Brendan McLaughlin

Robert Cannistra

Internetworking

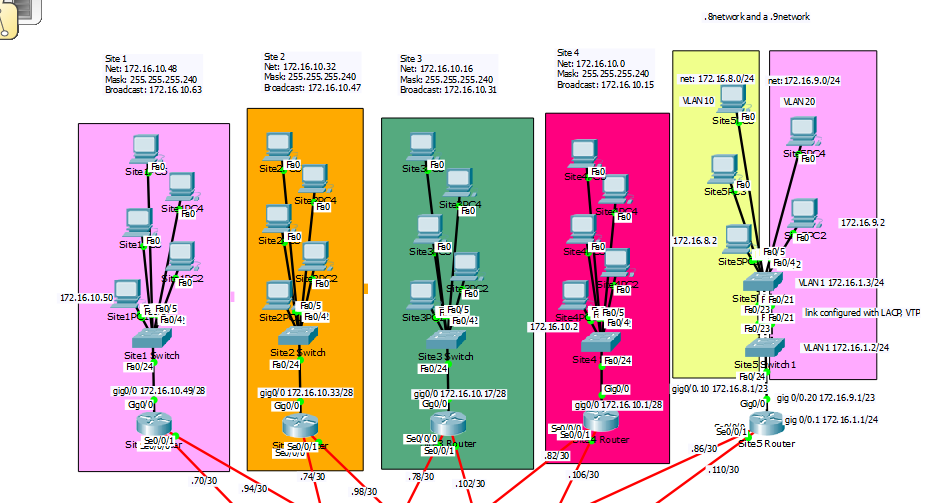
Spring 2016

Topology Design Project

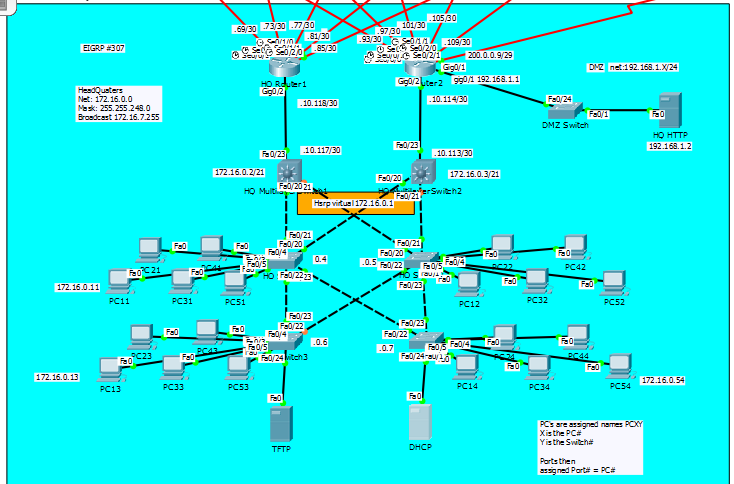
Unfortunately I was unable to fully complete my topology fully with DHCP in my HQ and a fully function DMZ ACL. I have completed a fully connected internal topology utilizing HSRP, EIGRP, static routing with route redistribution to my ISP, VLANS with VTP, LACP link aggregation and router on a stick, NAT and PPP. I was able to implement DHCP on my remote sites 1-4 but was receiving an APIPA error If I tried to implement it on my VLANS at site 5 or in my HQ where my HSRP on VLAN 1 reside. My DMZ extended ACL works however it unintentionally blocks my internal traffic attempting to reach my ISP which is clearly no useful in a real world network topology.

Additionally I did my network design all at once and was not documenting it throughout the process so some of my protocol examples are not from this topology.

