

# (VLAN) VIRTUAL LOCAL AREA NETWORK

## Implementing VLANS

### Abstract

A VLAN is a logical, software-defined subnetwork. It allows similar devices on the network to be grouped together into one broadcast domain, irrespective of their physical position in the network.

## CONTENTS

Virtual LANs.....	3
What is a VLAN?.....	3
The purpose of VLANs.....	3
Using routers to segment LANs .....	4
Using switches to segment LANs .....	4
Domain terminology .....	5
Using VLANs to segment LANs .....	5
Advantages of using VLANs:.....	6
Implementing VLANs .....	7
Port-based VLANs .....	7
Distributing a single VLAN across multiple switches .....	7
How does tagging work?.....	8
Mixing tagged and untagged packets on the same port .....	10
Configuring VLANs .....	10
Defaults .....	10
VLAN names .....	10
Access mode .....	11
Trunk mode .....	11
Deploying VLAN .....	12
What do I need to deploy VLANs on my network?.....	12
What kind of information is in a deployment plan? .....	12
What constitutes a deployment plan?.....	12
VLAN Configuration Exercise .....	15
Configure VLAN on core switch (VLAN20 – HS- Vlan).....	16
Configure VLAN Port Assignments on Core-Switch (VLAN 20 – High School Vlan) .....	17
Configure the building’s distribution-layer switch.....	19
Verify Access/Trunk Ports .....	21
DHCP Configuration .....	22
Adding New VLAN to Existing CISCO VLAN Database (new WiFi VLAN 40 and 50) .....	25
Create New VLAN Core-Switch (Student Wifi VLAN 40) .....	25
Configure BootP/DHCP Relay Core-Switch (IP Helper-Address).....	25
Add VLAN to Trunk Port Core-Switch .....	26

Create VLAN on Distribution Switch (HS-Switch).....	26
Add VLAN to Trunk Port Distribution Switch (HS-Switch).....	26
What to do with Wireless Access Points (Trunk or Access?) .....	26
Change wireless AP port to Trunk Port and assign vlans (HS-Switch) .....	27
Changing Wireless AP's Management VLAN Membership (Native VLAN) .....	27
Access Control.....	28
Access Control List Mechanics .....	28
Standard ACLs .....	28
Applying ACLs.....	30
Spanning Tree Protocol.....	33
What is Spanning Tree Protocol and what does it do? .....	33
Spanning Tree Exercise: .....	36
Appendix A .....	42
Configure Windows Server DHCP Scope .....	42

## VIRTUAL LANS

A VLAN is a logical, software-defined subnetwork. It allows similar devices on the network to be grouped together into one broadcast domain, irrespective of their physical position in the network. Multiple VLANs can be used to group workstations, servers, and other network equipment connected to the switch, according to similar data and security requirements.

### WHAT IS A VLAN?

In simple terms, a VLAN is a set of workstations within a LAN that can communicate with each other as though they were on a single, isolated LAN. What does it mean to say that they “*communicate with each other as though they were on a single, isolated LAN*”?

Among other things, it means that:

- Broadcast packets sent by one of the workstations will reach all the others in the VLAN.
- Broadcasts sent by one of the workstations in the VLAN will not reach any workstations that are not in the VLAN.
- Broadcasts sent by workstations that are not in the VLAN will never reach workstations that are in the VLAN.
- The workstations can all communicate with each other without needing to go through a gateway.  
For example, IP connections would be established by ARPing for the destination IP and sending packets directly to the destination workstation—there would be no need to send packets to the IP gateway to be forwarded on.
- The workstations can communicate with each other using non-routable protocols.

### THE PURPOSE OF VLANS

The basic reason for splitting a network into VLANs is to reduce congestion on a large LAN. To understand this problem, we need to look briefly at how LANs have developed over the years. Initially LANs were very flat—all the workstations were connected to a single piece of coaxial cable, or to sets of chained hubs. In a flat LAN, every packet that any device puts onto the wire gets sent to every other device on the LAN.

As the number of workstations on the typical LAN grew, they started to become hopelessly congested; there were just too many collisions, because most of the time when a workstation tried to send a packet, it would find that the wire was already occupied by a packet sent by some other device.

---

## USING ROUTERS TO SEGMENT LANS

The early solution to this problem was to segment the network using routers. This would split the network into a number of smaller LANs. There would be less workstations on each LAN, and so less congestion.

Of course, routable data being sent between LANs would have to be routed, so the layer 3 addresses would have to be organized so that each LAN had an identifiable set of addresses that could be routed to—such as an IP subnet or an AppleTalk zone. Non-routable protocols would have to be bridged, which is not quite so congestion-reducing, because bridges forward all broadcasts. But, at least for unicast packets, a bridge only forwards packets if it knows that the destination address is not in the originating LAN.

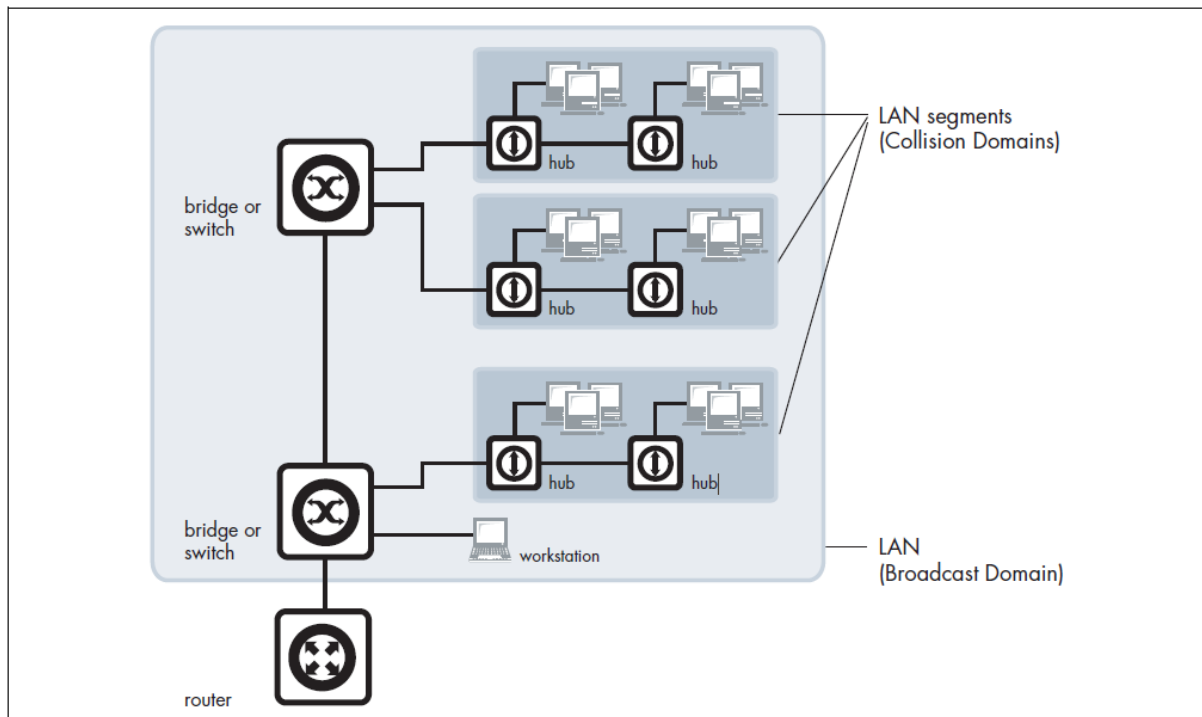
---

## USING SWITCHES TO SEGMENT LANS

As switches became more available, there was a move from chained hubs to a set of hubs connected to a switch. A switch only sends traffic to a given port if the traffic has to go to that port. So switches have the effect of reducing congestion at workstations, by stopping the workstations from seeing all the traffic from the other ports of the switch.

A simple switched network, though, still needs routers to set the boundaries of where broadcasts are sent. So, the typical LAN was set up as shown below in Figure 1.

Figure 1: Typical VLAN



---

## DOMAIN TERMINOLOGY

The above figure introduces the concept of a LAN segment. This is also referred to as a collision domain, because when a device is trying to send a packet, it can only collide with packets sent by other devices on the same segment. Each LAN segment consists of all the devices attached to a single switch port—the switch stops packets from different ports from colliding with each other.

The LAN itself is referred to as a broadcast domain, because if any device within the LAN sends out a broadcast packet, it will be transmitted to all devices in that LAN, but not to devices beyond the LAN.

---

## USING VLANS TO SEGMENT LANS

As LANs became larger, data rates became faster, and users desired greater flexibility, the routers in a network started to become a bottleneck. This is because:

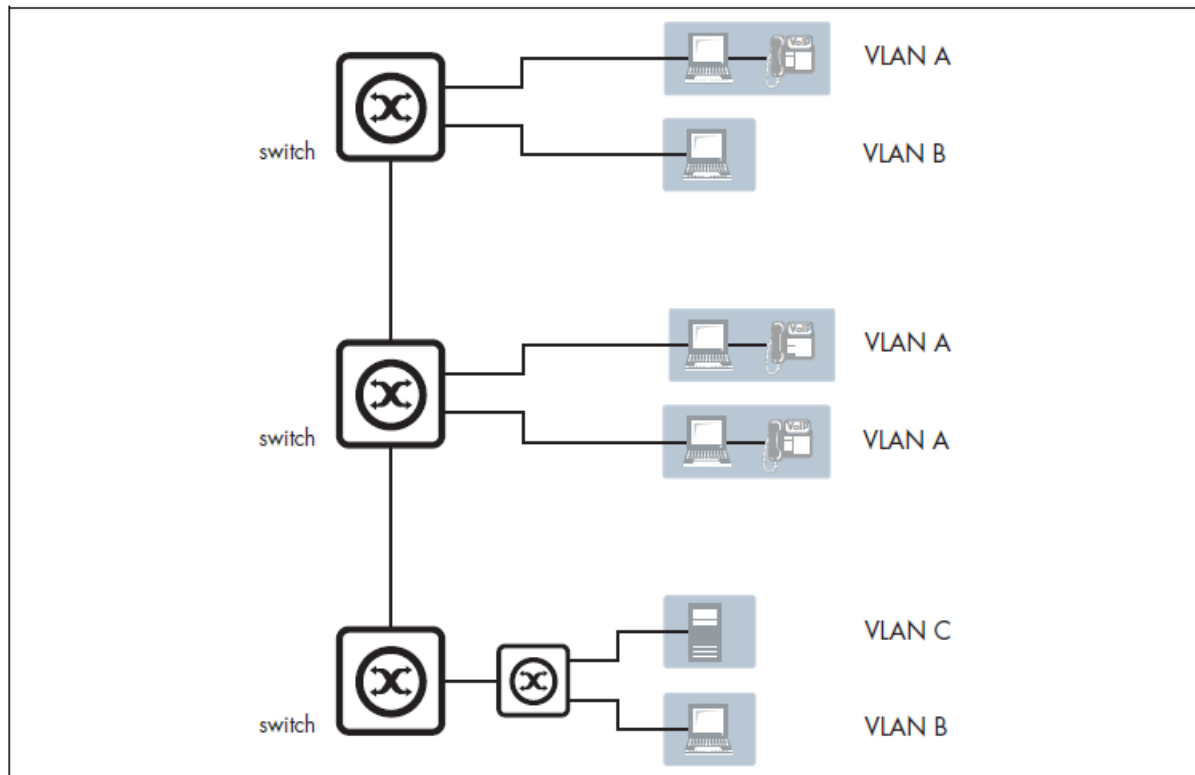
- Routers typically forward data in software, and so are not as fast as switches.
- Splitting up a LAN using routers meant that a LAN typically corresponded to a particular physical location. This became limiting when many users had laptops, and wanted to be able to move between buildings, but still have the same network environment wherever they plugged in.

Switch vendors started implementing methods for defining “virtual LANs”—sets of switch ports, usually distributed across multiple switches, which somehow interacted as though they were in a single isolated LAN. This way, workstations could be separated off into separate LANs without being physically divided up by routers.

At about the same time, hubs became less popular and have been largely replaced by L2 switches. This has made the whole concept of a collision domain somewhat historical. In modern networks, a “collision domain” mostly consists of a single device attached to an L2 switch port, or possibly a PC with something like an IP phone attached to it.

So, the layout of the LAN has become more like this next diagram:

Figure 2: Segmented VLAN



Instead of the LANs corresponding to physical areas divided from each other by routers, there are virtual LANs distributed across the network. For example, all the devices in the various areas labeled “LAN A” all belong to a single virtual LAN—i.e. a single broadcast domain.

#### ADVANTAGES OF USING VLANS:

1. **Performance.** As mentioned above, routers that forward data in software become a bottleneck as LAN data rates increase. Doing away with the routers removes this bottleneck.
2. **Broadcast Control.** Segment devices into smaller LAN broadcast domains to reduce overhead caused to each host in the VLAN. An average number of broadcasts should be 30 broadcasts per second, or less.
3. **Formation of virtual workgroups.** Because workstations can be moved from one VLAN to another just by changing the configuration on switches, it is relatively easy to put all the people working together on a particular project all into a single VLAN. They can then more easily share files and resources with each other.
4. **Greater flexibility.** If users move their desks, or just move around the place with their laptops, then, if the VLANs are set up the right way, they can plug their PC in at the new location, and still be within the same VLAN. This is much harder when a network is physically divided up by routers.

