

CCNP Switching Exam

The Basics

- **Ethernet**
 - IEEE 802.3
 - 10mbps
 - Possibility of collisions without switches
 - Switches create their own collision domains via each port having it's own.
 - Uses UTP
 - Length 100m
 - Mostly found at access layer
- **Fast Ethernet**
 - IEEE 802.3u
 - 100mbps – full duplex 200mbps
 - UTP / Fiber
- **Gigabit Ethernet**
 - 1000mbps
 - STP (shielded)
 - Multimode fiber (MMF) 50-62.5 micron core
 - Singlemode fiber (SMF) with an 8/9/50 micron core
- **10 Gib**
 - Only works with fiber
 - Only works with full duplex
- **Long range Ethernet**
 - Uses preexisting wiring (phone cabling) to provide Ethernet to sites
 - Cable length determines speed
- **MAC table**
 - Switch examines and learns sending MAC, then destination.
 - Sends to all ports, except that which sent it (unless it is in the MAC table
 - If there is an entry and the source and destination are found to be the same port, then the frame is dropped
 - Above is true unless the frame is a broadcast / multicast
 - MAC addresses stay in the table for 300 seconds (5 minutes)
 - Content Addressable Memory table – CAM – proper name for MAC table
- **Useful commands**
 - ***show mac-address-table*** – shows switches MAC table
 - ***show mac-address-table aging-time*** – shows the age of the stored mac address in seconds
 - ***mac-address-table aging-time*** – sets the aging time
 - ***sh int trunk*** – shows interfaces defined as trunks
 - ***interface fa 0/0***
 - switchport mode access***
 - switchport access vlan 99***
adds the port 0/0 to vlan 99
 - ***interface range fa 0/1 – 10***
 - range> speed 10***
 - range> switch mode access***
 - range> switch access vlan 99***
 - range> description ACCOUNTING VLAN***
defines a range of ports that are then set to 10mbps and added to VLAN 99
 - ***errdisable recovery cause xcausenamex/all***
errdisable recovery interval 3600 (<- note value in seconds)
defines a reason (or all) and time for the error disabled switchport to be re-enabled

VLANS

- **Why use VLANS?**
 - Prevents excess broadcast traffic (forwarded to all ports at layer 2)
 - Creates a logical group of hosts, irrespective of physical location
 - Creates a broadcast domain per VLAN
 - Cisco recommends one IP subnet per VLAN
 - Can improve security by hiding hosts
 - Layer 3 device required to route between VLANs
 - Broadcast traffic can traverse trunks!
- **Types of VLAN**
 - Static VLAN
 - § Port based assignment
 - Dynamic VLAN
 - § Dependant upon host MAC address – against a central DB assignment
 - § VMPS – VLAN membership policy server
 - § Remember switches check incoming MAC address before doing anything else, so this security mechanism can do that
 - § VMPS uses a TFTP server to help in this dynamic port assignment scheme.
 - § A database on the TFTP server maps source MAC addresses to VLANs and is downloaded to the VMPS server.
 - § The downloading occurs every time the VMPS server is rebooted.
 - § VMS uses UDP to listen for requests
 - § When a switchport receives a dynamic VLAN assignment, portfast is automatically enabled for that port. You can manually disable it.
 - § Must configure the server before configuring the ports as dynamic
 - § Do not use portsecurity with VMPS
 - § Trunk ports cannot belong to VMPS
 - § **Int fa 0/11**
Switch access VLAN dynamic – configures port for VMPS.
- **Configuring VLANS**
 - 2 Commands required, as by default ports are running in dynamic desirable trunking mode – looking to form a trunk, which would belong to all VLANs
 - § **Interface fa 0/1**
 - § **Switchport mode access/dynamic/trunk** – puts the port into a mode
 - § **Switchport access vlan 13** – creates the VLAN if it does not exist
- **VLAN Tagging - ISL / 802.1q**
 - Frames are tagged with a VLAN ID and the receiving switch examines the ID
 - Both ISL and 802.1q are point-to-point protocols between the trunks.
 - **ISL**
 - § Cisco proprietary
 - § Encapsulates frame with a header and a trailer
 - § ISL does not understand “native” or default VLANs, so all frames passing the trunk will be encapsulated
 - § 26 byte header is added contains the VLAN ID and a 4 byte trailer with a CRC.
 - § The additional 30 bytes may make the frame oversized – the limit for an Ethernet frame is 1518. Frames larger than this are called giants.
 - **802.1q**
 - § Does not encapsulate
 - § Adds a 4 byte header into the frame – maximum frame size of 1522
 - § If the frame is designated for the native VLAN, the header is not added
 - § This reduces the chance of oversized frames!

- For trunks to work, port speed and duplex settings need to be the same.
- Giants are frames greater than 1518 bytes – some switches can handle giants
- IEEE 802.3ac the frame length can be extended to 1522 bytes.
- The opposite of a giant is a runt, with is less than 64bytes
- Both switches must be in the same VTP domain – domain names are case sensitive
- Changing the native VLAN on one switch, does not change it on the remote partner
- **switchport trunk encapsulation dot1q/isl/negotiate** – port negotiates – if both ports support both, then ISL is selected.
- **VLAN database mode**
 - VLANs can also be configured by the VLAN database mode
 - **Vlan database**
 - **Vlan> vlan 75**
 - **Vlan> apply** – note this is required to apply changes made in VLAN database mode. Apply is automatically made when you type “exit”
 -
- **Dynamic trunking protocol**
 - Cisco proprietary
 - Attempts to negotiate a trunk line with the remote switch
 - DTP frames are transmitted ever 30 seconds (overhead!)
 - If the port is non-negotiable, then there is no need to use DTP
 - If the remote device does not understand trunking, then turn it off too.
 - Cannot disable DTP until the port is not in dynamic-desirable mode.
 - **int fa 0/11** – select the trunk port
 - **switchport mode trunk** – turns off dynamic
 - **switchport nonegotiate** – turns off DTP
 - **show DTP** – rarely used – shows DTP summary info.
 - **show DTP interface** – shows detailed DTP info
- **Default VLAN**
 - Aka native vlan
 - VLAN 1
 - The vlan that traffic / ports belong to, when there is no config.
 - Native vlan can be changed with the **switchport trunk native vlan** command. Be wary of changing it though – it does not change the default vlan on other switches!
 - **switchport trunk native vlan xx**
- **Summary**
 - **Trunk mode** – dedicated trunk mode (turn off DTP if you use this!)
 - **Dynamic desirable** – default setting. If the remote switch is running in trunk / dynamic desirable / dynamic auto, then a trunk will form
 - **Dynamic auto** – misleading title – is dynamic, but passive. Will not form a trunk with another dynamic auto device. It needs something to initiate a trunk.
 - **Access** – Turns trunking off, by entering client access mode.
- **Design considerations**
 - Let the distribution layer handle the little things in order to allow the core switches to switch – fast!
 - 2 Vlan designs end-to-end and local
 - § End-to end
 - Follow 80/20 rule – 80% of traffic stays locally. 20% traverses the core
 - Span entire network
 - Pain in the ass to configure
 - VLANs must be accessible on every access layer switch
 - § Local
 - Follows 20/80 rule. 80% crosses core with 20% staying local.

