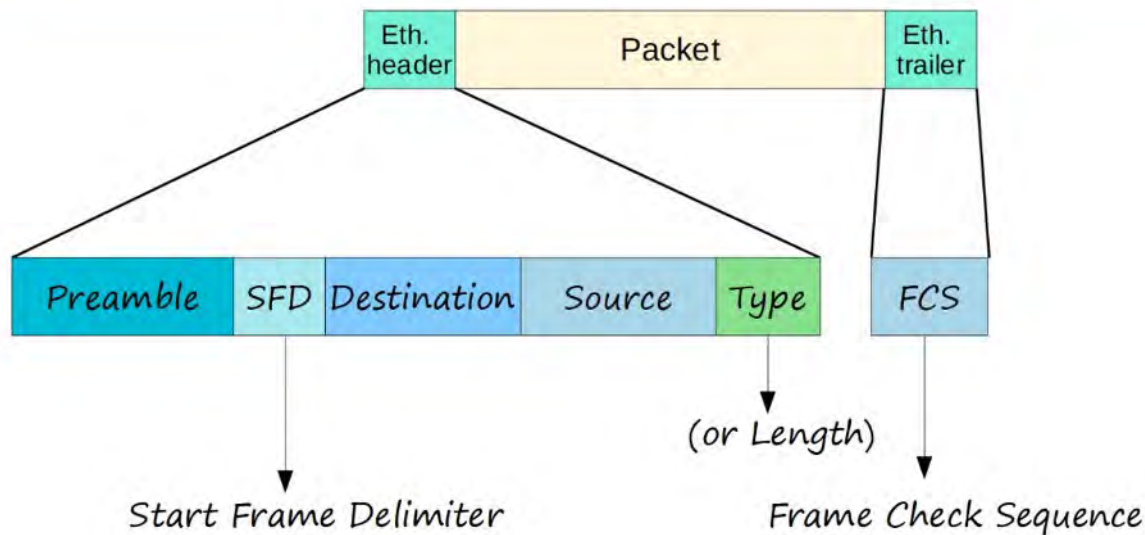
- A LAN is a network contained in a relatively small area.
- Routers are used to connect separate LAN's

Local Area Networks (LANs)



An ETHERNET FRAME looks like:

Ethernet Frame



Ethernet Trailer --- PACKET --- Ethernet Header

The Ethernet Header contains 5 Fields:

Preamble -- SFD -- Destination -- Source -- Type 7 bytes -- 1 byte -- 6 bytes -- 6 bytes -- 2 bytes

PREAMBLE:

- Length: 7 bytes (56 bits)
- Alternating 1's and 0's
- 10101010 * 7x
- Allows devices to synchronize their receiver clocks

SFD : 'Start Frame Delimiter'

- Length: 1 byte(8 bits)
- 10101011
- Marks end of the PREAMBLE and beginning of rest of frame.

DESTINATION AND SOURCE

- Layer 2 Address
- Indicates the devices sending / receiving the frame
- MAC = 'Media Access Control'
- = 6 byte (48-bit) address of the physical device

TYPE / LENGTH

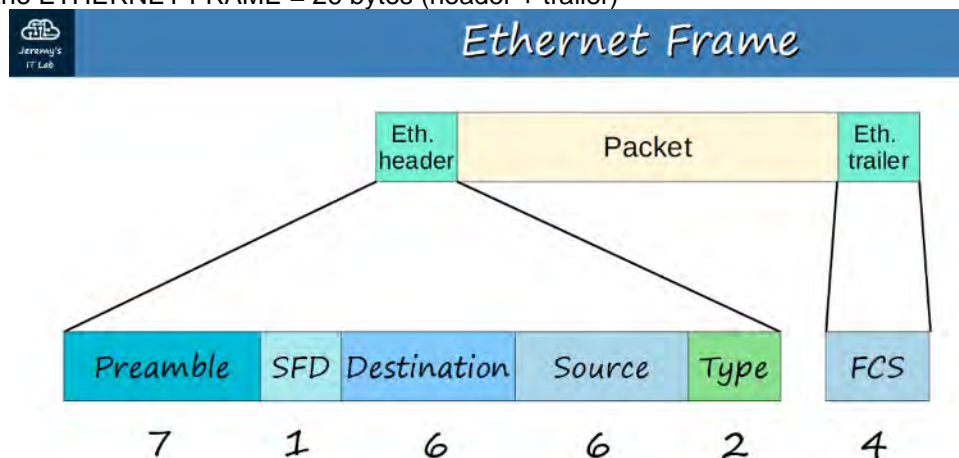
- 2 bytes (16-bit) field
- A value of **1500 or less** in this field indicates the LENGTH of the encapsulated packet (in bytes)
- A value of **1536 or greater** in this field indicates the TYPE of the encapsulated packet and length is determined via other methods.
- IPv4 = 0x0800 (hexadecimal) = 2048 in decimal
- IPv6 = 0x86DD (hexadecimal) = 34525 in decimal
- Layer 3 protocol used in the encapsulated Packet, which is almost always Internet Protocol (IP) version 4 or version 6.

The ETHERNET TRAILER contains:

FCS

- 'FRAME CHECK SEQUENCE'
- 4 bytes (32 bits) in length
- Detects corrupted data by running a 'CRC' algorithm over the received data
- CRC = "Cyclic Redundancy Check"

Altogether the ETHERNET FRAME = 26 bytes (header + trailer)



= 26 bytes (header + trailer)

MAC ADDRESS (48 bits long)

- 6-bytes (48-bits) physical address assigned to the device when it is made.
- AKA 'Burned-In Address' (BIA)
- Is globally unique

- First 3 bytes are the OUI (Organizationally Unique Identifier) which is assigned to the company making the device
- The last 3 bytes are unique to the device itself
- Written as 12 hexadecimal characters

Example:

E8:BA:70 // 11:28:74 OUI // Unique Device ID

HEXADECIMAL

Hexadecimal

DEC.	HEX.	DEC.	HEX.	DEC.	HEX.	DEC.	HEX.
0	0	8	8	16	10	24	18
1	1	9	9	17	11	25	19
2	2	10	A	18	12	26	1A
3	3	11	B	19	13	27	1B
4	4	12	C	20	14	28	1C
5	5	13	D	21	15	29	1D
6	6	14	E	22	16	30	1E
7	7	15	F	23	17	31	1F

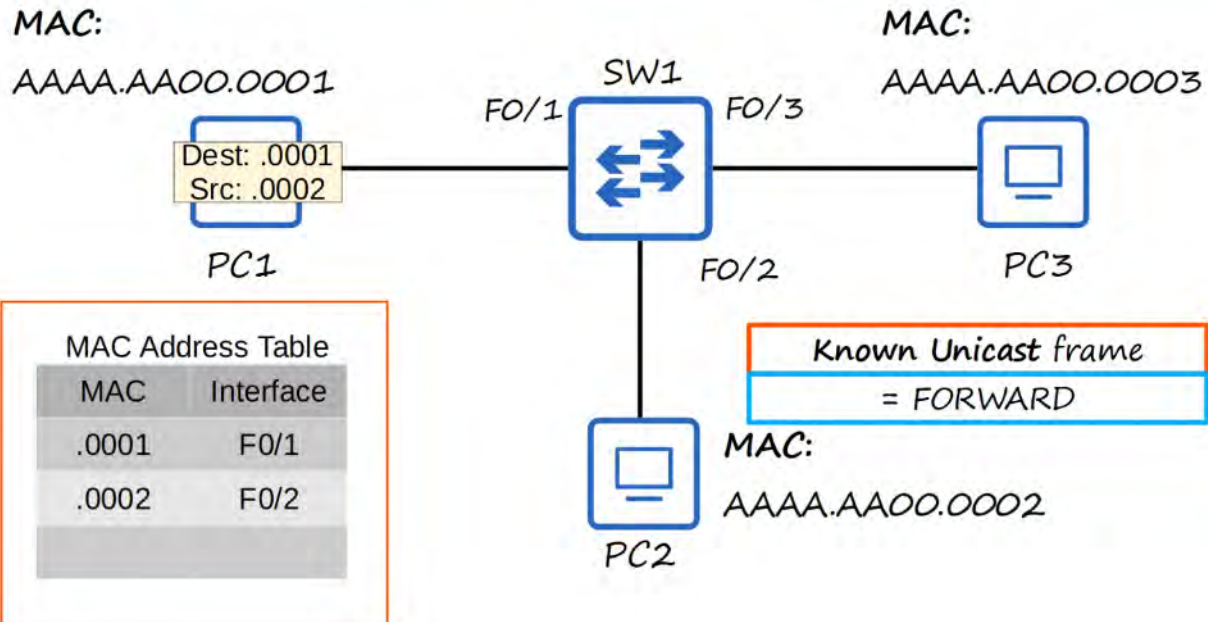
INTERFACE NAMES

F0/1, F0/2, F0/3... F stands for "Fast Ethernet" or 100 Mbps interfaces.