5. **Ease of partitioning off resources (Security).** If there are servers or other equipment to which the network administrator wishes to limit access, then they can be put off into their own VLAN. Then users in other VLANs can be given access selectively.

## IMPLEMENTING VLANS

### PORT-BASED VLANS

In the previous section, we simply stated that the network is split up into sets of virtual LANs. It is one thing to say this, it is quite another thing to understand how this is actually achieved.

Fundamentally, the act of creating a VLAN on a switch involves defining a set of ports, and defining the criteria for VLAN membership for workstations connected to those ports. By far the most common VLAN membership criteria is port-based. With port-based VLANs, the ports of a switch are simply assigned to VLANs, with no extra criteria.

Table 1: Port-based VLAN assignment

PORT	VLAN
1	1
2	1
3	2
4	1

All devices connected to a given port automatically become members of the VLAN to which that port was assigned. In effect, this just divides a switch up into a set of independent sub-switches.

---

### DISTRIBUTING A SINGLE VLAN ACROSS MULTIPLE SWITCHES

The figure "Segmented VLAN" on page 6 is an example of a VLAN-based network. It shows some of VLAN A connected to one switch, and some more of VLAN A connected to another switch. You may be asking "*Are these both part of the same VLAN A, or separate VLANs that all happen to be called VLAN A?*" The answer is that they are all parts of the same VLAN—there is a single VLAN A that is spread across two switches.

How is this achieved? How does one switch know that when it receives a broadcast packet that it associates to VLAN A that it must also forward that broadcast to other switches?



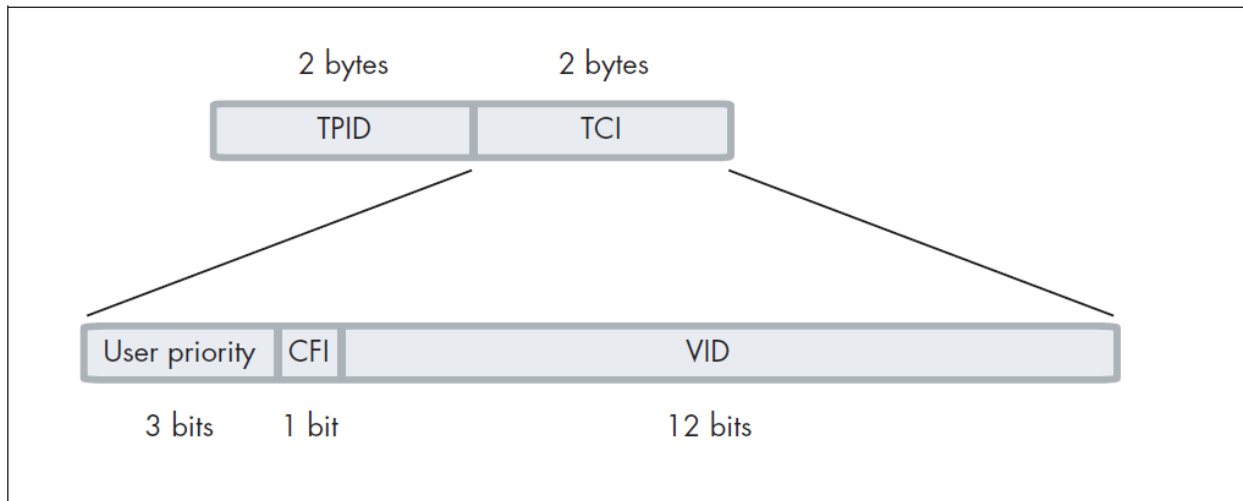
This can be done in a number of different ways, and in the early days of VLANs, just about every one of these ways was tried. Some vendors had their switches use a proprietary protocol to inform each other of their VLAN tables; some vendors used time-divided multiplexing in which different time slots were allocated to different VLANs; other vendors used frame tagging. In the end, frame tagging became the accepted standard. As we will see, in most respects this is a simple and elegant solution. However, it initially had one big downside: it required a fundamental change to the format of the Ethernet header. This split the world's Ethernet devices into those that recognized tagged headers and those that did not recognize tagged headers. In other words, a lot of Ethernet equipment was rendered obsolete.

---

#### HOW DOES TAGGING WORK?

Simply, 4 bytes are inserted into the header of an Ethernet packet. This consists of 2 bytes of Tag Protocol Identifier (TPID) and 2 bytes of Tag Control Information (TCI), as shown in the diagram below:

Figure 3: Tagged Ethernet packet



**TPID** is the tag protocol identifier, which indicates that a tag header is following and contains the user priority, Canonical Format Indicator (CFI), and the VLAN ID.

**User Priority** is a 3-bit field that allows priority information to be encoded in the frame. Eight levels of priority are allowed, where zero is the lowest priority and seven is the highest priority.

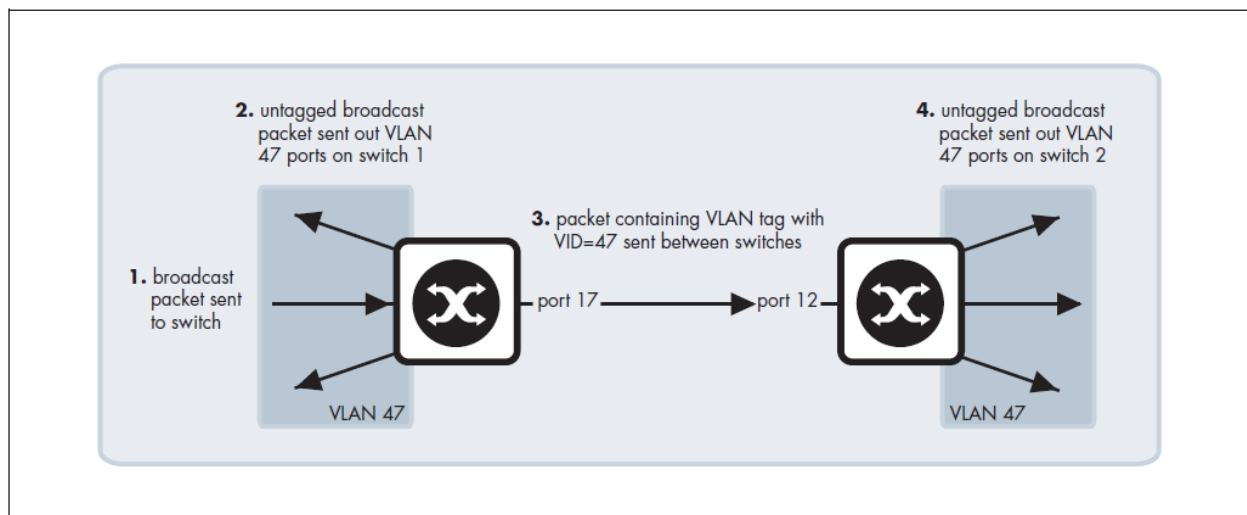
The **CFI** is a 1-bit indicator that is always set to zero for Ethernet switches. CFI is used for compatibility between Ethernet and Token Ring networks. If a frame received at an Ethernet port has a CFI set to 1, then that frame should not be bridged to an untagged port.

Then, the **VID** field contains the identifier of the VLAN. Actually, it is only the VID field that is really needed for distributing VLANs across switches—but the IEEE decided that while they were altering the format of the Ethernet header, they might as well add the User Priority and CFI too.

Let us see how this tag makes it simple to distribute VLANs across switches.

- Consider a broadcast packet arriving at a switch port. By some criterion, the packet is associated with VLAN 47, i.e. a VLAN with VLAN ID=47. Now, port 17 of this switch is connected to port 12 of another switch that also has some ports in VLAN 47.
- The network administrator needs to configure port 17 of switch 1 and port 12 of switch 2 as “tagged” member ports of VLAN 47. This tells switch 1 to send the broadcast out port 17 as a tagged packet, with VID=47 in the tag. And it tells switch 2 to accept that tagged packet and associate it with VLAN 47.
- Then, switch 2 will send the packet out all its member ports of VLAN 47, because that is what it does with broadcasts that it has associated with VLAN 47.

Figure 4: Distribute VLANs across multiple switches



The tag makes it very easy for the second switch to know what to do with the packet, because the tag marks this packet as belonging to VLAN 47, and switch 2 knows exactly what it should do with packets that belong to VLAN 47.

So, there really are only **two simple rules**:

- If a port is a tagged member of a VLAN, then any packets sent out that port by that VLAN must have a tag inserted into the header.
- If a tagged packet arrives in at a port, and the port is a tagged member of the VLAN corresponding to the VID in the packet's tag, then the packet is associated with that VLAN.

With these two simple rules, it is possible to distribute VLANs across multiple switches.

#### MIXING TAGGED AND UNTAGGED PACKETS ON THE SAME PORT

In the previous section, we discussed using tags to indicate the VLAN membership of packets that are transferred from one switch over to another. But, it is also possible that untagged packets will be transported across that link that joins the two switches.

For example, it could be that port 17 of switch 1 is an untagged member of VLAN 56, and port 12 of switch 2 is an untagged member of VLAN 56. In this case, if switch 1 needed to transport VLAN 56 packets over to switch 2, it would send them untagged.

When those untagged packets arrived at switch 2, what VLAN would switch 2 decide to associate these packets with, given that they do not have a tag to indicate their VLAN membership? Well, in fact, switch 2 would realize that VLAN 56 is the untagged VLAN on the receiving port, so untagged packets would be deemed to belong to VLAN 56.

Obviously, a port can be an untagged member of only one port-based VLAN, otherwise there would be uncertainty about what VLAN incoming untagged packets belonged to. This VLAN is often referred to as the **native** VLAN of the port.

Often, you might not want to associate a native VLAN with the port that connects a switch to another switch, so that all packets coming into that port must use a VLAN tag to indicate their VLAN membership. This stops the switch from accepting any untagged packets on the port.

## CONFIGURING VLANS

### DEFAULTS

By default, all switch ports are in access mode, are associated with the default VLAN (**vlan1**). You cannot delete **vlan1**.

### VLAN NAMES

When you create a VLAN (using the **vlan <ID>** command), you give it a numerical VLAN Identifier (VID) - a number from 2 to 4094, which is included in VLAN-tagged Ethernet frames to and from this VLAN. If tagged frames are transmitted from this VLAN, they will contain this VID in their tag. You may also give it an arbitrary alphanumeric name containing a meaningful description, which is not transmitted to other devices.

When referring to a VLAN, some commands require the VLAN to be specified by its VID while some commands require it to be specified by its interface name: `vlan<VID>`. In command output, the VLAN may be referred to by its VID, its interface name (`vlan<VID>`), or its VLAN name (the arbitrary alphanumeric string).

---

#### ACCESS MODE

A switch port in access mode sends untagged Ethernet frames, that is, frames without a VLAN tag. Each port is associated with one VLAN (the port-based VLAN, by default, `vlan1`), and when it receives untagged frames, it associates them with the VID of this VLAN. You can associate the port with another VLAN (using the **`switchport access vlan`** command or **`untagged <portID>`** command). This removes it from the default VLAN.

Use access mode (untagged) for any ports connected to devices that do not use VLAN tagging, for instance PC workstations, printers, etc.

---

#### TRUNK MODE

A switch port in trunk mode is associated with one or more VLANs for which it transmits VLAN-tagged frames, and for which it identifies incoming tagged frames with these VIDs.

To allow a switch port to distinguish and identify traffic from different VLANs, put it in trunk mode (using the **`switchport mode trunk`** command or **`tagged <portID>`** command), and add the VLANs (using the **`switchport trunk allowed vlan`** command). Use trunk mode (tagged) for ports connected to other switches which send VLAN-tagged traffic from one or more VLANs.

A trunk mode (tagged) port may also have a native VLAN (by default `vlan1`), for which it transmits untagged frames, and with which it associates incoming untagged frames (using the **`switchport trunk native vlan`** command).

---

### WHAT DO I NEED TO DEPLOY VLANS ON MY NETWORK?

- A current network diagram.
- VLAN compatible Layer 2 switches and a Layer 3 device (switch or router)
- A deployment plan.

---

### WHAT KIND OF INFORMATION IS IN A DEPLOYMENT PLAN?

- Device Names/Locations
- IP Addresses
- Port numbers
- Cabling connections

It is almost a given that all of you have the necessary equipment to implement basic VLANs. Cisco or HP switches are commonly designed for VLAN implementation.

A layer 3 device is required for routing and access control of traffic passing from one VLAN to another

---

### WHAT CONSTITUTES A DEPLOYMENT PLAN?

For traffic to pass from one VLAN to another, the traffic must pass through a Layer-3 routing device, such as a Layer-3 switch or a network router. Deploying only one VLAN on your network would be ineffective at reducing your broadcast domains, or providing more secure access to resources, this makes having a layer-3 device essential.