- Users are grouped by location in local VLANs
 - Increasingly, centralised data farms require 20/80 – ie local.
- **Adding and removing VLANs from a trunk**
 - **Switchport trunk allowed vlan add/remove/all/none/except**
 - **Switchport trunk allowed vlan except 1000**
 - **Switchport trunk allowed vlan add 1000**
 - See the impact with **show interface trunk**
- **Naming VLANS**
 - **vlan x123x name xMYVLANx**
 - **sh vlan brief** – to review
 - Be aware that this name cannot be referenced when configuring the VLAN – ie adding / removing ports.
- **Troubleshooting**
 - Check speed and duplex settings on the switch ports – should be consistent
 - Check MAC table –check hosts have an entry
 - show mac-address-table
- **Useful Commands**
 - **show vlan brief**
 - **show vlan id xvlan-namex**
 - **show interface trunk**
 - **show interface fast 0/12 trunk** – shows trunk info from a particular port
 - **no vlan 1234** – deletes VLAN 1234
 - **sh CDP neighbour detail** – shows Cisco neighbours

VTP

- Allows all switches to share information regarding VLANs
- **Revision numbers are crucial** – replacement switches should have the revision number reset to 0. Counters are stored in NVRAM that survives a reboot. This can be done by:
 - Change the VTP domain to a nonexistent domain, then change it back to the original
 - Change the mode to transparent, then back to server.
- Show VTP counters – shows number of advertisements sent/received
- 3 main types of VTP advertisements
 - **Summary advertisements** – sent by servers every 5 minutes, or upon a change in the VLAN database. These include:
 - VTP domain and version
 - Configuration revision number
 - MD5 hash
 - Timestamp
 - Number of subset advertisements to follow
 - **Subset advertisements** – sent by servers, contain specific information regarding the change:
 - VLAN created / deleted / activated / suspended
 - The new name of the VLAN
 - The new MTU
 - VLAN type (Ethernet / Token / FDDI)
 - **Client advertisement request** – sent by clients mostly when local database has been corrupted or deleted. Otherwise the summary advertisement would send this information without request.
- **VTP pruning**

- As trunk ports belong to all VLANs, this leads to problems with broadcasts and multicasts. The trunk port will forward broadcasts and multicasts for all VLANs it is aware of, irrespective of whether the remote switch has ports in that VLAN.
- VTP pruning allows switches to determine which broadcasts are actually required to be sent across the trunk to the remote device.
- Enabling pruning on x1 VTP server actually enables pruning for the entire domain.
- **Vtp domain xnamex**
- **vtp mode server/client**
- **vtp pruning**
- **show VTP status**
- VTP versions
 - V2 supports Token Ring Vlans and Token Ring.
 - V2 performs a consistency check when VLAN changes are made (names and numbers)
 - V2 switch in transparent mode will forward VTP advertisements received from VTP servers in that domain.
- Write erase – wipes switch, but not the vlan db. To do this delete the **flash** file vlan.dat
 - **delete vlan.dat**
- VTP passwords places the entire VTP domain into secure mode. Every switch requires a matching password.
 - **VTP password xpasswordx**
 - **Show VTP password**
- VTP tips
 - Be careful with transparent mode – don't use unless server / client don't meet your needs
 - VTP servers should be physically secure
 - Don't leave unnecessary switches as VTP servers

Spanning Tree Protocol

- Bridge Protocol Data Units
 - Transmitted every 2 seconds to a multicast MAC – 01-80-c2-00-00-00
 - Once a root bridge is elected, only that root bridge will originate BPDUs
 - Non root bridges will forward copies of the BPDU
 - BPDUs carry out the election for the root bridge
 - Each switch will have a Bridge ID priority value – BID
 - BID is a combination of a default priority value and the switches MAC, with the priority listed first.
 - Eg Priority of 32,768 and a MAC of 11-22-33-44-55-66 the BID would be 32768:11-22-33-44-55-66.
 - Therefore in the default situation, the MAC is the deciding factor in the root bridge election.
 - Lowest BID wins!!!
 - 2 types of BPDU
 - Topology change notification
 - Sent by any switch in the network when either:
 - Port goes into forwarding mode
 - Port goes from forwarding / or learning to blocking mode.
 - Does not provide detail of what has happened
 - Sent to root bridge
 - Each switch along the way sends an ACK

- Root bridge sends an ACK with the topology change flag on. This states that the mac address aging time will be moved from 5 minutes to the forward delay time (default of 15 seconds). Stays reduced for (forward delay + max age) – default of 35 seconds.
 - Portfast ports cannot generate TCNs – makes sense as there are generally client devices and changes in port status do not need passing up.
 - Configuration BPDUs
 - Used for the STP calculation
 - Sent by root bridge – below applies to this type!
- **Show spanning-tree vlan x123x**
 - Root = the root bridge
 - Bridge = this switch
 - For the root bridge much of the output is duplicated in the above sections
- **Root port**
 - It is the lowest cost port on the non root bridge leading to the root bridge
 - BPDU carries the root path cost
 - Cost increments as it traverses switches (as they are received)
 - Costs are locally significant only
 - Costs =
 - 100 for 10mbps
 - 19 for Fast Ethernet
 - 4 for 1Gb
 - 2 for 10Gb
 - Determined:
 - Superior BPDU – lowest BID (so from the same switch = a tie)
 - Lowest root path cost to the bridge
 - Choose the port receiving the BPDU with the lowest sender BPDU (same on a single receiving switch)
 - Choose the lowest port number
- Designated port – port allowed to forward frames
- Only 1 port in an pair p2p link will be in blocking
- Changing a ports path cost
 - Don't do it unless you really need to!
 - **Int fa 0/12**
 - **Int>spanning-tree cost x9x**
 - **Show spanning vlan x1x – check results!**
- Per VLAN port cost modification
 - Int fa 0/12
 - Spanning-tree vlan x1x cost xcostx
- Configuring vtp
 - Vtp domain xyzx
- **STP port states**
 - Disabled – doesn't appear, but is officially an STP state. Administratively down – STP disabled.
 - Once port is opened it moves to blocking state. Still receives BPDUs – does not block them.
 - Listening Mode – listening for BPDUs, Can also now send BPDUs. Still cannot forward or receive data frames
 - Learning Mode – LRN – Learning MAC addresses. Still not forwarding frames.
 - Forwarding mode – allows sending and receiving of frames, send and receive BPDUs, place MAC addresses in MAC table.

- To see STP mode of an interface use:
 - show spanning-tree interface or***
 - show spanning-tree vlan***
- Spanning tree timers
 - Don't change them!
 - If you do, you need to change them on the root bridge.
 - **Hello time – 2 seconds**
 - **Forward delay – length of both listening and learning stages – 15 seconds**
 - **Maximum age – time superior BPDUs is retained. 20 seconds. Eg a blocking port that does not receive a BPDU for 20 seconds will enter listening mode.**
 - ***Spanning-tree vlan x100x forward-time / hello-time / max-age / priority / root***
- STP and single root bridges
 - Not the best situation, but occurs by default.
 - Better to specify the root bridge and also to spread the workload per VLAN.
 - ***Spanning-tree vlan x123x root primary/secondary – allows specification per VLAN for root and also configuration of a secondary device. Works by setting priority values for you. Secondary command usually use 28672 for priority.***
- **Core switches make good STP switches – do not use access layer devices.**
- **Load sharing with port-priority command**
 - Can change a ports priorit for some VLANs and leave it default for others to perform load balancing over a trunk.
 - ***Int fast 0/12***
Spanning-tree vlan x15-20x port-priority x16x
 - This would assign VLANs 15-20 to the switch that the command is on, to travel on port 0/12. The other switch will block the other path for these VLANs – even if it is the root bridge for other VLANs.
- **Show interface fast 0/5 switchport** – shows loads of info about the port. – operational mode – shows status – down / access / etc.
- **Extended system ID feature**
 - ***Spanning-tree extem system-id***
 - Defined in IEEE 802.1t
 - Greatly extends the number of STP instances that can be supported by the switch – allowing the switch to support more VLANs. (4096).
- BID Priority
 - Defauly priority 32768 plus the system ID extension, which is the VLAN number
- Catos switches use catalyst and use “set” commands and not the standard IOS CLI.