MAC ADDRESS TABLE

Each Switch stores a DYNAMICALLY LEARNED MAC ADDRESS TABLE, using the SOURCE MAC ADDRESS of frames it receives.

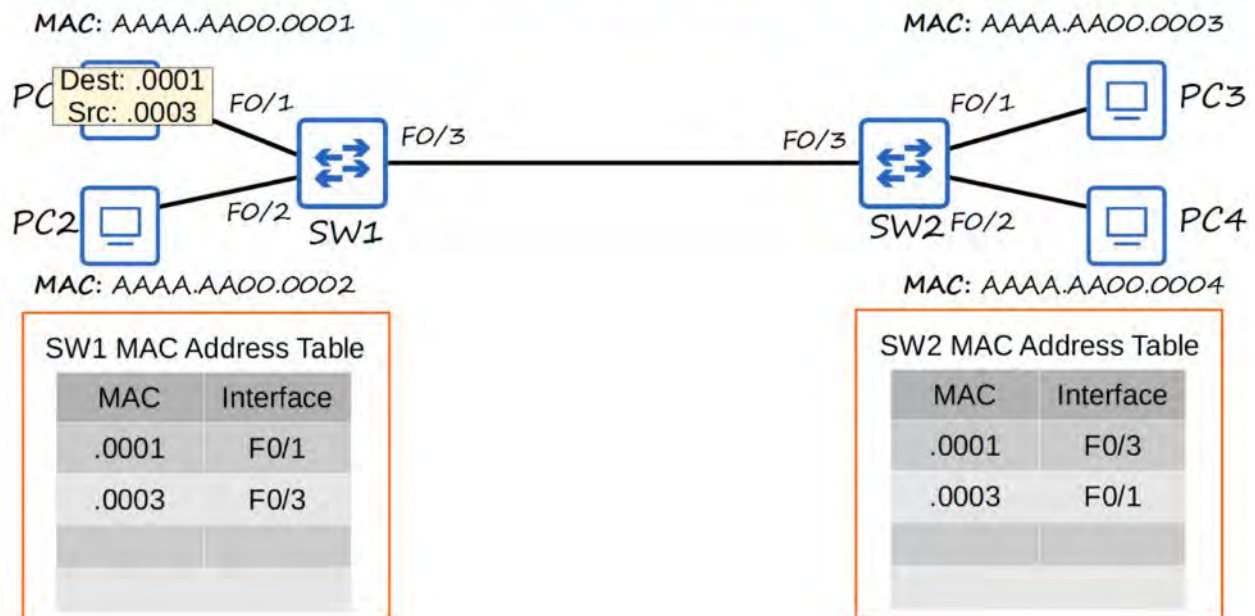
MAC Addresses



When a Switch doesn't know the DESTINATION MAC ADDRESS of a frame (UNKNOWN UNICAST FRAME), it is forced to FLOOD the frame - Forward the frame out of ALL it's interfaces, except the one it received the packet from.

When a KNOWN Unicast Frame is known (MAC Address is recognized by the entry in the MAC ADDRESS TABLE), the frame is FORWARDED like normal.

MAC Addresses

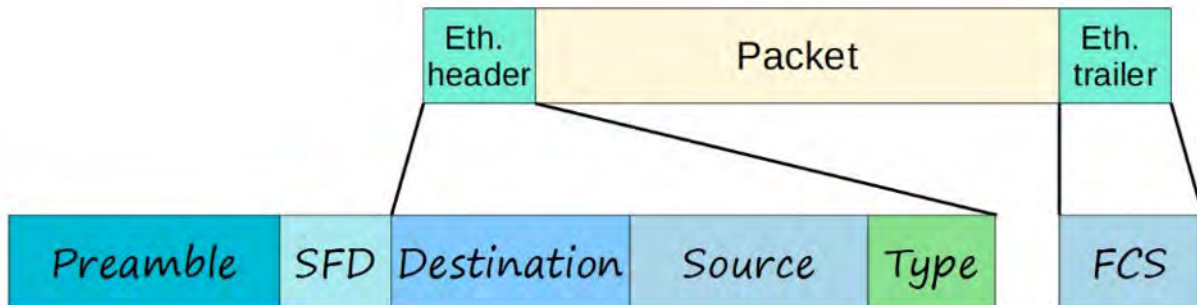


- Note: Dynamic MAC Addresses are removed from the MAC ADDRESS TABLE every 5 minutes of inactivity.

6. ETHERNET LAN SWITCHING : PART 2

An ETHERNET FRAME looks like:

Ethernet Header --- DATA (Packet) --- Ethernet Trailer



The Ethernet Header contains 5 Fields:

Preamble -- SFD -- Destination -- Source -- Type/Length 7 bytes -- 1 byte -- 6 bytes -- 6 bytes -- 2 bytes

Ethernet Trailer contains 1 Field:

FCS (Frame Check Sequence) = 4 bytes

- The PREAMBLE + SFD is not usually considered part of the ETHERNET HEADER.

THEREFORE the size of the ETHERNET HEADER + TRAILER is 18 bytes

(6 + 6 + 2 + 4 bytes for the FRAME CHECK SEQUENCE)

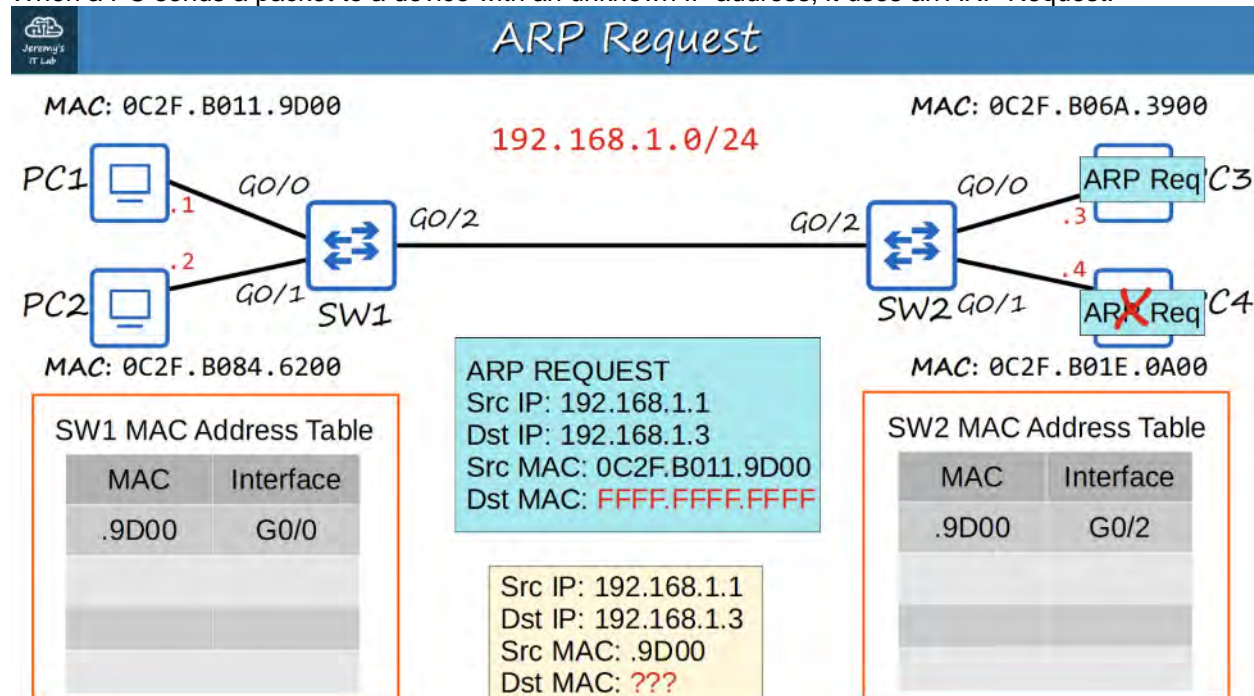
The MINIMUM size for an ETHERNET FRAME (Header + Payload [PACKET] + Trailer) is 64 BYTES.