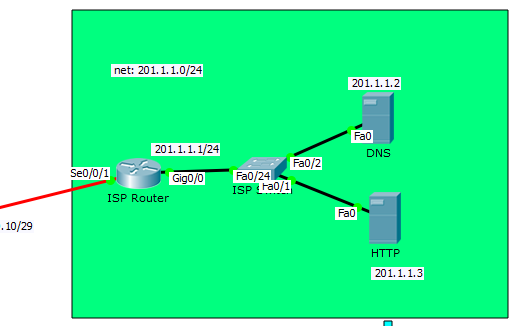
**Remote Sites**



**HQ**



**ISP hanging off of HQ\_Router 2**



**IP Addressing scheme**

**Headquarters**

Net: 172.16.0.0

Mask: 255.255.248.0

Broadcast: 172.16.7.255

Host Range: 172.16.0.1 – 172.16.7.254

\*Allows for 2046 possible hosts

**Site5 VLAN 10**

Net: 172.16.8.0

Mask: 255.255.255.0

Broadcast: 172.16.8.255

Host Range: 172.16.8.1 – 172.16.8.254

\*Allows for 254 possible hosts

**Site5 VLAN 20**

Net: 172.16.9.0

Mask: 255.255.255.0

Broadcast: 172.16.9.255

Host Range: 172.16.9.1 – 172.16.9.254

\*Allows for 254 possible hosts

**Site4**

Net: 172.16.10.0

Mask: 255.255.255.240

Broadcast: 172.16.10.15

Host Range: 172.16.10.1 – 172.16.10.14

\*Allows for 14 possible hosts

**Site3**

Net: 172.16.10.16

Mask: 255.255.255.240

Broadcast: 172.16.10.31

Host Range: 172.16.10.17 – 172.16.10.30

\*Allows for 14 possible hosts

**Site2**

Net: 172.16.10.32

Mask: 255.255.255.240

Broadcast: 172.16.10.147

Host Range: 172.16.10.33 – 172.16.10.46

\*Allows for 14 possible hosts

**Site1**

Net: 172.16.10.48

Mask: 255.255.255.240

Broadcast: 172.16.10.163

Host Range: 172.16.10.49 – 172.16.10.62

\*Allows for 14 possible hosts

**Router2 to ISP**

Net: 200.0.0.8

Mask: 255.255.255.248

Broadcast: 200.0.0.15

Host Range: 200.0.0.9 – 200.0.0.14

\*Allows for 6 possible host

**MultiLayer Switch 1 to Router 1**

Net: 172.16.10.116

Mask: 255.255.255.252

Broadcast: 172.16.10.119

Host Range: 172.16.10.117-172.16.10.118

\*Allows for two possible hosts

**MultiLayer Switch 2 to Router 2**

Net: 172.16.10.112

Mask: 255.255.255.252

Broadcast: 172.16.10.115

Host Range: 172.16.10.113-172.16.10.114

\*Allows for two possible hosts

**DMZ**

Net: 192.168.1.0

Mask: 255.255.255.0

Broadcast: 192.168.1.255

Host Range: 192.168.1.1 – 192.168.1.254

\*Allows for 254 Possible Hosts

**ISP**

Net: 201.1.1.0

Mask: 255.255.255.0

Broadcast: 201.1.1.255

Host Range: 201.1.1.1 – 201.1.1.254

\*Allows for 254 possible hosts

**Router1 to Site1**

Net: 172.16.10.68

Mask: 255.255.255.252

Broadcast: 172.16.10.71

Host Range: 172.16.10.69 – 172.16.10.70

\*Allows for 2 possible hosts

**Router1 to Site2**

Net: 172.16.10.72

Mask: 255.255.255.252

Broadcast: 172.16.10.75

Host Range: 172.16.10.73 – 172.16.10.74

\*Allows for 2 possible hosts

**Router1 to Site3**

Net: 172.16.10.76

Mask: 255.255.255.252

Broadcast: 172.16.10.79

Host Range: 172.16.10.77 – 172.16.10.78

\*Allows for 2 possible hosts

**Router1 to Site4**

Net: 172.16.10.80

Mask: 255.255.255.252

