1. Campus Network Overview
   * PDNTSPA
   * Protocol Data Units
     + 4 (Transport) 🡪 TCP Segment / TCP port
     + 3 (Network) 🡪 Packet / Router
     + 2 (Data Link) 🡪 Frame / Switch/bridge
   * Segment 🡪 broadcast domain
   * VLAN isolates broadcast domain
   * Each switched port is a collision domain
2. Modular Network Design
   * Switch block 🡪 group of access layer switches with their distribution switches. These all connect to the core block
   * Core block 🡪
   * Server farm block 🡪
   * Management block 🡪
   * Enterprise edge block 🡪
   * Service provider edge block 🡪
   * Don’t extend VLANs beyond distribution switches
3. Switch Operation
   * CAM table learns source MACs as frames arrive
   * CEF builds L3 destination information from routing tables and L2 data. This is stored in the Forwarding Information Base (FIB)
4. Switch Port Configuration
5. VLANs and Trunks
   * 801.1Q encapsulation adds 4-byte tag including 12-bit VLAN ID
6. VLAN Trunking Protocol
7. Aggregating Switch Links
   * Configuring Etherchannel

**lacp system-priority** priority

**int port-channel** port-channel-number

**int** media-type slot/port

**channel-group** number **mode {on | passive | active}**

**channel-protocol** lacp

**port-channel load-balance** load-balance-type

**show running-config interface port-channel** channel\_number

**show etherchannel** num **port-channel**

* Members must have same VLAN, STP, speed, duplex, settings to be bundled
* Load distribution is deterministic, but not necessarily equal
  + Determined by hashing algorithm
    - Can consist of combinations of source IP address, destination IP address, or a combination of source and destination IP addresses, source and destination MAC addresses, or TCP/UDP port numbers
    - The hash algorithm computes a binary pattern that selects a link number in the bundle to carry each frame
* EtherChannels can be negotiated between two switches to provide some dynamic link configuration. Two protocols are available to negotiate bundled links in Catalyst switches.
  + Port Aggregation Protocol (PAgP) 🡪 Cisco proprietary
    - Modes
      1. Auto 🡪 Default. Passive negotiating state. Responds to PAgP but does not initiate
      2. Desirable 🡪 Actively negotiates by sending PAgP packets
      3. On 🡪 Forces channel creation and does not exchange PAgP packets
  + IEEE 802.3ad Link Aggregation Control Protocol (LACP)
    - Modes
      1. Passive 🡪 Default. Passive negotiating state. Responds to LACP but does not initiate
      2. Active 🡪 Actively negotiates by sending LACP packets
      3. On 🡪 Forces channel creation
  + You also can configure an EtherChannel to use the on mode, which unconditionally bundles the links. In this case, neither PAgP nor LACP packets are sent or received.