64 BYTES - 18 BYTES (Header + Trailer size) = 46 BYTES

THEREFORE the MINIMUM DATA PAYLOAD (PACKET) size is 46 BYTES!

IF the PAYLOAD is LESS than 46 BYTES then PADDING BYTES are added (padding bytes are a series of 0's) until it equals to 46 BYTES.

When a PC sends a packet to a device with an unknown IP address, it uses an ARP Request.



- ARP stands for 'Address Resolution Protocol'.

- It is used to discover the Layer 2 address (MAC address) of a known Layer 3 address (IP address)
- Consists of two messages:
 - ARP REQUEST (Source message)
 - ARP REPLY (Destination message)
- ARP REQUEST is BROADCAST = sent to all hosts on network, except the one it received the request from.

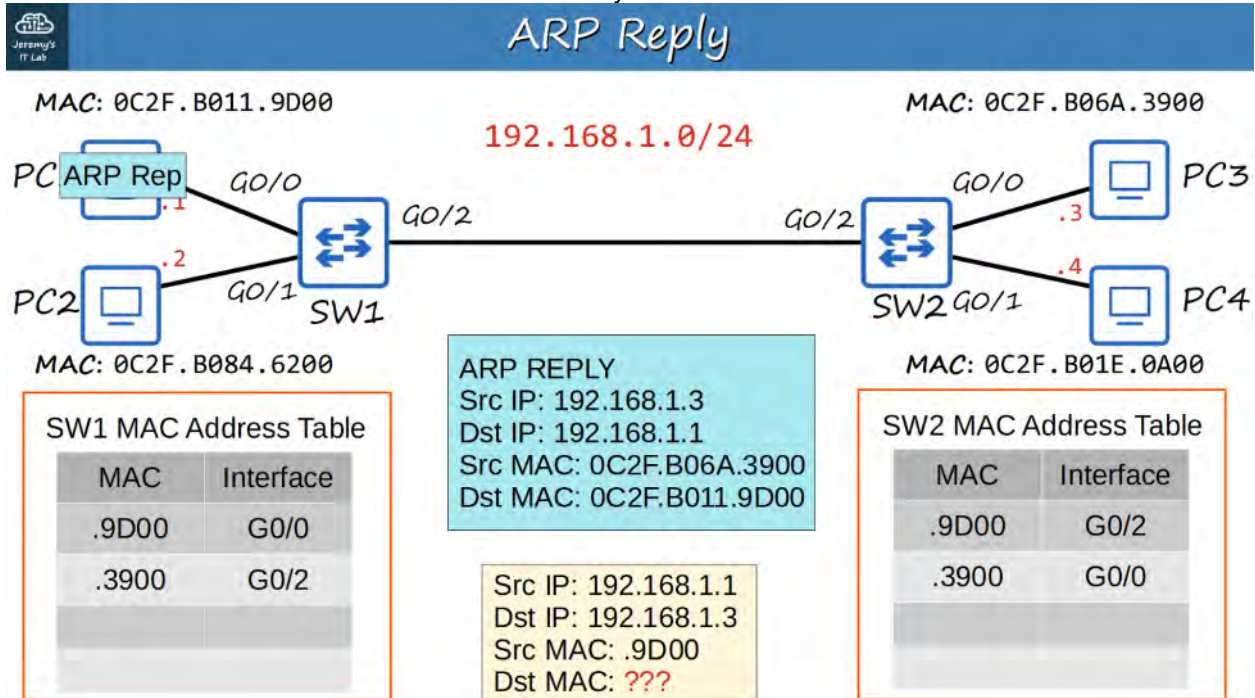
An ARP REQUEST frame has:

- Source IP Address
- Destination IP Address
- Source MAC address
- BROADCAST MAC Address - FFFF.FFFF.FFFF

An ARP REPLY frame has:

- Source IP Address
- Destination IP Address
- Source MAC address
- Destination MAC Address

ARP REPLY is a known UNICAST frame = Sent only to the host that sent the ARP REQUEST.



PING


- A network utility that is used to test reachability
- Measures round-trip time
- Uses two messages:
 - ICMP Echo REQUEST
 - ICMP Echo REPLY
- Is UNICAST
- Command to use ping:
 - ping

By Default, a CISCO IOS sends 5 ICMP requests/replies (Default size is 100-bytes)

- A period (.) is a failed ping
- An exclamation mark (!) is a successful ping

USEFUL CISCO IOS COMMANDS (from Privileged EXEC mode)

PC1# show arp // shows hosts ARP table



ARP Table


```
C:\Users\user>arp -a

Interface: 169.254.146.29 --- 0x9
Internet Address      Physical Address      Type
169.254.255.255       ff-ff-ff-ff-ff-ff    static
224.0.0.2             01-00-5e-00-00-02    static
224.0.0.22            01-00-5e-00-00-16    static
224.0.0.251           01-00-5e-00-00-fb    static
224.0.0.252           01-00-5e-00-00-fc    static
239.255.255.250       01-00-5e-7f-ff-fa    static
255.255.255.255       ff-ff-ff-ff-ff-ff    static

Interface: 192.168.0.167 --- 0xd
Internet Address      Physical Address      Type
192.168.0.1           98-da-c4-dd-a8-e4    dynamic
192.168.0.255         ff-ff-ff-ff-ff-ff    static
224.0.0.2             01-00-5e-00-00-02    static
224.0.0.22            01-00-5e-00-00-16    static
224.0.0.251           01-00-5e-00-00-fb    static
224.0.0.252           01-00-5e-00-00-fc    static
239.255.255.250       01-00-5e-7f-ff-fa    static
255.255.255.255       ff-ff-ff-ff-ff-ff    static
```

- Use arp -a to view the ARP table (Windows, macOS, Linux)
- Internet Address = IP address (Layer 3 address)
- Physical Address = MAC address (Layer 2 address)
- Type static = default entry
- Type dynamic = learned via ARP

SW1#show mac address-table // show the switches MAC table



MAC Address Table

```
SW1#show mac address-table
Mac Address Table
-----
Vlan    Mac Address      Type        Ports
----    -
1       0c2f.b011.9d00   DYNAMIC     Gi0/0
1       0c2f.b06a.3900   DYNAMIC     Gi0/2
Total Mac Addresses for this criterion: 2
SW1#
```

Will show:

Vlan --- MAC Address --- Type --- Ports(interfaces)
(Vlan = Virtual Local Area Network)

Clearing the MAC Address Table

```
SW1#show mac address-table
      Mac Address Table
```

```
-----
Vlan    Mac Address      Type    Ports
```

```
clear mac address-table dynamic
```

```
Total Mac Addresses for this criterion: 2
SW1#clear mac address-table dynamic
```

```
SW1#show mac address-table
      Mac Address Table
```

```
-----
Vlan    Mac Address      Type    Ports
```

```
-----
SW1#
```

SW1# clear mac address-table dynamic

// clears the entire switches MAC table. // IF the optional MAC address is used, it will clear the SPECIFIC MAC address.

SW1 #clear mac address-table dynamic interface

// clears the MAC table entry of the Switch by it's **INTERFACE** name.