Advanced spanning tree protocol

- **Portfast**
 - Suitable only for ports connected to a single host
 - Allows switch to go directly from blocking mode to forwarding mode
 - Allows users to receive DHCP address more quickly
 - Does not send BPDUs in blocking mode
 - Normally a 50 seconds process to enable a switch port
 - ***Int fa 0/5***
 - ***Int> spanning-tree portfast***

- Or to enable the entire switch and disable explicitly each trunk:
- **Config> spanning-tree portfast default**
- **Uplinkfast**
 - Portfast for trunk ports
 - 50 seconds to enable a blocked STP port on a trunk can be a long time!
 - Allows a backup port to be moved from blocking to forwarding immediately
 - Only use in access layer switches
 - Takes 1-3 seconds
 - Cannot be configured on a root switch
 - When enabled, it's enabled for all VLANs on the switch
 - When the root port comes back online, it becomes the root port again, but it uses a formula to determine the time:
 - $(2 \times \text{ForwardDelayTimer}) + 5 \text{ seconds}$
 - Uplinkfast will take action to ensure that a switch cannot become the root switch
 - Switch priority becomes 49,152
 - STP port cost will be increased by 3000, making it unlikely that this switch will be used to reach the root switch by any downstream switches
 - **spanning-tree uplinkfast** – enables globally
 - dummy frames are sent to a dummy MAC address
- **Backbonefast**
 - If a core switch detects an **indirect link failure** (evident from the receipt of an inferior BPDU)
 - For example, where the link between 2 downstream switches fail, prompting both to claim that they are the root. The root will tell the inferior switch “when MaxAge timer on root switch port for the inferior remote switch reaches 0.
 - BackboneFast skips the MaxAge stage. Delay is moved from 50 seconds to 30. Skips the 20 seconds of MaxAge default timer, but the 15 second listening and learning stages still run.
 - Uses the root link query protocol RLQ – series of requests and responses to detect indirect outages.
 - **RLQs are sent via the ports that would normally be receiving BPDUs**
 - **This is done to ensure connectivity to the root switch**
 - **RLQ identifies the root bridge that can be accessed by that port**
 - **If they are one and the same everything is fine (?)**
 - **Once RLQ is received a switch will answer**
 - **Either it is the root bridge in the request**
 - **Or the switch has no connectivity to the root bridge in the request, as it uses another root bridge**
 - **Or the RLQ is relayed to the root bridge**
 - Needs to be enabled on every switch on the network!!!
 - **conf>spanning-tree backbonefast**
 - **show spanning-tree backbonefast**
- **Rootguard**
 - Stops switches downstream of that port from becoming the primary or secondary STP root.
 - Will block/discard superior BPDUs, set port into inconsistent state.
 - When superior BPDUs stop, the port will be allowed to transition normally through port states. (**Does root inconsistent state block traffic too?**)
 - **int fa 0/10**
 - **int> spanning-tree guard root**
 - **sh int** – shows ports consistency state
 - **sh spanning-tree inconsistent** – shows inconsistent ports
- **BPDU guard**
 - Guards against BPDUs being received by a port with portfast enabled

- If BPDU is received on a port with BPDU guard enabled, it will be shutdown and disabled. This requires manual enablement.
- Can be applied to an interface
 - **Int fa 0/10**
 - **Int> spanning-tree bpduguard enable/disable**
- Can be enabled all ports that are portfast enabled **at the moment**
 - **conf> spanning-tree portfast bpduguard default**
- **Portfast BPDU filtering**
 - Stops BPDU guard disabling ports when BPDUs are detected
 - Works different when applied at global/interface levels!
 - **Globally**
Stops port running portfast
 - **Interface level**
BPDUs will be ignored and these will not receive responses
 - **Int fa 0/5**
 - **Int> spanning-tree bpduguard enable**
or for global
 - **conf> spanning-tree portfast bpduguard default**
- **show spanning-tree summary totals**
 - shows RLQ information
 - Uplinkfast stats
 - Etherchannel
 - Portfast status
 - And more!
- **Show spanning-tree fa 0/1 detail**
 - Shows port cost
 - ID
- **Unidirectional Link**
 - Fibre is susceptible to unidirectional traffic
 - UDLD (Unidirectional link detections)
 - Transmits UDLD frame across link – if response is received all is well. If not, it is considered unidirectional
 - Effectively a layer 2 ping
 - 2 modes
 - Normal
Flags a syslog message
 - Aggressive
Puts port into error disabled state after 8 replies are missed (1 per second). Shuts the port down and generates a message
 - **Conf> udld aggressive/enable – default is only to apply this to fiber ports or**
 - **Int fast 0/5**
int> udld port – this is normal mode
int> udld port aggressive – add keyword for aggressive
 - Needs to be enabled on both ports involved
 - If using aggressive mode a grace period is provided to allow configuration on the remote device. When remote device responds, then UDLD is active. Receipt of echo starts the timer!
- **Duplex mismatch and switching loops**
 - Duplex mismatch between trunking switches can lead to switching loops.
 - If you change one ports duplex settings – always change the partner port
 - The loop is caused by CSMA/CD – the full duplex port does not perform CSMA/CD, but the half duplex port will. The half duplex will listen and send, while the full duplex will just send and cause collisions. The HD port will backoff, but the FD port will likely continue to cause the HD port to continually backoff.

- **Loop guard**
 - Guards against switching loops(!)
 - If a switch tertiary link (naturally in blocked state due to STP) becomes unidirectional, preventing the receipt of BPDUs from the tertiary remote device, the switch will transition the port to forwarding, creating a switching loops (as both primary and tertiary links are now enabled)
 - Loop guard prevents this situation by placing the port into **Loop Inconsistent State – effectively blocking**
 - Rarer than other features as unidirectional traffic occurs less often.
 - **Int fa 0/5**
 - **Int> spanning-tree guard loop**
or
 - **Conf> spanning-tree loopguard default**
- **BPDU skew detections**
 - Detects BPDUs that are relayed sloooooowly
 - BPDUs are sent every 2 seconds, so can be delay sensitive if forwarding is delayed. Can result in unnecessary topology recalculations
 - This detects delayed BPDUs, but does not take corrective actions
 - Skew time = delay time
 - Syslog messages limited to 1 per 60 seconds
 - Critical is 10 seconds
- **Rapid spanning tree (RSTP)**
 - IEEE 802.1w and an extension of 802.1d
 - 30 seconds delay with listening and learning is a long time!
 - Root bridges are still elected
 - Port roles are different:
 - Alternate port – ALT - aka blocked port ALT
 - Backup port – BCK – provides redundant path to a destination
 - Designated port – ports with the best path
 - Root port – connected to the root switch on non root switch
 - Edge ports – single host ports – like an STP portfast port
 - Point to point ports – connected to another switch in full duplex
 - **Port states**
 - STP – disabled -> blocking -> listening -> learning -> forwarding
 - RSTP – discarding (STP 1,2,3 -DDL)-> Learning -> Forwarding
 - **Topology changes**
 - Edge port changes to not trigger topology change.
 - Switch sends BPDUs with the topology change bit set.
 - Edge ports are the same as portfast ports in STP, though behave differently when receiving BPDU:
 - RSTP – will move from edge role to normal rstp role status
 - **BPDU handling differences in RSTP**
 - STP
 - Root bridge sends a BPDU every 2 seconds
 - Non-root just forward when they receive it
 - MaxAge – specifies retention time of BPDU information. Default is 20 seconds.
 - BPDU uses topology change flag and v1 BPDUs
 - RSTP
 - All switches generate BPDU, irrespective of whether they have received a BPDU from the root bridge. Remains every 2 seconds.

