ICND1 -100-101 Study Guide (CCENT)

SECTION I (6%) – Purpose & Function of Various Network Devices (Routers, Switches, Bridges, Hubs)

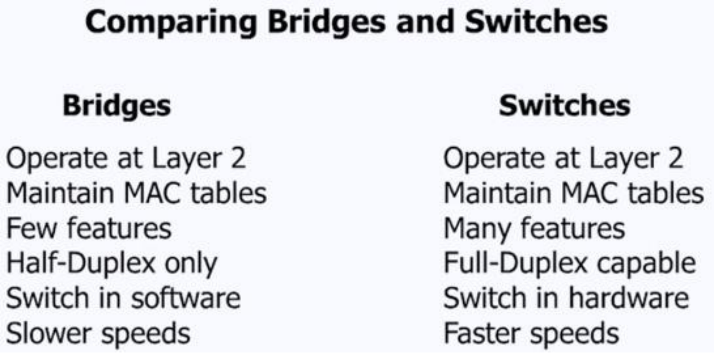
1.1 – Recognize Purpose & Functions of Various Network Devices (Routers, Switches, Bridges, Hubs)

a. Repeater
1) Purpose: unintelligent Layer 1 device to resolve attenuation (media distance constraints)
2) Function: 2-port device that regenerates signals to span greater than media distances allow; doesn't alter or interpret signal… just listens to signal & reproduces it

b. Hub
1) Purpose: multiport Layer 1 repeater to resolve attenuation issues (media distance constraints)
2) Function: Advg – extends networks like repeaters, allows connection of more than 2 devices (than repeaters), central point for cabling; Disadvg – creates more collisions (shared bandwidth), greater congestion, no traffic control (filtering), & only Half-Duplex (one-directional communication)

c. Bridges
1) Purpose: intelligent Layer 2 (Data Link) control
2) Function: joins/extends LAN segments, regenerates signals, reduces collisions, learn & filter traffic based on MAC Address; **Half-Duplex**
a) Floods Frames – sends out every port except port was received on; unknown unicast → no destination MAC address in Frame; Broadcasts; Multicasts
b) Forwards Frames – concept of MAC Address Tables (learns source MAC & Port the devices connected to it)
c) Filter – based off MAC Address Table entries, may drop Frames depending on source & destination addresses in a Frame

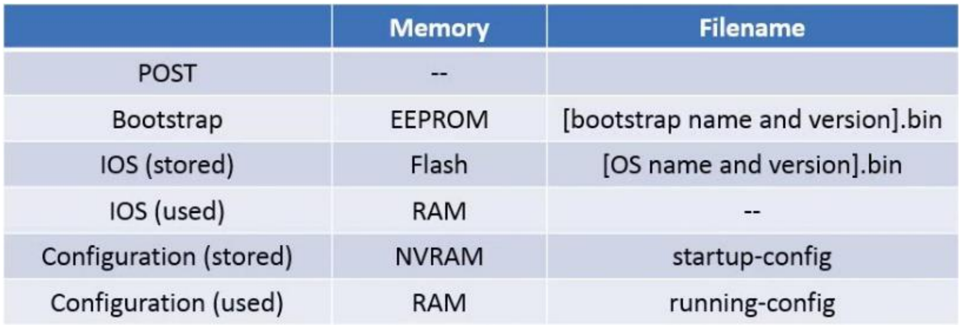
d. Switches
1) Purpose: multiport bridges with additional features; intelligent Layer 2
2) Function: **Full-Duplex** (bi-directional communication at same time); faster than Bridges (Gb); switching done in hardware
3) Switching Methods = Flood, Forward, Filter (see the Bridge Section above)
4) Benefits
a) Each port micro-segments LAN providing dedicated bandwidth to the connected device
b) Allows multiple simultaneous conversations between devices on different ports
c) Full-Duplex support, in effect doubling bandwidth available to connected device
d) Support for rate (speed) adaptation for devices configured with different speeds
5) Memory Types:
a) ROM – bootstrap executed upon bootup; POST; hard-coded onto Switch motherboard
b) NVRAM – non-volatile RAM; stores startup config permanently upon reboot
c) Flash – NVRAM type that stores IOS image; retained upon shutdown
d) RAM – active memory; running configuration, committed to NVRAM upon using "copy" cmd



Comparing Bridges and Switches

Bridges	Switches
Operate at Layer 2	Operate at Layer 2
Maintain MAC tables	Maintain MAC tables
Few features	Many features
Half-Duplex only	Full-Duplex capable
Switch in software	Switch in hardware
Slower speeds	Faster speeds

e. Routers
   1) Purpose: enables data to transfer from one IP network to a different IP network
   2) Function: allows “internal” networks to communicate with “external” networks (NAT)
   3) Bootup Process:
      a) Run POST
         1. Configuration Register (0x21xx) is checked; 0x2102 = Default; 0x2142 = Pwd change
      b) Find IOS in Flash (if none found, loads from ROM)
      c) Load IOS to RAM
      d) Find the (Startup) Config in NVRAM (if none found, broadcast to a TFTP server for Config)
      e) Load (Startup) Config to RAM
   4) Router Files:



	Memory	Filename
POST	--	
Bootstrap	EEPROM	[bootstrap name and version].bin
IOS (stored)	Flash	[OS name and version].bin
IOS (used)	RAM	--
Configuration (stored)	NVRAM	startup-config
Configuration (used)	RAM	running-config

1.2 – Select Components Required to Meet a Given Network Specification

a. This is relative; I would probably think in terms of the capabilities of each device, for example what creates or segments Broadcast vs Collision Domains

b. Where is a given device used (i.e. which Layer uses Repeaters, Hubs, Bridges, Switches, Routers)

c. Think of device function what it does or is capable of doing with network traffic

1.3 – Identify Common Applications & Their Impact on the Network

a. DNS – TCP/UDP 53; Domain Name Service, translating Hostnames to IPs
   1) Hosts/devices send DNS Requests to resolve a hostname to an IP Address
   2) Once hostname is resolved to an IP Address, the client begins TCP connection process

b. DHCP – UDP 67/68; Dynamic Host Configuration Protocol, assigns IPs & other options dynamically

c. FTP – TCP 20/21; File Transfer Protocol

d. HTTP (WWW) – TCP 80; Hyper-Text Transfer Protocol, web server access
1) Web clients send HTTP GET Requests for web page files; the web server sends an HTTP OK response
2) Ex. http://www.certskills.com/ICND1 → **http** = protocol used; **www.certskills.com** = hostname; **ICND1** = web page name

e. POP3 – TCP 110; Post Office Protocol, email access

f. SMTP – TCP 25; Simple Mail Transfer Protocol, email access

g. SNMP – UDP 161; Simple Network Management Protocol, device monitoring & management

h. SSH – TCP 22; Secure Shell, encrypted remote management (using PuTTy e.g.)

i. SSL – TCP 443; HTTPS, encrypted web server access

j. Telnet – TCP 23; unencrypted remote management (using PuTTy e.g.)

k. TFTP – UDP 69; basic version of FTP

l. QoS – Quality of Service; 4 QoS characteristics/components:
1) Bandwidth – volume of bits/second
2) Delay – amount of time it takes one IP Packet to flow from sender to receiver
3) Jitter – variation in delay
4) Loss - % of Packets discarded by network before they reach the receiver/destination

APPLICATION CATEGORY	DELAY	JITTER	LOSS
Web Browsing (Interactive)	Medium	Medium	Medium
VoIP	Low	Low	Low
Video Conferencing	Low	Low	Low

1.4 – Describe the Purpose & Basic Operation of Protocols in the OSI & TCP/IP Models

OSI MODEL

a. Application Layer (L7):
1) Interface between network & application software; user authentication; provides services to applications but is NOT the application itself
2) Data Unit = **Data**
3) Protocols = SMTP, POP3 – email; SSH – Secure Shell & Telnet for remote console access (Putty, WinSCP); DNS – resolves Hostname to IP & vice versa; FTP – file transfer protocol
4) Workstations, Servers, Firewalls

b. Presentation Layer (L6):
1) Defines data format & encryption; this Layer no longer in use

2) Data Unit = **Data**
3) Protocols = ASCII, JPEG, Binary, EBCDIC
4) Workstations, Servers

c. Session Layer (L5):
1) How to start & end conversations between endpoints; manages the point-to-point communication; this Layer no longer in use
2) Data Unit = **Data**
3) Protocols = NETBIOS
4) Workstations, Servers

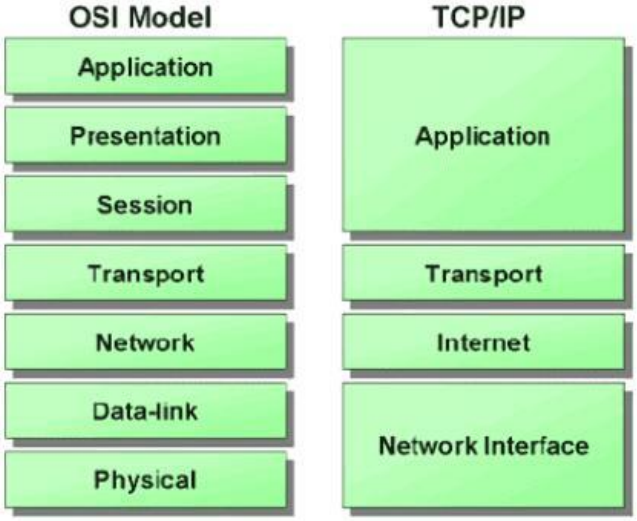
d. Transport Layer (L4):
1) Provides flow control & error recovery to prevent data loss; focuses on data delivery to other endpoints/devices
2) Data Unit = **Segment**
3) Protocols = TCP, UDP; also uses Port #'s (e.g. 21 [ftp], 22[ssh], 25[smtp], 53[dns], 80[http], 139[ldap], 443[https], etc.)
4) Connection-Oriented (TCP) – uses acknowledgment & flow control, sets up a virtual circuit; Connectionless-Oriented (UDP) – unreliable, uses best effort; fast; relatively no overhead; no virtual circuit; e.g. Radio, Streaming Video, TV
a) Three-Way Handshake → SYN > SYN, ACK > ACK
1. Sender sends a Segment incorporated with a Sequence Number
2. The Receiver responds with an ACK with its own Sequence Number & what the Sender's next Sequence Number should be
3. The Sender responds with a Segment with the data stream's next Sequence Number
b) Positive Acknowledgment & Retransmission (PAR) – continued guaranteed communication after the 3-Way Handshake process
1. Sender starts a timer when sending a Segment; retransmits Segment if timer expires before an ACK is received from the Receiver
2. Sender keeps record of all Segments sent & expects an ACK for each one sent
3. Receiver sends an ACK after each Segment indicating the expected next Sequence Number Segment
c) Sliding Window – process of a Receiver telling Sender to slow its Segment transfer rate if it (Receiver) is getting more Segment 'hits' than it can handle
1. Number of Segments a Sender can send in a transmission before Receiver sends an ACK; if Receiver isn't busy, the window size can be large; more congestion = smaller window size
2. Windows size is included in Segment headers and can change during conversation lifespan
5) Routers, Firewalls

e. Network Layer (L3):
1) Routing (forwarding), path determination, & logical addressing
2) Data Unit = **Packet**
3) Protocols = IP, ICMP
4) Routers, Layer-3 Switches

f. Data Link Layer (L2):
1) Provides media access control (MAC Address), error detection & assembles bits from frames & vice versa; rules to determine when a device can transmit data; defines formats of Frame

headers/trailers
2) Data Unit = **Frame**
3) Protocols = Ethernet, Serial, PPP, ATM, DOCSIS, DSL
4) Switches, Bridges, WAP (Wireless Access Point), Cable/DSL Modems

g. Physical Layer (L1):
1) Sends & receives bits and provides specification of voltage, wire speed, & cable pin-outs; bits (on/off electrical pulses); physical characteristics of transmission medium (connectors, pins, etc)
2) Data Unit = **Bits**
3) Protocols = Glass (Fiber), Copper (CAT-3/5/6), RJ45
4) Hubs, Repeaters, Cables, Radio Waves