7. IPv4 ADDRESSING : PART 1

OSI MODEL - NETWORK LAYER (Layer 3)

- Provides connectivity between end hosts on DIFFERENT networks (ie: outside of the LAN)
- Provides logical addressing (IP addresses)
- Provides path selection between SOURCE and DESTINATION
- ROUTERS operate at LAYER 3

ROUTING

SWITCHES (Layer 2 Devices) do not separate different networks. They connect and EXPAND networks within the same LAN.

By adding a ROUTER, however, between two SWITCHES, you create a SPLIT in the network; each with its own network IP address.

Example: 192.168.1.0/24 (255.255.255.0) 192.168.2.0/24 (255.255.255.0)



ROUTERS have unique IP Addresses for EACH of their interface connections, depending on their location.

The IP Address for the ROUTER's G0/0 Interface is: 192.168.1.254/24

The IP Address for the ROUTER's G0/1 Interface is: 192.168.2.254/24



The IP Address depends on network address of the LAN it is connects to.

The NETWORK portion of given IP Address will be the same for all HOSTS on a given LAN.

Example:

192.168.1.100 192.168.1.105 192.168.1.205

All of these addresses are on the SAME Network because the NETWORK PORTION of their IP Address is the same (192.168.1) while the HOST part (100,105,205) is UNIQUE!

When a BROADCAST message hits a ROUTER, it does NOT continue onward. It stays within the LOCAL LAN (Switch/Hosts).

IPv4 HEADER

IPv4 Header

IPv4 Header Format																																	
Offsets	Octet	0								1								2								3							
Octet	Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
0	0	Version				IHL				DSCP				ECN				Total Length															
4	32	Identification																Flags				Fragment Offset											
8	64	Time To Live								Protocol								Header Checksum															
12	96	Source IP Address																															
16	128	Destination IP Address																															
20	160	Options (if IHL > 5)																															
24	192																																
28	224																																
32	256																																

IP (or Internet Protocol) is the primary Layer 3 protocol in use today. Version 4 is the version in use in most networks.

IPv4 Headers contain MORE fields than the ETHERNET header.

IPv4 Headers contain a SOURCE IP Address and DESTINATION IP Address field.

This FIELD is 32-bits(4-bytes) in length (0-31)

192.168.1.254 (each decimal number represents 8 bits)

Translated to Binary:

11000000 . 10101000 . 00000001 . 11111110

EACH of these 8 bit groups are referred to as an OCTET

Since Binary is difficult to read for people, we use the Dotted Decimal format.

REVIEW of DECIMAL and HEXADECIMAL

Decimal & Hexadecimal

Decimal
(base 10)

3

$3 * 1000$

2

$2 * 100$

9

$9 * 10$

4

$4 * 1$

Hexadecimal
(base 16)

C

$C * 256$
(C = 12)

3072

D

$D * 16$
(D = 13)

208

E

$E * 1$
(E = 14)

14

Decimal (base 10)

Ex: $3294 = (3 * 1000) + (2 * 100) + (9 * 10) + (4 * 1)$

Hexadecimal (base 16)

Ex: 3294, would be CDE

C ($C * 256 / 12 * 256 = 3072$) // 256ths position

D ($D * 16 / D=13$ so $16*13 = 208$) // 16ths position

E ($E * 1 / E = 14$) // 1s position

Adding these up, we get 3294

So, how do we convert a BINARY NUMBER to a DECIMAL NUMBER? The same way we convert to Hexadecimal.

10001111

So:

$$1 * 128 = 128$$

$$1 * 8 = 8$$

$$1 * 4 = 4$$

$$1 * 2 = 2$$

$$1 * 1 = 1$$

Add them all up : $128 + 8 + 4 + 2 + 1 = 143$

The answer is 143.

Another example:

01110110

$$1 * 64 = 64$$

$$1 * 32 = 32$$

$$1 * 16 = 16$$

$$1 * 4 = 4$$

$$1 * 2 = 2$$

Add them all up: $64 + 32 + 16 + 4 + 2 = 118$

The answer is 118.

Another example:

11101100

$$1 * 128 = 128$$

$$1 * 64 = 64$$

$$1 * 32 = 32$$

$$1 * 8 = 8$$

$$1 * 4 = 4$$

Add them all up: $128 + 64 + 32 + 8 + 4 = 236$

The answer is 236.

So, how do we convert a DECIMAL NUMBER to a BINARY NUMBER?

Take the number 221.

We can take that number and start subtracting it from LEFT to RIGHT of our Binary slots.

221

$221 - 128 = 93$ so we place a 1 in the "128" slot

10000000

$93 - 64 = 29$ so we place another 1 in the "64" slot

$29 - 32$ isn't possible so we place a 0 in the "32" slot

$29 - 16 = 13$ so we place a 1 in the "16" slot

$13 - 8 = 5$ so we place a 1 in the "8" slot

$5 - 4 = 1$ so we place a 1 in the "4" slot

$1 - 2$ isn't possible so we put a 0 in the "2" slot

$1 - 1$ is possible so we put a 1 in the "1" slot

This, then, allows us to write out the BINARY number for 221.

It is : 11011101

Another example: 127

127 - 128 is not possible so 0 in "128"
127 - 64 is possible so 1 in "64"
63 - 32 is possible so 1 in "32"
31 - 16 is possible so 1 in "16"
15 - 8 is possible so 1 in "8"
7 - 4 is possible so 1 in "4"
3 - 2 is possible so 1 in "2"
1 is possible so 1 in "1"
So 127, in BINARY, is 0111 1111

Another example: 207

Alternatively, you can subtract the number from '255' (which is 1111111). The remainder, then, can be used to "find" where the 0's are in the binary number.

255 - 207 = 48 so ...

1 1 0 0 1 1 1 1 (32 + 16 = 48)

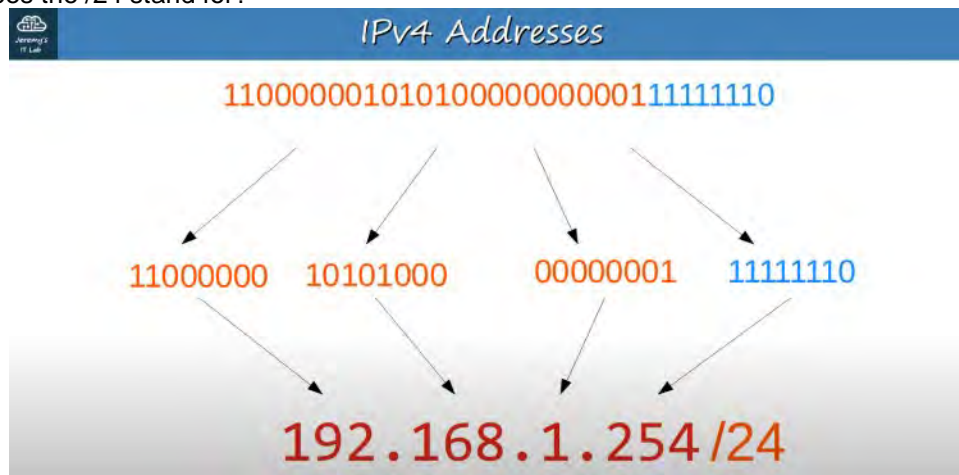
11001111 is the correct answer.