Broadcast: 172.16.10.83

Host Range: 172.16.10.81 – 172.16.10.82

\*Allows for 2 possible hosts

**Router1 to Site5**

Net: 172.16.10.84

Mask: 255.255.255.252

Broadcast: 172.16.10.87

Host Range: 172.16.10.85 – 172.16.10.86

\*Allows for 2 possible host

**Router2 to Site1**

Net: 172.16.10.92

Mask: 255.255.255.252

Broadcast: 172.16.10.95

Host Range: 172.16.10.93 – 172.16.10.94

\*Allows for 2 possible hosts

**Router2 to Site2**

Net: 172.16.10.96

Mask: 255.255.255.252

Broadcast: 172.16.10.99

Host Range: 172.16.10.97 – 172.16.10.98

\*Allows for 2 possible hosts

**Router2 to Site3**

Net: 172.16.10.100

Mask: 255.255.255.252

Broadcast: 172.16.10.103

Host Range: 172.16.10.101 – 172.16.10.102

\*Allows for 2 possible hosts

**Router2 to Site4**

Net: 172.16.10.104

Mask: 255.255.255.252

Broadcast: 172.16.10.107

Host Range: 172.16.10.105 – 172.16.10.106

\*Allows for 2 possible hosts

**Router2 to Site5**

Net: 172.16.10.108

Mask: 255.255.255.252

Broadcast: 172.16.10.111

Host Range: 172.16.10.109 – 172.16.10.110

\*Allows for 2 possible hosts

**Configurations**

**PPP –** point to point protocol

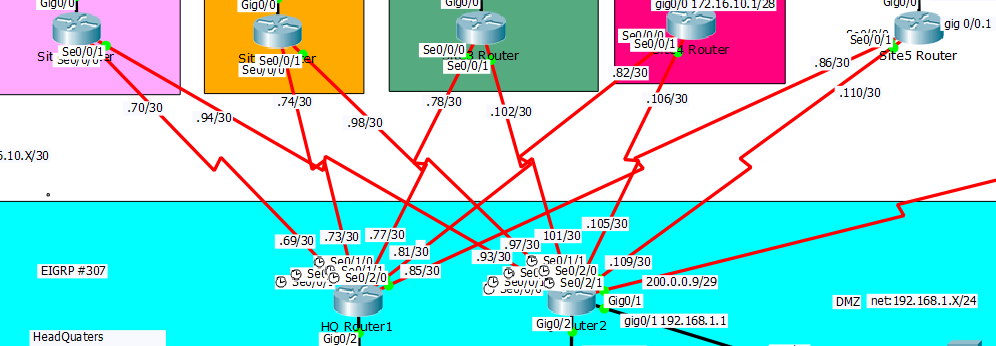
Provides a standard method for transporting multi-protocol datagrams over point to point links.

Configuration commands (From Global)

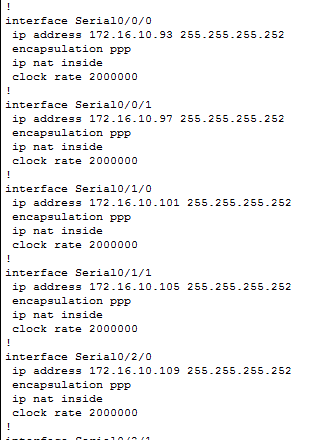
\*int {interface}

\*encapsulation ppp

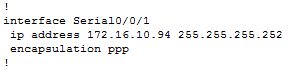
\*you must activate PPP on the other end of the link as well. All of the wan connections to my remote sites have been configured with PPP.



Router 2 Serial Interface Configuration



Accompanying Remote Router Configuration (Site 1)



**DHCP** – Dynamic Host Control Protocol

DHCP is a protocol which assigns and manages IP address from a specified pool within a Router to clients configured for DHCP.

Site 1 Router

Configuration Commands(from global)

\*Ip dhcp pool {pool name}



\*network {network address} {network mask}



\*default-gateway {default-gateway address}



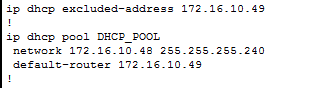
\*exit



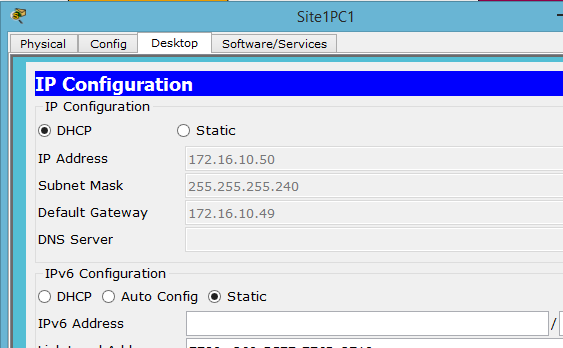
\*ip dhcp exclude 172.16.10.49



DHCP Router Configuration



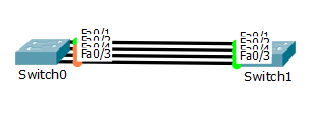
Activate DHCP on the client



**LACP** – Link Aggregation Control Protocol

LACP is used to bundle multiple physical ports together into a single logical link to create an ether channel. This ether channel has the combined bandwidth of all connections and is useful in the event one connection goes down the ether channel can remain up.