Encapsulation – process of lower level OSI or TCP/IP Model layers encapsulating data unit created at the upper layer levels; then transformed to 1s/0s (electrical impulses/voltages) at the physical layer

TCP/IP MODEL

a. Application
b. Transport
c. Internet
d. Network Interface

1.5 – Predict the Data Flow Between Two Hosts Across a Network

a. Begin with 4.1 a. below – Packet Forwarding process; otherwise, thinking through the below briefly discussed items should provide you with info needed to work through network traffic flow

b. Think about what happens from source Host to target Host, name resolution, MAC identification (ARP), to Subnet identification (Route lookup), to VLAN accessibility, to Internet access

c. Think of traffic flow in terms of protocol process; e.g. how is ARP handled on an Internetwork (i.e. LAN vs remote Subnet)

d. How are Frames handled by PCs, Switches, Routers; Encapsulation & De-encapsulation

1.6 – Identify the Appropriate Media, Cables, Ports, & Connectors to Connect Cisco Network Devices to Other Network Devices & Hosts in a LAN

a. Common Media Cable Types (Name, Notation, Cable, IEEE Notation, Speed, Length):

LAN

Ethernet Cable Type/Speed	Media	IEEE Notation	Length
10BASE-T / 10Mbps	CAT3 or better (2-pair)	802.3	100m
100BASE-T / 100Mbps	CAT5-UTP (2-pair)	802.3u	100m
1000BASE-T / 1000Mbps	CAT5e/6-UTP (4-pair)	802.3ab	100m
1000BASE-LX or SX	Multimode Fiber	802.3z	550m
1000BASE-LX	Single-Mode Fiber	802.3z	5km
10GBASE-T	CAT6a-UTP	802.3an	100m

CODE: T = Twisted Pair ; X = Fiber ; L = Long Wave Length ; S = Short Wave Length

b. Deprecated/Less Used Cables:
1) Thicknet; 10BASE-5; RG-8; 802.3; 10Mb; 500m
2) Thinnet (“cheaper net”); CATV coax, 10BASE-2; RG-58; 802.3; 10Mb; 185m

WAN

3) Leased Lines – uses 2 pairs of crossover in Full Duplex
   a) Names = Circuit; Serial (Link); Point-to-Point; T1; WAN Link
   b) Customer Site CPE (Customer Premise Equipment) = Host, Switch, Router w/Serial Interface, CSU/DSU (channel service unit/data service unit); Leased Line (LL) > Telco Switch > LL > CPE on Customer 'other side' (CSU/DSU > Router 2 > LAN Switch 2 > Host)
   c) Speeds – slower = multiples of 64Kbps; faster = multiples of 1.54Mbps
   d) Layer 2 Protocols Used – HDLC (High-Level Data Link Control) and PPP (Point-to-Point Protocol)
      1. HDLC – control correct delivery of data over a physical WAN link; HDLC Frame Fields = Flag, Address, Control, FCS; **NOTE:** Cisco adds a Type field
   e) Benefits = simple, widely available, private, high quality; Negatives = high cost, lengthy install
4) Ethernet – uses Fiber
   a) CPE > Fiber Ethernet Link > Service Provider (SP) Ethernet Switch > SP Point-of-Presence (POP) Ethernet (Fiber) WAN > SP POP Ethernet Switch > Fiber Ethernet Link > CPE
   b) EoMPLS – Ethernet over Multiprotocol Label Switching; does not use a serial interface at CPE
      1. PC > Switch > Router w/Ethernet Interface > EoMPLS (Fiber) Link > R2 w/EI> SW2 > PC2

c. Connectors
1) RJ45 (8P89C) – used with Copper cables
2) SFP (small form-factor pluggable) or GBIC (gigabit interface converter) – used by Fiber cables
3) ST – Straight tip Connector for Fiber/Optical; MMF
4) SC – Subscriber Connector for Fiber/Optical; MMF or SMF
5) LC – Lucent Connector for Fiber/Optical; most common fiber connector

d. Straight-Through Cable – pins are connected to same pins on each side of connection (1 → 1, 2 → 2, 3 → 3, 6 → 6)

1) Connect PC to Switch or Hub
2) Connect Router to Switch or Hub
3) Basically, when connecting "computers" (Workstations, Servers, Routers) to a Switch, Hub, or WAP

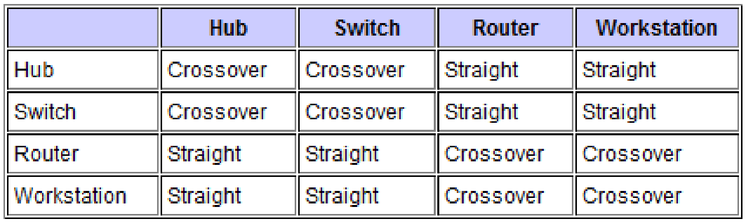
e. Crossover Cable – pins are crossed across sides (1 → 3, 2 → 6)

1) Connect PC to PC
2) Connect Switch to Switch
3) Connect Router to Router
4) Connect PC to Router
5) Connect Hub to Hub

f. Rollover Cable – pins are exact opposite from pins 1-8 (1 → 8, 2 → 7, 3 → 6, 4 → 5)

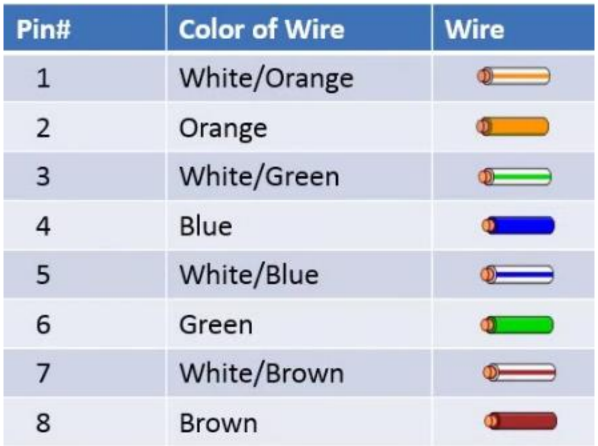
1) Connect directly to CONSOLE Port of Router or Switch from PC

TABLE TO SUMMARIZE DEVICE CONNECTIVITY REQUIREMENTS

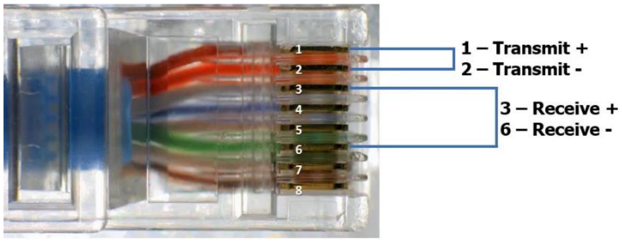


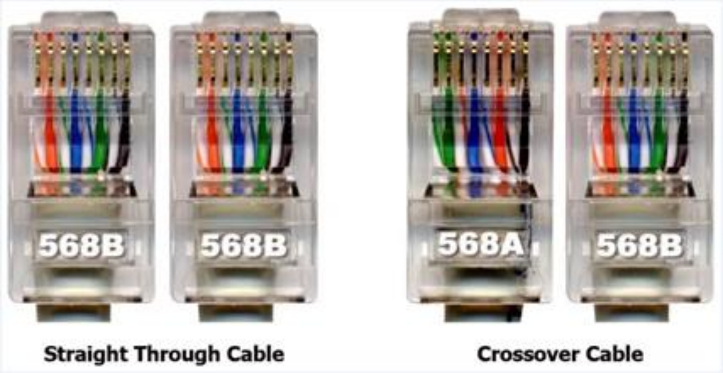
	Hub	Switch	Router	Workstation
Hub	Crossover	Crossover	Straight	Straight
Switch	Crossover	Crossover	Straight	Straight
Router	Straight	Straight	Crossover	Crossover
Workstation	Straight	Straight	Crossover	Crossover

PIN/WIRING STANDARD (T568B) FOR STRAIGHT-THROUGH & EXAMPLE



Pin#	Color of Wire	Wire
1	White/Orange	⬜🟧
2	Orange	🟧
3	White/Green	⬜🟩
4	Blue	🟦
5	White/Blue	⬜🟦
6	Green	🟩
7	White/Brown	⬜🟫
8	Brown	🟫





g. Ethernet (802.3) – defines WIRED LAN technology only, not wireless

1) Ethernet Address Types

a) Unicast – one workstation to one workstation communication

b) Broadcast – one workstation to all other workstations communication

c) Multicast – one workstation to a select group of workstations communication (Destination MAC Address Field starts with 0100.5exx.xxxx.xxxx)

2) Ethernet Frame Contents:

Preamble 7 bytes	SFD 1 byte	Dest MACAddr 6 bytes	SourceMAC Addr 6 bytes	Type (IPv4/6, →0800/86DD) 2 bytes	DATA 46-1500 bytes	FCS 4 bytes

a) Start Frame Delimiter (SFD) – signifies the next byte begins Destination MAC

b) Dest/Source MAC

1. MAC Address – 3 bytes (8bits) OUI assigned by IEEE, 3 bytes Vendor-assigned

c) Frame Check Sequence (FCS) – method for receiving NIC to determine if Frame had transmission errors

h. EMI – electromagnetic interference

1) Crosstalk: NEXT (near-end), FEXT (far-end), AXT (alien)

i. Attenuation – signal loses strength over distance

SECTION II (21%) – LAN Switching Technologies

2.1 – Determine the Technology & Media Access Control Method for Ethernet Networks

a. CSMA/CD – Carrier Sense Multiple Access with Collision Detection – CSMA = devices using shared medium (wire) for communication, analogous to landline home phone

→ **Half-Duplex** –sending traffic in one direction at a time [e.g. analogous to a one-way bridge]); legacy Hub-based or “Ethernet bus” networks

1) Workstations listen to the wire

2) If no one is sending data, data is sent by a workstation

3) If multiple workstations are sending at the same time, there will be a frame collision, causing a voltage spike on the "wire" (10volt spike)
4) Damaged Frames from colliding workstations will be discarded
5) Workstations will then send out a "jamming signal" to prevent other workstations from transmitting frames
6) Each workstation will then set a random timer to determine retransmission of frames (data)
7) When timers expire, process repeats starting from Step "1" above

