:::::::Notes::::::::

Virtual Local Area Networks/VLAN(VLAN=Broadcast Domain=SUBNET)

1.logically groups users/segmentation of routers without routers/no longer limited to physical location/can span multiple physical switches
2.segments broadcast domains (ports in different vlans are in different broadcast domains)
3.subnet correlation
4.access control/sort of layer security feature
5.QoS(Quality of service)

(Normal Switching)

1.one collision domain per port
2.broadcast sent to all ports by default (slows down network if big broadcast domain)
3.one subnet per vlan
4.very limited access control
5.hosts in the same vlan share the same broadcast domain
6.separate CAM table per VLAN
7.traffic inside the vlan is layer 2 switched and traffic outside the vklan is layer 3 routed

(Local vlan)

1.Local VLANs do not extend beyond the distribution layer (local vlan traffic routed to other destinations)
2.End-to-end VLANs (spans over the whole network)

(vlan numbering)

1.12-bit field (1-4094) (0 and 4095 are reserved)
2.1 is default ethernet vlan
3.normal vlans (1-1005)
4.extended vlans (1006-4094)
5.2950/2960 supports upto 255 vlans
6.3550/3560 support upto 1005 vlans
7.1002/1004 default token ring vlans
8.1003/1005 default fddi vlans

(VLAN membership)

1.once the vlans are created, membership is assigned at the port level
2.layer 2 "switcports" egenrally fall into three categories:
    1.access switchports (one vlan per port)
    2.trunk switchports (multiple vlans per port)
    3.dynamic switchports (automatically choose access or trunk)

(vlan trunk)

1.VLAN trunks (carry traffic from multiple vlans between switches on uplinks)

(VLAN Trunking)

1.aka tagging (passes multi-vlan information between switches)
2.places vlan information into each frame
3.Layer 2 feature
4.Vlans may need to span different physical switches

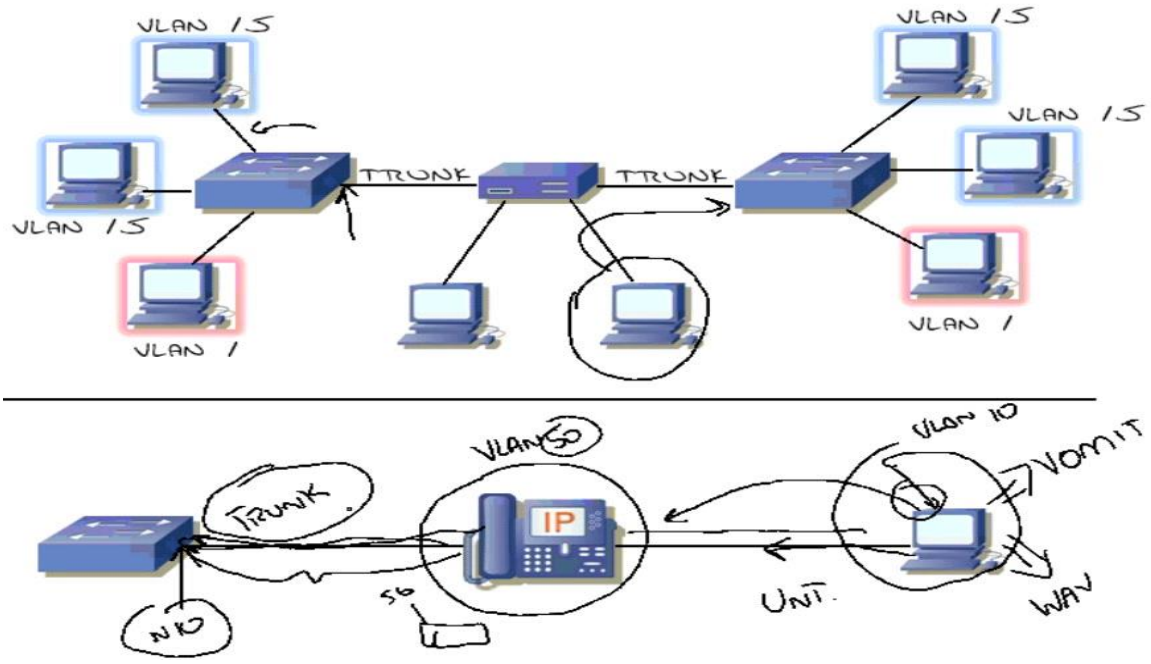
1.VLAN trunks allows hosts on different switches to share the same VLAN assignment (broadcast domain)
5.Accomplished by having ports between switches send traffic for multiple VLANs at the same time
1.VLAN number is included in the ethernet frame between switches