Example Topology

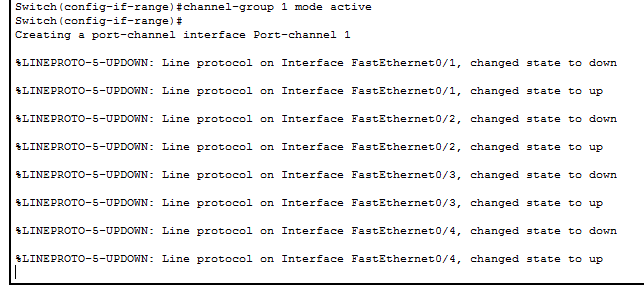


\*notice STP is preventing all the links from going up

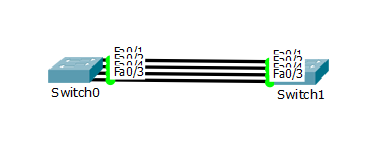
Configuration commands (From global)







\*Repeat steps for second router



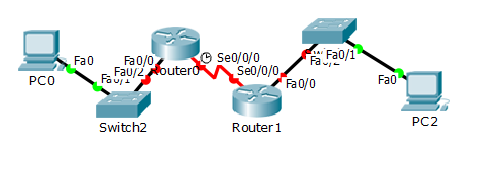
\*All Links are now up and functioning as a single port or etherchannel

**EIGRP –** Enhanced Interior Gateway Routing Protocol

EIGRP is an advanced hybrid of a distance vector routing protocol and a link state routing protocol. EIGRP uses hello packets to discover and maintain adjacencies between routers, uses a reliable protocol to transport routing updates, feasibility to select a loop-free path, the use of diffusing computations to compute new shortest path. EIGRP also invokes the use of vector metrics in computing which are Bandwidth, Load, Total Delay, Reliability, MTU, and Hop Count.

\*Un-configured network topology

We will use ip address 192.168.1.0/24 for PC0 network and 192.168.3.0/24 for the PC2 network. We will then configure 192.168.2.0/30 for our serial wan connection between routers.



Configuration commands(from global)

Router 0

\*int f0/0

\*ip address 192.168.1.1 255.255.255.0

\*no shutdown

\*int se0/0/0

\*ip address 192.168.2.1 255.255.255.252

\*no shutdown

\*router eigrp 307 (307 – represents autonomous system number)

\*network 192.168.1.0 0.0.0.255

\*network 192.168.2.0 0.0.0.3

Router 1

\*int f0/0

\*ip address 192.168.3.1 255.255.255.0

\*no shutdown

\*int se0/0/0

\*ip address 192.168.2.2 255.255.255.252

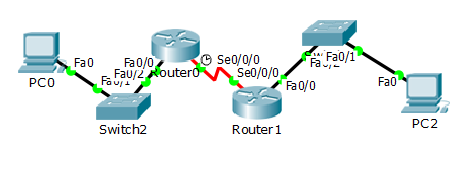
\*no shutdown

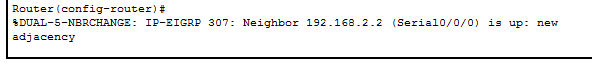
\*router eigrp 307 (autonomous system number must match)