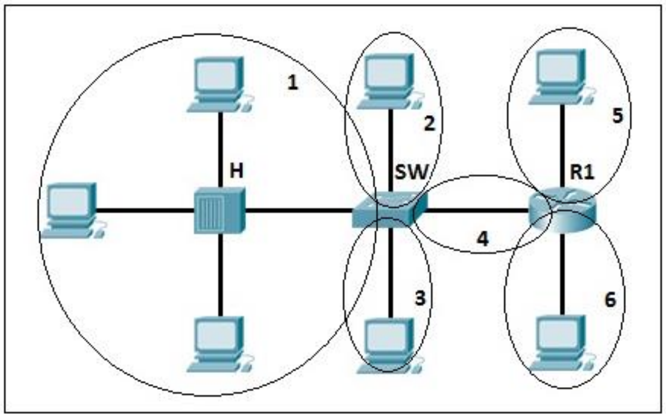
→ **Full Duplex** – can send traffic in both directions at same time (e.g. analogous to two-lane road); sending/receiving on NICs/ports at same time; PCs, Switches, Routers, Bridges

b. CSMA/CA (AppleTalk) – Carrier Sense Multiple Access with Collision Avoidance

2.2 – Identify Basic Switching Concepts & the Operation of Cisco Switches

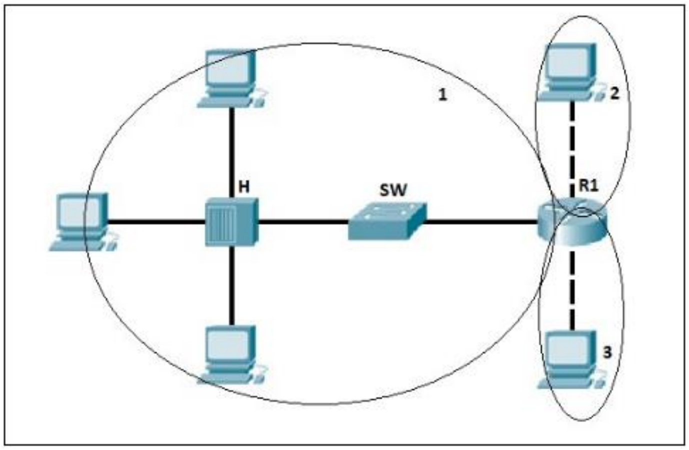
a. Collision Domains – collection of Interfaces on a LAN segment whose Frames can collide with one another, but not with other devices on the network not part of this collection
1) Hubs & Repeaters create 1 large Collision Domain because they join LAN segments
2) Bridges & Switches segment between Collision Domains; *microsegmentation*
3) This is a Physical Layer (Layer1) "issue"; a voltage spike on the wire (> 10v)
a) To mitigate propagation of Collisions, need to implement a Layer 2 (Data Link Layer) device → Switch

COLLISION DOMAIN EXAMPLE WITH HUBS (**H**), SWITCHES (**SW**), & ROUTERS (**R**)



b. Broadcast Domains – collection of devices on a LAN segment which will receive a Broadcast Frame when a neighboring device sends one; other devices not in the collection will not receive the Broadcast Frame
1) Hubs & Repeaters create 1 large Broadcast Domain (in addition to Collision Domain)
2) Bridges & Switches create 1 large Broadcast Domain (due to Flood Switching Method)
3) Routers & Layer-3 Switches segment between Broadcast Domains
4) This is a Data Link Layer (Layer2) "issue"; all "Fs" in the Destination MAC Address field of the Ethernet Frame Header
a) To mitigate propagation of Broadcasts, need to implement a Layer 3 device → Router
5) Try to keep the # of Hosts in a Broadcast Domain to around 500

BROADCAST DOMAIN EXAMPLE WITH HUBS (**H**), SWITCHES (**SW**), & ROUTERS (**R**)



c. Ways to Switch

1) Store and Forward – most common Switching method; Switch doesn't forward (switch) a Frame until the full Frame is received.
2) Cut Through – Switch forwards Frame upon receiving part of the Frame Header containing the Destination MAC Address; reduces latency but propagates errors because FCS not checked
3) Fragment-Free – Frame is forwarded after the 1st 64 bytes of Frame are read
4) CAM Table – Content Addressable Memory (or MAC) Address Table; stores *source* MAC Addresses in the Table upon receiving a new Frame into a switch port/interface
5) Frame Flooding – Switch sends incoming Frame out all ports, *except* the source port the Frame entered, for unknown unicast Frame (Destination MAC Address not yet in the MAC Address Table) or a Broadcast Frame (ARP Requests)

d. VLANs – Virtual LANs

1) Creates separate Broadcast Domains within a Switch

e. Switch Layers:

1) Access – end-user access; layer 2 connectivity; not connected to 1 another but dual-connected to Distribution
2) Distribution – aggregation between access switches; uplinks to the Core
3) Core – high speed switching between Distribution Switches; reliability & speed are key; DC

f. Models:

1) Compact – SOHO type (desktop FF); "c" after model number (e.g. 2560c); low port #'s
2) Stackable – rack mountable (1.75"; 1RU FF); standalone; leading model # = L2 or L3 type
3) Single Stack – 1RU FF; stackable via special cabling
4) Chassis Based – 7-20RU FF; Enterprise Core
5) Layer-Based – if the model number begins with a "2", it's Layer 2; begins with a "3", it's Layer 3

2.3 – Configure & Verify Initial Switch Configuration Including Remote Access Management

a. Hostname:

1) Upon first logging into Switch, in Exec Mode designated by ">" next to hostname (i.e. **`sw1>`**)
2) Go into Privileged/Enabled Mode – **`sw1> en`**
3) Go into Global Configuration Mode – **`sw1#config t`**
4) Change Hostname as needed – **`sw1(config)#hostname switch1`**

b. Management IP Address

1) Done on a virtual interface, on VLAN1; i.e. there is no NIC/physical port associated with this interface
2) Config Mode: **`conf t`**
3) Sub-Config Mode: **`interface vlan 1`**
4) Assign IP/Subnet: **`ip address 192.168.10.2 255.255.255.0`**
5) Bring Interface "up": **`no shut`**

c. IP Default Gateway

1) From Global Config Mode – **`sw1(config)#ip default-gateway 10.10.1.1`**(e.g. upstream Router

d. Local User & Password

1) From Global Config Mode – **`sw1(config)#username shane password cisco`**

e. Enable Secret "Password" – encrypting password used to get into Privileged Mode

1) From Global Config Mode – **`sw1(config)#enable secret SomePwd`**

f. Console & VTY Logins

1) Console (direct Console access to Cisco device via Console or AUX ports on device):
   a) From Global Config Mode –**`sw1(config)#line con 0`**
   b) **`logging sync`** (log msgs displayed more user-friendly when configuring device)
   c) **`password somepassword`**
   d) **`login`**
   e) **`privilege level 1`** (to make sure NOT going into Privileged Mode directly when logging into the console port of the device, if not already set)
2) VTYs (Virtual Console access to Cisco device – PuTTy TCP connection for Telnet or SSH):
   a) From Global Config Mode – **`sw1(config)#line vty # #`** (individual number, or range [as shown here])

g. Exec-Timeout – timeout period for each "line" session (Console, AUX, or VTYs)

1) From Sub-Config Mode for line configuring for – **`sw1(config-line)#exec-timeout # #`**, where # & # are minutes & seconds, respectively

h. Service Password Encryption – enabling basic encryption on Console, AUX, & VTY when not using login local

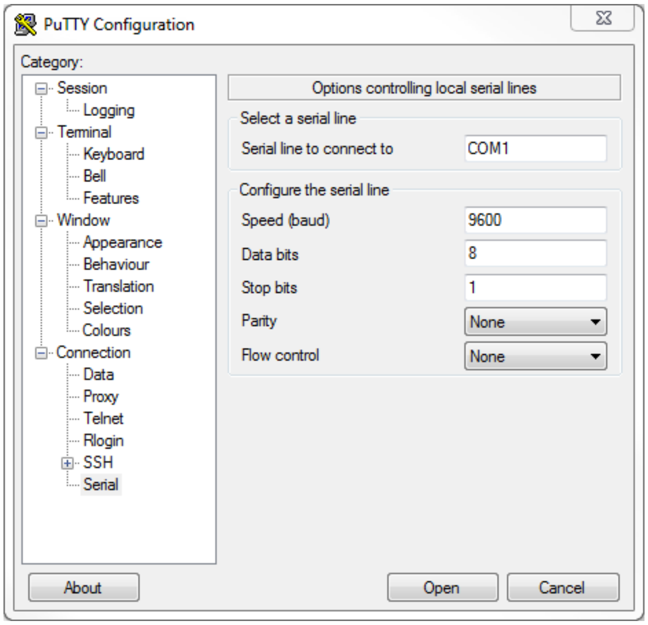
1) From Global Config Mode – **`sw1(config)#service password-encryption`**
2) Verify "lines" have pwds encrypted– **`sw1(config)#do show run`** (or simply **`show run`** from Privileged Mode)

i. Copy Run Start – saving changes made to Running Configuration (RAM) to Startup Configuration (NVRAM) so changes made to Running Config will be available if device is rebooted (reloaded)

1) From Privileged Mode – **`sw1#copy running-config startup-config`**

j. Ways to connect to Cisco devices:

1) Console/AUX – Serial connection to RJ45 port on back of Cisco device, with the following settings in PuTTy (9600 speed/data rate; No flow control; 8-bit ASCII/data; No parity; 1 stop bit):



PuTTY Configuration

Options controlling local serial lines

Select a serial line

Serial line to connect to: COM1

Configure the serial line

Speed (baud): 9600

Data bits: 8

Stop bits: 1

Parity: None

Flow control: None

2) VTY – SSH/Telnet session using PuTTy & configured IP of Cisco device for remote CLI

3) TFTP – remote server used to push/upload images & configs from/to the device

3) Web-based – end of life (Cisco Security Device Manager/Cisco Configuration Professional)

k. Other Notable Switch Commands:

1) Reload – reboot Switch, only from Global Config Mode

2) Keyboard shortcuts – CTRL-A = cursor to beginning of line; CTRL-E = cursor to end of line; CTRL-SHIFT-6 = interrupts current cmd (e.g. Telnet session to other device); Up/Down arrows = command history (buffer)

3) **`show version`** – displays general device info such as IOS version, uptime, reload info, # of interfaces, model/serial numbers

4) **`write erase`** or **`erase startup-config`** – erases startup-config file

5) **`ip host name IPAddress`** – adds a DNS entry to the device Hosts file

6) "Piping" command output – i.e. for show command → **`show running-config | section vty`** (or **`begin lin vty`**); shows cmd output only from the beginning of the piped output

2.4 – Verify Network Status & Switch Operation Using Basic Utilities Such As:

a. Ping – send ping to Switch (SW1) from PC or to PC from SW1 or to other Switch (or Router) from SW1

b. Telnet – test Telnet session to SW1 using PuTTy (only if Telnet is not disabled); or from User Mode of device (e.g. **`telnet IPAddressOfDevice`**)

c. SSH – test SSH session to SW1 using PuTTy or from User Mode of device (e.g. **`ssh –l username IPAddressOfDevice`**)