- Important as it allows all switches to have a role in detecting link failures. Also allows for faster detection of link failures. All switches expect a BPDU from their neighbour and cuts them off after 3 missed hellos (6 seconds)
 - BPDU format is the same as STP, but RSTP uses all flag bits available for various purposes. Also uses type 2 v2 BPDUs, this allows RSTP to identify older switches.
 - Uplinkfast / Backbonefast / Portfast are built into RSTP
- **Common Spanning Tree**
 - A single instance of STP for all VLANs
- **Per-VLAN Spanning Tree (PVST)**
 - Runs a separate instance of STP for each VLAN
 - Allows for much better fine tuning of STP
 - Performance hit!
 - Cisco proprietary
- **PVST+**
 - PVST does not work well with CST (common spanning tree – when all VLANs run the same or common instance of STP) so PVST+ uses dot1q rather than ISL – allows the PVST to act as an intermediary between PVST and CST. PVST+ is also proprietary.
- **Rapid Per VLAN Spanning Tree Plus (RPVST+)**
 - If you configure a switch running PVST+ to use RSTP, you end up with RPVST+
- **Multiple Spanning Tree - MST**
 - Allows you to reduce the number of STP instances, without moving all the way back to 1
 - Defined in 802.1s
 - Logically divides switches into “regions”
 - Must agree on:
 - Configuration name
 - Instance to VLAN mapping table
 - Configuration revision number
 - Sends MST BPDUs that contain configuration name / revision number and a digest derived from the mapping table (rather than send the entire mapping table for agreement).
 - If values are not agree, the devices are classed as being in a different region.
 - MST purpose is to map multiple VLANs to a lesser number of STP instances
 - CST operates between the MST subnets, whilst MST looks after each “region” or “internal spanning tree – IST”
 - The IST in each region – the IST must keep the MST region loop free
 - Up to 16 MST instances (MSTIs) can exist in each region 0-15. MSTI 0 is reserved for the IST instance and only IST is going to send MST BPDUs.
 - VTP does work under MST and each switch must be configured manually. Each switch must be configured with the VLAN mappings in MST, as they are not advertised.
 - **conf> spanning-tree mode mst**
 - **conf> spanning-tree mode mst configuration**
 - **conf-mst > name region1**
 - **conf-mst> revision 1**
 - **conf-mst> instance 1 10, 13, 14-20** – maps these VLANs to MST instance 1.
- **Why does anyone run CST rather than PVST**
 - CST 100 VLANs in one STP process
 - PVST 100 VLANs results in 100 STP processes, but allows for greater flexibility with trunk usage (per VLAN load balancing for eg)

- **What are they?**
 - Logical bonding of 2-8 connections.
 - Aka aggregation / link aggregation
 - Avoids delays associated with independent link failures – arising from STP.
 - STP isn't aware of the individual links and thus there is no reconfiguration / recalculation if a component link goes down.
 - Uses Exclusive Or – XOR to determine which channel in the EC is used to transmit data to the remote switch.
 - Data being routed over a failed component link, will be transparently rerouted within a millisecond.
 - Represented as "Po" – or Portchannel
 - Will lower the cost of the new Po virtual port, as a result of the increased bandwidth
- x2 Standards that can be used:
 - Link Aggregation Control Protocol LACP
 - **802.3ad**
 - **Assigns a priority value to each port that has etherchannel capability.**
 - **You can actually assign up to 16 ports to belong to an LACP-negotiated etherchannel, but only 8 ports with the lowest port priority will take part – the remaining ports will only be bundled if the primary ports fail.**
 - Port Aggregation Protocol – PAgP
 - Cisco proprietary
 - Sends PAgP packets between switches via ports that have the capacity to be placed into an etherchannel.
 - First the packets check the capabilities of the remote ports against those of the local switch ports for 2 values
 - Remote port group number must match local port group number
 - Device ID of all remote ports must be the same (all bonded ports must terminate on the same switch – not switches).
 - PAgP also has the capability of changing a characteristic of the etherchannel as a whole if one of the ports in the etherchannel is changed. If you change the speed of one of the ports in an etherchannel, PAgP will allow the etherchannel to adapt to this change. Eg if you change the speed of one of the ports.
 - Differences between the 2- also the "on" mode
 - PAgP has a dynamic mode and an auto mode.
 - Dynamic initiate bundling with a remote switch,
 - Auto mode waits for the remote switch.
 - LACP uses active and passive modes
 - Active ports initiate bundling
 - Passive wait for the remote device
 - On mode – no auto negotiation at all!!!
 - Configuring etherchannels
 - See which ports are trunking :
show interface trunk
 - **Int fa 0/11**
 - **Int > channel-protocol lacp/pagp**
 - **or**
 - **Int > channel-group 1 mode active / passive – LACP**
auto / desirable – PagP
on – no auto
 - Repeat on the other local channel ports
 - **Then enable on the remote switch:**

- **show inter port-channel1** - effectively same as **show int x**
- hardware is listed as "etherchannel"
- **show pagp channexchannelnumberx neighbour** - shows neighbour details and mode - also exists for lacp too
- **show etherchannel brief** - shows summary of etherchannel - also whether it is L2 or L3
- **show etherchannel details** - lots of info about etherchannel
- **show etherchannel summary** - channel, port channel, protocols in use etc
- **show spanning-tree vlan 1**
- Layer3 etherchannels can have a virtual IP too!
- Troubleshooting etherchannels
 - Changing the VLAN assignment mode to dynamic - ports configured for VMPS cannot remain or join an EC
 - The allowed range of VLANS for the EC must match those of the ports.

Securing Switches

- **AAA (authentication / authorisation and accounting)**
 - **Authentication - Local basics**
 - **enable password** - plaintext
 - **enable secret** - encrypted
 - **service password-encryption** - configures device to encrypt all devices
 - **line vty 0 15 password xpasswordx** - telet password plaintext
 - **line vty 0 15 privilege level 15** - sets privilege mode for vty lines to 15 - enable mode
 - Creating a local database of users and passwords with different privileges
 - **conf> username privilege 15 password xpasswordx**
- privilege 15
 - **conf> username privilege 15 password xpasswordx**
- user exec mode
 - **line vty 0 15 login local / tacacs** - local = use local router database
 - **Authentication - RADIUS / TACACS**
 - **aaa new-model**
 - **radius-server host x.x.x.x**
or
 - **tacacs-server host x.x.x.x**
 - **aaa authentication login default local group radius** - sets local switch db to be checked first, then the radius server
 - **line vty 0 15**
 - **line> login authentication default**
 - **Authorisation**
 - Assigning the right to conduct tasks
 - RADIUS is limited, TACACS+ can be configured to allow more granular commands
 - **aaa authorization**
 - **Accounting**
 - Uses a radius / tacacs server to track activity
 - Can be used to bill time too
- Port security
 - Uses MAC address restrictions

- Ports must in access mode - not dynamic desirable
- If a maximum is set and no mac-addresses are defined, the switch learns the MAC of the devices connecting
- conf> int fa 0/1
- int> switchport mode access
- int> switchport port-security - see below for options:
 - aging -
 - maximum - number of MAC addresses
 - violation - action port takes upon violation
 - protect - drops offending frames
 - restrict - drops offending frames and syslog/snmp
 - shutdown - default - error disabled state and syslog/snmp
 - mac-address - secure MAC definition
- Example config
 - **int fa 0/5**
 - **int > switchport port-security maximum 1** - 1 MAC remember
 - **int > switchport port-security mac-address aa-aa-aa-aa-aa-aa** - specify the MAC address to remember
 - **exi**
 - **conf> errdisable recovery interval 30** - resets port after 30 seconds
- To re-enable a port,
 - remove the offending device,
 - shutdown the local port
- show port-security - shows enabled ports, counts of MACs and violations.
 - sh port-security interface fa 0/5 - as above, but also shows MAC addresses and VLAN membership
- Cannot be run on
 - Trunk ports
 - Ports in an etherchannel
 - Destination SPAN port
 - 802.1x ports
- **Dot1x Port based authentication**
 - Host (workstation) and switches must be 802.1x EAPOL enabled - Extensible Authentication Protocol Over Lans
 - Authentication server must be RADIUS - not TACACS
 - Until user is authenticated, only EAPOL / STP / CDP can pass
 - Once user is authenticated, all traffic can travel
 - Eg
 - **conf> aaa new-model**
 - **conf> dot1x system-auth-control**
 - **conf> int fa 0/1**
 - Options:
 - Force-authorized - default - forces port to authorise any host - no security
 - Force-unauthorised - disable any attempt to connect
 - Auto - Enables dot1x on the port
- **SPAN**
 - Allows switch to mirror traffic from the source port to the destination port on which a sniffer is attached.
 - Locations of source ports required, determines the version of SPAN to implement
 - All source and destination on a single switch = Local SPAN
 - Remote switch ports = Remote SPAN
 - All on a VLAN = VSPAN
 - Eg SPAN