\*network 192.168.3.0 0.0.0.255

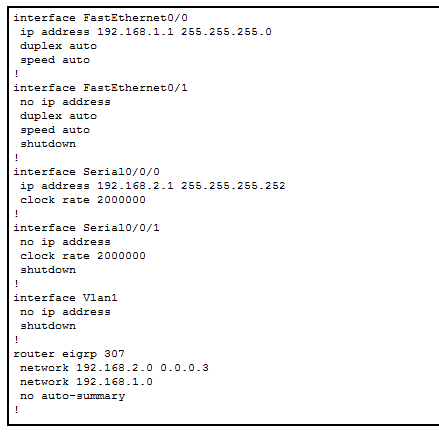
\*network 192.168.2.0 0.0.0.3

\*Links coming up

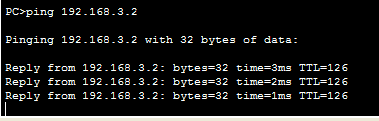




\*Router configuration



\*Ping

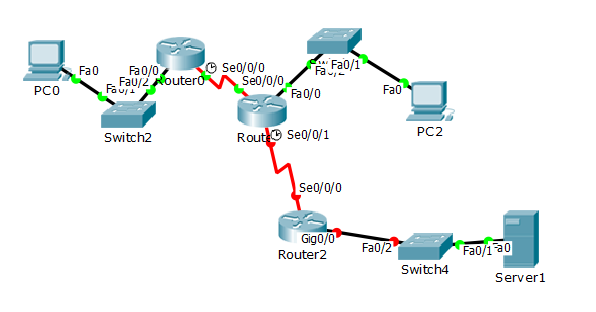


**Static route to ISP**

ISP connections must contain a static route, i.e. no eigrp

Updated topology with added ISP Router

We will add on a static route to Router 1 which will serve as our edge router and a static route to our router 2 which we will pretend is our ISP router. Our ISP server will be put on the 192.168.4.0/24 network and we will use a public network address of 200.0.0.0/29 for our ISP connection.



Configurations

Router 2

\*int gig0/0

\*ip address 192.168.4.1 255.255.255.0

\*no shutdown

\*int se 0/0/0

\*ip address 200.0.0.2 255.255.255.252

\*no shutdown

\*exit – go back to global

\*ip route 0.0.0.0 0.0.0.0 200.0.0.1 - allow any ip address

Router 1

\*int se 0/0/1

\*ip address 200.0.0.1 255.255.255.252

\*no shutdown

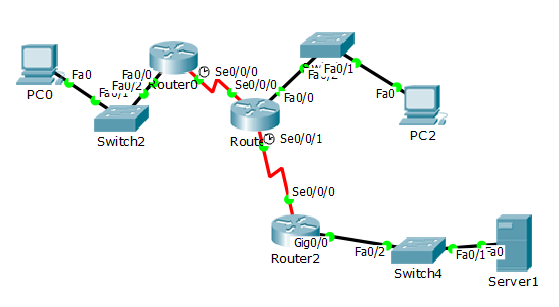
\*exit

\*ip route 0.0.0.0 0.0.0.0 200.0.0.2

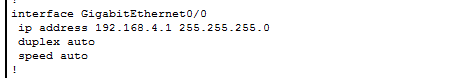
\*\*\*\*\*\*\*\* very important\*\*\*\*\*\*\*\*\*

\*redistribute static – distribute new static ip into eigrp

All links are up



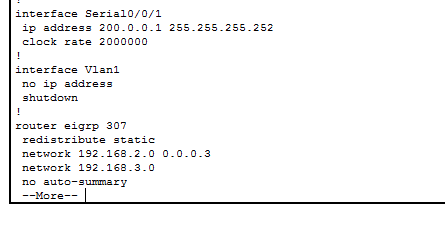
Router 2 Configuration



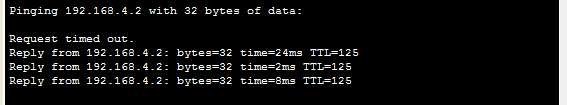




Router 1 updated Configuration



Ping from PC0 to ISP server



**ACL** – Access control list

ACLs are lists of IOS global configuration commands that can match which packets to discard and which to allow through a router on a specified interface. When implementing Standard ACLs, one must first plan the location of the router and interfaces, and decide upon the direction, in or out, on the interface. keep in mind that the ACL list is searched sequentially using first-match logic.

Using the same topology above configuration for router 1 ACL(from global)

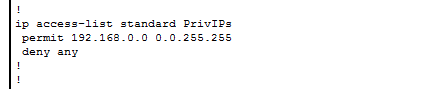
\*ip access-list standard PrivIPs – PrivIPs is the variable name for the ACL

Now you are in the ACL

\*permit 192.168.0.0 0.0.255.255 – we want to permit any traffic from the internal network

\*deny any – deny anything else

The ACL is created and our configuration now displays our ACL



Instead of assigning this ACL to an interface we are going to use it to create our NAT

**NAT** – Network Address Translation

Network Address Translation translates internal network ip addresses into public ip addresses so users can reach out to the internet. Nat can also be configured to hide many internal ip addresses behind one public ip address increasing network security.

Using the same topology we created above we will now configure nat so our private ip address will be translated into a public one when we reach out to our ISP server.