2.5 – Describe How VLANs Create Logically Separate Networks & the Need For Routing Between Them

a. Explain Network Segmentation & Basic Traffic Management Concepts

1) VLANs created separate Broadcast Domains, thus logical separate network segments
2) Because VLANs create separate Broadcast Domains, Layer 3 devices are only solution to route traffic between VLANs
   a) Need different Subnets per VLAN
3) Increase number of Broadcast Domains, but decrease each Broadcast Domain size, decreasing network Broadcast congestion
4) Flexible – in the use of Cisco equipment; less equipment to buy (e.g. Routers)
5) Ease of administration due to less equipment to manage
6) Allows creation of logical group of users by function, not necessarily location
7) User adds, moves, etc. easier

2.6 – Configure & Verify VLANs

a. Configure VLAN:

1) From Global Config of Switch: **`vlan #`**
2) Optionally give a description (name) for the vlan "group": **`name SomeName`**
3) Assign Interfaces to the VLAN:
   a) From Global Config: **`config t`**
   b) **`interface f#/# - #`** (for a range of interfaces, or just use **`f#/#`** for single interface)
   c) Assign the Interface(s) to the VLAN: **`switchport access vlan #`**
   d) To disable Interface(s) from being a trunk (optional): **`switchport mode access`**

SECOND OPTION

4) Go into Interface or Interface Range (range command shown): **`interface f#/# - #`**
5) Assign the Interface(s) to the VLAN: **`switchport access vlan #`**
   NOTE: This command creates the vlan if not already created

b. Verify:

1) Display Interfaces &/or VLANs: **`show vlan brief`** , **`show vlan #`** , or **`show running-config`** (not displayed as intuitively)
2) Verify Interface Mode configuration & VLAN association: **`show running-config | begin interface f#/#`** to show only the part of the config beginning with the Interfaces, or **`| section f#/#`** to only show the entered interface section explicitly

2.7 – Configure & Verify Trunking on Cisco Switches

a. DTP (Topic) – Dynamic Trunking Protocol

1) Protocol used to dynamically determine which trunking protocol type to use – ISL or 802.1Q
2) If both Cisco Switches support ISL & 802.1Q, ISL is used; otherwise, use the protocol both support (NOTE: ISL is mostly deprecated in today's Switches)
3) Manual Trunk Protocol configuration: **`switchport trunk encapsulation { dot1q | isl | negotiate }`**

b. Auto-Negotiation/Trunking Modes:

1) Trunk – switchport mode command to Administratively configure an Interface as a trunk port
2) Dynamic Desirable – Switches initiate trunk negotiation message
3) Dynamic Auto – Switches passively wait to receive trunk negotiation messages; this is the default Administrative Mode Interface setting
4) Administrative Mode – configuration setting of an Interface; uses **`switchport mode`** command
5) Operational Mode – what Mode the Interface is currently running as (access, trunk)
6) To display Interface Administrative/Operational Mode: **`show interfaces f#/# switchport`** (Interface number is optional, if only wanting to display for a single Interface)

c. Configure Trunking:

1) From Global Config: **`interface g#/#`**
2) **`switchport mode trunk`** or **`switchport mode dynamic desirable`** (if another Switch is connected to this port, the negotiation process will cause this port to initiate a trunk message to the other Switch, causing trunking to be enabled on this Interface & the Interface this is connected to on SW2)

TABLE OF EXPECTED PORT OPERATIONAL MODE BASED ON 2 SWITCHES ADMINISTRATIVE MODE CONFIGURATION

ADMINISTRATIVE MODE	ACCESS	DYNAMIC AUTO	TRUNK	DYNAMIC DESIRABLE
Access	Access	Access	Do Not Use	Access
Dynamic Auto	Access	Access	Trunk	Trunk
Trunk	Do Not Use	Trunk	Trunk	Trunk
Dynamic Desirable	Access	Trunk	Trunk	Trunk

NOTE: The 1st row & column lists each Switches Admin Mode config; the Mode in the subsequent boxes state the Operational Mode of the Port based on the 2 Admin Mode Switch Port configuration

d. Verify Trunking:

1) See Interfaces configured as a trunk: **`show interfaces trunk`**

e. Reasons Trunk won't pass certain VLAN traffic

1) VLAN not configured on Switch (verified by the **`show vlan`** command)
2) VLAN has been disabled (**`shutdown vlan #`**)
3) VLAN has been removed from the allowed VLAN list (**`switchport trunk allowed vlan except | remove #`**, where **`#`** is the number of the vlan created wanting removed)

SECTION III (11%) – IP Addressing

3.1 – Describe the Operation & Necessity of Using Private & Public IP Addresses (IPv4)

a. Private – "Internal", to an org, IP addresses, not used/routable in Public space

b. Public – All other IPs, not used for Private & not Class D or E

c. Operation – Private > Public accessibility is done using NAT (Network Address Translation) on Routers to forward Packets from internal Private IP Addresses to Internet (Public space)

d. Necessity (for using Private) – to conserve Public IP Addresses; build a layer of security

3.2 – Identify the Appropriate IPv6 Addressing Scheme to Satisfy Addressing Requirements in a LAN/WAN Environment

a. Subnet IPv6
1) Take the Global Prefix (e.g. /48) and use those bits as the first part of the IPv6 Address
2) For now, 64 bits are used as (Host) Interface ID part of the IPv6 Address (2^{64} possible Host Addresses)
3) Remaining bits (16 bits in this case) are used for Subnets (or 2^{16} = 65536 possible Subnets)
a) To find all Subnets, basically just use all possible (HEX) decimal combinations in the 4th "quartet" (8 quartets in an IPv6 Address; 128bits total; see Section 3.5 for more info)
4) So, an org generally uses /64 as Prefix Length (/48 assigned + /16 Subnetted); this length supplies more Subnets (Prefixes) and Hosts than an org needs & keeps the math simple
5) Generally, the Subnets begin with the Global Routing Prefix(e.g. 2001:0DB8:0000:1111), then: 0000, 0001, 0002, 0003, 0004, … 000A, 000B, … 000F; 0010, 0011, 0012, … ; then end ::/64
6) Using Unique Local (Private) Address – begin with **FD**, then any next 10 HEX digits (40 bits) & using the next 4 HEX (16 bit) digits for Subnetting as was done using a Global Routing Prefix

3.3 – Identify the Appropriate IPv4 Addressing Scheme Using VLSM & Summarization to Satisfy Addressing Requirements in a LAN/WAN Environment

IP CLASS	CLASS A	CLASS B	CLASS C	CLASS D	CLASS E
First Octet Range	**1-126**	**128-191**	**192-223**	**224-239**	**240-255**
Valid Network #'s	1.0.0.0 – 126.0.0.0	128.0.0.0 – 191.255.0.0	192.0.0.0 – 223.255.255.0	Reserved (Multicast)	Reserved (Experiment)
Total Networks	126 (2^7 – 2)	16,384 (2^{14})	2,097,152 (2^{21})	–	–
Hosts Per Network	16,777,214 (2^{24} – 2)	65,534 (2^{16} – 2)	254 (2^8 – 2)	–	–
Network Bits	8	16	24	–	–
Host Bits	24	16	8	–	–
Default Mask	255.0.0.0	255.255.0.0	255.255.255.0	–	–

a. CIDR – Classless Inter-domain Routing
1) Represented by: /## after an IP Address
a) Defines total number of "prefix" (1s) bits; Network + Subnet bits = P
b) All 1s bits are "to the left"; remaining bits are all 0s & are the remaining bits to the right of the total 32 bit address
c) Divides a Subnet in two parts → a *prefix* and *host* part (e.g. pppppppp.pppppppp.pphhhhhh.hhhhhhhh = /18)
1. Can be used to calculate # of subnets in a Classful Network → 2^S (S = # of Subnet bits)
2. Can be used to calculate number of Hosts per Subnet → 2^H – 2 (H = # of Host bits)

d) Is the Subnet Mask – e.g. /18 = 255.255.192.0 = 11111111.11111111.11000000.00000000

SUBNETTING CHART (Learn or be able to recreate for the Exam)

	Third Octet							.	Fourth Octet							
CIDR	/17	/18	/19	/20	/21	/22	/23	/24	/25	/26	/27	/28	/29	/30	/31	/32
Mask	128	192	224	240	248	252	254	255	128	192	224	240	248	252	254	255
Incrmnt	128	64	32	16	8	4	2	1	128	64	32	16	8	4	2	1
Hosts	32766	16382	8190	4094	2046	1022	510	254	126	62	30	14	6	2	-	-
SubB	2	4	8	16	32	64	128	256	512	1024	2048	4096	8192	16384	-	-
SubC	-	-	-	-	-	-	-	-	2	4	8	16	32	64	-	-

BINARY TO CIDR CONVERSION

BINARY (3rd & 4th Octet)		CIDR
10000000	00000000	/17
11000000	00000000	/18
11100000	00000000	/19
11110000	00000000	/20
11111000	00000000	/21
11111100	00000000	/22
11111110	00000000	/23
11111111	00000000	/24
11111111	10000000	/25
11111111	11000000	/26
11111111	11100000	/27
11111111	11110000	/28
11111111	11111000	/29
11111111	11111100	/30
11111111	11111110	/31
11111111	11111111	/32

b. Given an IP Address and Mask, calculate the Subnet ID, IP Range, & Broadcast

1) Convert Mask into in prefix/host format (255.255.252.0 → pppppppp.pppppppp.pppppphh.hhhhhhhh)
2) Convert IP Address into Binary – e.g. 130.4.102.1 (10000010.00000100.01100110.00000001) & write it below the p/h Mask format
3) Draw a line down, starting at the converted Mask, between the p & h in the Mask & down through the Binary IP Address
4) To the left of the line, copy the bits exactly as they are in the IP Address… 1 for 1s, 0 for 0s
5) To the right of the line, make all bits 0s (10000010.00000100.01100100.00000000)
6) Regardless of where the line is drawn, now convert each Octet of the new Binary into Decimal; This is the Subnet ID (64+32+4 = 100, so ID => 130.4.100.0)
7) To find the Broadcast Address for the Subnet, from Step 5, instead of placing all 0s to the right of the line, place all 1s to the right; as in Step 6, then convert each Octet of the new Binary to Decimal (64+32+4+2+1 = 103, so Subnet Broadcast = 130.4.103.255)
8) The **range of IP Addresses** for the Subnet is **1 more than the Subnet ID** & **1 less than the Subnet Broadcast Address** (130.4.100.1 - 130.4.100.254)
9) Faster conversion:
   a) Look at the Subnet Mask

b) Where there is a 255 in the Mask, simply copy the IP Address Decimal for that Octet
c) Where there is a 0 in the Mask, simply place a 0 for that Octet
d) Where there is neither a 255 or 0, subtract this "Interesting Octet" part of the Mask by 256 (256 - #) to get some number, which we will call a "multiple" (i.e. 128, 64, 32, 16, 8, 4, 2, 1)
e) Starting at 0, go in multiples of the number found in Step "d" up to the number in the IP Address Octet corresponding to the Interesting Octet in the Mask WITHOUT going over the number in the IP Address (e.g. 0, 32, 64, 96, etc.)
f) The number stopped at is the last number used in the Subnet ID
g) For example, given IP Address 130.4.102.1 and Mask 255.255.240.0, find Subnet & IP Range:
1. Since 255 is in 1st two Octets & 0 is in last Octet of Mask corresponding to the IP Address, we know part of the Subnet ID (see Step "b" & "c" above) → 130.4.?.0
2. To find the last number of the Subnet: 256 – 240 (Mask "Interesting Octet" number) = 16; so, go in multiples of 16 up to BUT NOT GOING OVER 102, the 3rd Octet DDN of the IP Address – 0, 16, 32, 48, 64, 80, 96, 112
3. Since 96 is the closest number without going over 102, the last number in the Subnet ID is 96 → 130.4.96.0 is the full Subnet ID
4. To find the Subnet Broadcast, take the octet number just found for the Subnet ID (96 in this ex.), add the number used as a multiple (16 in this ex.) then subtract 1 (96 + 16 = 112 – 1 = 111); the Subnet Broadcast = 130.4.111.255 (where there is a 0 in the Mask, place a 255)