- **conf> monitor session x1-66spansessionnox source interface/remote/VLAN fast 0/1- 5**
 - **conf> monitor session x1-66spansessionnox destination interface/remote fa 0/10**
- EG RSPAN
 - All intermediate switches need to be configured for RSPAN
 - Need something to carry cloned frames across the trunk
 - Create a VLAN for carrying the mirrored frames
 - VTP treats RSPAN vlan like any othr VLAN.
 - If configured on the VTP server it will be propogated throughout the network
 - Otherwise, it needs to be manually configured throughout the network
 - VTP will also prune RPSAN as any normal VLAN
 - MAC address learning is disabled for the RSPAN VLAN
 - Source and Destination must be defined on both the source and destination switch - but the commands are different on each switch
 - **host conf> vlan 40**
 - **host vlan> remote-span**
 - **host vlan> exit**
 - **host conf> moitor session 2 source interface fast 0/1 - 5**
 - **host conf> monitor session 2 destination remote vlan 40**
 - **dest conf> monitor session 2 source remote vlan 40**
 - **dest conf> vlan 40**
 - **dest conf> monitor session 2 destination interface fast 0/10**
 - **show monitor** - shows SPAN config
- Source port notes
 - Source port can be monitored in multiple sessions
 - Can be part of an etherchannel
 - Can be any port type (Fa / Gb etc)
 - Source ports cannot be destination ports!
- Destination port notes
 - Can be any port type
 - Cannot be part of an ether channel
 - Cannot be a source port
 - Can only participate in 1 span session
 - Doesn't participate in STP/CDP/VTP/PaGP/LACP/DTP
- **VLAN ACLs - VACLs**
 - Allow traffic filtering bewteen hosts on the same VLAN
 - This is because the CAM (content addressable memory) holds MACs that the switch has learned, but the TCAM (ternary content addressable memory) cuts down on the number of lookups require to compare a packet against an ACL. The TCAM limits ACL filtering to inter VLAN traffic.
 - VACL allow intra VLAN filtering
 - Bridged traffic as well as no IP/IPX traffic require VACLs to filter
 - VACLs run top to bottom with an implecit deny at the end
 - Only one VACL per VLAN
 - Sequence numbers allow subsequent editing whilst running!
 - ACL configs can cause poor performance as can:
 - Excessive logging
 - ICMP unreachable messages
 - Eg
 - **conf> ip access-list extended NO_123_CONTACT**
 - **acl> permit ip 172.10.10.0 0.0.0.3 172.10.10.0 0.0.0.255 -**
identify traffic from 10.1/2/3 to any other in subnet

- conf> vlan access-map NO_123 10 - 10 is sequence number
- access-map> match ip address NO_123_CONTACT
- access-map> action drop
- conf> vlan access-map NO_123 20
- access-map> action forward
- conf> vlan filter NO_123 vlan 10

- **Private VLANs**

- Hosts can be placed into a secondary VLAN with one of two results
 - Community Private VLAN
Host can communicate with hosts in the primary and secondary VLAN - but not in other secondaries
 - Isolated Primary VLAN
Host can only communicate with hosts in the primary VLAN - not even the hosts in it's own secondary VLAN
- Promiscuous mode
 - Allows communication to any primary/secondary VLAN
- NOTE - VTP must be running and in transparent mode
- Eg
 - Create private VLAN
 - conf> vlan 75
 - vlan> private-vlan community/isolated
 -

Multilayer switching

- Multilayer switching
 - Where switching and routing occurs in the switch hardware itself.
 - Results in much faster performance. Upto 10x as fast as a router
 - Cisco Catalyst switches use a route processor (or L3 engine)
 - The L3 engine must download routing information to the hardware
 - To do this Cat switches run either the older MLS (Multilayer switching) or the newer CEF (Cisco Express forwarding.)
 - ASICs rewrite the L2 packets. IP source and destination do not change, but MACs do.
- Route caching / Cisco Express forwarding
 - Older Multilayer Switching (MLS) aka netflow switching
 - Routing processor routes a flows first packet
 - Switching engine snoops that packet and destination and then takes over, forwarding the rest of the packets in the flow.
 - A flow is a unidirectional stream of packets from a source to a destination.
 - Packets on the same flow share the same protocol
 - New Cisco Express Forwarding (CEF)
 - Designed for backbone switches
 - Toplogy based switching method
 - Requires special hardware - at least a 3550 (not 3500xl)
 - Topology based switching
 - Uses Forwarding information base (FIB)
 - Also Adjacency table (AT)
 - Have same routing information as a router, but in a different format
 - Routing information is ket in the FIB, used to make L3 prefix decisions
 - FIBs contents is the same as the IP routing table
 - **show IP CEF**
 - AT table will handle the L2 table, this keeps L2 next hop information

- MLS will make as a normal router, including changing the L2 address to the next hop. L3 address does not change - as per norm. The L2 source address will change as well, to the MAC on the MLS transmitter.
 - enabling CEF - on by default! Cannot turn it off as it's hardware based
 - however, you must have a routing table for CEF to work!
 - No IP routing gives - **CEF not running**
 - **config> ip routing**
 - CEF does support per-packet and per-destination load balancing, but not all switches. Check before you buy.
 - Two logical planes exist in CEF MLS.
 - Control Plane (or L3 engine)
 - Builds the ARP and IP routing tables
 - This makes the FIB and AT table possible
 - Data Plane (or hardware engine or ASIC)
 - This places data in the L3 switches memory
 - Performs any necessary encapsulation
 - Forward data to the next hop
 - Exception packets that cannot be hardware switched
 - Packets with IP header options (Not TCP header options)
 - Packets that will be fragmented before transmission, as they exceed the MTU
 - 802.3 ethernet packets
 - Fastest switching options
 - Distributed CEF (DCEF) - workload distributed over multiple CPUs
 - CEF
 - Fast switching
 - Process switching
- Inter-VLAN routing and SVIs
 - Router on a stick
 - Router on a stick requires a fast ethernet port. Configure sub interfaces and place each in a vlan.
 - Router on a stick gets inter-vlan routing but may not guarantee bandwidth for each VLAN if lots of vlans involved
 - Router on a stick places a load on the router - needs to be powerful enough to cope
 - Router on a stick relies on a single port - single point of failure
 - Internal Router processor / Route Switch Module (RSM)
 - Eg catalyst 5000s RSM takes the place of a router
 - Switched Virtual Interface - eg VLAN1 not a real port
 - In each VLAN configure an ip address (int vlan xx)
 - **Enable IP Routing**
 - If you need an external router as the gateway, you use a routed port (or routing port)
 - A port that is routing rather than switching
 - **int fast 0/1**
 - **no switchport**
 - **ip address 192.168.0.1 255.255.255.0**
 - Port can now ping
 - Add a routing protocol on to enable connectivity to remote hosts connected to the L3 device
 - **router eigrp 100**
 - **no auto summary**
 - **network 192.168.0.0 0.0.0.255**
 - **network 191.1.1.0 0.0.0.255**
 - **network 190.1.1.0 0.0.0.255**