Configuration

To begin we must first reassign our wan connection subnet. We will assign the same ip addresses however use the /29 network. The /29 network allows for 6 total hosts. We will use these extra hosts to store our public address pool.

Router 1(from global)

\*int se 0/0/1

\*ip address 200.0.0.1 255.255.255.248

Router 2(from Global)

\*int se 0/0/0

\*ip address 200.0.0.2 255.255.255.248

Router 1 from global

\* ip nat pool Public 200.0.0.3 200.0.0.6 netmask 255.255.255.248 – creates pool called ‘Public’ of public addresses for nat use beginning with 200.0.0.3 and ending with 200.0.0.6, i.e. our leftover public addresses.

\* ip nat inside source list PrivIPs pool Public – this uses our acl that we just created to define our internal addresses and binds it to our available public address pool. It says permit any internal traffic to be translated into one of our public ip addresses.

Now we need to define which interfaces on our router are on the internal network and which is on the public external network. We do this by going into the individual interfaces.

From Global

\*int f0/0

\*ip nat inside

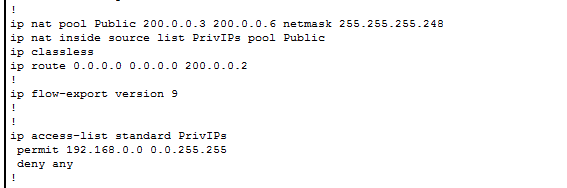
\*int se0/0/0

\*ip nat inside

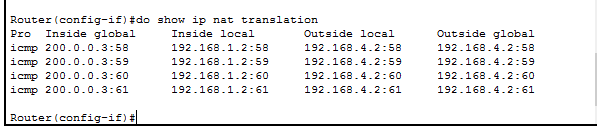
\*int se0/0/1

\*ip nat outside

NAT configuration with ACL



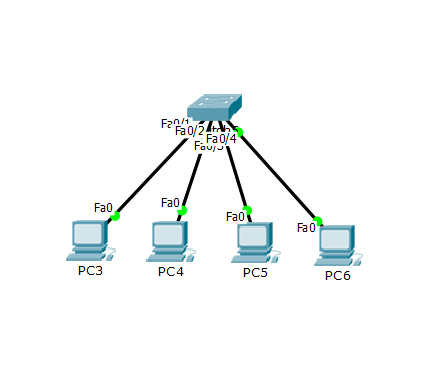
\*show ip nat translation



**VLANS –** Virtual Lans

VLANS allow the grouping of hosts even if they are not on the same physical lan. They are also capable of breaking up one physical LAN into multiple VLANS to prevent hosts on the same virtual LAN from taking to each other.

Network topology



Switches use layer 2 mac addresses to manage the flow of traffic. All ports connected to a layer 2 switch are by default assigned to vlan 1.

Configurations(from global)

Create new vlans

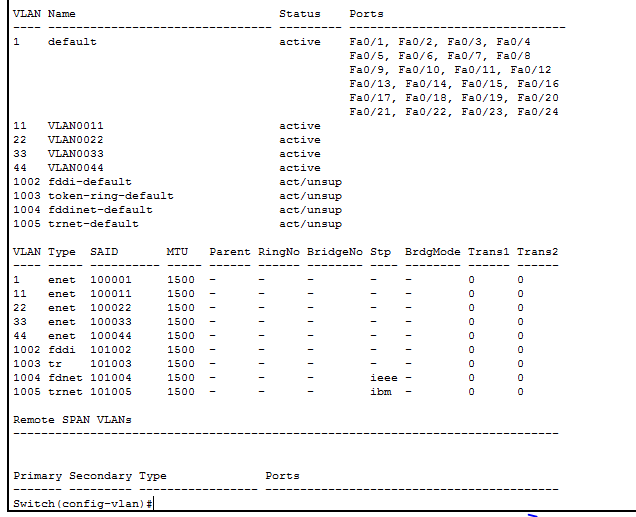
\*vlan 11

\*vlan 22

\*vlan 33

\*vlan 44

\*do show vlan – displays vlan interfaces



Assigning vlans to specific interfaces(from global)