c. When Using VLSM, Verify No Overlap/Add Additional Subnet ID
1) Given an IP or Subnet ID and Mask, calculate all Subnets and Subnet Broadcast Addresses
2) Determine if there is any overlap with current networking scheme
3) If none, then calculate new Subnets that may be added (lowest Subnet ID or highest Subnet ID) without overlap with current network scheme
a) To find a Subnet to be used with current network scheme, typically are given Masks to try
b) Calculate all Subnet IDs with the given Mask(s) and see if there is any overlap
c) Use the lowest or highest Subnet ID that doesn't overlap with current networking scheme, based on requirements of what the new/added VLSM Subnet should be

d. Route Summarization – decreases size of Routing Table by combining multiple (similar) subordinate Subnets into 1 encompassing Subnet with a new Mask ; increases Router performance; decreases Router resource utilization; saves bandwidth
1) Given Subnet IDs and Mask CIDR), calculate a new Mask that incorporates all Subnets into a single route
2) If Mask doesn't incorporate all Subnets in its IP Range of Addresses, decrease CIDR by 1 (i.e. go from /23 to /22) and recalculate to verify if all fall under the range

e. VLSM-Capable Routing Protocols
1) EIGRP, OSPF, RIPv2, IS-IS, BGP

f. IPv6 Routing Protocols
1) EIGRPv6, OSPFv3, RIPng

3.4 – Describe Technological Requirements For Running IPv6 in Conjunction With IPv4

a. Dual Stack – simply, running IPv4 & IPv6 in an environment simultaneously
1) IPv4 is enabled by default on Routers

2) To run IPv6, IPv6 must be enabled

a) From Global Config: `ipv6 unicast-routing` ; this is to enable routing for IPv6… has nothing to do with ability to add an IPv6 address, but added here cuz it's often overlooked

b) From the Interface, configure IPv6 Address: `ipv6 address ####:####:###:#::#/64` (NOTE: Address can be full 32bit HEX number or abbreviated; letters can be capitalized or lower case; the prefix length [/64 in this case] must be used & no space between the address & the prefix length)

c) Alternative Address cmd: `ipv6 address ####:####:###:#::#/64 eui-64` ; The #'s represent the 64-bit Prefix (Subnet), then the `eui-64` parameter tells the Router to auto-generate the remaining Interface ID part of the IPv6 Address from the device MAC

3) Verify IPv6 Config: `show ipv6 interface g0/0` (includes the prefix config)

b. NAT-PT Translation

c. 6-to-4 Tunneling

3.5 – Describe IPv6 Addresses

HEX > BINARY CONVERSION

HEX	BINARY	HEX	BINARY
0	0000	8	1000
1	0001	9	1001
2	0010	A	1010
3	0011	B	1011
4	0100	C	1100
5	0101	D	1101
6	0110	E	1110
7	0111	F	1111

a. IPv6 Addressing

1) Terms

a) Global Routing Prefix – block of Public addresses an org can request for routing use

b) Prefix – IPv6 Subnet ID

1. All devices must have the GRP as the first part of its full IPv6 Address

c) Prefix Length – Mask in CIDR notation; defines the part of the IPv6 Address that is the Subnet ID or Prefix

1. To find the IPv6 Subnet/Prefix, given Prefix Length, P → copy 1st P bits of the IPv6 Address, then change the rest of the bits to 0s (remember, each digit represents 4 bits)
2. If Prefix Length is multiple of 4, can think in terms of 1 (HEX) digit (i.e. digit is copied or 0)
3. For example, if Prefix Length = /64, divide by 4 = 16; so, 1st 16 digits are kept, rest are all 0s
4. Once Prefix (Subnet ID) is determined, it can be abbreviated as noted in item "2" below

d) Interface ID – Host portion of the IPv6 Address

e) Quartet – group of four digits in an IPv6 Address; 8 total quartets; each digit = 4 bits (e.g. 1 > 0001; A = 1010; etc), 16 bits per quartet x 8 quartets = 128 bits total per IPv6 Address

2) Address Abbreviation

a) Remove leading 0s to the LEFT of side of quartet (e.g. 0050 = 50; 0000 = 0)

b) Replace consecutive quartet of 0s with double colon [::] (e.g. AB45.0000.0000… = AB45::)

1. Can use only ONCE in the IPv6 Address
2. If there are multiple consecutive quartet of 0s, replace with :: where there is more consecutive quartets if separated by regular digits)
i. For example: AB01.0000.0000.4500.0000.0000.0000.0010 = AB01.0.0.4500::10 ; the double quotes were used where there were 3 consecutive 0 quartets, not in the beginning of the address where there were only 2

c) When abbreviating Prefix (Subnet ID), same rules apply EXCEPT if the IPv6 Address is broken up in the middle of a quartet (string of 4 HEX digits), the last 2 HEX digits are 0s and are written, THEN the :: are used (e.g. 0200.1234.5678.ABC9.1234.5678.ABCD.1111 /56 => 200.1234.5678.AB**00**::/56; 56 is a multiple of 4, which needs us to keep 14 HEX digits, which means we stop in the middle of the 4th quartet; we don't write …AB::… /56, but rather keep the trailing 0s in that 4th quartet, then use :: and then the Prefix Length of /56

b. Global Unicast – Public IPv6 IP Address
1) Unicast – concept that each address is used by only 1 Interface
2) Block of IPv6 Addresses assigned to an org by IANA, RIR, or ISP
3) Global Prefix

c. Multicast
1) Packet sent to multiple, but not all, devices on a Subnet
2) **FF02::1** = packet sent to all Host devices with IPv6 enabled
3) **FF02::2** = packet sent to all Routers with IPv6 enabled
a) **FF02::A** = packet sent to all Routers that only use IPv6 and EIGRP
b) **FF02::5** and **FF02::6** = packet sent to all Routers that only use IPv6 and OSPF
4) Solicited-Node Address = **FF02::1:FF** , then the last 6 HEX digits

d. Link Local
1) Addresses not used for normal packet flow; not routed
2) Used by some overhead protocols
3) Unicast address
4) Used by Neighbor Discovery Protocol (NDP), which is the IPv6 version of ARP
5) Auto-generated by a Host, always starting with FE80::/10 (i.e. **FE80**, **FE90**, **FEA0**, **FEB0**), then the next 3 HEX quartets are binary 0 > **FE80:0000:0000:0000:0000**
6) Interface ID portion (last 4 quartets) is auto-generated by eui-64 on Cisco devices; MS OS uses other randomized method

e. Unique Local – Private IPv6 IP Address
1) First 2 Hex numbers start with **FD**
2) Can use any of the next 40 bits (**10 HEX numbers**) as the Global ID
3) Last 64 bits of Address used for Interface ID (i.e. Hosts)
4) Middle 16 bits (4th "quartet") used for Subnetting

f. EUI 64 – Extended Unique Identifier, using a 64-bit prefix only
1) Auto-creation of Interface ID (Host) part of an IPv6 Address
a) Take the device MAC and split in 2 (6 digits on left, 6 digits on right)
b) Put **FFFE** in between the 2 sides (e.g. 1612.3456.789A > 161234 **FFFE** 56789A)
c) With the new 16 digit HEX number, this is the new/auto-generated Interface ID
d) Last step – write out the 1st 2 digits in binary (4 bits each), then invert the 7th bit

1. 1 6 = 0001 0110 ; invert the 7th bit ("1" in red) to 0, then convert back to HEX (0001 0100 > 14) NOTE: a Router Serial Interface doesn't have a MAC, thus a Router uses MAC for lowest numbered Interface that has a MAC (i.e. f0/0)

e) Final HEX Interface ID = 1412.34FF.FE56.789A

g. Auto-Configuration – Stateless Address Autoconfiguration (SLAAC)

1) Sub-interface Cmd: ipv6 address autoconfig

h. Unknown IPv6 Address – denoted by **::** (all 0s); analogous to a Host using an APIPA (169…) Address before it knows its own IPv6 Address

i. Loopback - ::1

j. Neighbor Discovery Protocol (NDP)

1) Router Solicitation (RS) – sent by a Host to FF02::2 to learn Router Link Local Address (Gateway/Default Router Info) & Prefix and Prefix Length
2) Router Advertisement (RA) – sent by Routers to FF02::1 in response to RSs
3) Neighbor Solicitation (NS) – sent by a Host requesting a target Host MAC; "neighbor" = same Subnet or 'Data Link' (VLAN)
   a) Duplicate Address Detection (DAD) – NS msg sent to check for duplicate IPv6 Address on Subnet
4) Neighbor Advertisement (NA) – response to NS with the target Host MAC
   a) NA also in response to NS for "DAD"; only a NA if there is a duplicate detected/used

SECTION IV (26%) – IP Routing Technologies

4.1 – Describe Basic Routing Concepts

a. Packet Forwarding (PC1 > PC2)

1) PC1 sends ARP request to Switch
2) Switch sends ARP Broadcast to its LAN (Frame Flood)
3) If Destination device is on LAN, the device (PC2) replies with MAC Address to be included in PC1's Destination MAC Address piece of the Ethernet/Layer2 Frame Header
4) Message is transmitted over the "wire"
5) If Destination device (PC2) is NOT on LAN, PC1's Default Router (R1) receives the ARP message & replies with its receiving interface MAC Address; or PC1 uses its ARP Table for R1's MAC
6) R1's MAC Address is then added to PC1's Destination Address part of the Layer2 Frame Header
7) Message is transmitted to R1
8) Frame Header is stripped from the Packet (IP info + Data) on R1
9) R1 reconstructs Frame using it's Routing Table and ARP Table
10) If Destination IP in message Packet is directly connected to R1, R1 sends ARP to Destination device (PC2) for its MAC Address; or R1 uses the MAC for PC2 in its ARP Table
11) PC2 MAC Address is then used in the message Framer Header
12) Message sent on the "wire" to PC2 and appropriate action is taken
13) If Destination IP in message Packet is NOT directly connected to R1, R1 uses its Routing and ARP Tables to reconstruct the Frame Header to include the MAC Address of the "next hop Router" (e.g. R2)

14) Step 13 process is repeated until the correct Destination is reached; Step 10 process is then implemented

b. Router Lookup Process

1) PC – open CMD Prompt in Admin Mode, then enter `route print` to display local Host routing table; enter `arp -a` to display ARP table
2) Router – from Privileged Mode, enter `show ip route` to display Router routing table; enter `show ip arp` to display ARP table

c. Process Switching/Fast Switching/CEF – Cisco Router internal routing logic variations