(Two Types of Trunks/Two Trunking Protocols)
->used to encode the VLAN number on trunks
1.IEEE 802.1q/dot1q (open standard/industry standard/common implementation/inserts tag into frame rather than encapsulating)
2.ISL/Inter Switch Link (being deprecated/encapsulates the entire frame/cisco proprietary)

(trunking protocol 802.1q/dot1q)
(DstMAC|SrcMAC|4byte dot1q tag|Ethernet Frame|FCS)
1.open standard
2.inserts tag into frame rather than encaosulating
3.4 byte tag except for 'native' vlan (inserts between src/dst MAC and the len/ethertype fields)(native vlan sent untagged)
4.rebuilds trailer (FCS) since frame is modified
5.4 byte Tag includes 3 bits priority field and VLAN tag
6.dot1q is efficient compared to ISL
7.you can run ISL and 801.21q in different parts of the networks but you can't combine them. Swicthes encapsulates on switch by switch basis.
8.QinQ support: multiple tags on a single frame

(trunking protocol ISL/Inter switch link)
(26 byte ISL tag|Ethernet Frame|4 byte CRC ISL trailer)
1.proprietary
2.encapsulates the entire frame
3.30 bytes encapsulation for all frames(26 bytes header+4 bytes trailer/FCS)
4.does not modify the original frame

(Native VLAN)
1.untagged frames (frames received on a trunk that are not tagged)
2.native vlans must be the same on uplink ports to other switches to avoid native vlan mismatch problem i.e. to avoid Vlan seepage/Vlan leaking problems
3.by default vlan 1 is native vlan
4.this concept is now being used with cisco IP phones creating a small trunk between IP phone and the switch(cisco IP phones can tag vlans to frames), whereas the untagged frames from PC are put in the native vlan.



(Negotiating Trunk)

1.Switches can auto-negotiate trunk connections using the dynamic trunking protocol (DTP)
2.Five different modes:
    1.Access(one vlan/secure)
    2.Trunk(multiple vlan/dtp sent)
    3.Dynamic auto(passively listens/no dtp sent)
    4.Dynamic desirable(dtp negotiation/dtp sent/default switchport mode)
    5.non-negotiate (no dtp sent/most secure)
3.Only these modes work:
    1.auto-desirable
    2.desirable-desirable
    3.auto-on
    4.on-on
    5.on-desirable
->(auto-auto does not work)

(DTP negotiation/Dynamic Switchports)(Dynamic desirable/Dynamic auto)

1.Dynamic Trunking Protocol(DTP) is cisco proprietary that negotaites the trunking status of a switchport
2.switchport non-negotiate
3.switchport mode access
4.by default every switchport is dynamic desirable thus sending DTP packets
5.PC can spoof DTP packets trying to negotiate trunk with the switch in auto,dynamic desirable or trunk mode, thus sniffing all the packets using a packet sniffer (security remification)
6.disabled with switchport nonnegotiate or switchport mode access or dot1q-tunnel
7.you should disable DTP if you are trunking between switch and a router

or switch and a server
8.you will run DTP in case you need to plug IP phones which also sends DTP packets to negotiate a turnk and then run voice vlan for voice traffic and data vlan for data traffic
9.you should know by design which ports will be trunk especially if you have ip phones and access points
10.Mostly switches (e.g. 2950/2960/3550) have all their switchports in dynamic desirable by deafult but 3560 has it in dynamic auto

(Allowed list)
1.by default trunk ports carry traffic for all vlans called 'allowed list' but can be manually filtered
2.used to reduce broadcast transmission, unknown multicasts and stp overhead
3.can be automatically filtered using vtp pruning

(Ethernet interface types)
1.Layer 2 switchports
    1.Access (one vlan)
    2.Trunk (multiple vlans)
    3.Tunel (transparent layer 2 vpn)
    4.Dynamic (dtp negotiation)
2.Layer 3 Ports
    1.Switched Virtual Interface (SVI)
    2.Native Routed Interfaces