IPv4 ADDRESSES

So we now know that IP Addresses are the Dotted Decimal conversion of a series of BINARY NUMBERS (broken up into 4 OCTETS) like so:

192.168.1.254/24

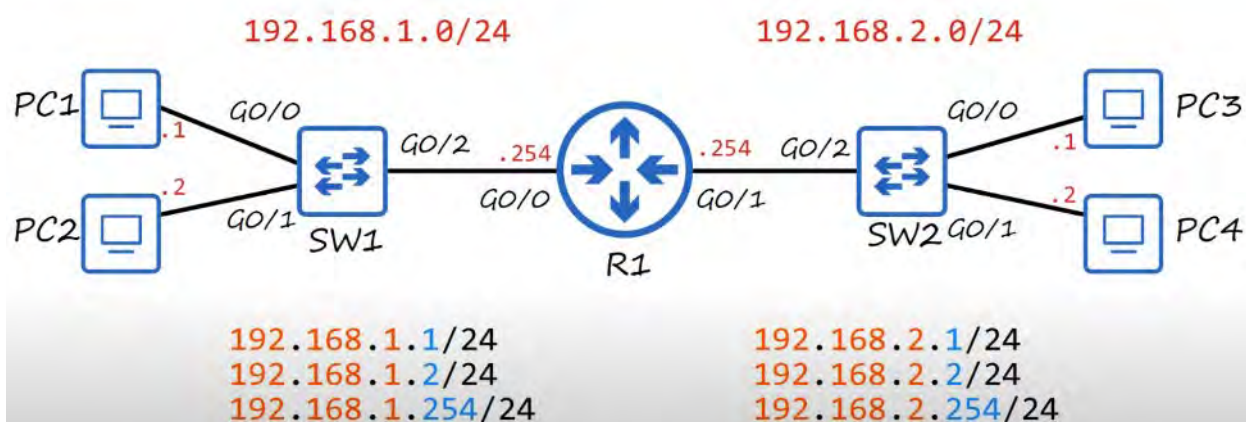
But what does the /24 stand for?



It means the FIRST 24 BITS of this address represent the NETWORK portion of the address.

192.168.1 is the NETWORK PORTION (the first 3 OCTETS)

.254 is the HOST PORTION (the last OCTET)



CONVERT this BINARY number into an IPv4 Address:

10011010010011100110111100100000

10011010 . 01001110 . 01101111 . 00100000

Octets:

1. 128 + 16 + 8 + 2 = 154
2. 64 + 8 + 4 + 2 = 78
3. 64 + 32 + 8 + 4 + 2 + 1 = 111
4. 32

The IPv4 address is: 154.78.111.32/16

154.78 is the NETWORK PORTION 111.32 is the HOST PORTION

Another Example:

0000110010000000111101100010111

00001100 . 10000000 . 11110111 . 00010111

Octets:

1. 8 + 4 = 12
2. 128
3. 255 - 4 = 251
4. 16 + 4 + 2 + 1 = 23

The IPv4 address is: 12.128.251.23/8

12 is the NETWORK PORTION 128.251.23 is the HOST PORTION

IPv4 ADDRESS CLASSES

IPv4 ADDRESSES are split up into 5 different 'classes'. The class of an IPv4 is determined by the FIRST OCTET of the address.

CLASS FIRST OCTET FIRST OCTET NUMERIC RANGE

A 0xxxxxxx 0-126 + 127 'loopback' B 10xxxxxx 128-191 C 110xxxxx 192-223 D 1110xxxx 224-239 E 1111xxxx 240-255

From the above chart, if the FIRST OCTET STARTS with 0, the numeric RANGE of possible first DOTTED DECIMAL is between 0-127.

The CLASSES we will be focusing on are CLASS A to CLASS C.

IPv4 Address Classes

Class	Leading bits	Size of network number bit field	Size of rest bit field	Number of networks	Addresses per network
Class A	0	8	24	128 (2^7)	16,777,216 (2^{24})
Class B	10	16	16	16,384 (2^{14})	65,536 (2^{16})
Class C	110	24	8	2,097,152 (2^{21})	256 (2^8)

D CLASS are reserved for 'MULTICAST' ADDRESSES

E CLASS are reserved for 'EXPERIMENTAL' ADDRESSES

A CLASS USUALLY have a range of 1-126? WHY?

Because 127 is usually reserved for 'loopback addresses'

127.0.0.0 to 127.255.255.255 are used to test the network.

- Used to test the 'Network stack' (OSI & TCP/IP model) on the local device.

IPv4 Address Classes

Class	First octet	First octet numeric range	Prefix Length
A	0xxxxxxx	0-127	/8
B	10xxxxxx	128-191	/16
C	110xxxxx	192-223	/24

The PREFIX LENGTH is the LENGTH of the NETWORK PORTION of the Address.

From the examples above:

12.128.251.23/8 is a CLASS A Address 154.78.111.32/16 is a CLASS B Address 192.168.1.254/24 is a CLASS C Address

Because the NETWORK portion of CLASS A is so short, it means there are a LOT more potential Hosts.

Because the NETWORK portion of CLASS C is so long, it means fewer potential Hosts.

NETMASK

Netmask

Class A: /8 255.0.0.0

(11111111 00000000 00000000 00000000)

Class B: /16 255.255.0.0

(11111111 11111111 00000000 00000000)

Class C: /24 255.255.255.0

(11111111 11111111 11111111 00000000)

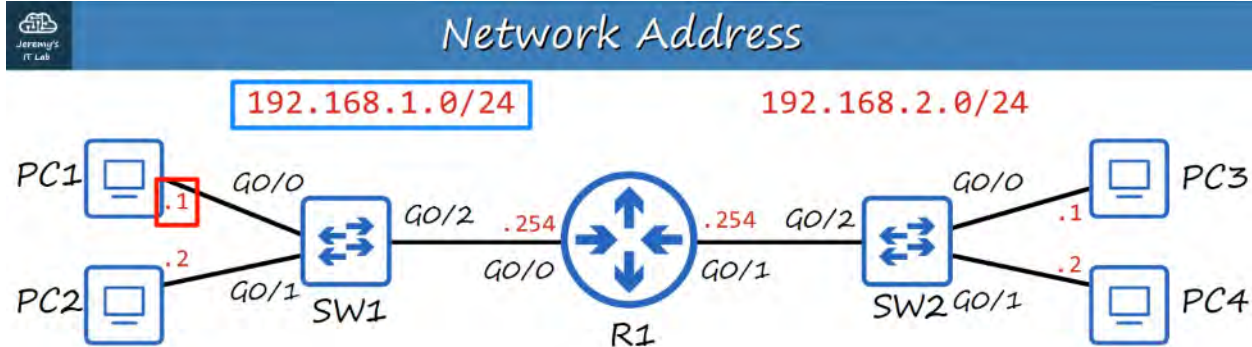
A NETMASK is written like a Dotted Decimal IP Address

CLASS A: /8 = 255.0.0.0

CLASS B: / 16 = 255.255.0.0

CLASS C: /24 = 255.255.255.0

NETWORK ADDRESSES



Host portion of the address is all 0's = Network Address

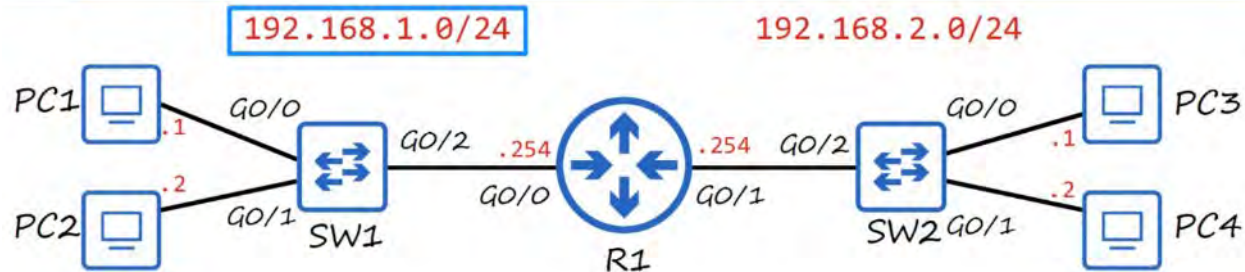
The network address CANNOT be assigned to a host.

If the HOST PORTION of an IP ADDRESS is ALL 0's, it means it is the NETWORK ADDRESS = the identifier of the network itself.

Example: 192.168.1.0/24 = THIS is a NETWORK ADDRESS.

A NETWORK ADDRESS cannot be assigned to a HOST. A NETWORK ADDRESS is the FIRST ADDRESS.

Broadcast Address



Host portion of the address is all **1**'s = Broadcast Address

The **broadcast address** CANNOT be assigned to a host.