1) Process Switching – legacy; no routing optimizations
2) Fast Switching – more optimal routing logic that caches data link headers & uses other table than routing table
3) Cisco Express Forwarding (CEF) – default logic now; same as Fast Switching but also uses table organization in a tree structure

4.2 – Configure & Verify Utilizing CLI to Set Basic Router Configuration

a. Hostname – From Global Config Mode: `R1(config)#hostname SomeName`

b. Local User & Password – From Global Config Mode: `R1(config)#username shane password cisco` (or use `secret` to replace `password` to encrypt the local user password)

c. Enable Secret Password – from Global Config Mode: `R1(config)#enable secret SomePwd`

d. Console & VTY Logins – from Global Config Mode: `R1(config)#lin con 0` and for PuTTy sessions (i.e. VTY) : `R1(config)#lin vty 0 4`

1) Then do same procedure as stated in Section 2.3f. above, Configuring a Switch

e. Exec-Timeout – same as Section 2.3f. above, Configuring a Switch

f. Service Password Encryption – same as Section 2.3f. above, Configuring a Switch

g. Interface IP Address

1) From “sub” Config Mode (i.e. Int) – `R1(config-int)#ip address #.#.#.# #.#.#.#` NOTE: the first group of #s is the IP Address & 2nd group of #s is the Subnet Mask
2) Configure interface wanting to connect to – (e.g.) `R1(config)#int g0/0` or `f0/0` (e.g.)
3) Enter IP info – `R1(config-int)#ip address 10.100.10.1 255.255.255.0`
4) Bring interface up – `R1(config-int)#no shut`

h. Banner – Message displayed upon logging into User Mode on Router

1) MOTD – shown before the login prompt when connecting to Cisco device; intended as a temporary message (i.e. `Router down for maintenance`) ; from Global Config Mode – `R1(config)#banner #Some text here#`
2) Login – shown before the login prompt when connecting to Cisco device but after MOTD; used for permanent messages (i.e. `Unauthorized Access Prohibited`) ; from Global Config Mode – `R1(config)#banner login #Some text here#`
3) Exec – shown after logging in, intended to provide info hidden from unauthorized users; from

Global Config Mode – **R1(config)#banner exec #Some text here#**

i. Copy Run Start – saving changes made to Running Configuration to Startup Configuration so changes made to Running Config will be available if Cisco device is rebooted

1) From Privileged Mode – **sw1#copy run-config startup-config**

j. Enable SSH on Routers

1) Set domain name, from Global Config Mode: (**config t**)# **ip domain-name domain.local**
2) Create local user/pwd, if not already created (see item "d" above for procedure)
3) Generate cryptography key for Cisco device (requires device hostname + domain name) to encrypt traffic between Cisco device & SSH client : **crypto key generate rsa general-keys modulus 1024**
4) Connect to VTY lines: **lin vty #   #**
5) Configure the VTYs to logon locally: **login local**
6) Enable latest SSH version (more secure): **ip ssh version 2**
7) Enable SSH (i.e. disable Telnet): **transport input ssh**

4.3 – Configure & Verify Operation Status of an Ethernet Interface

a. Configuration

1) From Global Config, enter: **int f0/1** (e.g., or whatever interface wanting to configure & its annotation)
2) Assign IP Address/Subnet Mask, if required: **ip address #.#.#.# #.#.#.#**
3) Bring interface up/online: **no shut**
4) Set speed/duplex if desired: **speed 100** then **duplex full**

b. Verification – this can be done by using **show interface x#/#** (for appropriate interface)

4.4 – Verify Router Configuration & Network Connectivity Using

a. Ping – self-explanatory; used to test Layer1 & 2 issues to a destination

1) Extended Ping – in IOS; type **ping** then press Enter & prompted for extra options to enter

b. Traceroute – testing to isolate where network issue(s) exist in the network chain

c. Telnet – connect to other Cisco devices directly from current Cisco device (**telnet #.#.#.#**)

1) Once connected, press CTL+SHIFT+6 at same time, then X to suspend a session
2) Type **show sessions** to see what sessions are open; hit Enter or press 1 then Enter to resume the session; if multiple sessions open, press the # of session then Enter to resume that session
2) Enter **disconnect** to close Telnet sessions

d. SSH – connect to other Cisco devices directly from current Cisco device (**ssh –l username #.#.#.#**)

1) Suspend session (CTL+SHIFT+6, then X first) then enter **show session** to show what sessions are open; **resume #**, or Enter to resume last session

e. Show CDP Neighbors – list details about neighboring Cisco devices (Switches & Routers)

1) Device type → Router or Switch
2) IOS Version
3) Router/Switch type (2950, etc.)
4) IP Address
5) Shows only direct-connected devices, not tertiary devices connected to direct-connected devices

4.5 – Configure & Verify Routing Configuration For a Static or Default Route Given Specific Routing Requirements

a. Configure Static Route
1) From Global Config (**`config t`**): **`ip route 10.0.10.0 255.255.255.0 172.16.2.1`** (IPs in cmd → remote Network & Mask, Next Hop Router IP; or can define local Router outbound Interface)

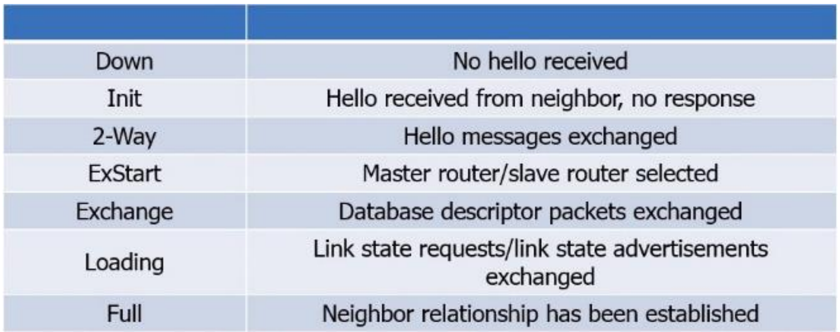
b. Configure Default Route
1) From Global Config (**`config t`**): **`ip route 0.0.0.0 0.0.0.0 172.16.2.1`**

c. To verify these routes, you can do “layered” ping tests, from a Workstation; first, ping to the Workstation Default Gateway (Router), then, if successful, ping out from the Router to the Subnet of the IP route configured on the Router
1) show ip route will display all routes in the Routing Table, including Connected (IPs assigned to Interfaces), Static, Default, & Dynamic Routes

4.6 – Differentiate Methods of Routing & Routing Protocols

a. Static vs Dynamic
1) Static – manually entered/created “ip route” to routing table
2) Dynamic – Internal Gateway Protocol (IGP); routes dynamically learned/added to routing table

b. Link State vs Distance Vector
1) Distance Vector (DV)
a) Routing Information Protocol (RIPv1) – based off number of (router) hops
b) Enhanced Interior Gateway Routing Protocol (EIGRP) – Cisco designed; Advanced DV Protocol
2) Link State
a) Open Shortest Path First (OSPF)
1. Neighbor Table – uses HELLO msgs to build ‘neighbor’ (Router) relationships by letting other Routers know its current State; Neighbor Table “states” shown below:



Down	No hello received
Init	Hello received from neighbor, no response
2-Way	Hello messages exchanged
ExStart	Master router/slave router selected
Exchange	Database descriptor packets exchanged
Loading	Link state requests/link state advertisements exchanged
Full	Neighbor relationship has been established

2. Link State Database (LSD) – built by propagating an LSA (Link State Advertisement) to neighbors based on info in the Neighbor Table
   i. LSA is a packet containing info about directly connected networks to the Router
   ii. Received by neighboring Router, add info to its LSD, increment a Seq # to LSA & fwd LSA on to its neighboring Routers
   iii. LSA Types → 1 = Router connections (i.e. Serial PPP); 2 = Network connections (i.e. Ethernet or Broadcast segment); others (3, 5, 7)
3. Routing Table – built from the LSD using Dijkstra/SPF algorithm

b) IS-IS – not really used

c. Next Hop – simply, next Router; metric used by RIP

d. IP Routing Table – database of routes for destination IP Addresses in routing Packets
1) Administrative Distance (AD) – measure of trustworthiness a router assigns to how a route to a network was learned; different protocols have a pre-determined AD:
   a) Connected = 0
   b) Static = 1
      1. NOTE: If the AD is manually entered in a Static Route cmd, it is referred to as a Floating Route; adding an AD to a Static Route is commonly done when adding a "backup" route
   c) EIGRP = 90 (external EIGRP = 170)
   d) OSPF = 110
   e) RIP = 120
   f) When more than 1 route exists & choosing a 'best' route, the Router will choose the protocol with the lowest AD value (i.e. in order shown above > Connected, Static, EIGRP, OSPF, etc.)
2) Metric – algorithm a protocol uses to determine the best path to a network; the metric is the "outcome value" of the algorithm; each protocol uses a different metric:
   a) EIGRP = feasible distance > bandwidth & delay default, reliability & load
   b) OSPF = cost → 10^8 (reference bandwidth #)/Interface bandwidth in bps
      i. OSPF Cost Ex's: 10Mbps = 10; 100Mps = 1; 1.544Mbps (T1 line) = 64; 64Kbps = 1562 (100,000,000/10Mps, etc.)
   c) RIP = hop count → # of routers a packet has to pass through to reach the destination
3) `show ip route` lists Route Table with Interface-assigned IPs (L), Connected networks (C), Dynamically-learned networks (O, D, R) & their AD & Metric [#/#], the remote network learned & its Mask, and the Interface out which the remote network was learned from

e. Passive Interfaces (How They Work) – OSPF; Router Interface that may not need to send/receive Neighbor Hello msgs or form Neighbor relationship, maybe because there is no Router off that Interface (i.e. Interface used for Trunking [i.e. ROAS])
1) Globally make all Interfaces passive in the OSPF Sub-Config, then enable on individual Interfaces: **`passive-interface default`** then enter **`no passive-interface s0/0/0`**
2) Or, just make individual interfaces passive: **`passive-interface g0/1`**
3) To see passive interfaces: **`show running-config`** or **`show ip ospf interface g#`**

4.7 – Configure & Verify OSPF (Single Area)

a. Benefits of Single Area
1) Simplicity

b. Configure OSPFv2 in a Single Area

1) From Global Config – `R1(config)#router ospf #` (# = some process ID number 1-65535)
2) Advertise directly connected routes from OSPF sub-router config
   - a) Enter `R1(config-router)#do show ip route connect` to see all directly connected routes
   - b) Enter routes to advertise – `R1(config-router)#network #.#.#.# #.#.#.# area #` where 1st # = directly-connected IP Address/Route, & 2nd # = *wildcard* Mask (i.e. inverse of the Mask – 255.255.255.0 = 0.0.0.255); area, at least for ICND1/CCNA studies will always = 0
3) Another way to configure OSPF is per Interface:
   - a) e.g. `int f0/0` or `int s0/1/0`
   - b) Sub-Interface cmd – `ip ospf 10 area 0`
4) Configure on ALL Routers so Neighbor Table can be built; Route Table should be similar on all Routers
5) To verify routes
   - a) See if neighbor relationships formed: `show ip ospf neighbor`
   - b) Check the LSD: `show ip ospf database` (router or network for "external" or Ethernet LAN)
   - c) Check Routing Table: `show ip route` [`connect`] ; also can use `show ip protocols`
6) Designated Router (DR)/Backup Designated Router (BDR)
   - a) Lowest "Priority" # = DR (default # = 1; if every Router has 1, highest Router-ID # is DR)
   - b) Configured: `ip ospf priority #` (NOTE: # = 0 means Router can never be a DR or BDR)
   - c) Updates sent to DR/BDR from ancillary Routers via multicast IP of **224.0.0.6**
   - d) Updates sent from DR to all Routers via multicast IP of **224.0.0.5**

c. Configure OSPFv3 in a Single Area

1) From Global Config, enable IPv6 Routing: `ipv6 unicast-routing`
2) From Global Config – `ipv6 router ospf #`
   - a) If Router doesn't have a RID, *must* configure RID: `router-id #` (# = 32 bit number)
   - b) If want to make an Interface "passive" (not use OSPF): `passive-interface g0/0`
3) Enable on each Interface: `ipv6 ospf # area 0`
4) For verification, same `show` cmds as with IPv4 are used but with `ipv6` in the cmd (e.g. `show ipv6 ospf database`)

d. Router ID – 32bit ID to define a Router identity used by OSPF; how determined what is used:

1) Manually configured is used 1st; if not configured, then next is…
2) If loopback interfaces (virtual interfaces) are configured AND shown as up, the highest IP Address of all loopback interfaces is used; if not configured/used, then next is…
3) Highest IP of any other interface used by the Router where at least the 1st status of the interface is shown as "up" (i.e. interfaces with status as up/down are considered)

e. Passive Interface – see previous Section (4.6)

4.8 – Configure & Verify inter-VLAN Routing (Router on a Stick or ROAS)

a. Sub Interfaces

1) To configure VLANs on Routers, create a subinterface to the "main" Interface (i.e. G0/0.10 for

VLAN 10); NOTE: the `.#` can be any number but for consistency, just use same # as VLAN ID #

a) **`interface g0/0.10`** (for trunking, create other interfaces as needed, i.e. **`g0/0.20`**, etc.)
b) **`encapsulation dot1q 10`** (NOTE: the "10" here denotes the actual VLAN ID; MUST be VLAN ID…this not an optional/random # as when configuring sub-Interfaces)
c) **`ip address 10.0.10.0 255.255.255.0`** (NOTE: use an IP from the VLAN subnet)
d) **`no shut`**

b. Upstream Routing – this simply means to not forget to configure other Router Interfaces static routes, as well as Routing protocols so packets can get forwarded properly

c. Encapsulation – protocol used for Trunking; dot1q or ISL; ISL is Cisco-proprietary & rarely used

4.9 – Configure SVI Interfaces

a. Switched Virtual Interfaces (SVI) – virtual interfaces on Layer3 Switches used/configured for Routing; also called VLAN Interfaces
1) Enable hardware support for routing on the Switch – **`sdm prefer lanbase-routing`** ; then **`reload`** (reboot)
2) Enable routing 'globally' – **`ip route`**
3) Create a VLAN for each VLAN to be routed on the Switch – **`interface vlan ##`**
4) Configure an IP Address/Mask for each VLAN Interface – **`ip address #.#.#.# #.#.#.#`**
5) "Enable" the interface - **`no shut`**

SECTION V (8%) – IP Services

5.1 – Configure & Verify DHCP (IOS Router)

a. Configuring Router Interfaces to Use DHCP – from Subinterface (i.e. **`f0/0`**): **`ip address dhcp`**

b. DHCP Options (Basic Overview & Functionality); From DHCP Pool (**`ip dhcp pool name`**)
1) Define the subnet supported: **`network #.#.#.# #.#.#.#`** or **`/##`** (IP then Mask or CIDR)
2) Define Default Gateway/Router of Subnet: **`default-router #.#.#.#`** (# = Router IP)
3) Define DNS Server of Subnet: **`dns-server #.#.#.#`** (# = DNS Server IP)

c. Excluded Addresses
1) From Global Config: **`ip dhcp excluded-address #.#.#.# #.#.#.#`** (first, last of IP range)

d. Lease Time
1) From DHCP Pool (**`ip dhcp pool name`**): **`lease # # #`** (days hours minutes)

e. Verification:
1) **`show ip dhcp pool`** – lists DHCP Pool info
2) **`show ip dhcp binding`** – lists IP Lease info
3) **`show ip dhcp conflict`** – lists which IPs Router was going to Lease but upon (3) pings found already in use on the network
4) **`show ip dhcp statistics`** – lists DHCP Resource usage (i.e. RAM, etc.) and # of DHCP msgs

5.2 – Describe the Types, Features, & Applications of ACLs

a. Standard (Editing & Sequence Numbers) – Named or Numbered
1) Numbers used: 1-99, 1300-1999
2) Filters by looking only at Source IP
3) Cisco best practice is to enable as close to filter *destination* as possible so as not to filter traffic not intending to

b. Extended
1) Numbers used: 100-199, 2000-2699
2) Can filter on Source or Destination IP, Source or Destination Ports, & Protocol
3) Cisco best practice is to enable as close to filter *source* as possible

c. Named
1) Standard or Extended
2) Used instead of Numbers for better recognition/description
3) Can modify ACL per line

d. Numbered – See "a." & "b." above for a bit more info; see below (Section 5.3) for configuration

e. Log Option – to simply add the **`log`** parameter at the end of the **`access-list`** cmd

5.3 – Configure & Verify ACLs in a Network Environment

a. Named
1) Standard (remember, filters only on source IP):
a) From Global Config: **`ip access-list standard ACLNAME`** to enter **`config-std-nacl`** sub Config Mode
2) Extended:
a) From Global Config: **`ip access-list extended ACLNAME`** ; then enter a **`config-ext-nacl`** sub Config Mode
3) Enter permit & deny entries (Extended lists have more options)
a) Standard:
1. Example – **`deny 10.0.0.11`** or **`permit 10.0.0.0 0.0.0.255`**
2. The end is an "implied" deny any; recommend to add this cmd explicitly to receive deny counters
b) Extended:
1. Example – **`deny ip 192.16.10.10 0.0.0.0 10.128.50.1 0.0.0.0`** (any packet from specific Host)
2. Example – **`permit udp 192.16.10.0 0.0.0.255 host 10.128.50.202 eq 53`** (allows DNS traffic *only* from whole 192 Subnet to specific DNS server)
3. NOTE: **`host`** parameter in cmd or Wildcard Mask of `0.0.0.0` = explicit Host
4) Assign to proper Interface & proper direction (in/out): **`ip access-group ACLNAME out`** (or **`in`**)
a) Do **`show ip access-lists`** to see the list with counters; if counters show, clear them for testing purposes (From Global Config Mode: **`clear access-list counters`**)

5) General Cisco rules ir: Access Lists
a) Standard Lists → apply Rule close to DESTINATION Address (prevent unwanted Filtering)
b) Extended Lists → apply Rule close to SOURCE Address (save Bandwidth)

b. Numbered
1) From Global Config: **`access-list # permit | deny sourceIP wildcardMask`**
e.g. **`access-list 1 permit 10.1.1.0 0.0.0.255`** (no WC Mask needed for single Host)
a) Or, can enter: **`ip access-list standard #`**
1. NOTE: the **`#`** parameter must match the numbering scheme for Std (i.e. 1-99, 1300-1999)
b) When using the "IP Format", remaining configuration is in an ACL Sub-config as noted in the "Named" Section above; enter permit & deny cmds like was stated above
c) The **`#`** in the **`access-list`** cmd determines if the ACL is Standard or Extended
2) Add as many more entries as needed; the end is an "implied" **`deny any`** (all); it is recommended to explicitly add **`access-list deny any`** as a last item to view metrics as the implied deny isn't listed with show commands
3) Lines can be added if forgotten or ACL needs modified, but are added at the end of the list unless sequence (line) number is explicitly entered in the access list cmd (same for Named lists)

c. Log Option – as stated in Section 5.2, just add the log parameter at the end of the access list cmd

5.4 – Identify the Basic Operation of NAT

a. Purpose – simply to enable Private-IP internal org devices to communicate externally to the Internet

b. Pool – assigning a pool/range of public IPs to internal Local devices

c. Static – manually adding address translation rules, mapping 1 Inside Local (Private) Address to an Inside Global (Public) IP Address

d. 1 to 1 – Basically, this is Static NAT mapping an Inside Local (Private) Address to an Inside Global (Public) IP Address; for an Enterprise with large amount of devices needing to communicate on the Internet, this defeats the purpose of NAT & conserving IP Addresses

e. Overloading – Dynamic NAT with Overload (Port Address Translation [PAT]) that allows NAT to scale for large number of internal devices to communicate on the Internet using Ports, with only few Public Internet Addresses needed

f. Source Addressing – know NAT terminology
1) Inside Local – this is the Private IP of the internal computing device (Host or Server)
2) Inside Global – this is the Internet-routable Public IP assigned to an org by the ISP
3) Outside Local – this is the IP of the destination device
4) Outside Global – this is also the IP of the destination

g. One-Way NAT – I think what Cisco is suggesting here is a 1-to-1 Static NAT assignment (see below)

5.5 – Configure & Verify NAT For Given Network Requirements

STATIC NAT

a. Configure

1) Determine inside & outside network (LAN & Internet) on Router interfaces
2) Go into the Sub-Interface configuration mode for the inside Interface
   a) Make the Interface the "inside" (LAN) part of NAT: `ip nat inside`
3) Go to the Internet-facing Interface & make it the outside part of NAT: `ip nat outside`
4) From GLOBAL Config Mode, enter the rule(s): `ip nat inside source static 192.168.10.100 203.0.113.70` (Inside Local Address, Inside Global Address respectively)

b. Verify:

1) `show ip nat translations`
2) `show ip nat statistics`

DYNAMIC NAT

a. Configure

1) Assign NAT inside & outside to Router Interfaces: `ip nat inside` ; `ip nat outside`
2) Create a standard ACL to identify traffic to be NAT'd
   a) `access-list 1 permit 10.1.1.2`
   b) `access-list 1 permit 10.1.1.3`
3) Create a pool of Public IPs for the NAT
   a) `ip nat pool PoolName PublicIP#1 PublicIP#2 netmask 255.255.255.248`
      The 1st & 2nd Public IP #'s represent a RANGE of IP Addresses used for the Pool
4) Enable Dynamic NAT associating the ACL created in #2 with the Pool created in #3
   a) `ip nat inside source list 1 pool PoolName`
5) To start with fresh NAT stats: `clear ip nat translation *`

b. Verify – again, can use the `show ip nat translations` cmd to view the traffic

1) From Global Config Mode: `clear access-list counters` to start from a fresh list to really verify traffic is working as intended

"PAT" (OVERLOAD)

a. Configure

1) Since PAT (Dynamic NAT with Overload) can use over 65000 concurrent connections on a single Inside Global (Public) IP, it is generally best to not use a Pool in the PAT Configuration steps
2) To configure, use same steps as regular Dynamic, but skip Step #3 (Pool creation)
3) Change the Step 4 cmd to resolve the ACL solely to 1 IP, or more specifically, to the Router Interface which is configured with the Inside Global (Public) IP, then add "overload" parameter
   a) `ip nat inside source list 1 interface S0/0/0 overload`