(Vlan trunking protocol/VTP)
1.Vlan replication protocol/Replicates vlans to other switches in the same vtp domain
2.solves the administration problem of vlans
3.eases management of vlans in the network
4.used to automatically learn the vlan names and numbers in the network
5.cisco proprietary (cannot run in mixed environment)
6.has a small database that keeps record of revision numbers when vlans are added/removed, and updates that revision number to other switches
7.vtp pruning
8.it's important to set the domain and password to avoid the replication of vlan to switches in the production network especially when new switches are added
9.a port that is assigned to a vlan that doesn't even exist, that port can't communicate with any other port
10.if you make an error, mistakenly removing a vlan on a server switch that will be immidiately replicated amongst the switches

(Three VTP modes)
1.Server(default)
    1.allows addition, deletion, and modification of vlans information
    2.changes on server overwrites the rest of the domain/send and received vtp updates
    3.saves vlan configuration
2.Client

1.can not add, remove or modify vlan information
2.listens for advertisements originated by a serve, install them, and passes them on/sends and receives vlan updates
3.does not save vlan configuration

3.Transparent(separate database but passes vtp info)

1.keeps a separate vtp database from the rest of the domain/saves vlan configuration/power to change vlan information
2.does not originate advertisements
3.transparently passes received advertisements through without installing them but it must be in the same domain name/forwards (passes though) vtp updates
4.needed for some applications like private vlans
5.if there are transparent switches in the transit way of vtp pruning then we can end up into a situaiton that some switches won't even forward even important vlans

(if you wishes to use transparent switches in the network it need to be in the same domain) (DTP tries to prevent this by generating a log message and drops the frames if the transparent switch in-transit is on a different domain)

(in v1 vtp transparent switch inspects vtp messages for the domain name and the version and forwards the messages only if the version and domain macthes. In v2 it forwards vtp messages in transparent mode without inspecting the version and the domain.)

(How VTP works)

1.vtp domain
1.which devices can exchange vtp advertisements
2.different vtp domains that share the same vlan numbers are still in the same broadcast domain. so vtp does not define the broadcast domain
3.defaults to null value and server mode. when first vtp domain is set it is replicated to other servers automatically. so when it is null it is susceptible, it sets the domain it gets
2.vtp mode (server(default)/client/transparent)
3.vtp revision nmumber (higher revision number means newer database)
4.update identity and update timestamp
5.MD5 vlan configuration
6.vtp will not work/replicate on a link that is not a trunk
7.2950 supports 128 vlans for vtp
8.v3 is run only on enterprise networks using extended vlan range and enhanced security/v2 supports token ring

(VTP security)

1.susceptible to attacks or misconfiguration where vlans are deleted
2.MD5 (one way hashing algo/can't be reverse engineered) authentication prevents against attacks but does not prevent against misconfiguration (transparent mode recommended)

(VTP pruning)

1.negotiates trunking allowed lists automatically

2.all unwanted vlans are pruned (removed) off the trunk
3.keeps unneccessary broadcasts, unknown multicasts and unknown unicasts traffic from crossing trunk links and improves bandwidth
4.only works on vtp servers and supported on client mode
5.does not work for transparent mode
6.VLANs not in 'prune eligible list' can not be pruned
7.if a switch is in a transit path of a vlan it will not prune it as it could be requested by the switch connected to it
8.VLANs 2 - 1001 are "prune eligible" (able to prune)
9.VLANs not in the "prune eligible list" cannot be pruned (traffic will always be sent recevied for them)
10.as vtp is cisco proprietary and if there is a non cisco device in the network not supporting vtp then the vtp pruning will not work as the non cisco device will request for all the vlans to be sent/recevived as it doesn't support pruning. So we will have to edit the prune eligible list manually as a solution.