In beginning to decide what VLANs we will deploy, we need to identify what we want to accomplish with the VLAN's we will create. The purposes stated above as reasons to use VLANs are not meant to be all inclusive. For example, you may want to use VLANs to segregate IP phone traffic so that it is logically separated from other network traffic.

If one of our intentions is to reduce the size of broadcast domains, we need to consider how we will group devices on each VLAN. Where possible we should limit the number of devices on a single VLAN so that broadcast traffic would remain under the estimate of 30 broadcasts per second.

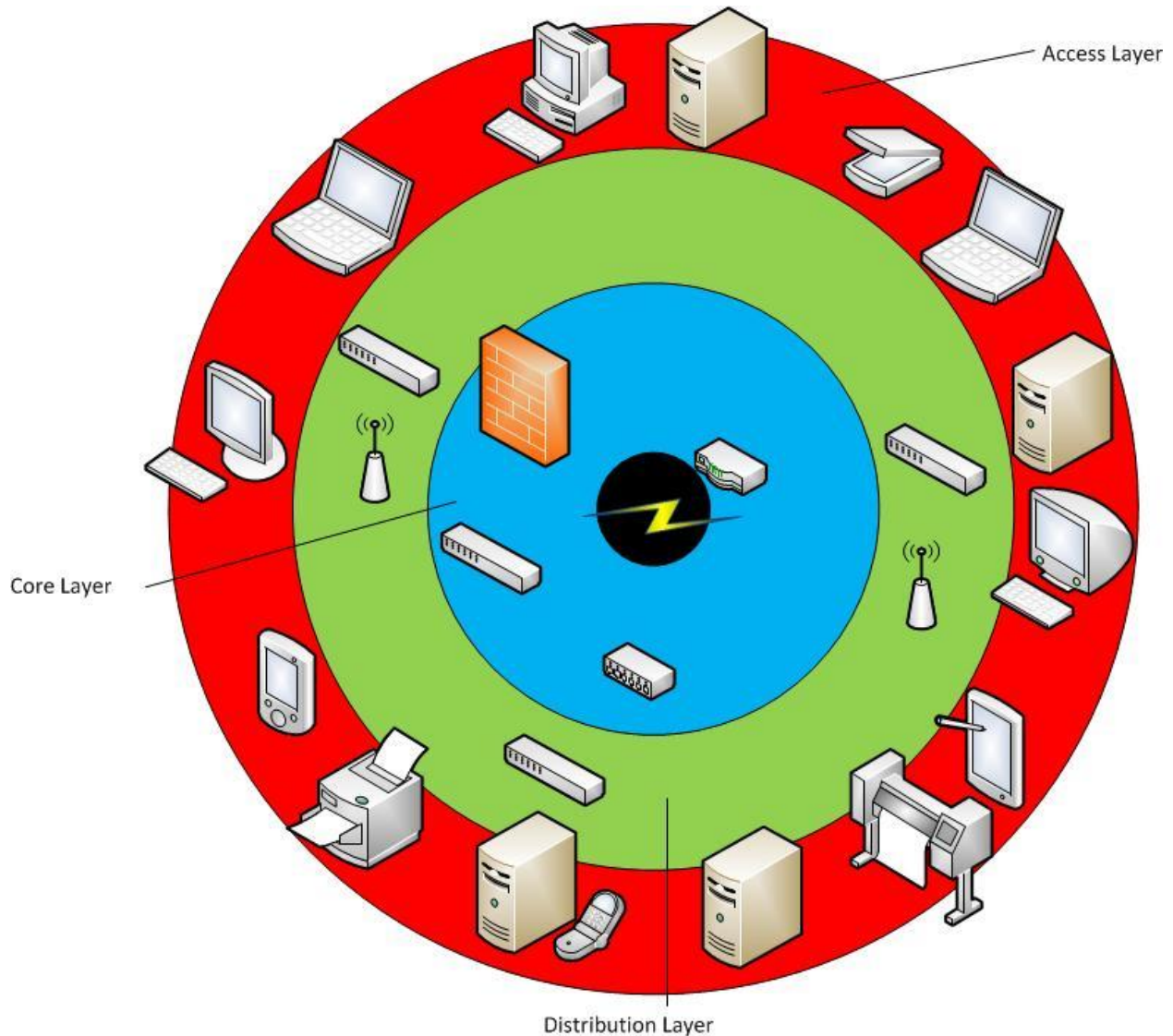
For the purpose of grouping resources, we need to try to put as many of the resources needed by a device on the same VLAN with the device. Traffic having to be transferred through a layer-3 device could introduce additional latency.

In terms of serving IP management, the opposite would be true. Servers on their own subnet, and clients on a different subnet, would require routing through a layer-3 device, but the IP space would be more easily managed.

As a rule of thumb, keep the number of VLANs to a minimum. Each additional VLAN significantly increase the overhead for management.

In your handout is a deployment plan worksheet. Using the network diagram, the pertinent information on user devices and switches can be entered and information about the prospective VLAN structure can be developed based upon this data.

In developing your deployment plan it is useful to look at your network structure as a form of bulls-eye.



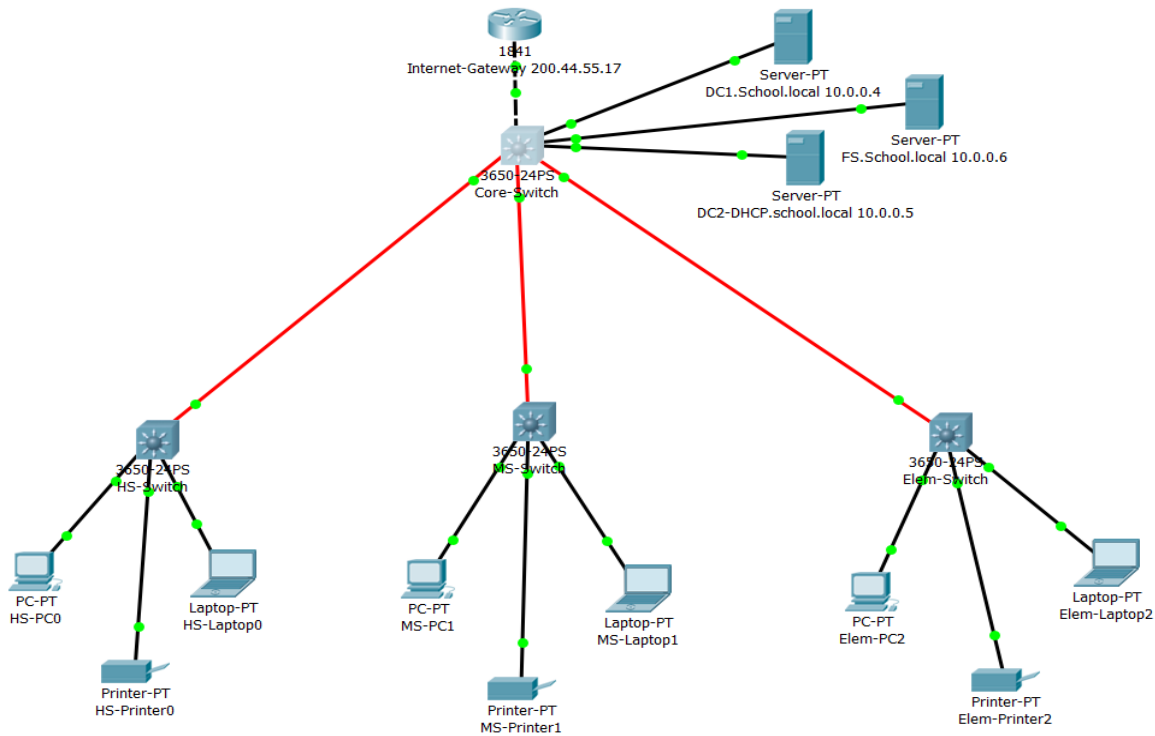
In the outermost ring you have user devices such as PC's and network printers, servers, scanners, smartphones, network enabled pads, network enabled copy machines etc. This is often referred to as the access layer.

In the next inward ring you have network connectivity devices like secondary switches and wireless access points. This is often referred to as the distribution layer.

The third ring contains core devices that serve the entire network such as: Firewall appliances, Bandwidth Aggregators, Web filters and network routers. This is often called the Core layer.

For VLAN development, we first need to assess what groupings need to be established in the access layer devices. Criteria for grouping devices can be whatever natural segregation exists between device users, or device uses on your network. For example a natural segregation for most school districts users exists between, building locations, or types of users such as: Administrators, Faculty, Staff and Students. These require differing access to shared resources, and access to different network resources.

## VLAN CONFIGURATION EXERCISE



High School 325 devices

Middle School 210 devices

Elementary school 130 devices

Device	Uplink Port	Upstream device	Downlink ports	Downstream device
<b>Core-Switch</b>	Gig1/0/1	Internet Gateway	Gig1/1/1	HS-Switch
			Gig1/1/2	MS-Switch
			Gig1/1/3	EL-Switch
			Gig1/0/10	DC1
			Gig1/0/11	FS1
			Gig1/0/12	DC2-DHCP
<b>HS-Switch</b>	Gig1/1/1	Core-Switch	Gig1/0/1	HS-PC0
			Gig1/0/2	HS-LT0
			Gig1/0/3	HS-Ptr0
<b>MS-Switch</b>	Gig1/1/1	Core-Switch	Gig1/0/1	MS-PC1
			Gig1/0/2	MS-LT1
			Gig1/0/3	MS-Ptr1
<b>EL-Switch</b>	Gig1/1/1	Core-Switch	Gig1/0/1	Elem-PC2
			Gig1/0/2	Elem-LT2
			Gig1/0/3	Elem-Ptr2
School	IP Address Range		Subnet Mask	
<b>High School</b>	10. . .0		255.255. .0	
<b>Middle School</b>	10. . .0		255.255. .0	
<b>Elementary School</b>	10. . .0		255.255. .0	

DNS Server 10.0.0.4 Device password : cisco



It has been determined that the changes to the private IP address space will be limited to establishing separate address ranges for each school building.

It is a goal of the configuration that no VLAN should serve more than 600 devices.

---

#### CONFIGURE VLAN ON CORE SWITCH (VLAN20 – HS- VLAN)

- Core-Switch>
- Core-Switch>**en**
- Password:
- Core-Switch#
- Core-Switch#**configure t**  
Enter configuration commands, one per line. End with CNTL/Z.
- Core-Switch(config)#**vlan 20**
- Core-Switch(config-vlan)#**name "High School VLAN"**
- Core-Switch(config-vlan)#**exit**
- Core-Switch(config)#**interface vlan 20**
- Core-Switch(config-if)#**description "High School VLAN"**
- Core-Switch(config-if)#**ip address 10.20.0.1 255.255.255.0**
- Core-Switch(config-if)#**exit**
- Core-Switch(config)#**exit**
- Core-Switch#**write memory**  
Building configuration...  
Compressed configuration from 7383 bytes to 3601 bytes[OK]  
[OK]
- Core-Switch#

Repeat these steps for each of the VLAN's you wish to create (**VLAN 30 –Elementary, VLAN 60 – Middle School**). Once you have these steps down you can easily add new VLAN's as needed. We will be adding VLAN's for Wireless and VoIP later in the course (**VLAN 40 – Student Wireless, VLAN 50 – Faculty Wireless, and VLAN 10 –VoIP**).

VLANs have now been created on the core device, but no port assignments have been made. Configure Core-Switch and make the necessary port assignments.