- Add a default gateway to force the routing to forward unknown traffic
 - Remember
 - To create the VLAN before the SVI
 - To use "no switchport" for routing ports
 - VLANs and SVI are not the same thing - the interface is the interface for the VLAN
- Fallback bridging
 - CEF does not support IPX / SNA / LAT and Appletalk. SNA and LAT are non routable protocols.
 - Fallback bridging can be used to get this traffic from one VLAN to another.
 - conf> bridge-group 1 (create a bridge group)
 - conf> inter VLAN 10
 - conf-if>bridge-group 1X
- Router redundancy protocols (HSRP / IRDP / VRRP / GLBP)
 - Requires a secondary router, also requires protocol to manage which one is active.
 - IRDP (ICMP - router discovery protocol)
 - Defined in RFC 1256.
 - Commonly used by windows DHCP clients and unix. extension to ICMP (ping!)
 - IRDP routers will generate router advertisement packets that will be heard by hosts on that segment.
 - If a host hears from more than one IRDP router, it will choose one as its primary and will use the other for failover
 - IRDP does not use a virtual router
 - Hosts generate router solicitation messages - usually at startup - asking for router advertisement packets.
 - To enable IRDP on a routers interface
 - conf> interface serial 0
 - conf-if> ip irdp
 - HSRP
 - Defined in 2281
 - Cisco proprietary
 - Routers are put into a HSRP router group
 - Along with dynamic routing protocols and STP, is considered a high availability service.
 - One router is chosen as primary and that primary will handle the routing while the other routers are in standby.
 - Hosts see a virtual MAC and IP address, representing the HSRP group
 - By configuring multiple HSRP groups on a single interface, HSRP load balancing can be achieved. Not true load balancing.
 - **conf-if# standby 1 ip 192.168.0.10**
 - **conf-if# standby (group) ip (virtualIP)**
 - Do the same on both routers
 - **show standby**
 - HSRP creates a virtual MAC address using a well known MAC address 00-00-0c-07-ac-xx, where xx is the group ID in hex. So 11 is 17 in decimal.
 - You can change timers :
 - standby 1 timers xxxx
 - Group timers must be consistent
 - Priority - highest priority becomes primary router
 - **conf-if# standby 1 priority 255**
 - changes in priority are not enacted unless the existing primary fails, or unless the preempt keyword is used

- **conf-if# standby 1 priority 255 preempt or**
 - **conf-if# standby 1 preempt**
 - Not using preempt is a common cause of problems
 - Can manually change the virtual MAC too
 - **conf-if# standby 1 mac-address aaaa.aaaa.aaaa**
 - HSRP Load Balancing
 - 2 HSRP devices
 - Create 2 groups for the 1 VLAN
 - Each router is a primary for a HSRP group
 - Hosts are then configured 50/50 for each virtual router
 - Not really load balancing really eh?
 - HSRP update authentication
 - **standby 1 authentication** (optional MD5) **password**
 - HSRP on a L3 switch
 - On a SVI
 - Routed port
 - L3 Port channel (etherchannel with an IP address)
 - HSRP requires the "enhanced multilayer software image" EMI, to run on an L3 switch. Gig ethernet switches have this, but fast ethernet will have either EMI or Standard Multilayer image. SMI is similar to L2. You can upgrade software though.
 - HSRP states
 - **Disabled** - not running HSRP
 - **Initial** - HSRP enabled, but not yet running
 - **Learn** - Router waiting to hear from partner routers and also does not know partners IP address
 - **Listen** - Router knows virtual IP, listening for hello - not yet active or passive
 - **Speak** - Router sending Hello messages and is active in the election process
 - **Standby** - Is candidate to become active and still sending hellos
 - **Active** - Is forwarding packets sent to the groups virtual IP address.
 - Note: HSRP does not send hellos until it reaches the speak state
 - An interface may participate in multiple HRP groups on most routers. Some 2500/3000/4000 routers do not support this.
 - HSRP Interface Tracking
 - Enables HSRP process to monitor an additional interface
 - Status of this interface will change the priority of a group.
 - Needs to be configured with the preempt option
 - Means that router used can change depending on the status of a network port. Can be used to track multiple ports.
 - **conf-if# standby 1 track serial 0 x (where x is the decrement is the value removed from this routers priority when this router goes down - default is 10)**
 - Bear in mind this means that priority spacing must be less than a10 for this to have an impact
 - Most common problem with interface tracking is the priority decrement values.
 - debug standby
 - shows hellos coming in and out, priority changes etc
- VRRP (Virtual router redundancy protocol)

- Defined in RFC 2338
- Open standard equivalent to HSRP
- Master router = active / IP address owner
 - Router that has the virtual routers IP as a real address on its interface
- Backup = standby router
- Physical routers combine to form a virtual router
- VRRP advertisements use 224.0.0.18
- MAC address of VRRP router is 00.00.5e.00.01.xx (where xx is the group number in hex)
- Preempt is a default setting for VRRP routers
- As of IOS 12.3(2)T, VRRP supports object tracking (similar to interface tracking in HSRP)
- HSRP and VRRP
 - does not support accurate load balancing
- GLBP (Gateway Load Balancing Protocol)
 - Allows for true load balancing
 - Round robin format for all devices in the group
 - When a host sends a ARP address for the MAC of the router, one physical router answers with it's physical address - so the host has the virtual IP of the router, but a physical MAC address of the router.
 - Router with the highest GLBP priority is the AVG (active virtual gateway) this router sends back virtual MAC addresses (assigned by the AVG) If priorities are the same, the highest MAC address becomes the AVG.
 - Each router has the same L3 address, but different L2 address
 - AVG assigns virtual MACs to the AVFs (active virtual forwarders)
 - AVF is backed up by the backup AVG
 - Hellos between routers detect which routers are online
 - GLBP load balancing can be fine tuned, through different methods of MAC address assignment.
 - Default is round robin
 - Host dependant load balancing provides the same MAC gateway for consistency
 - Weighted MAC assignments affect the percentage of traffic sent to each AVF. The higher the weight, the more traffic is passed to that router.
 - conf-if# glbp 1 ip 192.168.1.1
 - conf-if# glbp 1 priority 150
 - conf-if# glbp 1 preempt
- Server load balancing
 - SLB represents multiple servers to a host as a single physical server
 - conf# ip slb serverfarm serverfarmname
 - conf-slb# real x.x.x.x (for each server in the farm)
 - conf-slb# inservice (for each server in the farm)
 - Now create the virtual server
 - conf#ip slb vserver virtualservername
 - conf-slb# serverfarm serverfarmname
 - conf-slb# virtual x.x.x.x
 - conf-slb# inservice
 - Control which hosts can connect to the virtual server. If hosts /subnets are defined, these will be the only devices allowed to connect to the virtual device. This command uses wildcard masks.
 - conf-slb-vserver# client 192.168.0.0 0.0.0.255

IP Telephony

- 3 Ports on an IP phone
 - One for switch
 - One for the phone asic
 - One for the PC - PC acts normally as if connected to the switch
- **Switch and phone link can be a trunk / access**
 - Access - shared data/voice = bad
 - Trunk allows creation of a voice VLAN, in turn allowing QoS - 4 choices:
 - **Configure the link as a trunk and use 802.1p**
 - Voice traffic has high priority
 - Voice traffic is sent through the native voice VLAN, VLAN 0.
 - **switchport voice vlan dot1p**
 - **Configure the link as a trunk and do not tag voice traffic**
 - **Configure the link as a trunk and specify a voice VLAN - preferred**
 - **switchport voice vlan x** (x = vlan number)
 - When VLAN is configured on a port, portfast is automatically enabled. Portfast is not disabled when voice VLAN is removed from the port!
 - Recommended that you use QoS on the switch and switch port connected to the IP phone, to be set to trust incoming CoS values. The commands to perform this are **mls qos** and the interface command **mls qos trust cos**
 - Can configured voice VLANs on ports running port security or 802.1x authentication.
 - CDP must be running on the port leading to the IP phone.
 - Voice VLANs are supported on only L2 Ports
 - Make sure traffic total traffic does not exceed 75% of overall available bandwidth (voice / video / data/)
 - Voice and video should not exceed 33% of a links bandwidth - allows network control traffic to flow through the network - helps to prevent jitter too.
 - CDP spoofing can cause problems as voice requires CDP. Attack pretends to be the IP telephone
- 3 problems with IPT
 - Jitter
 - Delay
 - Packet loss
- **Best-effort-delivery - no QoS. Works okay for UDP but not for voice.**
- **Integrated services model (IntServ)**
 - Users Resource Reservation Protocol (RSVP) - creates a high-priority path in advance of the voice traffic's arrival.
 - Transmitter waits for this path before transmitting
 - Creation of this path is sometimes referred to as Guaranteed Rate Service / Guaranteed Service (GRS).
 - Not scalable - reserved paths for all hosts would be demanding on bandwidth
- **Differentiated Services Model (diffserv) - most popular**
 - Diff serv makes QoS decisions on a per hop basis as the flow traverses the network - Per hop behaviour (PHB)
 - Most popular of the available options
 - Core tasks of diffserve QoS are marking and classification
 - Marking = tagging data with a value
 - Classification = taking the appropriate approach to queuing and transmitting according to the data value
- Frames sent from a tagged switch to a host are not tagged! Tagging is for the relevant switch devices.
- Frames sent across trunks can have CoS values as well as VLAN tagging