(Manual Trunk Pruning)
1.Trunk's 'allowed list' controls what VLANs will forward over the link
2.all VLANs (1-4094) by default

(Extended VLANs)
1.normal vlans range is 1-1005
2.extended vlans is range 1006 -4094
3.requires vtp transparent mode unless VTPv3 is supported (eliminates the ability to run vtp pruning)

:::::::Commands/Configs:::::::

(vlan database)
1.vlan.dat file contains all vlan/vtp information
2.(to delete the vlans completly from the switch)
(there is no information stored in running config for vlans as its all stored in vlan.dat file)
delete flash:vlan.dat
(need rebooting it after this as it will still show in running config as they are memory resident)
3.sh flash:
(vlans are saved in vlan.dat)
->(sometimes the vlan.dat file can not handle the changes in the vlans and could get corrupted. So if you are having trouble with vlan/vtp even after everything is set up correctly, it's better removing vlan.dat file and re-do config)

(Old way of creating vlans/Creating VLANs in database mode/this mode is being deprecated)
S#vlan database
S(vlan)#vlan 100 name IT
S(vlan)#exit

(Cisco preferred way of creating vlans/Global configuration)
S(config)#vlan 100

```
S(config-vlan)#name IT
S(config-vlan)#exit
S(config)#vlan 30,40,50-55
S(config-vlan)#exit
S(config)#no vlan 30,40,50-55
->VLANs must be created even before they are assigned to a port as some
platforms create the vlan for you but some platforms don't and sometimes
there is no error checking
```

(VLAN Membership/Assigning vlan to a port/associating a port to a vlan)

```
S(config)#interface range fastethernet 0/1 - 10
S(config-if-range)#switchport mode access
S(config-if-range)#switchport access vlan 100

->by default all switchports are dynamic desirable(converts to access port
if PC is plugged in(no DTP packets) or converts to trunk if a switch is
plugged in(DTP packets)), which is a security concern.
```

:::::VLAN Verification/TSHOOT/Debug/Show commands::::::

```
1.sh vlan   (shows the individual ports in particular vlans)
2.sh vlan brief
3.sh interfaces fa0/1 switchport (mode and the vlan)
4.sh arp (to check mac addresses)
5.sh mac-address-table dynamic vlan 10 (to check particular vlan cam table)
6.sh mac-address-table dynamic (to check per vlan cam tables)

7.sh vlan id 1
8.sh vlan name marketing
9.sh interfaces vlan 1
10.sh run interface fa0/5
11.sh cdp nei
12.sh ip int bri
13.sh run
```

(Trunking/Trunk links/Dynamic Trunking Protocol(DTP))

```
S(config)#int fa0/15
S(config-if)#switchport trunk encapsulation dot1q (some switches like 2950
don't have trunk encapsulation command as they support only dot1q and not
ISL. 3550/3560 supports both)
or
S(config-if)#switchport trunk encapsulation isl
or
S(config-if)#switchport trunk encapsulation negotiate

S(config-if)#switchport mode trunk (other modes are dynamic
desirable/dynamic auto/access) (by default every switchport is dynamic
desirable thus sending DTP packets)
->if it comes up with a message: an interface whose trunk encap is 'auto'
can not be configured to 'trunk' mode, you will have to set the encap
dot1q/isl

S(config-if)#switchport nonnegotiate (stops sending DTP packets)
```

->you set the switchport as trunk and then nonnegotiate. you do it only for high availability applications like VoIP, for example you don't need to add DTP overhead as voice is calculated in ms. if switchport is in dynamic desirable it won't allow it. i.e negotiation of trunking will now be off. so if you restart the other side even if its in trunk this nonnegotiate side will not be sending dtp and no trunk will be formed (layer2 loop problem)(both sides must always agree trunking)

(Native Vlan)
S(config-if)#switchport trunk native vlan 1
S(config)#vlan dot1q tag native (will tag native vlan too)(will apply to all trunk links. By default native vlan is 1)

:::::Trunking Verification/TSHOOT/Debug/Show commands::::::
1.sh interfaces fa0/13 switchport (admin mode(set by command)/operational mode(after negotiation)/negotiation of trunk 'on/off'/trunking encapsulation/trunking native vlan/VLANs allowed/voice vlan)

2.sh interfaces status (to see if it's a trunk)
3.sh interface trunk (even to check native vlan number/n-isl default/'n' means negotiated)(shows which vlans are allowed/ vlans allowed and active in management domain)(vlans in spa forwarding and not pruned)

4.sh arp
5.sh mac address-table dynamic vlan 1
6.sh cdp neighbor
7.sh interface fastethernet 0/15 trunk
8.sh spa vlan 1 (if root bridge info matches that means trunk is working)(to check what ports are forwarding)

9.sh run int fa0/15
10sh interfaces switchport (admin mode/operational mode/encapsulation/negotiation('on' means sending DTP))
11.sh spa vlan 1
12.sh spa interface fa0/15 (same root bridge for all vlans if no priority is changed)(0 being the best spa priority)
13.sh int switchport