5.6 – Configure & Verify NTP as a Client

a. Configure

1) Config Mode: `ntp server #.#.#.# version 4` , where the #'s is the IP of the NTP Server

b. Verify

1) `sho ntp status` or `sho ntp associations`

SECTION VI (15%) – Network Device Security

6.1 – Configure & Verify Network Device Security Features

a. Device Password Security – covered already; use **`enable secret`**, **`security password-encrypt`**, and **`username secret`** for device password security

b. Enable Secret vs Enable – (Password for entering Privileged Mode); if both are configured, password associated with the Secret command overrides one associated with password command
1) The **`enable secret`** cmd in Config Mode when logging into EXEC Mode stores pwd in MD5
2) Use **`username secret SomePwd`** when creating local device accounts (though it's recommended to use some external authentication server [RADIUS]) for user authentication

c. Transport
1) Disable Telnet – from VTY, **`transport input ssh`** (or **`none`** instead of using "ssh"; but disables both SSH & Telnet)
2) SSH – to enable SSH, see Section 4.2, "k" above

d. VTYs
1) Enable SSH & use password, preferably one that is encrypted when viewing running-config (i.e. **`username name secret pwd`**)
2) Use an ACL to limit access to only Networking Staff
   a) Create a Standard ACL allowing IPs or Subnet access (e.g. **`access-list 3 permit 10.1.1.0 0 0.0.0.255`**)
   b) In **`lin vty`** Mode, enter **`access-class 3 in`** , referring to the access-list created in "a."

e. Physical Security
1) This just has to deal with having adequate security in a facility/rooms that house network equipment → locks, codes, badges, alarms, etc.
2) Physical access should be limited to only authorized personnel

f. Service Password
1) From Global Config Mode – **`R1(config)#service password-encryption`**
2) Verify "lines" have pwds encrypted – **`R1(config)#do show run`** (or simply **`show run`** from Privileged Mode
3) Good practice of Cisco device services to disable (from Global Config Mode):
   a) **`no ip source-route`**
   b) Disable "Finger" – **`no ip finger`** then **`no service finger`**
   c) **`no ip bootp server`**
   d) Disable Web Server – **`no ip http server`**
   e) Disable DNS – **`no ip name-server`**
   f) Disable SNMP – **`no snmp-server`**
   h) Disable "small" services (Echo, e.g.): **`no service tcp-small-services`** then same cmd using **`udp`** instead of **`tcp`**
   i) Disable CDP on an Interface: **`int g0/1`** then **`no cdp enable`**

g. Describe External Authentication Methods – this is simply enabling authentication via AAA (authentication , authorization, & accounting) server; i.e. an LDAP server using RADIUS/TACACS

h. DHCP Snooping

1) Go into Config Mode on Switch & enable DHCP Snooping: **`ip dhcp snooping`**
2) Create a VLAN to put "clients" (devices connected to Ports) in: **`vlan 10`** , then exit VLAN Config Mode
3) Enable Snooping in the "clients" VLAN from Global Config: **`ip dhcp snooping vlan 10`**
4) Place "clients" Ports in the VLAN: **`int range f0/1 – 4`** ; **`switchport access vlan 10`**
5) Give "trust" DHCP access to the Port connected to a valid DHCP Server: **`int f0/3`** , for ex.
6) "Trust" DHCP for the Port: **`ip dhcp snooping trust`**

6.2 – Configure & Verify Switch Port Security

a. Sticky MAC – Switch dynamically learns/discovers MAC on each Port

1) From Subinterface config mode:
   a) **`switchport mode access`**
   b) **`switchport port-security`**
   c) **`switchport port-security mac-address sticky`**

b. MAC Address Limitation

1) From Subinterface config mode:
   a) **`switchport mode access`**
   b) **`switchport port-security`**
   c) **`switchport port-security maximum 2`**

c. Static/Dynamic

1) Static – **`switchport port-security mac-address 0200.1111.1111`**
2) Dynamic – **`switchport mode access`** then **`switchport port-security`**

d. Violation Modes

1) Err Disable – shuts down interface; result of Shutdown violation; to re-enable, issue **`shut`** then **`no shut`**
2) Shutdown – this is the actual name of the Mode, but the interface is placed in an err-disabled state; recovering from this state is listed in "1" above
3) Protect
4) Restrict

Violation Mode/Action	Protect	Restrict	Shutdown
Discards offending traffic	Yes	Yes	Yes
Sends log/SNMP message	No	Yes	Yes
Disables interface	No	No	Yes

e. Shutdown Unused Ports – go into a range of interfaces & shutdown:

1) **`interface range f0/4 - 15`**
2) **`shutdown`**
3) Verify from Global Config Mode – **`show interfaces status`**

f. Err Disable Recovery

1) **`shut`** command on Port/Interface, then **`no shut`**

g. Assign Unused Ports in Unused VLANs
1) Create some unused VLAN: **`vlan 99`**
2) Give the VLAN a name for added description: **`name unused`**
3) Go into multiple Interfaces (range & select ones): **`int range f0/1 - 24 , g0/1 , g0/2`** (spaces needed between dashes and commas)
4) Now, assign the selected Ports to the unused VLAN: **`switchport access vlan 99`**

h. Putting Native VLAN to Other Than VLAN1
1) Check the Interface Trunking is configured on: **`sho int trunk`**
2) Go into Config Mode: **`conf t`**
3) Go into the Interface that is Trunk'd: **`int f#/#`**
4) Change the Native VLAN: s**`witchport trunk native vlan ##`** , where ## is the new native VLAN number

6.3 – Configure & Verify ACLs to Filter Network Traffic

a. Configure – review Sections 5.2 & 5.3 for Standard & Extended ACLs

b. Verify – review Sections 5.2 & 5.3; use the **`show access-lists`** cmd, which also shows counters

6.4 – Configure & Verify ACLs to Limit Telnet & SSH Access to the Router

a. Create an ACL allowing IPs or a Subnet access: **`access-list 3 permit 10.1.1.0 0.0.0.255`**
b. From **`lin vty`** Mode, enter **`access-class 3 in`**, referring to the access-list created in "a."
1) If using **`out`** parameter, the IP that is to be matched to filter is the Destination IP, not Source

c. To Verify ACL, simply test if able to connect to Router via SSH or Telnet

SECTION VII (13%) – Troubleshooting

7.1 – Troubleshoot & Correct Common Problems Associated With IP Addressing & Host Configurations

a. Potentially Useful Show Commands
1) **`show int g#/#`** – shows interface status (up, down, etc) and line status (up, down, etc.) which corresponds to Layer 1 and Layer 2 status respectively
2) **`show cdp neighbors detail`** – shows info related to directly connected Cisco devices
3) **`show mac-add`** / **`show mac-add dynamic`**
4) **`show arp`**
5) **`show version`** – shows software version, uptime, model #, Layer-based IOS image, memory info, S/N of device, & configuration register (3 register code options: 0x10F,)
6) Check IPs are in the correct Subnet
7) Check that Subnets do not overlap

8) Check Default Gateway IP (in right Subnet? is it even configured?)

b. NAT

1) Use of `show ip nat translations` or `show ip nat statistics` cmds

2) If wanting to look at fresh counters – `clear ip nat translation*` for Dynamic only

7.2 – Troubleshoot & Resolve VLAN Problems

a. Identify VLANs Are Configured

1) List VLANs and their “state” (active/lshut) – `show vlan` or `show vlan brief`

2) Check ‘allowed’ VLANs on a Trunk – `show interfaces g#/# trunk`

3) Check Admin vs Operational config of Interfaces – `show int f#/# switchport`

4) If you notice an inactive VLAN, enable it – `no shut` Subinterface cmd or `no shut vlan #`

5) Is Trunking enabled on the appropriate Interface

b. Verify Port Membership is Correct

1) Again, use `show vlan` to see what VLAN each port is assigned to

2) If port in wrong VLAN, add to proper one – `switchport access vlan #` sub interface cmd

3) Add VLANs to Trunk Port if needed – `switchport access vlan add #,# - #`

c. Correct IP Address is Configured

1) Simply, use `show interface vlan #` on vlan configured the mgmt IP on

7.3 – Troubleshoot & Resolve Trunking Problems on Cisco Routers

a. Verify Trunk States

1) Use `show` cmds: `show interfaces trunk` or `show vlan` will show state (Active, etc.)

2) Correct Switch Interface enabled?

3) Router enabled for ROAS?

a) Subinterface(s) configured appropriately?

b) Subnet IP Address appropriate?

b. Verify Correct Encapsulation is Configured

1) `show interfaces trunk` will show the Native VLAN & Encapsulation type

2) If not properly set, go into the Interface using as Trunk: `int f0/24`

3) Set Encapsulation: `switchport trunk encapsulation dot1q`

4) Explicitly (i.e. not Auto) set the Port Mode to Trunk: `switchport mode trunk`

c. Correct VLANs Are Allowed

1) Again, the `show interfaces trunk` will also show the allowed VLANs

2) To allow: `switchport trunk allowed vlan #` (range: `# - #` ; explicit group: `#,#`)

3) If missed allowing a VLAN, to add: `switchport trunk allowed vlan add #`

7.4 – Troubleshoot & Resolve ACL Issues

a. Troubleshooting Steps

1) Use `show access-list #` cmd to display ACL stats

2) Correct Network and/or Mask in the ACL?

3) Applied on the correct Interface, and in correct direction (`in` or `out`)? `sho run int f0/0`

a) When applied to Interface & using Named, make sure of ACL Name spelling or won't work

b) When applied to Interface, is cmd syntax correct (ip **access-group** …), & the list # in the cmd

4) Correct usage?

a) Standard – 1-99 or 1300-1999 numbers used in ACL statements?

b) Extended – 100-199, 2699

c) Because of implicit 'deny all' at end of ACLs, if ACL has nothing but 'deny' statements, all traffic will be denied

b. Tests

1) Ping, Traceroute, nslookup tests

7.5 – Troubleshoot & Resolve Layer 1 Problems

a. Framing – not entirely sure what Cisco is inferring here

b. CRC – counter increments when received frames do not pass the FCS math, caused by collisions; Frame trailer check

c. Runts – a Frame doesn't meet the minimum size requirement (64bytes); increment = collisions; could be a duplex mismatch indicator

d. Giants – Frame exceeds maximum size requirement (1500bytes)

e. Dropped Packets – counter, means what it says; host of reasons for this (loose cable, EMI, etc.)

f. Late Collisions – if counter increases, typically means duplex mismatch

g. Input/Output Errors

1) Input – total of several counters → runts, giants, CRC, overrun, & ignored

2) Output – # of Frames port tried to transmit but unable to due to some problem

3) If CRC errors grow, but collision counter does not, more than likely cable interference issue

h. Why a Ping won't work?

1) Cable loose

2) Cable plugged in wrong Port

3) Routing issue > Gateway not set somewhere (PC, Switch)

INTERFACE LINE/PROTOCOL STATUS

Interface Status	Line Protocol Status	Resulting Link State
Up	Up (connected)	Functioning
Up	Down	Layer 2 Issue > HDLC vs PPP config; clock rate
Down	Down	Unplugged; remote end not enabled/on
Administratively Down	Down	Locally disabled

Created by Shane Williford 19 October 2014. If you reference this Study Guide, please give credit to the author.