If the HOST PORTION of an IP ADDRESS is ALL 1's, it means it is the BROADCAST ADDRESS for the network.

A BROADCAST ADDRESS cannot be assigned to a HOST.

DESTINATION IP : 192.168.1.255 (Broadcast IP address) DESTINATION MAC : FFFF.FFFF.FFFF (Broadcast MAC address)

Because of the two 'reserved' addresses, the range of USABLE HOST ADDRESSES is 1 to 254.

8. IPv4 ADDRESSING : PART 2

MAXIMUM HOSTS PER NETWORK

Let's take a Class C Network:

192.168.1.0/24

(gives a range of 0 ---> 255)

Said another way, the HOST portion (the .0) is equal to 8 bits so...

Host portion = 8 bits = $2^8 = 256$

HOWEVER, since the Network Address (Network ID)

192.168.1.0 is Reserved

AND

192.168.1.255 (BROADCAST ADDRESS) is ALSO reserved.

The MAXIMUM Hosts per Network = $2^8 - 2 = 254$ hosts

What about a Class B Network ?

172.16.0.0/16 ----> 172.16.255.255/16

Host portion = 16 bits = $2^{16} = 65,536$

Maximum hosts per network = $2^{16} - 2 = 65,534$ hosts

What about a Class A Network ?

10.0.0.0/8 -----> 10.255.255.255/8

Host portion = 24 bits = $2^{24} = 16,777,216$

Maximum hosts per network = $2^{24} - 2 = 16,777,214$ hosts

THEREFORE:

The formula for calculating the number of HOSTS on a network is:

$2^N - 2$ (2 to the power of N - 2)

where N = number of HOST bits

FIRST / LAST USABLE ADDRESSES

Class C Network

192.168.1.0/24 (NETWORK ADDRESS)

Add 1 so the Host Portion = 00000001

192.168.1.1/24 = FIRST USABLE ADDRESS

192.168.1.255/24 (BROADCAST ADDRESS)

Subtract 1 from the BROADCAST ADDRESS = 11111110

192.168.1.254/24 = LAST USABLE ADDRESS

Class B Network

172.16.0.0/16 (NETWORK ADDRESS)

Add 1 to Host portion so 0000 0000 0000 0001

172.16.0.1/16 is the FIRST USABLE ADDRESS

172.16.255.255/16 (BROADCAST ADDRESS)

Subtract 1 to Broadcast Address so 1111 1111 1111 1110

172.16.255.254/16 is the LAST USABLE ADDRESS

Class A Network

10.0.0.0/8 (NETWORK ADDRESS)

Add 1 to Host portion so 00000000 00000000 00000001

10.0.0.1/8 is the FIRST USABLE ADDRESS

10.255.255.255/8 (BROADCAST ADDRESS)

Subtract 1 to Broadcast Address so 1111 1111 1111 1110

10.255.255.254/8 is the LAST USABLE ADDRESS

CISCO CLI DEVICE CONFIGURATION

R1> enable R1# show ip interface brief

Lists the Interfaces, IP Addresses, Method, Status, and Protocol.

Interfaces:

- What port interfaces are available/connected

IP Addresses

- Self explanatory. What IP Address is assigned.

Method

- What method was the IP address assigned?

Status (Layer 1 Status)

- Current status of interface
- 'administratively down' = Interface has been disabled with the 'shutdown' command

Administratively down is the DEFAULT status of Cisco Router interfaces.

Cisco Switch interfaces are NOT administratively down by DEFAULT.

Protocol (Layer 2 Status)

- Cannot operate if Status (Layer 1) is down

Interface	IP-Address	OK?	Method	Status	Protocol
GigabitEthernet0/0	unassigned	YES	unset	administratively down	down
GigabitEthernet0/1	unassigned	YES	unset	administratively down	down
GigabitEthernet0/2	unassigned	YES	unset	administratively down	down
GigabitEthernet0/3	unassigned	YES	unset	administratively down	down

- administratively down: Interface has been disabled with the 'shutdown' command.
- This is the default Status of Cisco router interfaces.
- Cisco switch interfaces are NOT administratively down by default.

// configure terminal cmd

R1# conf t

// This enters interface configuration mode

R1(config)# interface gigabitethernet 0/0

This can be shortened to 'g0/0' like they are listed in physical network maps.

```
R1(config-if)#ip address 10.255.255.254 ?
A.B.C.D IP subnet mask

R1(config-if)#ip address 10.255.255.254 255.0.0.0
R1(config-if)#no shutdown
R1(config-if)#
*Dec 7 08:29:08.937: %LINK-3-UPDOWN: Interface GigabitEthernet0/0, changed state to up
*Dec 7 08:29:09.938: %LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/0, changed state to up
R1(config-if)#
```

// This sets the IP ADDRESS and SUBNET MASK of device

R1(config-if) #ip address 10.255.255.254 255.0.0.0

```
// This enables the device
R1(config-if) #no shutdown
```

Two messages should appear showing the state has changed to 'up' (Status). Second message should show line protocol is now 'up' (Protocol).

// 'do' allows you to run a Privileged EXEC command from outside the mode.

```
R1(config-if) #do show ip interface brief
```

Good to confirm that the device/interface you have configured is up and running.

More 'show' CLI Commands

show interfaces [interface]

```
R1#show interfaces g0/0
GigabitEthernet0/0 is up, line protocol is up
  Hardware is iGbE, address is 0c1b.8444.f000 (bia 0c1b.8444.f000)
  Internet address is 10.255.255.254/8
  MTU 1500 bytes, BW 1000000 Kbit/sec, DLY 10 usec,
    reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation ARPA, loopback not set
  Keepalive set (10 sec)
  Auto Duplex, Auto Speed, link type is auto, media type is RJ45
  output flow-control is unsupported, input flow-control is unsupported
  ARP type: ARPA, ARP Timeout 04:00:00
  Last input 00:00:06, output 00:00:05, 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
    167 packets input, 30159 bytes, 0 no buffer
    Received 0 broadcasts (0 IP multicasts)
    0 runs, 0 giants, 0 throttles
    0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored
    0 watchdog, 0 multicast, 0 pause input
    350 packets output, 39097 bytes, 0 underruns
    0 output errors, 0 collisions, 2 interface resets
    105 unknown protocol drops
    0 babbles, 0 late collision, 0 deferred
    1 lost carrier, 0 no carrier, 0 pause output
    0 output buffer failures, 0 output buffers swapped out
```

'show interfaces '

- Shows Layer 1 and Layer 2 information about the interface and some Layer 3.
- Shows MAC Address (or BIA address)
- IP Address
- ... and so much more

'show interfaces description'

- Allows you to add descriptions for interfaces.

Example:

```
// Configure mode for interface Gigabyte Interface 0/0
```

```
R1(config) #int g0/0
```