- CoS = Code of Service
- CoS is used by the receiving switch to make QoS decisions on the frame.
- ISL and 802.1q handle CoS differently
 - ISL = 4 bit user field indicate the the CoS value - 0-7 range
 - 1q = 3 1p priority bits that make up the CoS value of 0-7
- Untagged frames can be tagged if they are destined for a voice / priority VLAN
- Type of Service (TOS) byte - used by diffserv at L3
 - ip precedence value - ip prec - 3 bits
 - tos value - 4 bits
 - a zero - 1 bit
- Diffserv uses this 8 bit value too, but refers to this as the Doff Serv (DS) field.
 - DS code point value - 6 bits (DSCP - RFC 2474) - Dividied into:
 - Class selector value - 3 bits
 - class 7 (111)- Network control
Used for network control traffic - STP / routing protocol etc
 - class 6 (110) Internetwork control
As above
 - class 5 (101) Expedited forwarding (EF - RFC 2598)
For voice and other time sensitive traffic. Almost guaranteed not to be dropped.
 - classes 1-4 (001 - 100) Assured forwarding (AF, RFC 2597)
Allow QoS to be defined for non-critical time sensitive traffic.
 - class 0
best effort! default!
 - Drop precedence value - 3 bits - RFC 2597
 - High 3
 - Medium 2
 - Low 1
 - Configured as :
 - AF (class number) (drop precedence)
 - eg AF 31
 - This is class 3 and a drop precedence of 1 or Low
 - Explicit Congestion Notification Value (ECN - RFC 2481)
 - Trusted values with QoS between switches
 - Trusted Qos vlaue, frames are processed according to value
 - Non trusted values, receiving switch can assign a pre-configure value.
 - Normally internal switches are trusted
 - Values can help provide information about interfaces for the admin!
 - **Int fast 0/1**
 - **description Leads to IP phone**
 - Enabling Qos
 - **conf# mls qos**
 - Configuring the trust after enabling QoS
 - **int fast 0/1**
 - **mls qos trust cos**
 - **mls qos trust device cisco-phone** (trust value if remote device is cisco phone)
 - **switchport priority extend cos 0-7** (allows definition of class of service for traffic coming from the PC connected to the IP phone)
 - **switchport priority extend trust** (allows trusting of traffic priority defined by the PC connected to the IP phone)
- Troubleshooting

- **show mls qos interface fast 0/1**
shows trusted state of interface
 - QoS traffic should be marked as close to the source as possible - effectively access switches, not just to take the task from core devices, but to ensure end-to-end QoS.
 - Compressing the RTP header can also improve VOIP performance - this takes the IP/UDP/RTP header from 40 bytes to 2-4 bytes.
 - ip rtp header-compression**
-passive is an optional parameter that only compresses outgoing headers if the incoming packets are compressed
- Cisco AVVID (Architecture for Voice, Video and Integrated Data).
 - It is concerned with:
 - High availability
 - Quality of Service
 - Security
 - Enterprise mobility
 - Scalability
 - Good PDF on the Cisco AVVID site. Read more on this.
 - Storage networking is also growing - now available as a CCIE!
- Power over ethernet
 - IEEE 802.3af for
 - High power version proposed as 802.3at
 - www.poweroverethernet.com
 - Not all switches are POE compliant
 - POE ports on capable switches attempt to find a device needing power
 - **conf# int fa 1/1**
 - **config-if# power inline auto|consumption|never|static**
 - auto = default
 - consumption = milli-watts used by device
 - never = disables POE
 - Commands for POE vary between switches!

Wireless Networking

- Infrastructure Wireless Local Area Network
 - Traditional Access Point / Client model
 - Also called a BSS - basic service set
 - Each AP provides an access cell
 - Clients form an association with the AP
 - SSID can be up to 32 characters in length
 - Users can roam between cells
 - Standards for roaming cutover is bespoke to each supplier.
 - 2 methods for finding new APs
 - Active scanning
Client sends probe request frames and waits to hear probe responses. Client chooses best AP
 - Passive scanning
Client listens for beacon frames from APs
- Independent Basic Service Set (IBSS)
 - Ad-hoc networks
 - These are created between client
- Open system and shared key
 - Wired Equivalent Privacy - static key
 - Open system
- EAP - Extensible Authentication Protocol (RFC 3748)
 - Original for PPP, adapted to wifi
- LEAP - Lightweight Extensible Authentication Protocol
 - 2 way auth between AP and client

- AP uses a radius server for auth
 - Keys are dynamic, not static
- WPA - wifi protected access
 - wi-fi.org - WiFi Alliance created standard
 - IEEE 802.11i wifi alliance released WPA2
- 802.11a
 - 25mbps, but can reach 54mbps
 - Range is 100ft
 - Frequency is 5Ghz
- 802.11b
 - 6.5mbps, but can reach 11mbps
 - Range is 100ft
 - Frequency is 2.4Ghz
 - Microwaves also use 2.4ghz!
- 802.11g
 - 25mbps, with a 54mbps peak
 - Range is 100ft
 - Frequency is 2.4Ghz
 - Microwaves also use 2.4ghz!
- 802.11n
 - 200mbps, peak of 540mbps
 - Range is 160ft
 - Frequency can be either 2.4 or 5Ghz
- IRDA - Infra Red Data Association
 - 115k speed
 - 1 mt range
 - v1.1 allows 4mbps
 - Compatibility between v1 and v1.1 is not great
- Antenna types
 - Yagi antenna - Yagi-Uda antenna
 - aka directional antenna
 - aka p2p antenna
 - Requires alignment
 - Good for point to point links
 - Omni direction antenna
 - aka point to multipoint antenna
 - good for hosts
- WLANs
 - Can't listen and send - they are half duplex!
 - They use CSMA/CA - carrier sense multiple access, with collision avoidance
 - CSMA/CA core is the "distributed coordination function" (DCF).
 - stations wait for for the "distributed interframe space" (DIFS) time interval to expire before doing so - the DIFS is a random backoff timer
- Cisco Compatible Extension Website - CCX
 - List compatible Cisco wireless devices and features
 - 22.22
- Cisco Unified Wireless Network
 - WLAN controller allows for central management of Lightweight Access Points (LAP)
 - This takes place by the LWAPP - Lightweight Access Point Protocol
 - Sends security policy / QoS policy / Mobile user policy etc
 - Many aironet AP's can operate autonomously or as an LAP
 - 1230 AG series
 - 1240 AG series
 - 1130 AG series
 - Aironet System Tray Utility
 - sits in system tray

- information is non-intuitive
- Red - connection in place but low signal
- Yellow - fair connection
- Green - Good
- Light gray - AP present, but not authenticated
- Dark gray - No connection to AP
- White - adapter is disabled