### CONFIGURE VLAN PORT ASSIGNMENTS ON CORE-SWITCH (VLAN 20 – HIGH SCHOOL VLAN)

1. Examine the Core-Switch configuration to learn the VLAN IDs that correspond to each building's VLAN using the command **show vlan**

```
Core-Switch#
Core-Switch#sh vlan
```

VLAN Name	Status	Ports
1 default	active	Gig1/0/2, Gig1/0/3, Gig1/0/4, Gig1/0/5 Gig1/0/6, Gig1/0/7, Gig1/0/8, Gig1/0/9 Gig1/0/10, Gig1/0/11, Gig1/0/12, Gig1/0/13 Gig1/0/14, Gig1/0/15, Gig1/0/16, Gig1/0/17 Gig1/0/18, Gig1/0/19, Gig1/0/20, Gig1/0/21 Gig1/0/22, Gig1/0/23, Gig1/0/24, Gig1/1/1 Gig1/1/2, Gig1/1/3, Gig1/1/4
20 High School VLAN	active	
30 Elem VLAN	active	
60 Middle School VLAN	active	
201 Internet-Uplink	active	Gig1/0/1
1002 fddi-default	active	
1003 token-ring-default	active	
1004 fddinet-default	active	
1005 trnet-default	active	

VLAN	Type	SAID	MTU	Parent	RingNo	BridgeNo	Stp	BrdgMode	Trans1	Trans2
1	enet	100001	1500	-	-	-	-	-	0	0
20	enet	100020	1500	-	-	-	-	-	0	0
30	enet	100030	1500	-	-	-	-	-	0	0
60	enet	100060	1500	-	-	-	-	-	0	0
201	enet	100201	1500	-	-	-	-	-	0	0
1002	fddi	101002	1500	-	-	-	-	-	0	0
1003	tr	101003	1500	-	-	-	-	-	0	0
1004	fdnet	101004	1500	-	-	-	ieee	-	0	0
1005	trnet	101005	1500	-	-	-	ibm	-	0	0

VLAN	Type	SAID	MTU	Parent	RingNo	BridgeNo	Stp	BrdgMode	Trans1	Trans2
1	enet	100001	1500	-	-	-	-	-	0	0
20	enet	100020	1500	-	-	-	-	-	0	0
30	enet	100030	1500	-	-	-	-	-	0	0
60	enet	100060	1500	-	-	-	-	-	0	0
201	enet	100201	1500	-	-	-	-	-	0	0
1002	fddi	101002	1500	-	-	-	-	-	0	0
1003	tr	101003	1500	-	-	-	-	-	0	0
1004	fdnet	101004	1500	-	-	-	ieee	-	0	0
1005	trnet	101005	1500	-	-	-	ibm	-	0	0

Remote SPAN VLANs

Primary	Secondary	Type	Ports

Core-Switch#

2. Locate the Core-Switch Downlink ports with appropriate VLAN. Use the **show cdp neighbors** command to find the correct Downlink ports.

```
Core-Switch#sh cdp neighbors
Capability Codes: R - Router, T - Trans Bridge, B - Source Route Bridge
                  S - Switch, H - Host, I - IGMP, r - Repeater, P - Phone
Device ID        Local Intrfce  Holdtme    Capability   Platform    Port ID
MS-Switch        Gig 1/1/2      126        R             3650         Gig 1/1/1
HS-Switch        Gig 1/1/1      126        R             3650         Gig 1/1/1
Elem-Switch      Gig 1/1/3      126        R             3650         Gig 1/1/1
Internet-Gateway
                  Gig 1/0/1      126        R             C1841        Fas 0/0
Core-Switch#
```

3. Configure Core-Switch Downlink ports as Trunk (Tagged) ports (repeat commands for each VLAN and Downlink Interface. You have examples for VLAN 20 and VLAN 60).

#### **(Cisco Layer3 Switch Commands)**

➤ Core-Switch#

➤ Core-Switch#conf t

Enter configuration commands, one per line. End with CNTL/Z.

- Core-Switch(config)#
- Core-Switch(config)#**interface gigabitEthernet 1/1/1**
- Core-Switch(config-if)#**description "High School VLAN"**
- Core-Switch(config-if)#**switchport trunk encapsulation dot1q**
- Core-Switch(config-if)#**switchport mode trunk**
- Core-Switch(config-if)#**switchport nonegotiate**
- Core-Switch(config-if)#**switchport trunk allowed vlan 20**
- Core-Switch(config-if)#**exit**
- Core-Switch(config)#**exit**
- Core-Switch#**write memory**
  
- Core-Switch#**conf t**  
Enter configuration commands, one per line. End with CNTL/Z.
- Core-Switch(config)#
- Core-Switch(config)#**interface gigabitEthernet 1/1/2**
- Core-Switch(config-if)#**description "Middle School VLAN"**
- Core-Switch(config-if)#**switchport trunk encapsulation dot1q**
- Core-Switch(config-if)#**switchport mode trunk**
- Core-Switch(config-if)#**switchport nonegotiate**
- Core-Switch(config-if)#**switchport trunk allowed vlan 60**
- Core-Switch(config-if)#**exit**
- Core-Switch(config)#**exit**
- Core-Switch#**write memory**
  
- (HP Commands)**
- Core-Switch#**conf t**
- Core-Switch(config)#**vlan 20**
- Core-Switch(config-vlan)#**tagged 1**
- Core-Switch(config-vlan)#**exit**
- Core-Switch(config)#**exit**
- Core-Switch#
  
- Core-Switch#**conf t**
- Core-Switch(config)#**vlan 60**
- Core-Switch(config-vlan)#**tagged 2**
- Core-Switch(config-vlan)#**exit**
- Core-Switch(config)#**exit**
- Core-Switch#

Now that the Core-Switch is configured, we need to configure the Distribution Layer Switches. We will need to add the VLAN to the switch.

---

## CONFIGURE THE BUILDING'S DISTRIBUTION-LAYER SWITCH

1. Create appropriate VLAN.

**(Cisco Command and HP Command)**

- HS-Switch>
- HS-Switch>en
- Password:
- HS-Switch#
- HS-Switch#**confi t**  
Enter configuration commands, one per line. End with CNTL/Z.
- HS-Switch(config)#**vlan 20**
- HS-Switch(config-vlan)#**name "High School VLAN"**
- HS-Switch(config-vlan)#**exit**
- HS-Switch(config)#**exit**
- HS-Switch#
  
- MS-Switch#**confi t**  
Enter configuration commands, one per line. End with CNTL/Z.
- MS-Switch(config)#**vlan 60**
- MS-Switch(config-vlan)#**name "Middle School VLAN"**
- MS-Switch(config-vlan)#**exit**
- MS-Switch(config)#**exit**
- MS-Switch#
  
- Elem-Switch#**confi t**  
Enter configuration commands, one per line. End with CNTL/Z.
- Elem-Switch(config)#**vlan 30**
- Elem-Switch(config-vlan)#**name "Elementary VLAN"**
- Elem-Switch(config-vlan)#**exit**
- Elem-Switch(config)#**exit**
- Elem-Switch#

2. Configure the uplink port of the distribution-layer switch as a 'tagged' port (ie. 'trunk')

**(Cisco Commands)**

- HS-Switch#
- HS-Switch#**config t**  
Enter configuration commands, one per line. End with CNTL/Z.
- HS-Switch(config)#**interface gigabitEthernet 1/1/1**  
\*\*\*"switchport trunk encapsulation dot1q" only required on Layer3 Cisco Switches\*\*\*
- HS-Switch(config-if)#**switchport trunk encapsulation dot1q**
- HS-Switch(config-if)#**switchport mode trunk**
- HS-Switch(config-if)#**switchport nonegotiate**
- HS-Switch(config-if)#**switchport trunk allowed vlan 20**
- HS-Switch(config-if)#**exit**

**(HP Commands)**

- Core-Switch#**conf t**
  - Core-Switch(config)#**vlan 20**
  - Core-Switch(config-vlan)#**tagged 1**
  - Core-Switch(config-vlan)#**exit**
  - Core-Switch(config)#**exit**
  - Core-Switch#
3. Configure the port that is connected to the PC/Printers as an 'untagged' (ie. 'access') port for the VLAN.

**(Cisco Commands)**

- HS-Switch>**en**
- HS-Switch#**conf t**
- Enter configuration commands, one per line. End with CNTL/Z.
- HS-Switch(config)#**interface range gigabitEthernet 1/0/1 - gigabitEthernet 1/0/10**
- HS-Switch(config-if-range)#**switchport mode access**
- HS-Switch(config-if-range)#**switchport access vlan 20**
- HS-Switch(config-if-range)#**exit**
- HS-Switch(config)#**exit**
- HS-Switch#

**(HP Commands)**

- HS-Switch#**conf t**
- HS-Switch(config)#**vlan 20**
- HS-Switch(config-vlan)#**untagged 10-18**
- HS-Switch(config-vlan)#**exit**
- HS-Switch(config)#**exit**
- HS-Switch#

## VERIFY ACCESS/TRUNK PORTS

Now we need to verify that all of the ports are correctly assigned to the appropriate VLAN as well as verify that all of the Uplink/Downlinks are tagged/trunked appropriately.

1. Verify the ports are 'tagged' port (ie. 'trunk') using **show interface trunk** command

```
Core-Switch#
Core-Switch#
Core-Switch#sh interface trunk
Port          Mode          Encapsulation  Status        Native vlan
Gig1/1/1      on            802.1q         trunking      1
Gig1/1/2      on            802.1q         trunking      1
Gig1/1/3      on            802.1q         trunking      1

Port          Vlans allowed on trunk
Gig1/1/1      20
Gig1/1/2      60
Gig1/1/3      30

Port          Vlans allowed and active in management domain
Gig1/1/1      20
Gig1/1/2      60
Gig1/1/3      30

Port          Vlans in spanning tree forwarding state and not pruned
Gig1/1/1      20
Gig1/1/2      60
Gig1/1/3      30
```

```
Core-Switch#
```

2. Verify the ports are untagged ports (ie "access") using **show vlan** command

```
HS-Switch#sh vlan
VLAN Name                Status    Ports
-----
1    default                  active    Gig1/0/11, Gig1/0/12, Gig1/0/13, Gig1/0/14
Gig1/0/15, Gig1/0/16, Gig1/0/17, Gig1/0/18
Gig1/0/19, Gig1/0/20, Gig1/0/21, Gig1/0/22
Gig1/0/23, Gig1/0/24, Gig1/1/2, Gig1/1/3
Gig1/1/4
20   High School VLAN         active    Gig1/0/1, Gig1/0/2, Gig1/0/3, Gig1/0/4
Gig1/0/5, Gig1/0/6, Gig1/0/7, Gig1/0/8
Gig1/0/9, Gig1/0/10
30   Elementary VLAN          active
60   Middle School VLAN       active
1002 fddi-default             active
1003 token-ring-default     active
1004 fddinet-default         active
1005 trnet-default          active

VLAN Type  SAID      MTU   Parent RingNo BridgeNo Stp  BrdgMode Transl Trans2
-----
1    enet     100001   1500  -     -     -     -     -     0     0
20   enet     100020   1500  -     -     -     -     -     0     0
30   enet     100030   1500  -     -     -     -     -     0     0
60   enet     100060   1500  -     -     -     -     -     0     0
1002 fddi     101002   1500  -     -     -     -     -     0     0
1003 tr      101003   1500  -     -     -     -     -     0     0
1004 fdnet   101004   1500  -     -     -     ieee  -     0     0
1005 trnet   101005   1500  -     -     -     ibm   -     0     0

VLAN Type  SAID      MTU   Parent RingNo BridgeNo Stp  BrdgMode Transl Trans2
-----

Remote SPAN VLANs
-----

Primary Secondary Type          Ports
-----

HS-Switch#
```

3. Save the configuration **write memory**
4. Repeat steps for each building/distribution-layer switch.

---

## DHCP CONFIGURATION

When configuring your DHCP server to serve a subdivided IP address space, you need to create a separate scope for each of the address blocks corresponding to your VLANs.

**Packet Tracer Instructions ...** use these instructions only for Packet Tracer Exercise. See Appendix A for Windows Server DHCP Configuration.

1. Click on the DHCP server to open its GUI. Click the services tab. Choose DHCP. You will see the scope for the Default VLAN has already been created.
2. Ensure the DHCP service is turned on.
3. Configure the IP Range to start at 10.20.0.100 and end at 10.20.0.254.
4. Choose the ADD button and create scope entries for each of the remaining three VLANs. The gateway for each scope will be the IP address of the virtual interface you created on the Core-Switch that corresponds to the related VLAN.

The DHCP server's IP address is statically assigned. It is on the Default VLAN, which is in a different subnet from the workstations.

Switches, by default, will not forward broadcast packets from one VLAN to another. Since DHCP client messages use the destination IP address of 255.255.255.255 (all Nets Broadcast), DHCP clients will not be able to send requests to a DHCP server on a different subnet unless the switch is configured to forward the request.

To forward the BootP/DHCP request from the client to the DHCP server, the ip helper-address interface command is used.

### **Configure BootP/DHCP (Cisco)**

- Core-Switch>
- Core-Switch>**en**
- Password:
- Core-Switch#
- Core-Switch#**config t**  
Enter configuration commands, one per line. End with CNTL/Z.
- Core-Switch(config)#**interface vlan 20**
- Core-Switch(config-if)#**ip helper-address 10.0.0.5**
- Core-Switch(config-if)#**exit**
- Core-Switch(config)#**exit**
- Core-Switch#**write memory**  
Building configuration...  
Compressed configuration from 7383 bytes to 3601 bytes[OK]
- Core-Switch#

This will need to be configured for each of the virtual interfaces you created on the Core-Switch that corresponds to the related VLAN.

## Verification

Once you have completed the steps above, use the following procedure to verify the VLAN configuration.

1. Open HS-PC0 and go to the desktop/IP Configuration screen.
2. Select the DHCP option.
3. If the IP address and gateway populate with the correct addresses for the High School the configuration was successful.
4. Repeat the actions for MS-PC1 and Elem-PC2.
  - a. If any of the PC's fail to get the correct DHCP address and gateway, begin troubleshooting on the appropriate distribution-layer switch.
5. Open a command prompt on HS-PC0.
6. Ping the following IP addresses
7. 10.0.0.1
8. 10.20.0.1
9. 192.168.100.2
10. Ping MS-PC1 from Elem-PC2

## Troubleshooting

If a workstation fails to obtain a DHCP address:

1. Check the connected port on the distribution-layer switch to ensure that it has been configured as an 'untagged' (i.e. 'access') port, and that it has been assigned to the proper VLAN.
2. If the port is correctly configured and assigned, check the uplink port of the distribution-layer switch to confirm that it is configured as a 'tagged' (i.e. 'trunk' port)
3. If the uplink port of the distribution-layer switch is correctly configured, check the connected port on the Core-Switch to ensure that it is configured as a 'tagged' (i.e. 'trunk') port.
4. If the Core-Switch port is correctly configured, check the virtual interface corresponding to the proper VLAN to ensure that it has the proper ip-helper address.
5. If the ip-helper address is correct and present, open the DHCP configuration page on the DHCP server and ensure that the relevant scope is configured with the correct range.