\*int f0/1

\*switchport mode access – configures specific port as an access port

\*switchport access vlan 11

\*int f0/2

\*switchport mode access

\*switchport access vlan 22

\*int f0/3

\*switchport mode access

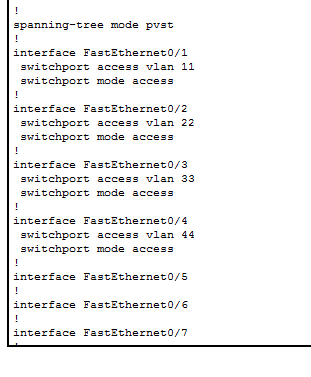
\*switchport access vlan 33

\*int f0/4

\*switchport mode access1

\*switchport access vlan 44

\*do show run – displays running config



Bring up VLAN 1 (administrative vlan)

\*int vlan 1

\*no shutdown

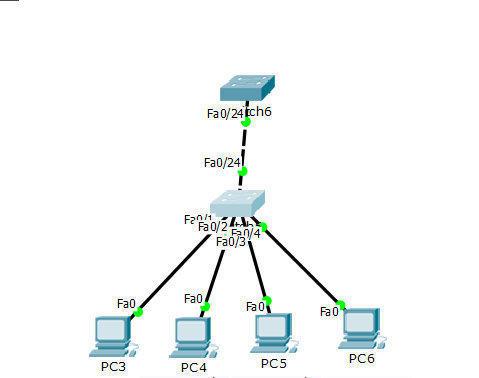
\*do show run



**VTP** – VLAN Trunking Protocol

VTP is a cisco proprietary protocol that propogates the definition of VLANS throughout the entire network by linking them to a VTP domain. In other words rather then having to go into every switch and set up the vlans, VTP will bring up your vlan interfaces throughout the network automatically through the use of VTP servers which updates vlans, VTP clients which receive updates, and VTP transparent devices which remain isolated from VTP propagation but still pass on VTP updates.

Updated topology

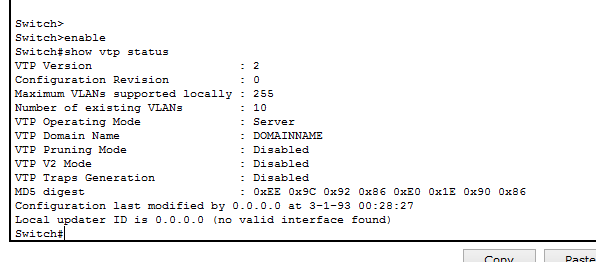


Configuration(from global)

Switch5(bottom swtich)

\*vtp domain DOMAINNAME – creates vtp domain, named DOMAINNAME

\*do show vtp status – displays vtp status



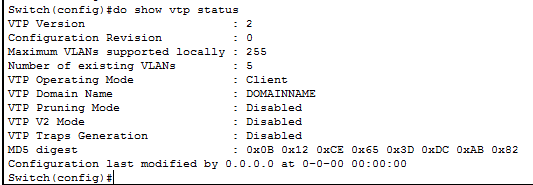
Notice the vtp domain name has been configured to DOMAINNAME and it has discovered our existing vlans. If we only created 5 vlans why does it say we have 10? This is because 5 vlans exist by default for token ring and telnet. Also notice this switch is already configured to operating mode server. This is the default mode for vtp. Now we have to configure our new switch to receive these vlans.

On switch 6 from global

\* vtp domain DOMAINNAME

\*vtp mode client.

\*do show vtp status



Notice this hasn’t been updated with our vlans. This is because we need to set up trunking on our interface that connects the two switches. A trunk port unlike an access port allows the flow of multiple vlans across is.

On switch 6 from global

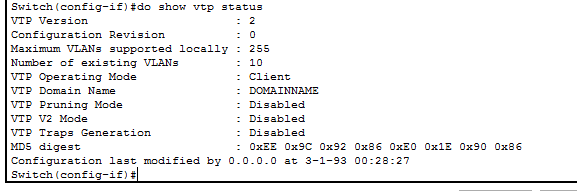
\*int f0/24

\*switchport mode trunk – this will automatically bring up a trunk on the other side of the connection.

If we want we can only allow certain vlans across the trunk

\*switchport trunk allowed 1-99

\*do show vtp status



Our switch 6 is now recognizing those vlans from switch 5

Additional technologies exist within my actual topology such as inter vlan routing with routing on a stick, hsrp and DMZ