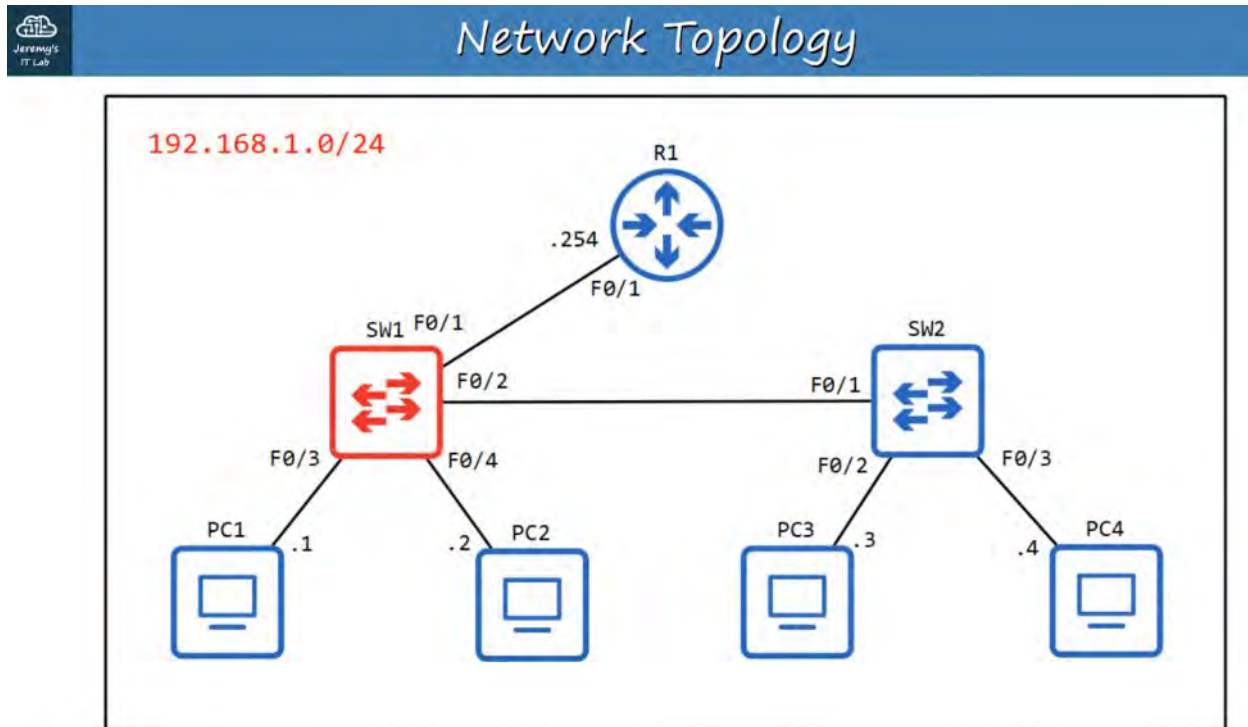
```
R1(config) #description ## to SW1 ##
```

This sets the 'Description' column to display:

```
Interface Description
```

```
Gi0/0 ## to SW1 ##
```

9. SWITCH INTERFACES



CISCO CLI for SWITCHES

Jeremys IT Lab

show ip interface brief

```
SW1>en
SW1#sh ip int br
```

Interface	IP-Address	OK?	Method	Status	Protocol
Vlan 1	unassigned	YES	unset	up	up
FastEthernet0/1	unassigned	YES	unset	up	up
FastEthernet0/2	unassigned	YES	unset	up	up
FastEthernet0/3	unassigned	YES	unset	up	up
FastEthernet0/4	unassigned	YES	unset	up	up
FastEthernet0/5	unassigned	YES	unset	down	down
FastEthernet0/6	unassigned	YES	unset	down	down
FastEthernet0/7	unassigned	YES	unset	down	down
FastEthernet0/8	unassigned	YES	unset	down	down
FastEthernet0/9	unassigned	YES	unset	down	down
FastEthernet0/10	unassigned	YES	unset	down	down
FastEthernet0/11	unassigned	YES	unset	down	down
FastEthernet0/12	unassigned	YES	unset	down	down

// enter Privileged EXEC mode

SW1>enable

// Show all interfaces of Switch 1.

SW# show ip interface brief

This will show the interfaces currently on Switch 1. It has the same information structure as Cisco Routers.

Notice the Status (Layer 2) and Protocol (Layer 1) columns are showing "up/up".

Unlike ROUTERS, SWITCHES do no DEFAULT to 'administrative down/down'(shutdown).

Unconnected devices will show as "down" and "down" (not connected to another device)



show interfaces status

```
SW1#show interfaces status
```

Port	Name	Status	Vlan	Duplex	Speed	Type
Fa0/1		connected	1	a-full	a-100	10/100BaseTX
Fa0/2		connected	trunk	a-full	a-100	10/100BaseTX
Fa0/3		connected	1	a-full	a-100	10/100BaseTX
Fa0/4		connected	1	a-full	a-100	10/100BaseTX
Fa0/5		notconnect	1	auto	auto	10/100BaseTX
Fa0/6		notconnect	1	auto	auto	10/100BaseTX
Fa0/7		notconnect	1	auto	auto	10/100BaseTX
Fa0/8		notconnect	1	auto	auto	10/100BaseTX
Fa0/9		notconnect	1	auto	auto	10/100BaseTX
Fa0/10		notconnect	1	auto	auto	10/100BaseTX
Fa0/11		notconnect	1	auto	auto	10/100BaseTX
Fa0/12		notconnect	1	auto	auto	10/100BaseTX

// Show the status of all interfaces on SW1

SW1#show interfaces status

This will list:

- Ports
- Name (which is description)
- Status (connection status)
- Vlan (can be used to divide up LANs) - Vlan 1 is the default.
- Duplex (can the connection send/receive at same time?) - Auto is default
- Speed (speed in bps) - Auto is default
- Type (what medium is being used, speed of interface)



Configuring interface speed and duplex

```
SW1#conf t
Enter configuration commands, one per line. End with CNTL/Z.
SW1(config)#int f0/1
SW1(config-if)#speed ?
10          Force 10 Mbps operation
100         Force 100 Mbps operation
auto        Enable AUTO speed configuration
SW1(config-if)#speed 100
SW1(config-if)#duplex ?
auto        Enable AUTO duplex configuration
full        Force full duplex operation
half        Force half-duplex operation
SW1(config-if)#duplex full
SW1(config-if)#description ## to R1 ##
```


Configuring interface speed and duplex

```
SW1#sh int status
```

Port	Name	Status	Vlan	Duplex	Speed	Type
Fa0/1	## to R1 ##	connected	1	full	100	10/100BaseTX
Fa0/2		connected	trunk	a-full	a-100	10/100BaseTX
Fa0/3		connected	1	a-full	a-100	10/100BaseTX
Fa0/4		connected	1	a-full	a-100	10/100BaseTX
Fa0/5		notconnect	1	auto	auto	10/100BaseTX
Fa0/6		notconnect	1	auto	auto	10/100BaseTX
Fa0/7		notconnect	1	auto	auto	10/100BaseTX
Fa0/8		notconnect	1	auto	auto	10/100BaseTX
Fa0/9		notconnect	1	auto	auto	10/100BaseTX
Fa0/10		notconnect	1	auto	auto	10/100BaseTX
Fa0/11		notconnect	1	auto	auto	10/100BaseTX
Fa0/12		notconnect	1	auto	auto	10/100BaseTX

INTERFACE RANGE

Unused Interfaces can pose a security risk so it's a good idea to deactivate them.

However, if you have 28+ interfaces not in use, do you have to do them one at a time?

Answer: No! There is a command to apply configurations to a range of interfaces.

Inside Global Config Mode (config t):

interface range

```
SW1(config)#interface range f0/5 - 12
SW1(config-if-range)#description ## not in use ##
SW1(config-if-range)#shutdown
00:42:36: %LINK-5-CHANGED: Interface FastEthernet0/5, changed state to administratively down
00:42:36: %LINK-5-CHANGED: Interface FastEthernet0/6, changed state to administratively down
00:42:36: %LINK-5-CHANGED: Interface FastEthernet0/7, changed state to administratively down
00:42:36: %LINK-5-CHANGED: Interface FastEthernet0/8, changed state to administratively down
00:42:36: %LINK-5-CHANGED: Interface FastEthernet0/9, changed state to administratively down
00:42:36: %LINK-5-CHANGED: Interface FastEthernet0/10, changed state to administratively down
00:42:36: %LINK-5-CHANGED: Interface FastEthernet0/11, changed state to administratively down
00:42:36: %LINK-5-CHANGED: Interface FastEthernet0/12, changed state to administratively down
SW1(config-if-range)#
```

SW1(config)#interface range f0/5 - 12 // Choose all interfaces from 0/5 to 0/12

SW1(config-if-range)#description ## not in use ##

SW1(config-if-range)#shutdown

<< this will list all the interfaces being set to administratively down >>

Confirm with 'show interface status' in Privileged EXEC mode or if in CONFIG mode, use 'do show interface status'

```
SW1(config-if-range)#do sh int status
```