(vtp config)(domian/password/mode/version/vtp pruning)
S(config)#vtp domain CBTNugget (if a switch hasn't got a domain name it will be in a susceptible state/case sensitive/updates all other switches which are in a null state when set first time)

S(config)#vtp password cisco (VTP authentication)
S(config)#vtp mode server (deafult is server/ other modes are client/transparent)
S(config)#vtp version 2 (in version 1 transparent switches absorb the vtp rev updates rather than passing them on)(v3 can use extended vlans)
S(config)#vtp pruning
S(config)#no vtp pruning

->(now when you start adding/removing vlans it will keep increasing

revision number)
->(when device is in client mode it cannot create or modify or delete vlans)
->(vlan.dat file can sometimes get corrupt in case it wasn't previously deleted properly
casues problems to VTP)
->(order of operation problem can be eliminated by adding and deleting a vlan/force an update)

->(if you assign a vlan number to a port which doesn't even exit it won't communicate and there won't be any spa instance or mac address/switch parser doesn't do that)
sh spa vlan 200
sh mac-address-table dynamic vlan 200

:::::VTP Verification/TSHOOT/Debug/Show commands::::::
1.sh vtp status (shows revision number/max vlans supported/number of existing vlans/mode/domain/last modification server address)
->shows MD5 checksum mismtach as (vtp does authentication on everything as null authentication/the hash or the digest doesn't match that could happen in order of operation problem/solution to add/remove vlan to force update and check revision number)

2.sh vtp password
3.sh vtp status | include MD5
4.sh vtp status | include Revision|MD5 (revision and hash should match on all)

5.sh interface pruning (shows pruned vlans as not requested by nei/shows vlans traffic requested by nei)

6.sh interface trunk
7.sh vtp counters
8.sh interface switchport
9.sh spa interface fa0/5
10.sh interface fa0/5 pruning
11.sh flash:
12.sh vlan
13.sh mac-address-table dynamic vlan 101 (shows whether the frames are going to be forwarded or not for a particular vlan)
14.sh arp

(Defining Allowed vlans/VLAN Pruning/Manual Trunk Pruning/Controlling or allowing what vlans to send over the trunk link compared to VLAN pruning)
S(config-if)#switchport trunk allowed vlan 10,20,30
(all/except/none/add/remove keywords can also be used)
1.sh interface trunk
2.sh interface switchport

(VTP pruning)
S1(config)#vtp pruning
S(config-if)#switchport trunk pruning vlan remove 4, 20-30 (i.e prune

2-3,10-19,31-1001)

S(config-if)#switchport trunk pruning vlan except 40-50 (to edit prune eligible list)(i.e prune 2-39,51-1001)

->(only vlans included in the pruning-eligible list can be pruned. vlans 2 through 1001 are pruning eligible by default on trunk ports)
->(reserved vlans and extended--range vlans cannot be pruned)
1.sh interface trunk (vlans active in management domain/vlans in spa forwarding state and not pruned)

2.sh interface pruning
3.sh interface switchport
4.sh int fa0/5 pruning
5.sh int fa0/13 switchport (pruning vlans enabled)(trunking vlans enabled)

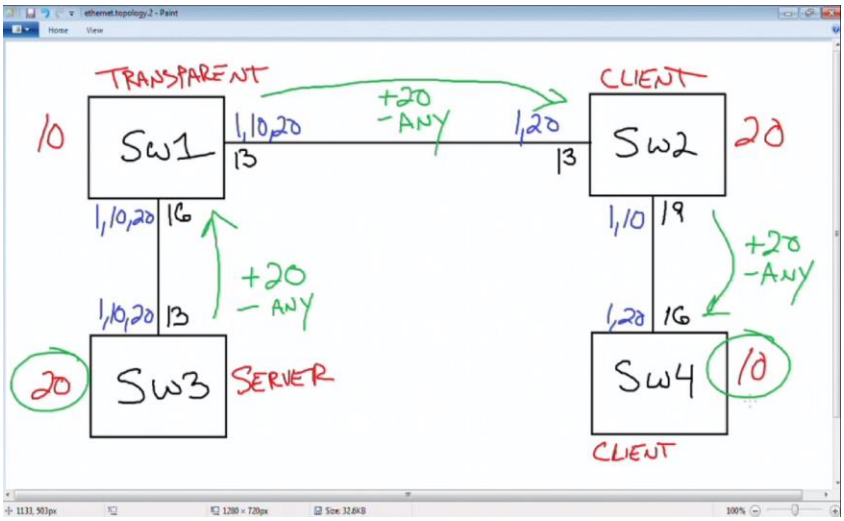
(VTP Pruning Problem)
-> (Specific problem to this sort of scenario) SW3 requests for only vlan 20 and transparent switch simply passes request further to SW2 and then further down to SW4. SW4 on the other hand requests for vlan 10 but doesn't send vlan 10 messages out as it was never requested by SW3 or SW2, but the problem here is that SW1 has got a vlan 10 for which we need the traffic to transmitted from SW4 towards SW1.
So always check using these commands:

1.sh int fa0/16 trunk (shows vlans in spa forwarding and not pruned) (shows all active vlans in management domain)

2.sh int fa0/16 pruning (shows vlans pruned for lack of request by nei) (shows vlan traffic requested of nei)

(check mac addresses in CAM tables)
3.sh arp
4.sh mac address-table dynamic vlan 10

Solution: not to use vtp pruning or manually edit prune eligible list



Topology Diagrams:
http://ccieccie.wordpress.com/category/ine-volume-1-lab-diagrams/

Layer 2 Access Switchports

Using the diagram for reference configure access VLAN assignments on SW1, SW2, SW3, and SW4 to obtain basic connectivity between the devices with Ethernet segments with the exception of R6.

Do not use VTP to accomplish this.

Configuration

```
SW1:
vlan 7,58,67,79,146
!
interface FastEthernet0/1
 switchport access vlan 146
!
interface FastEthernet0/5
 switchport access vlan 58

SW2:
vlan 8,22,43,58
!
interface FastEthernet0/2
 switchport access vlan 22
!
interface FastEthernet0/4
 switchport access vlan 43
!
interface FastEthernet0/24
 switchport access vlan 22

SW3:
vlan 5,9,43,79
!
interface FastEthernet0/5
 switchport access vlan 5
!
interface FastEthernet0/24
 switchport access vlan 43

SW4:
vlan 10,146
!
interface FastEthernet0/4
 switchport access vlan 146
```

Rack1SW1#ping 155.1.79.9

Rack1SW1#ping 155.1.37.3

Rack1SW2#ping 155.1.58.5

Rack1R1#ping 155.1.146.4

Rack1R2#ping 192.10.1.254

Rack1R4#ping 204.12.1.254

Rack1SW1#show interface status

Rack1SW2#show interface status

Rack1SW3#show interface status

SW4#show interface status

Layer 2 Dynamic Switchports

Configure all inter-switch links on SW2, SW3, and SW4 to be in dynamic auto state.

Configure all inter-switch links on SW1 to be in dynamic desirable state.

Using the CAM table verify that all layer 2 traffic between devices in the same VLAN, but not attached to the same switch, is transiting SW1.

Configuration

```
SW1:
interface range FastEthernet0/13 - 21
 switchport mode dynamic desirable

SW2:
interface range FastEthernet0/13 - 21
 switchport mode dynamic auto

SW3:
interface range FastEthernet0/13 - 21
 switchport mode dynamic auto

SW4:
interface range FastEthernet0/13 - 21
 switchport mode dynamic auto
```

```
Rack1R4#ping 155.1.146.6

Rack1R4#show arp

Rack1SW4#show mac address-table dynamic address 0011.2031.4461

  Rack1SW2#show mac address-table dynamic address 000f.24da.2220

Rack1SW2#show mac address-table dynamic address 0011.2031.4461

Rack1SW4#show mac address-table dynamic address 000f.24da.2220
```

ISL Trunking

Statically set the trunking encapsulation of SW1's inter-switch links to ISL.

Verify that SW2, SW3, & SW4 are negotiating ISL as the trunking encapsulation to SW1, and that SW1 is not negotiating ISL to SW2, SW3, and SW4.

Configuration

```
SW1:
interface range FastEthernet0/13 - 21
 switchport trunk encapsulation isl
```

```
Rack1SW1#show interface trunk

Rack1SW2#show interface trunk

Rack1SW3#show interface trunk

Rack1SW4#show interface trunk
```

802.1q Trunking

Change the trunking encapsulation on SW1's inter-switch links from static ISL to static 802.1q.