If a workstation obtains an incorrect IP address:

1. Verify that the proper VLAN is configured on the distribution-layer switch.
2. Verify that the connected port on the distribution-layer switch is assigned to the proper VLAN.
3. Verify that the virtual interface on the Core-Switch has the proper gateway IP address for the VLAN configured on the distribution-layer switch to which the PC is connected.

If a workstation obtains an incorrect gateway address:

1. Verify that the appropriate gateway address is entered for the DHCP scope that matches the VLAN to which the PC is connected.

## ADDING NEW VLAN TO EXISTING CISCO VLAN DATABASE (NEW WIFI VLAN 40 AND 50)

Now we are going to look at the steps needed to add new VLAN to an already existing configured VLAN switch. Thus far we have seen setting up a single VLAN Trunk/Tagged port. You will find the need to add new VLANs to your network as your network grows and as technology changes. The process for HP switches will not look any different than what was seen when adding the Trunk/Tagged ports earlier. Because the process for the Cisco includes different commands, we need to look at the process.

---

### CREATE NEW VLAN CORE-SWITCH (STUDENT WIFI VLAN 40)

- Core-Switch>
- Core-Switch>**en**
- Password:
- Core-Switch#
- Core-Switch#**configure t**  
Enter configuration commands, one per line. End with CNTL/Z.
- Core-Switch(config)#**vlan 40**
- Core-Switch(config-vlan)#**name "Student Wifi VLAN"**
- Core-Switch(config-vlan)#**exit**
- Core-Switch(config)#**interface vlan 40**
- Core-Switch(config-if)#**description "Student Wifi VLAN"**
- Core-Switch(config-if)#**ip address 10.40.0.1 255.255.255.0**
- Core-Switch(config-if)#**exit**
- Core-Switch(config)#**exit**
- Core-Switch#**write memory**  
Building configuration...  
Compressed configuration from 7383 bytes to 3601 bytes[OK]  
[OK]
- Core-Switch#

---

### CONFIGURE BOOTP/DHCP RELAY CORE-SWITCH (IP HELPER-ADDRESS)

- Core-Switch>
- Core-Switch>**en**
- Password:
- Core-Switch#
- Core-Switch#**config t**  
Enter configuration commands, one per line. End with CNTL/Z.
- Core-Switch(config)#**interface vlan 40**
- Core-Switch(config-if)#**ip helper-address 10.0.0.5**
- Core-Switch(config-if)#**exit**
- Core-Switch(config)#**exit**
- Core-Switch#**write memory**  
Building configuration...  
Compressed configuration from 7383 bytes to 3601 bytes[OK]

---

#### ADD VLAN TO TRUNK PORT CORE-SWITCH

When you add the VLAN to the trunk port you have to make sure you use the command **switchport trunk allowed add vlan**. If you leave the word **ADD** off of the command you will overwrite the allowed VLAN. With the **ADD** command you are adding new allowed VLAN to the port. You can also use the command **switchport trunk allowed vlan 20,40,50**. The key to this command is to redo the entire command or use the **ADD** command to add extra VLANs.

- Core-Switch#
- Core-Switch#**conf t**  
Enter configuration commands, one per line. End with CNTL/Z.
- Core-Switch(config)#
- Core-Switch(config)#**interface gigabitEthernet 1/1/1**
- Core-Switch(config-if)#**switchport trunk allowed add vlan 40**  
or you can use the following **#switchport trunk allowed vlan 20,40**
- Core-Switch(config-if)#**exit**
- Core-Switch(config)#**exit**
- Core-Switch#**write memory**

---

#### CREATE VLAN ON DISTRIBUTION SWITCH (HS-SWITCH)

- HS-Switch#**confi t**  
Enter configuration commands, one per line. End with CNTL/Z.
- HS-Switch(config)#**vlan 20**
- HS-Switch(config-vlan)#**name "High School VLAN"**
- HS-Switch(config-vlan)#**exit**
- HS-Switch(config)#**exit**
- HS-Switch#

---

#### ADD VLAN TO TRUNK PORT DISTRIBUTION SWITCH (HS-SWITCH)

- HS-Switch#**config t**
- Enter configuration commands, one per line. End with CNTL/Z.
- HS-Switch(config)#**interface gigabitEthernet 1/1/1**
- HS-Switch(config-if)#**switchport trunk allowed vlan add 40**  
or you can use the following **#switchport trunk allowed vlan 20,40**
- HS-Switch(config-if)#**exit**

#### WHAT TO DO WITH WIRELESS ACCESS POINTS (TRUNK OR ACCESS?)

When you are adding new VLANs to your network that are for Wireless Access you will need to treat all of your Access Points as you would a Downlink/Uplink Port.

You will have all of the ports that have an AP plugged in set as a Trunk/Tagged port for every VLAN that is included for the Wireless system (Student Wireless VLAN 40, Faculty Wireless VLAN 50, Guest Wireless VLAN 70, ect).

---

#### CHANGE WIRELESS AP PORT TO TRUNK PORT AND ASSIGN VLANS (HS-SWITCH)

- HS-Switch#**conf t**  
Enter configuration commands, one per line. End with CNTL/Z.
- HS-Switch(config)#
- HS-Switch(config)#**interface gigabitEthernet 1/0/16**
- HS-Switch(config-if)#**description "Wireless AP"**
- HS-Switch(config-if)#**switchport trunk encapsulation dot1q**
- HS-Switch(config-if)#**switchport mode trunk**
- HS-Switch(config-if)#**switchport nonegotiate**
- HS-Switch(config-if)#**switchport trunk allowed vlan 20,40,50,70**
- HS-Switch(config-if)#**exit**
- HS-Switch(config)#**exit**
- HS-Switch#**write memory**

Depending on what VLAN the management IP of the Access Point belongs to might add to some more configuration considerations of the Access Point port. If the AP's belong to the "Default VLAN (10.0.0.0/24)" then you will not have to make changes to the port VLAN membership (Untagged- Native VLAN). All ports on your switches are automatically a member of the "Default VLAN".

If you Wireless AP's have a management IP of 192.168.99.0/24 (VLAN 99), you can change the VLAN port membership using the following command. This will allow for the VLAN 99 to pass traffic on that port without requiring the VID Tag in the packet.

---

#### CHANGING WIRELESS AP'S MANAGEMENT VLAN MEMBERSHIP (NATIVE VLAN)

- HS-Switch#**conf t**  
Enter configuration commands, one per line. End with CNTL/Z.
- HS-Switch(config)#**interface gigabitEthernet 1/0/16**
- HS-Switch(config-if)#**switchport trunk native vlan 99**
- HS-Switch(config-if)#**exit**

## ACCESS CONTROL

In the current configuration, any device on any VLAN can discover and/or communicate with any device on any other VLAN. This is not always ideal for the management and security of your network.

Inter-VLAN communication happens at the layer 3 switch or router. That makes this device the most natural location to apply any restrictions on inter-VLAN communication.

Here we can configure rules regarding other VLANS with which a device on a specific VLAN can communicate.

This is accomplished through the configuration of access control lists(ACLs). Access control lists are groups of statements in the configuration specifying actions to be applied to data coming into or leaving an interface.

---

### ACCESS CONTROL LIST MECHANICS

Traffic that comes into or out of an interface to which an Access Control List is applied is compared to entries in the applied list based on the order that the entries occur in the configuration.

New statements are added at the end of the Access Control List.

The device continues to look until it has a match. If no matches are found by the time the end of the list is reached, the traffic is denied. Due to this, you must have at least one permit statement in an Access Control List or all traffic is blocked.

---

### STANDARD ACLS

Standard ACLs are the oldest type of ACL. Standard ACLs control traffic by the comparison of the source address of the IP packets to the addresses configured in the ACL.

This is the command syntax format of a standard ACL.

**access-list** access-list-number {**permit**|**deny**}

{**host**|source source-wildcard |**any**}

The access-list-number can be anything from 1 to 99. Later in their development, standard ACLs begin to use additional numbers (1300 to 1999). These additional numbers are referred to as Expanded IP ACLs. Even later versions added the ability to use list name in standard ACLs.

The source address can be an individual IP address or a range of IP addresses. If you are specifying a range of IP address, you would enter the network address of the range i.e. 10.1.100.0, for the range 10.1.100.1 – 10.1.100.253 (mask 255.255.255.0) and an inverse mask of 0.0.0.255, or 10.1.32.0 for the range 10.1.32.1-10.1.35.253 (mask 255.255.224.0) inverse mask 0.0.31.255.

ACLs use what is referred to as an 'inverse' or 'wildcard mask for IP addresses. The values in the mask work in a reverse or 'inverse' manner to those mask values used in regular IP address configuration.

For example, a 0.0.0.0 mask associated with regular IP address configuration would include all possible IP addresses. An inverse mask of 0.0.0.0 would limit the configuration to a single IP address.

Standard Mask	Value	Inverse Mask Value
0		255
254		1
252		3
248		7
240		15
224		31
192		63
128		127

A source/source-wildcard setting of 0.0.0.0/255.255.255.255 can be specified as **any**. The wildcard can be omitted if it is all zeros. Therefore, host 10.1.1.2 0.0.0.0 is the same as host 10.1.1.2.

After the ACL is defined, it must be applied to the interface (inbound or outbound. The direction must be specified.

**interface** <interface>

**ip access-group** number {in | out}

This is an example of the use of a standard ACL in order to block all traffic except that from source 10.1.1.x.

```
interface Ethernet0/0
ip address 10.1.1.1 255.255.255.0
ip access-group 1 in
access-list 1 permit 10.1.1.0 0.0.0.255
```

The following Access list has been configured on the core switch.

```
access-list 101 remark allow DHCP requests*****
access-list 101 permit udp any eq 68 host 255.255.255.255 eq 67
access-list 101 remark ***allow DNS requests***
access-list 101 permit udp any host 10.10.100.12 eq domain
access-list 101 remark *** allow internet access ***
access-list 101 permit ip any host 10.0.0.5
access-list 101 remark *** allow file server access ***
access-list 101 permit ip any host 10.10.100.11
```

Although the access list is on the core switch, it is not currently controlling any traffic. It will have to be applied to an interface before it becomes active.

---

## APPLYING ACLS

You can define ACLs without applying them. But, the ACLs have no effect until they are applied to the interface.

When you try to block traffic from source to destination, you can apply an inbound ACL to an inbound interface instead of an outbound list to an outbound interface.

An access-list has a **deny ip any any** implicitly at the end of any access-list. If traffic is related to a request is not explicitly permitted, the traffic is dropped because when you look at the request in IP, the source address is not specifically permitted.

So, you should permit this kind of traffic in your access-list or the traffic is dropped due to implicit deny at the end of the statement.

**Out**—Traffic that has already been through the router or switch and leaves the interface. The source is where it has been, on the other side of the router or switch, and the destination is where it goes.

**In**—Traffic that arrives on the interface and then goes through the router or switch. The source is where it has been and the destination is where it goes, on the other side of the router or switch.

**Inbound** —If the access list is inbound, when the router or switch receives a packet, the Cisco IOS software checks the criteria statements of the access list for a match. If the packet is permitted, the software continues to process the packet. If the packet is denied, the software discards the packet.

**Outbound**—If the access list is outbound, after the software receives and routes a packet to the outbound interface, the software checks the criteria statements of the access list for a match. If the packet is permitted, the software transmits the packet. If the packet is denied, the software discards the packet.

The in ACL has a source on a segment of the interface to which it is applied and a destination off of any other interface. The out ACL has a source on a segment of any interface other than the interface to which it is applied and a destination off of the interface to which it is applied.

We now need to decide which interface Access List 101 needs to be applied to and whether it will be better to inspect the data as it arrives or leaves the interface.

All the rules in access-list 101 function to permit communication from all our workstations to specific applications or devices. Our objective would be to maximize our rules effectiveness while minimizing the additional processing on the core switch's CPU.

There are three interfaces physical that represent the source of all workstation related traffic to the core switch, Fa0/22, Fa0/23 and Fa0/24. These are the 'tagged' ports, they carry traffic from more than one VLAN. Attempting to apply the access-list here would block VLAN 1 traffic, as well as the VLAN we have assigned to the building. This is not desirable. However, if we apply the access-list to the virtual interface associated with each individual VLAN, we can control just the workstation traffic without impacting the management VLAN. These are the



Vlan110, Vlan120 and Vlan130 interfaces. To minimize CPU processing we would like to apply the rules to inbound traffic. This allows the core switch to drop packets that it is configured to deny before going through calculating their routing information.