Network Design and models

- Cisco model
 - Core
 - **Job is just to switch**
 - Locate root bridges here
 - Switches are likely to be multilayer switches
 - QoS is performed at the core
 - QoS is high speed queing
 - Core should have redundancy
 - Distribution
 - Aggregates uplinks from access layer
 - Powerful multilayer switches working at L2 and L3
 - Should be redundant
 - **Routing occurs here**
 - QoS can also be found here
 - Acts as a boundary for broadcast traffic
 - Access
 - Handles VLAN membership
 - Basic QoS (marking traffic is handled here, classification further up)
 - Uplinks should be scalable for future growth
 - Port density should be scalable
 - MAC filtering
 - Collision domains are also formed at this layer
- Enterprise Composite Network Model
 - Switch blocks

Units of access layer / distribution layer devices. These work together to provide access to a bring access to a unit of the network
 - Core blocks

Allow switch blocks to communicate
 - Actual design depends on physical and logical factors
 - 3 Parts to this model
 - Enterprise campus
 - Enterprise edge
 - Service provider edge
 - Enterprise campus consists of :
 - Campus infrastructure module
 - Building access module (access devices)
 - Building distribution module (distribution devices)
 - Campus backbone (interconnects multiple distribution modules)
 - Server farm module
 - Network management module
 - Enterprise edge
 - Dual core - Redundant links to the core
 - Collapsed core - core and distribution switches are merged to reduce cost
 - Server farm block - represents a discrete switch block with it's own access/distribution switches.

- Combination of access/distribution and core layers is known as the campus infrastructure
- Network management block - AAA servers / syslog servers / network monitoring servers. As per the server block
- Enterprise connectivity block - Locally controlled, point of demarc to the WAN and internet
- Service provider block - managed by service provider
- Find network diagram of this model??? and tidy section

Queueing

- First in first out - FIFO
 - Not really queueing
 - Not suited to time sensitive traffic
- Weighted Fair Queueing - WFQ
 - Prevents any one stream of traffic from using most or all of the bandwidth
 - Flows are defined by WFQ and require no access lists
 - Flow based WFQ takes packet flows and classifies them into conversations.
 - WFQ gives priority to the interactive, low bandwidth conversations.
 - WFQ then splits the remaining bandwidth fairly between non-interactive high bandwidth conversations.
 - Enabled by default on all interfaces running at or below E1 speed
 - **config-if# fair-queue 1-4096** (number is discard threshold - number of packets that can be held in a queue)
 - **show queue serial 0** - shows queueing strategy, queue lengths and bw
 - **config-if# fair-queue 1-4096 16-409** (2nd parameter is the number of reservable conservation queues - queues are used for specialised queues, like the resource reservation protocol - RSVP).
 - **show queueing fair**
- Class maps and policy maps
 - Advanced form of WFQ allows manual specification of which flows should be transmitted first. Class Based WFQ (CBWFQ)
 - First step is to create an ACL matching the source
 - access-list 100 permit tcp 172.10.10.0 0.0.0.255 any**
 - access-list 110 permit tcp 172.20.20.0 0.0.0.255 any** FTP matches traffic from a source for FTP data
 - Create a class-map - name can be the subnet for clarity (up to 64 classes allowed)
 - conf# class-map 17210100**
 - conf cmap# match access-group 100**
 - conf# class-map 17220200**
 - conf cmap# match access-group 110**
 - Can match against lots of things in a class-map statement - not just ACLs.
 - call another class-map (nested)
 - input interface
 - IP values
 - mpls
 - protocol
 - qos-group etc
 - Next apply them to a policy map
 - conf# policy-map mypolicy**
 - conf pmap# class 17210100**
 - conf pmap-c# bandwidth 400** (can be kb value or %)
 - conf pmap-c# queue-limit 50**
 - conf pmap# class 17220200**
 - conf pmap-c# queue-limit 25**

- Finally apply the policy map!
conf if# service output mypolicy
 - Note: you may have to disable WFQ with "no fair-queue"
 - **show policy-map**
 - **show class-map**
- Can't assign over 75% of an interfaces bandwidth via CBWFQ as 25% is reserved for network control / routing traffic. This can be changed with the "max-reserved-bandwidth x" command on the interface - not recommended!
- If packets are dropped from CBWFQ, then there are options to how this occurs:
 - **taildrop** is the default - drops last items in the queue - arbitrary approach. Also causes problems with TCP global synch - where senders reduce transmission rate and then increases it again. Creates a rollercoaster of congestion.
 - **random early detection**
RED proactively drops packets before the queue gets full, but the packets dropped is still random
 - **weighted random early detection**
Like RED, but WRED uses a packets IP precedence or differentiated service code point (DSCP) to decide which packets to drop. Drops packets from queues other than the priority queue first.
 - **Configuring RED**
conf# policy-map mypolicy
conf-pmap# class 17210100
conf-pmap-c# random detect
 - Both RED and WRED only work with TCP traffic
- Low latency queueing (LLQ) is an add on to CBWFQ that creates such a strict priority for voice traffic it allows us to avoid jitter.
 - Cisco Recommend that LLQ is used only with voice traffic
 - WRED and LLQ don't work together. WRED is TCP and LLQ is primarily used for voice traffic - which is UDP.
 - LLQ doesn't have strict queue limits, so the queue-limit and priority commands are mutually exclusive
 - Bandwidth and priority commands are also mutually exclusive
 - **access-list 155 permit udp 210.1.1.0 0.0.0.255 220.1.1.0 0.0.0.255 range 17000 18000**
 - **conf# class-map voice_priority**
 - **conf-cmap# match accessp-group 155**
 - **conf# policy-map voice_policy**
 - **conf-pmap# class voice_class_priority**
 - **conf-pmap-c# priority 45 (45k)**
 - **conf-pmap-c# class class-default (defines what happens to traffic not matching other classes - the rest gets fair queued see below)**
 - **conf-pmap-c# fair-queue**
 - **conf-int# service-policy output voice_policy**
- Priority queueing (PQ)
 - Uses 4 pre defined queues:
 - High aka strict priority queue -20 packet limit
 - Medium - 40 packet limit
 - Normal - 60 packet limit
 - Low - 80 packet limit
 - Traffic is placed into these queues through the use of ACLs and priority lists

- PQ does not use round robin - packets in the HQ are sent before packets in any other queue! Potential to drown out low priority traffic - this is known as packet starvation / traffic starvation
- Configuring
 - Incoming interface / protocol can be used to determine queue
 - **conf# priority-list x protocol|interface|queue-limit high|medium|normal|low list|protocol|size (where x is the list number 1-16)**
 - **priority list 1 protocol ip high list 110 (where 110 is the acl)**
 - **priority-list 1 protocol ip medium tcp 53**
 - **priority-list 1 interface ethernet0 normal**
 - **priority-list 1 queue-limit a b c d** (where a/b/c/d are the values for each queue)
 - Apply this with **conf if# priority-group 1**
 - **show queueing**
 - **show queueing priority**
- Custom Queueing (CQ) extends PQ functionality
 - It allows you to define how many bytes are forwarded by each queue upon its turn
 - CQ has 17 queues
 - 1-16 are configurable
 - 0 is for network traffic, routing protocols / syslog msgs / STP etc
 - Default packet limit for each queue is 20 packets and each will send 1500 bytes
 - Uses round robin, each queue then sends its limit, or until it's empty
 - To configure:
 - Define the size of queues
 - Define what packets go into the queues
 - Define the custom queue list by applying it to the interface
 - For example:
 - define queue lists:
 - conf# queue-list 1 queue 1 limit 100
 - with ACLs:
 - conf# access-list 110 permit ip 100.1.1.0 0.0.0.127 200.2.2.0 0.0.0.15
 - queue-list 1 protocol ip 2 list 110
 - by interface
 - queue-list 1 interface ethernet0 4
 - limit byte count
 - queue-list 1 queue 4 byte-count 3000
 - default queue for all other traffic
 - queue-list 1 default 5
 - by protocol
 - queue-list 1 protocol ip 2 tcp 80
 - apply this to the interface
 - int# custom-queue-list 1
 - use below to troubleshoot
 - show queueing custom