Verify that SW2, SW3, & SW4 are negotiating 802.1q as the trunking encapsulation to SW1, and that SW1 is not negotiating 802.1q to SW2, SW3, and SW4.

Configuration

```
SW1:
interface range FastEthernet0/13 - 21
 switchport trunk encapsulation dot1q
```

```
Rack1SW1#show interface trunk

Rack1SW2#show interface trunk

Rack1SW3#show interface trunk

Rack1SW4#show interface trunk
```

802.1q Native VLAN

Modify the native VLAN on the 802.1q trunks of SW1 so that traffic between devices in VLAN 146 is not tagged when sent over the trunk links.

Configuration

```
SW1:
interface range FastEthernet0/13 - 21
 switchport trunk native vlan 146

SW2:
interface range FastEthernet0/13 - 15
 switchport trunk native vlan 146

SW3:
interface range FastEthernet0/13 - 15
 switchport trunk native vlan 146

SW4:
interface range FastEthernet0/13 - 15
 switchport trunk native vlan 146
```

```
Rack1SW1#show interface trunk

Rack1SW2#show interface trunk

Rack1SW3#show interface trunk

Rack1SW4#show interface trunk
```

Disabling DTP Negotiation

Disable Dynamic Trunking Protocol on the trunk links of SW1.

Verify that trunking is still occurring between SW1 & SW2, SW1 & SW3, and SW1 & SW4 without the use of DTP.

Configuration

```
SW1:
interface range FastEthernet0/13 - 21
 switchport trunk encapsulation dot1q
 switchport mode trunk
 switchport nonegotiate

SW2:
interface range FastEthernet0/13 - 15
 switchport trunk encapsulation dot1q
 switchport mode trunk
 switchport nonegotiate

SW3:
interface range FastEthernet0/13 - 15
 switchport trunk encapsulation dot1q
 switchport mode trunk
 switchport nonegotiate

SW4:
interface range FastEthernet0/13 - 15
 switchport trunk encapsulation dot1q
 switchport mode trunk
 switchport nonegotiate
```

```
Rack1SW1#show interface fa0/13 switchport | include Negotiation
 Rack1SW1#show interface trunk
Rack1SW2#show interface trunk
Rack1SW3#show interface trunk
Rack1SW4#show interface trunk
```

VTP

Configure all inter-switch links on SW2, SW3, and SW4 to be in dynamic auto state.

Configure all inter-switch links on SW1 to be in dynamic desirable state.

Configure SW2 as a VTP server in the domain CCIE.

Configure SW1, SW3, and SW4 as VTP clients in the domain CCIE.

Configure necessary VLAN definitions on SW2 using the diagram for reference

Configure access VLAN assignments on SW1, SW2, SW3, and SW4 to obtain basic connectivity between the devices with Ethernet segments.

Configure router-on-a-stick between SW2 and R6 per the diagram so R6 has reachability to devices on VLANs 67 and 146.

Configuration

```
R6:
interface FastEthernet0/0.67
 encapsulation dot1q 67
 ip address 155.1.67.6 255.255.255.0
!
interface FastEthernet0/0.146
 encapsulation dot1q 146
 ip address 155.1.146.6 255.255.255.0

SW1:
vtp domain CCIE
vtp mode client
!
interface range FastEthernet0/13 - 21
 switchport mode dynamic desirable
!
interface FastEthernet0/1
 switchport access vlan 146
!
interface FastEthernet0/5
 switchport access vlan 58

SW2:
vtp domain CCIE
vlan 5,7,8,9,10,22,43,58,67,79,146
!
interface FastEthernet0/2
 switchport access vlan 22
!
interface FastEthernet0/4
 switchport access vlan 43
!
interface FastEthernet0/6
 switchport trunk encapsulation dot1q
 switchport mode trunk
!
```

```
interface FastEthernet0/24
 switchport access vlan 22
!
interface range FastEthernet0/13 - 21
 switchport mode dynamic auto
```

```
SW3:
vtp domain CCIE
vtp mode client
!
interface FastEthernet0/5
 switchport access vlan 5
!
interface FastEthernet0/24
 switchport access vlan 43
!
interface range FastEthernet0/13 - 21
 switchport mode dynamic auto
```

```
SW4:
vtp domain CCIE
vtp mode client
!
interface FastEthernet0/4
 switchport access vlan 146
!
interface range FastEthernet0/13 - 21
 switchport mode dynamic auto
```

```
Rack1SW1#show vtp status
Rack1SW2#show vtp status
Rack1SW3#show vtp status
Rack1SW4#show vtp status
Rack1SW1#show vlan brief
Rack1SW2#show vlan brief
Rack1SW3#show vlan brief
Rack1SW4#show vlan brief
```

VTP Transparent

Configure SW1 in VTP transparent mode and remove all previous VLAN definitions from it.