Port	Name	Status	Vlan	Duplex	Speed	Type
Fa0/1	## to R1 ##	connected	1	full	100	10/100BaseTX
Fa0/2	## to SW2 ##	connected	trunk	a-full	a-100	10/100BaseTX
Fa0/3	## to end hosts ##	connected	1	a-full	a-100	10/100BaseTX
Fa0/4	## to end hosts ##	connected	1	a-full	a-100	10/100BaseTX
Fa0/5	## not in use ##	disabled	1	auto	auto	10/100BaseTX
Fa0/6	## not in use ##	disabled	1	auto	auto	10/100BaseTX
Fa0/7	## not in use ##	disabled	1	auto	auto	10/100BaseTX
Fa0/8	## not in use ##	disabled	1	auto	auto	10/100BaseTX
Fa0/9	## not in use ##	disabled	1	auto	auto	10/100BaseTX
Fa0/10	## not in use ##	disabled	1	auto	auto	10/100BaseTX
Fa0/11	## not in use ##	disabled	1	auto	auto	10/100BaseTX
Fa0/12	## not in use ##	disabled	1	auto	auto	10/100BaseTX

FULL / HALF DUPLEX

HALF DUPLEX:

- Device cannot send / receive data at the same time. If it is receiving a frame, it must wait before sending a frame.

FULL DUPLEX:

- Device CAN send / receive data at the same time. It does NOT have to wait.

MOST modern SWITCHES support FULL DUPLEX.

WHERE is HALF DUPLEX used? Almost nowhere.

In the past, LAN HUBS used HALF DUPLEX.

When multiple packets were received by the HUB, the HUB would simple FLOOD the connections with frame data, causing a COLLISION (on the interface), and hosts would not receive the frame intact.

All devices connected to a HUB are called a COLLISION DOMAIN.

To DEAL with COLLISIONS, Ethernet devices use a mechanism called CSMA/CD.

CSMA/CD = CARRIER SENSE MULTIPLE ACCESS with COLLISION DETECTION.

- Before sending frames, devices 'listen' to the collision domain until they detect that other devices are not sending.
- IF a collision occurs, the device sends a jamming signal to inform the other devices that a collision happened.
- Each device will wait a random period of time before sending frames again.
- The process repeats.

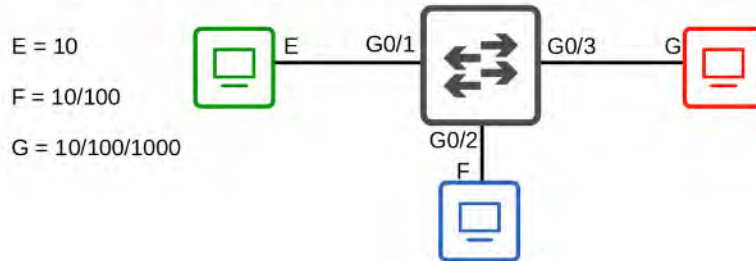
SWITCHES are more sophisticated than HUBS.

HUBS are Layer 1 Devices - Collisions are common and use CSMA/CD. SWITCHES are Layer 2 Devices - Collisions RARELY occur.



Speed/Duplex Autonegotiation

- Interfaces that can run at different speeds (10/100 or 10/100/1000) have default settings of speed auto and duplex auto.
- Interfaces 'advertise' their capabilities to the neighboring device, and they negotiate the best speed and duplex settings they are both capable of.



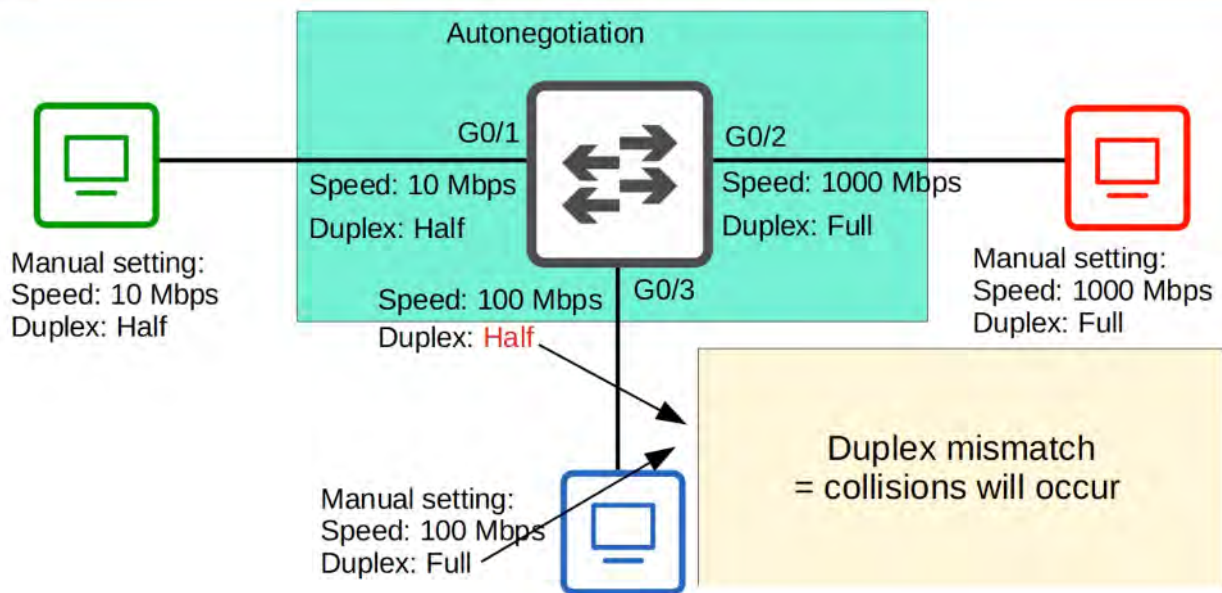
SPEED / DUPLEX AUTONEGOTIATION

- Interfaces that can run at different speeds (10/100 or 10/100/1000) have a default setting of SPEED AUTO and DUPLEX AUTO.
- Interfaces 'advertise' their capabilities to the neighbouring device, and they negotiate the best SPEED and DUPLEX settings they are both capable of.

WHAT if AUTONEGOTIATION is DISABLED on the device connected to the SWITCH ?



Speed/Duplex Autonegotiation



- SPEED: The SWITCH will try to send at the speed that the other device is operating at. If it fails to send the speed, it will use the slowest supported speed (ie: 10 Mbps on a 10/100/1000 interface).
- DUPLEX: If the speed is 10 or 100 Mbps the SWITCH will use HALF DUPLEX. If the speed is 1000 Mbps or great, it will use FULL DUPLEX.

INTERFACE COUNTERS AND ERRORS

Show using the:

// Privileged EXEC mode

SW1#show interfaces
Error stats will be at the bottom.

Interface Errors

```
SW1#show interfaces f0/2
FastEthernet0/2 is up, line protocol is up
  Hardware is Fast Ethernet, address is 000C.3168.8461 (bia 000C.3168.8461)
  Description: ## to SW2 ##
  MTU 1500 bytes, BW 100000 Kbit, DLY 100 usec,
    reliability 255/255, txload 1/255, rxload 1/255
  Auto-duplex, Auto-speed
  Encapsulation ARPA, loopback not set
  ARP type: ARPA, ARP Timeout 04:00:00
  Last input 02:29:44, 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
  Queuing 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
    269 packets input, 71059 bytes, 0 no buffer
    Received 6 broadcasts, 0 runts, 0 giants, 0 throttles
    0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored
    7290 packets output, 429075 bytes, 0 underruns
    0 output errors, 3 interface resets
    0 output buffer failures, 0 output buffers swapped out
```

Packets Received / Total bytes received.

Runts: Frames that are smaller than the minimum frame size (64 bytes)

Giants: Frames that are larger than the maximum frame size (1518 bytes)

CRC: Frames that failed the CRC check (in the Ethernet FCS trailer)

Frame: Frames that have an incorrect format (due to an error)

Input errors: Total of various counters, such as the above four

Output errors: Frames the SWITCH tried to send, but failed due to an error