Apply access list 101 to the appropriate interfaces using the configuration commands, then save the configuration.

The next step is to power off one of the workstations from the GUI. Power it back on and determine if it receives a DHCP address from the DHCP server.

Ping the following IP addresses

10.0.0.5

10.10.100.12

10.10.100.11

Ping PC2 from PC0

What were your results, and how did they differ from before the access list was applied.

Since Spanning Tree and VLAN's interact based on ports, it could be helpful to know what Spanning Tree will do behind the scenes in the event of the failure of a link or switch.

---

### WHAT IS SPANNING TREE PROTOCOL AND WHAT DOES IT DO?

Spanning Tree Protocol is a feature of layer 2 and 3 network switches that is intended to prevent network loops due to redundant physical connections. This feature can be enable or disabled on individual switches.

Often in a network, we would like to build in the capability to maintain connectivity between our switches in the event that a specific link should fail. To do this, additional interconnections between network devices are built into the LAN infrastructure. Two common methods of achieving redundant connections are to connect two devices with two separate physical connections, or to interconnect devices at the distribution and/or access layers, in addition to their uplinks to the core layer.

While this redundant connectivity allows the network to be more resilient to failures of specific devices and/or cables, unless properly managed, it can bring a network to its knees.

Packets can loop endlessly when there is more than one path to take across the network. This behavior can clog your network with packets until nothing else moves. Broadcast traffic is particularly problematic in this type of situation.

In order to counter this, Spanning Tree Protocol was developed and implemented in network devices. Cisco and Juniper among others, developed proprietary versions of Spanning Tree Protocol before the IEEE standards committee produced the current industry standard.

The current the current industry standard for Spanning Tree Protocol with regard to networks with VLAN configurations is Multiple Spanning Tree Protocol. The specifications for this standard are in IEEE 802.1Q though Cisco still recommends its own Rapid Spanning Tree Protocol on its devices.

Implementing VLANs on a network that is running a version of Spanning Tree Protocol, (and vice versa), can change the way that both the VLANs and the Spanning Tree work. These two configurations both are tied to ports on the device and can alter the way that traffic flows in unexpected ways.

When the Spanning Tree feature is turned on, the device will send specially formatted packets about itself and its links to other devices on the network. These packets are called Bridge Protocol Data Units or BPDU.

Spanning Tree limits connectivity between devices to the least costly, active path from point A to point B. The cost associated with a specific interface is determined by the protocol based upon the speed of the connection. The slower the connection the higher the cost. Spanning Tree alters how ports on a device operate. Ports on a device are placed into one of three states, listening, forwarding or blocking.

In the listening state, the port forwards network traffic while analyzing BPDU packets from other devices on the network, in order to evaluate the state that each of its ports should be in.

In the forwarding state, the port forwards network traffic normally, while collecting and forwarding any BPDU packets it receives

In the blocking state, the port does not forward network traffic on the link connected to it, while still collecting and passing BPDU packets.

When Spanning Tree is initially enabled, all ports are in a listening state. Spanning Tree Protocol uses an algorithm to evaluate the information in the BPDU packets to decide which ports need to put in either the forwarding or blocking state to leave the single path from A to B.

This operation is referred to as 'Convergence'. The first step in the process is to determine which device on the network is going to be the starting point that network paths are measured from. This device is referred to as the Root Switch. Each device is assigned an 8 byte value called a Bridge ID. This value is composed of two parts. The first is Priority. This value has a default of 32,769. The Priority value can be set to any desired value when configuring Spanning Tree on a device. The second part of the Bridge ID is the MAC address of the device. The MAC address portion cannot be altered and insures that the Bridge ID of a device is unduplicated on the network.

Example 32769:0200:0001:0001 or 0:0200:0001:0001

Each device sends out a BPDU packet containing its Bridge ID. Every switch on the network compares the Bridge ID's it receives with each other and with its own Bridge ID to elect one switch as the root switch. The first judgment is made on the Priority value. The lowest Priority value will be selected as the root switch. If more than one Bridge ID has the same priority value, (such as the default value), then the MAC address portion is compared with the lowest value MAC address being elected.

Example:

22:0200:0001:0001 vs 32769:0100:cd01:00ba - 22:0200:0001:0001 gets elected.

32769:0200:0001:0001 vs 32769:0100:cd01:00ba -32769:0100:cd01:00ba gets elected.

Based on this information, all the switches will come to the same conclusion as to which device will be considered the 'root' switch. The importance of which switch is elected to be the root switch comes from the way that the next step in convergence is conducted.

Once the root switch is determined, all other devices identify which port connects back to the root switch that has the least associated cost. Port cost is an integer value assigned to each port on a device by the STP to choose which interfaces to add to the STP topology. Devices send the port cost associated with an interface connected to the link in regular BPDU packets called Hello packets. Each device calculates the cost of its interface(s) connected back to the root switch by adding the cost it receives in Hello packets through that port to the port cost assigned to that port, so that the cost of the path back to the root switch is the sum of all the port costs associated with the path. The port that the switch identifies as its lowest cost link back to the root switch is called its 'root port'.

The third and final step in Spanning Tree determines which of the two device ports that are connected to the link between switches will be placed in a forwarding state, and which will be placed in a blocking state. The port with the lowest cumulative cost back to the root switch will be placed in the forwarding state. The port on the other device attached to the link will be placed in the blocking state. If both links are of equal cost, the port on the device with the lowest Bridge ID will be selected as the designated port. The designated port will be put in the forwarding state, and the other port will be placed in the blocking state.

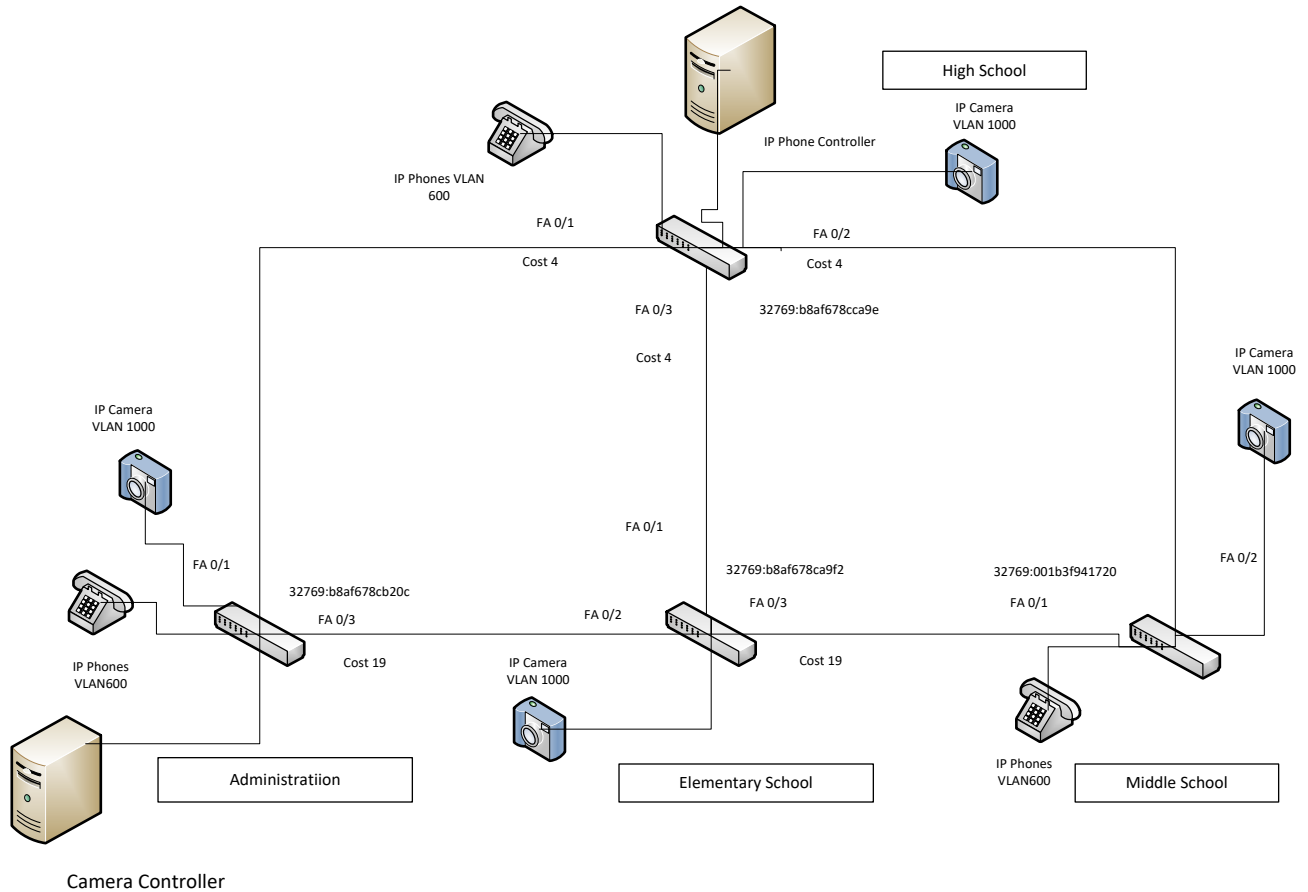
This means that traffic from the device that has the blocking port that would normally pass to the device at the other end will no longer flow. If you have workstations on the device with the blocking port that normally communicate with workstations, printers or other devices on the switch at the other end of the link, that communication will have to take a different path that may introduce more delay, differing VLAN configurations and other potential complications.

In summary, STP can break VLAN communications. It is important that when you plan deployment of either VLANs or Spanning Tree you take into consideration how they might affect each other.

In our network Spanning Tree would not be a factor in normal operations as we have no redundant connectivity. However, should someone accidentally or maliciously create a looped connection, enabling Spanning Tree could prevent network performance from being degraded.

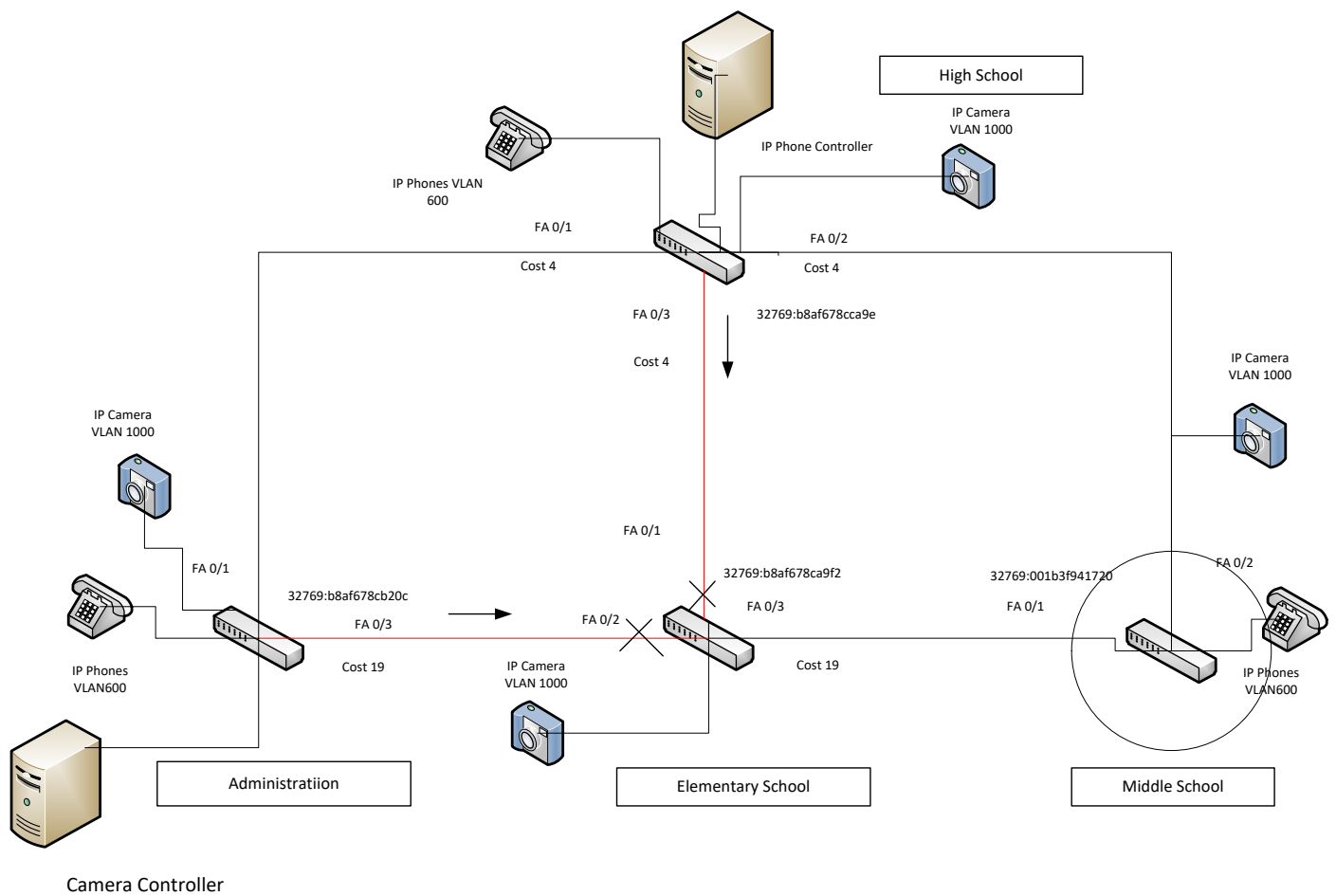
The following exercise illustrates a possible scenario.