Configure SW1 with only the VLAN definitions necessary to obtain basic connectivity between the devices with Ethernet segments.

Configuration

```
SW1:
vtp mode transparent
no vlan 2-1000
vlan 7,43,58,67,79,146
```

```
Rack1SW1#show vtp status
```

```
Rack1SW1#show vlan brief
```

```
Rack1SW2#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Rack1SW2(config)#vlan 123
Rack1SW2(config-vlan)#end
Rack1SW2#
```

```
Rack1SW2#show vlan | include ^123
123  VLAN0123                         active
123  enet  100123     1500  -      -      -        -    -        0      0
```

```
Rack1SW3#show vlan | include ^123
123  VLAN0123                         active
123  enet  100123     1500  -      -      -        -    -        0      0
```

```
Rack1SW1#show vlan | include ^123
```

VTP Pruning

Configure SW1 in VTP client mode.

Enable VTP pruning in the layer 2 network so that inter-switch broadcast replication is minimized.

Verify this configuration is functional through the **show interface trunk** output

Configuration

```
SW1:
vtp mode client

SW2:
vtp pruning
```

```
Rack1SW1#show interface fa0/16 pruning
Rack1SW3#show interface fa0/13 pruning
Rack1SW1#show vtp status
Rack1SW2#show vtp status
Rack1SW3#show vtp status
Rack1SW4#show vtp status
Rack1SW1#show interface trunk | begin pruned
Rack1SW2#show interface trunk | begin pruned
Rack1SW3#show interface trunk | begin pruned
Rack1SW4#show interface trunk | begin pruned
```

VTP Prune-Eligible List

Edit the prune-eligible list to ensure that traffic for VLAN 7 is carried on all active trunk links in the layer 2 network.

Verify this configuration is functional through the **show interface trunk** output.

Configuration

```
SW1:
interface FastEthernet0/13
 switchport trunk pruning vlan 2-6,8-1001
!
interface FastEthernet0/14
 switchport trunk pruning vlan 2-6,8-1001
!
interface FastEthernet0/15
 switchport trunk pruning vlan 2-6,8-1001
!
interface FastEthernet0/16
 switchport trunk pruning vlan 2-6,8-1001
!
interface FastEthernet0/17
 switchport trunk pruning vlan 2-6,8-1001
!
interface FastEthernet0/18
 switchport trunk pruning vlan 2-6,8-1001
!
interface FastEthernet0/19
 switchport trunk pruning vlan 2-6,8-1001
!
interface FastEthernet0/20
 switchport trunk pruning vlan 2-6,8-1001
!
interface FastEthernet0/21
 switchport trunk pruning vlan 2-6,8-1001

SW2:
interface FastEthernet0/13
 switchport trunk pruning vlan 2-6,8-1001
!
interface FastEthernet0/14
 switchport trunk pruning vlan 2-6,8-1001
!
interface FastEthernet0/15
 switchport trunk pruning vlan 2-6,8-1001
```

```
SW3:
interface FastEthernet0/13
 switchport trunk pruning vlan 2-6,8-1001
!
interface FastEthernet0/14
 switchport trunk pruning vlan 2-6,8-1001
!
interface FastEthernet0/15
 switchport trunk pruning vlan 2-6,8-1001

SW4:
interface FastEthernet0/13
 switchport trunk pruning vlan 2-6,8-1001
!
interface FastEthernet0/14
 switchport trunk pruning vlan 2-6,8-1001
!
interface FastEthernet0/15
 switchport trunk pruning vlan 2-6,8-1001
```

```
Rack1SW1#show interface trunk | begin pruned
Rack1SW2#show interface trunk | begin pruned
Rack1SW3#show interface trunk | begin pruned
Rack1SW4#show interface trunk | begin pruned
```