**Display Function**

Current EtherChannel status of each member port

Time stamps of EtherChannel changes

Detailed status about each EtherChannel component

Load-balancing hashing algorithm

Load-balancing port index used by hashing algorithm

EtherChannel neighbors on each port

LACP system ID

**Command Syntax**

**show etherchannel summary show etherchannel port**

**show etherchannel port-channel**

**show etherchannel detail**

**show etherchannel load-balance**

**show etherchannel port-channel**

**show {pagp | lacp} neighbor**

**show lacp sys-id**

1. Traditional STP
   * Traditional spanning tree defined by IEEE 802.1D
   * Single STP instance referred to as Common Spanning Tree (CST)
   * Cisco proprietary version Per-VLAN Spanning Tree (PVST) runs one instance per VLAN
     + Requires ISL, doesn’t interoperate with CST
   * Cisco proprietary PVST+ allows interoperability between CST and PVST
   * Bridge ID is an 8-byte value composed of:
     + 2-byte bridge priority 🡪 0 – 65,635 [default is 32,768]
     + 6-byte MAC address 🡪 from the pool of 1,024 on the switch
   * Root bridge 🡪 elected as top of network tree by lowest bridge ID
   * Port roles
     + Root port 🡪
     + Designated port 🡪
   * STP States
     + Disabled 🡪 Can’t send or receive data
     + Blocking 🡪 Receives BPDUs only. Ports enter this state to avoid a loop
     + Listening 🡪 Sends and receives BPDUs but can’t send data or learn MAC addresses
     + Learning 🡪 Sends and receives BPDUs and learns MAC addresses but can’t send data
     + Forwarding 🡪 Sends and receives BPDUs, learns MAC addresses, and sends and receives data
   * Steps in STP process
     + Elect root bridge
     + Each switch selects root port
     + Each segment elects designated port
     + Block all ports with loops (any non-root or non-designated port gets blocked)
   * Timers 🡪 Once set on the root bridge, these get propagated throughout via BPDUs
     + Hello timer (default 2s) 🡪 triggers periodic “hello” BPDU sent from the elected root to the other bridges
     + Forward delay timer (default 15s) 🡪 amount of time a port stays in Listening state before moving to Learning state and how long it stays in Learning before moving to Forwarding
     + Max age timer (default 20s) 🡪 Max length of time a BPDU can be stored without receiving an update. Expiration indicated an indirect failure with designated or root bridge
2. STP Configuration
   * Always force configuration of primary and secondary root bridges
     + **spanning-tree vlan** 10,20,30 **priority** priority
     + **spanning-tree vlan** 10,20,30 **root** {**primary** | **secondary**} [**diameter** diameter]
   * Manually configuring STP timers
     + **spanning-tree** [**vlan** vlan-id] **hello-time** seconds
     + **spanning-tree** [**vlan** vlan-id] **forward-time** seconds
     + **spanning-tree** [**vlan** vlan-id] **max-age** seconds
   * PortFast 🡪 Set for access ports to immediately move into forwarding state, but still run STP. Also prevents TCN BPDUs from being sent. Automatically enabled for switchport mode access
   * UplinkFast 🡪 Used on access layer switches to enable fast uplink port failover
   * BackboneFast 🡪 Used on core for fast convergence after STP topology change
3. Protecting STP
   * Port Roles
     + Root port 🡪
     + Designated port 🡪
     + Blocking port 🡪
     + Alternate port 🡪
     + Forwarding port 🡪
   * Root Guard 🡪 Used to prevent switches from becoming STP root
   * BPDU Guard 🡪 Detects BPDUs received on a port that shouldn’t receive BPDUs and shuts the port down to prevent any loops. Automatically enabled by PortFast
   * Loop Guard 🡪 Tracks BPDU activity on non-designated ports and moves them into loop-inconsistent state if BPDUs stop.
   * UDLD 🡪 If a link becomes unidirectional (fiber optic RX removed), it will stop receiving BPDUs and could come out of blocking and form a loop. UDLD echoes an L2 UDLD frame that it expects to hear back to make sure the link is bidirectional.
4. Advanced STP
   * RSTP 802.1w
     + Uses BPDU version 2 to distinguish from 801.2D (can coexist)
     + Port roles
       1. Root port 🡪
       2. Designated port 🡪
       3. Alternate port 🡪 A port that has an alternate, less desirable path to the root
       4. Backup port 🡪
     + Port states
       1. Discarding 🡪 Essentially combines disabled, blocking, and listening states of 802.1D. Drops all frames
       2. Learning 🡪 Incoming frames are dropped, MACs learning
       3. Forwarding 🡪 Frames forwarded, MAC learning
     + Port types
       1. Edge port 🡪
       2. Root port 🡪
       3. P2P port 🡪
5. Multilayer Switching
6. Router/Supervisor/Power Redundancy
7. IP Telephony
8. Securing Switch Access
9. Securing with VLANs
10. Wireless LAN Overview
11. Wireless Architecture and Design
12. Cisco Unified Wireless Network

* L3 High Availability

Packet-Forwarding Review  When a host must communicate with a device on its local subnet, it can generate an Ad- dress Resolution Protocol (ARP) request, wait for the ARP reply, and exchange packets directly. However, if the far end is located on a different subnet, the host must rely on an intermediate system (a router, for example) to relay packets to and from that subnet.  A host identifies its nearest router, also known as the default gateway or next hop, by its IP address. If the host understands something about routing, it recognizes that all packets destined off-net must be sent to the gateway’s MAC address rather than the far end’s MAC address. Therefore, the host first sends an ARP request to find the gateway’s MAC address. Then packets can be relayed to the gateway directly without having to look for ARP entries for individual destinations.  If the host is not so savvy about routing, it might still generate ARP requests for every off- net destination, hoping that someone will answer. Obviously, the off-net destinations can- not answer because they never receive the ARP request broadcasts; these requests are not forwarded across subnets. Instead, you can configure the gateway to provide a proxy ARP function so that it will reply to ARP requests with its own MAC address, as if the destination itself had responded.  Now the issue of gateway availability becomes important. If the gateway router for a sub- net or VLAN goes down, packets have no way of being forwarded off the local subnet. Several protocols are available that allow multiple routing devices to share a common gate- way address so that if one goes down, another automatically can pick up the active gate- way role.

* + HSRP default priority is 100
  + HSRP election is based on a priority value (0 to 255) that is configured on each router in the group
  + Default priority is 100
  + The router with the highest priority value (255 is highest) becomes the active router for the group
  + If all router priorities are equal or set to the default value, the router with the highest IP address on the HSRP interface becomes the active router
* Switch(config-if)# standby group priority [priority]