SPANNING TREE EXERCISE:

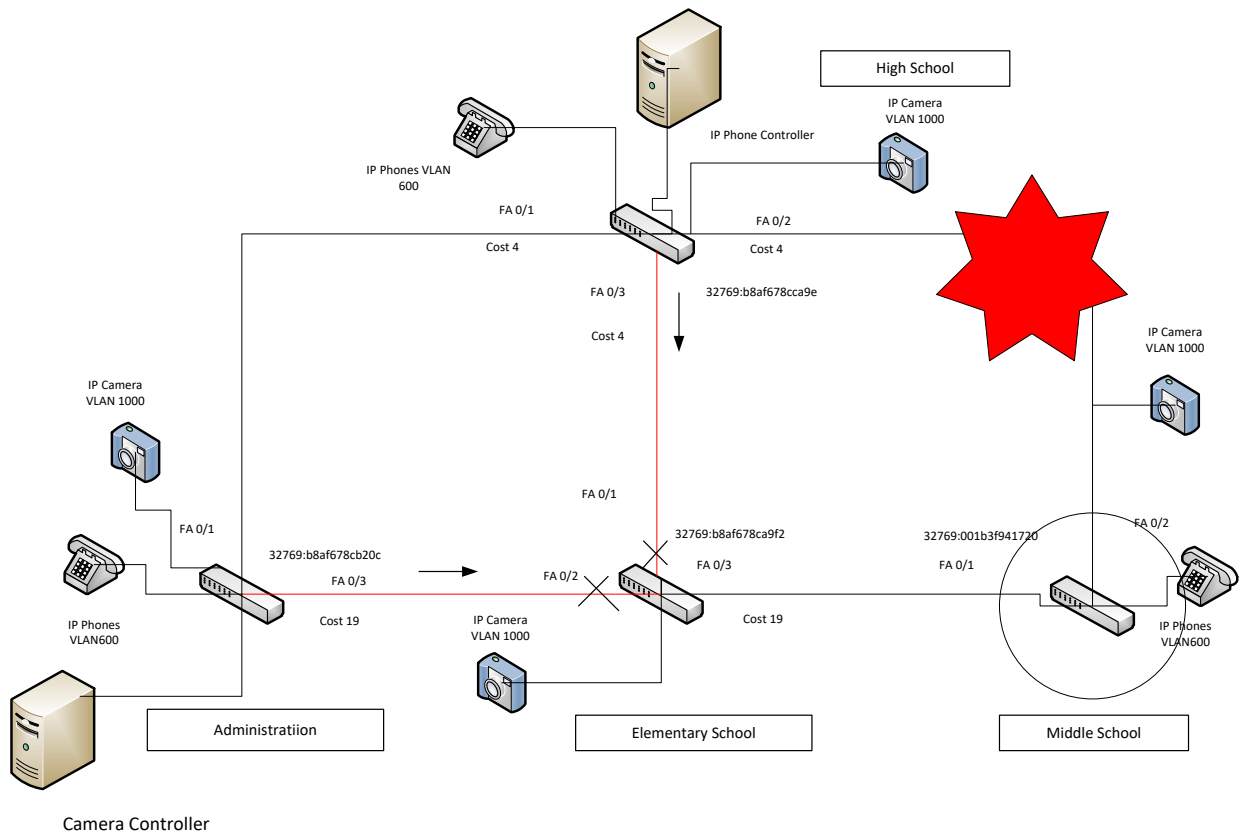


Given the information in the diagram, can we determine:

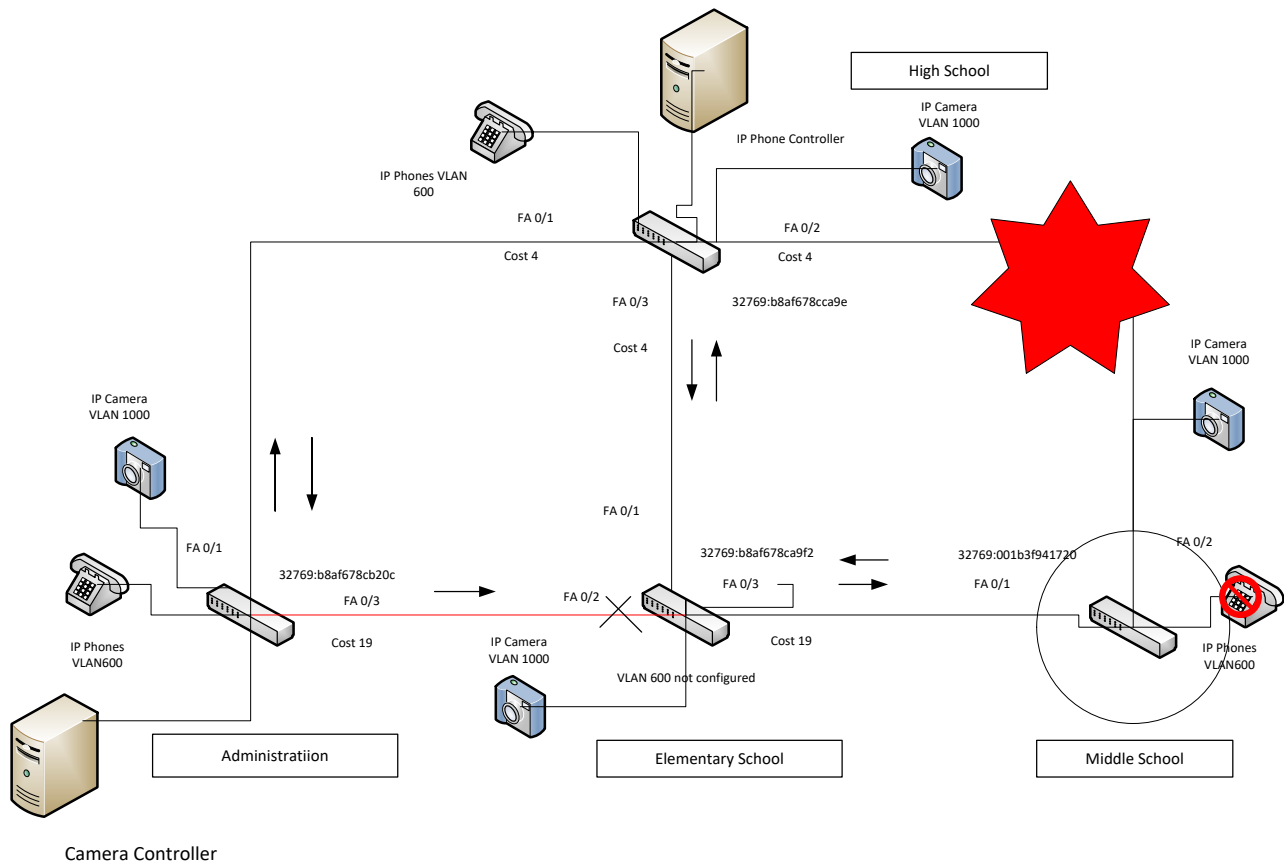
- 1 Which device will be elected as the 'root switch.'
- 2 What will the 'root port' be for each switch?
- 3 What will be the designated port and blocking port for the two segments that will be blocking?



IP phones are installed at the High School, Middle School and Admin. The Elementary does not use IP phones. The IP phone controller is located in the high school. In order to insure that the IP phone system runs smoothly, VLAN 600 has been configured on the LAN network segments with IP phones to segregate the voice traffic and provide separate routing. Security cameras are installed at all locations. The video recorder is located in the Administration building. Six months later, the TC decides to implement Spanning Tree Protocol. All is well, until the link between the High School and the Middle School is cut by workers digging a trench for new sewer lines.



Here is a diagram showing how the Spanning tree protocol reacts to the loss of the High School – Middle School fiber link.

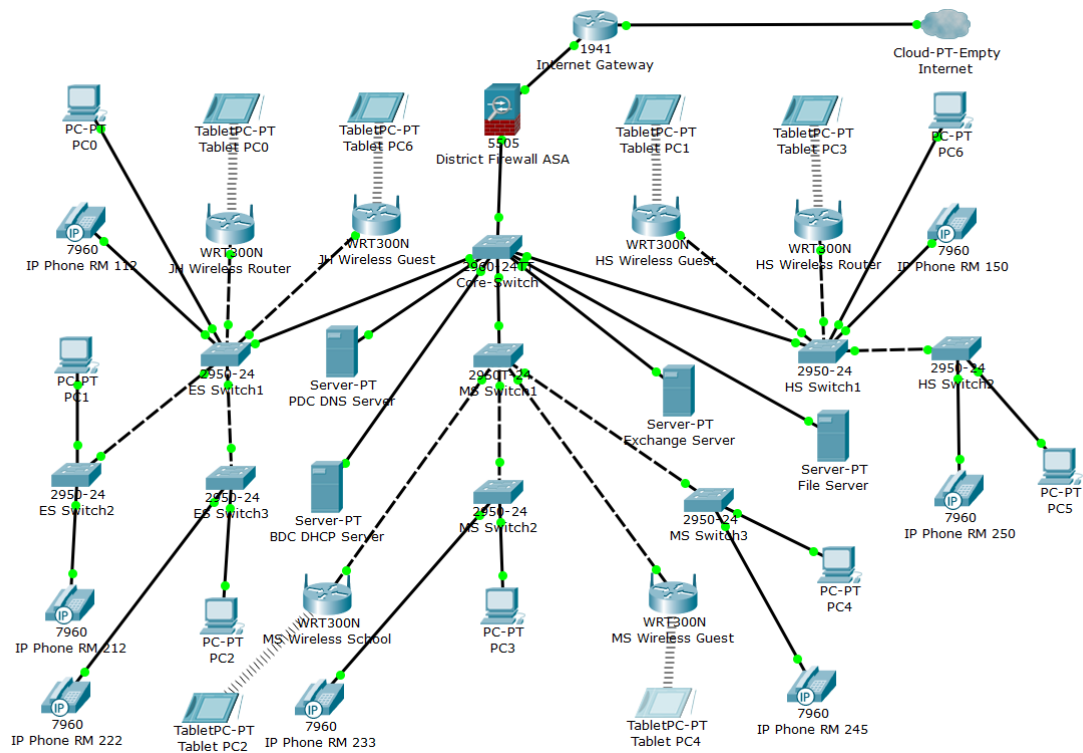


All the IP Phones at the Middle School are still down. The Security Cameras have come back up and are communicating with the Video recorder. What happened, and how do we fix it.

Because the VLAN for the IP phones was configured in advance of the implementation of Spanning Tree Protocol, it was not necessary at that time to configure VLAN 600 on the elementary school switch. When Spanning Tree Protocol was first configured, there was no immediate impact on IP phone performance. Without the low cost link between the High School and Middle School switches, Spanning Tree reconverged forcing all traffic through the Elementary School switch at some point. When the network topology changes, due to switch replacement, connectivity change or outage, the protocol adjusts without the need for any outside involvement. This is great for maintaining network resilience, but not so well for troubleshooting the network.



## Section IV: Planning Exercise



## Planning Exercise Criteria

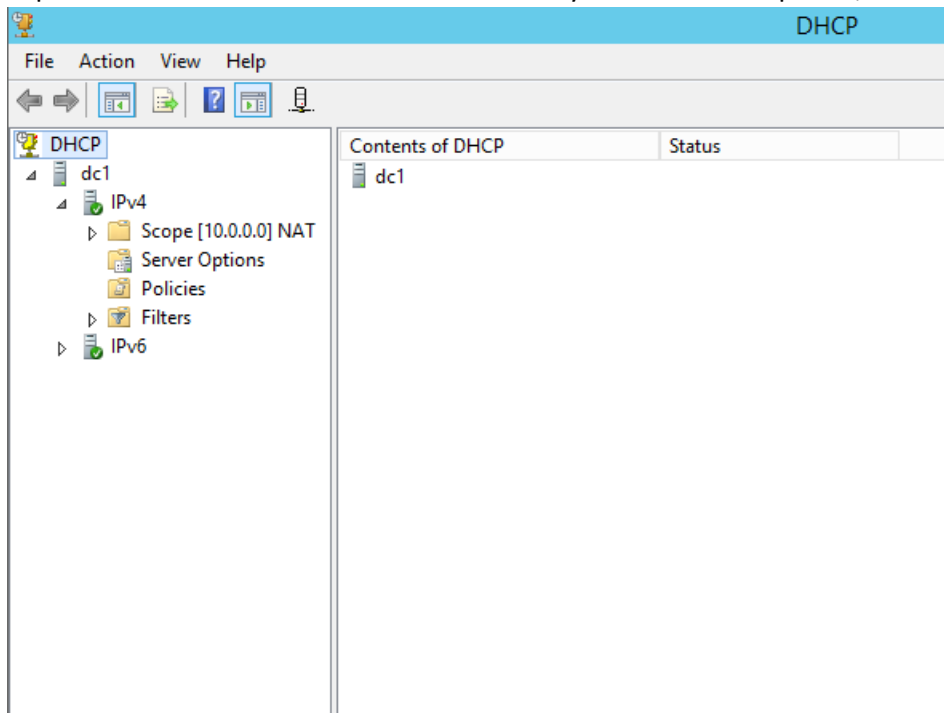
1. Provide for IP Phones
  - a. Choose location for phone controller on the network
  - b. Establish a VLAN to serve phones
2. Limit broadcast domains to < 600 devices
  - a. Divide IP address space into blocks of 512 addresses.

- b. Configure DHCP scopes to correspond to segmenting of IP address space
- 3. Establish a 'guest' wireless network that is internet only.
  - a. Create a VLAN for guest wireless access
  - b. Configure access list for guest wireless access to internet gateway only
  - c. Apply the access list to the appropriate network interface(s)
- 4. Provide the necessary structure to filter traffic based on device grouping
  - a. Establish VLANs based on connectivity type and location.
  - b. Assign IP address ranges to each established user type
- 5. Create an access list that limits inter-VLAN communication
  - a. Limit guest wireless to internet only
  - b. Allow DNS and DHCP access for all VLANs
  - c. Deny IP traffic other than IP phones between buildings

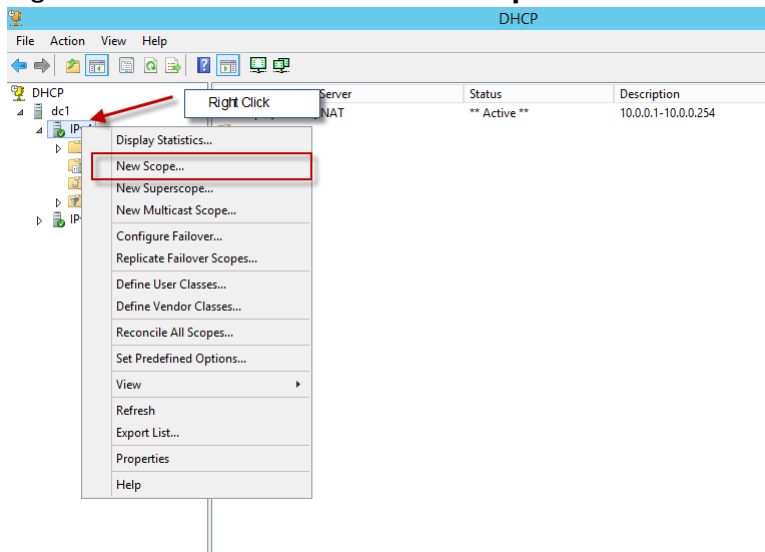
## Appendix A

### Configure Windows Server DHCP Scope

1. Expand the server node and **IPv4** node until you see Server Options, Policies.



2. Right click on **IPv4** and select **New Scope**.



3. On the **Scope Name** screen enter the Scope name and description you want to use for this scope e.g. HS-VLAN

The screenshot shows the 'New Scope Wizard' dialog box with the 'Scope Name' step selected. The title bar reads 'New Scope Wizard'. Below the title, the section is titled 'Scope Name' with a folder icon. The text says: 'You have to provide an identifying scope name. You also have the option of providing a description.' Below this, a larger text block says: 'Type a name and description for this scope. This information helps you quickly identify how the scope is to be used on your network.' There are two input fields: 'Name:' with the value 'HS-VLAN' and 'Description:' with the value '10.20.0.1-10.20.1.254'. At the bottom, there are three buttons: '< Back', 'Next >', and 'Cancel'.

4. On the **IP Address Range** screen type in the starting and ending IP address for this scope along with the subnet mask. This is the range of IP addresses this DHCP server will be issuing. Click **Next**.

The screenshot shows the 'New Scope Wizard' dialog box with the 'IP Address Range' step selected. The title bar reads 'New Scope Wizard'. Below the title, the section is titled 'IP Address Range' with a folder icon. The text says: 'You define the scope address range by identifying a set of consecutive IP addresses.' Below this, there are two sections. The first is 'Configuration settings for DHCP Server' with the instruction 'Enter the range of addresses that the scope distributes.' It contains two input fields: 'Start IP address:' with the value '10 . 20 . 0 . 100' and 'End IP address:' with the value '10 . 20 . 1 . 254'. The second section is 'Configuration settings that propagate to DHCP Client' with two input fields: 'Length:' with the value '23' and 'Subnet mask:' with the value '255 . 255 . 254 . 0'. At the bottom, there are three buttons: '< Back', 'Next >', and 'Cancel'.

***\*\*It is recommended to leave a few numbers at the start of the scope for static assignment e.g. if the IP range is 10.20.0.100 - 10.20.0.99 enter 10.20.0.100 for the Starting IP Address and 10.20.1.254 for the Ending IP***

**Address to leave 100 IP's at the beginning of your IP range for static assignment.**

5. On the **Exclusion** screen enter the IP addresses you want to be excluded from the DHCP range defined in the previous step and then click **Next**.

The screenshot shows the 'New Scope Wizard' window with the 'Add Exclusions and Delay' step selected. The title bar reads 'New Scope Wizard'. Below the title bar, the section is titled 'Add Exclusions and Delay' with a sub-header 'Exclusions are addresses or a range of addresses that are not distributed by the server. A delay is the time duration by which the server will delay the transmission of a DHCP OFFER message.' To the right of this text is a folder icon. The main area contains instructions: 'Type the IP address range that you want to exclude. If you want to exclude a single address, type an address in Start IP address only.' Below this are two input fields: 'Start IP address:' and 'End IP address:', each followed by a small 'Add' button. Underneath is a larger 'Excluded address range:' text box with a 'Remove' button to its right. At the bottom right of the main area is a 'Subnet delay in milli second:' label with a spin box containing the value '0'. At the very bottom of the window are three buttons: '< Back', 'Next >', and 'Cancel'.

6. On the **Lease time** screen take the default values unless required otherwise and Click **Next**.

The screenshot shows the 'New Scope Wizard' window with the 'Lease Duration' step selected. The title bar reads 'New Scope Wizard'. Below the title bar, the section is titled 'Lease Duration' with a sub-header 'The lease duration specifies how long a client can use an IP address from this scope.' To the right of this text is a folder icon. The main area contains explanatory text: 'Lease durations should typically be equal to the average time the computer is connected to the same physical network. For mobile networks that consist mainly of portable computers or dial-up clients, shorter lease durations can be useful. Likewise, for a stable network that consists mainly of desktop computers at fixed locations, longer lease durations are more appropriate.' Below this is the instruction: 'Set the duration for scope leases when distributed by this server.' Underneath is the label 'Limited to:' followed by three spin boxes: 'Days:' with the value '1', 'Hours:' with the value '0', and 'Minutes:' with the value '0'. At the very bottom of the window are three buttons: '< Back', 'Next >', and 'Cancel'.

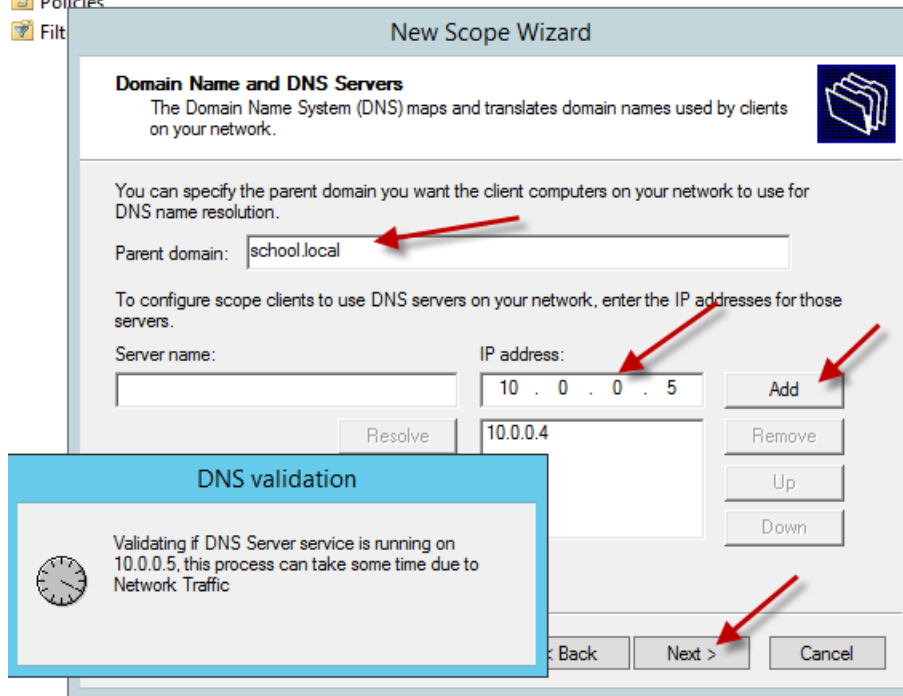
7. On the **Configure DHCP options** screen select **Yes, I want to configure these options now** and click **Next**.

The screenshot shows the 'New Scope Wizard' window with the 'Configure DHCP Options' step. The title bar reads 'New Scope Wizard'. Below the title bar, the section is titled 'Configure DHCP Options' with a folder icon. The text says: 'You have to configure the most common DHCP options before clients can use the scope.' Below this, it explains: 'When clients obtain an address, they are given DHCP options such as the IP addresses of routers (default gateways), DNS servers, and WINS settings for that scope. The settings you select here are for this scope and override settings configured in the Server Options folder for this server.' A question is asked: 'Do you want to configure the DHCP options for this scope now?'. There are two radio button options: 'Yes, I want to configure these options now' (which is selected) and 'No, I will configure these options later'. At the bottom, there are three buttons: '< Back', 'Next >', and 'Cancel'.

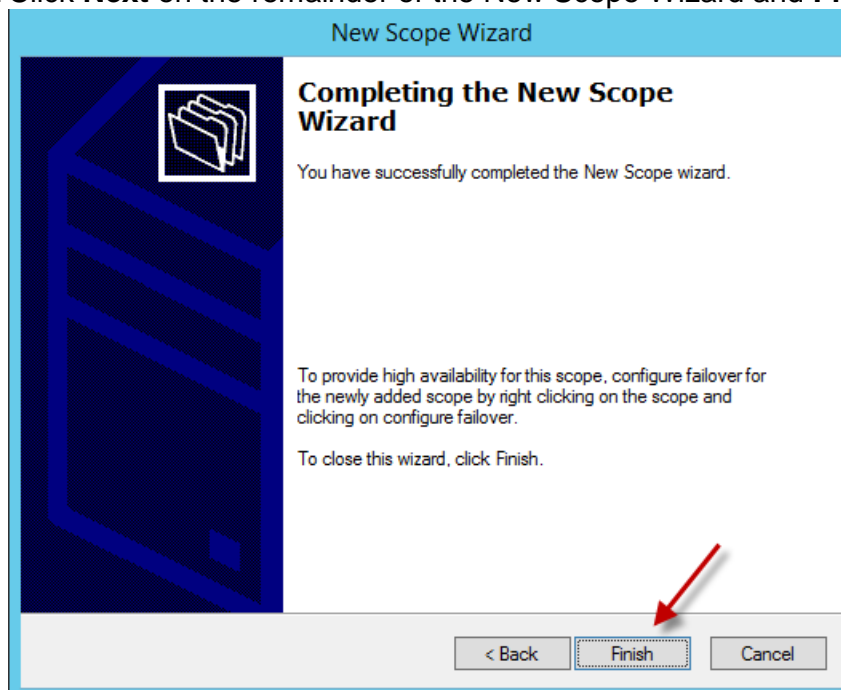
8. On the Router (Default Gateway) add your new VLAN gateway **10.20.0.1** and Click **Add** and click **Next**

The screenshot shows the 'New Scope Wizard' window with the 'Router (Default Gateway)' step. The title bar reads 'New Scope Wizard'. Below the title bar, the section is titled 'Router (Default Gateway)' with a folder icon. The text says: 'You can specify the routers, or default gateways, to be distributed by this scope.' Below this, it says: 'To add an IP address for a router used by clients, enter the address below.' There is an 'IP address:' label followed by a text input field containing '10 . 20 . 0 . 1'. To the right of the input field is an 'Add' button, which is highlighted with a red arrow. Below the input field is a list box containing '10.20.0.1'. To the right of the list box are four buttons: 'Remove', 'Up', and 'Down'. At the bottom, there are three buttons: '< Back', 'Next >', and 'Cancel'.

9. On the Domain Name and DNS Server options, enter your Domain Name **school.local** and local DNS servers **10.0.0.4** and **10.0.0.5**. Click Next



10. Click **Next** on the remainder of the New Scope Wizard and **Finish** at the end.



11. Repeat these steps for each of the new VLAN's you have created.

