**Networking Lab**

**Other Things:**

* Port channel — aggregates multiple physical links (up to 8) into a single logical link
  + - provides for increased bandwidth, load balancing, aggregation, and redundancy
    - int Po1 -> no shut
    - int Fa0/23 -> channel-group 1 mode [on | off | desirable]
      * + “on” means to force channeling on, other side needs to be set to “on” as well
        + “desirable” means to negotiate channeling, other side needs to also be “desirable”
        + spanning-tree vlan 1 port-priority 112 (second int)
* Using “switchport mode trunk” without specifying VLANs will by default trunk all VLANs
* Use Ctrl+Shift+6 to exit out of “waiting” prompts
* DHCP and HSRP should be set at the multi-layer switches, not the routers
* Router0 (the primary router on top of the first MLS) handles all route redistributions

**Enabling SSH and Secure Passwords:**

* service password-encryption
* ip domain-name [domain]
* crypto key generate rsa
  + - * + username admin secret cisco
* line vty 0 4
  + - login local
    - transport input ssh

**VLANs, VTP and Trunking (Router-on-a-stick):**

* vlan 5
* int vlan 5
* Access mode:
  + - switchport mode access
    - switchport access vlan 5
    - spanning-tree portfast
* Trunking mode:
  + - switchport trunk encapsulation dot1q
    - switchport mode trunk
    - switchport trunk vlan allowed 5,6,10
* Router on a stick:
  + - int Fa 0/1 -> no shut
    - int Fa 0/1.100 -> encap dot1q 5, ip add, no shut
* VTP:
  + - Allows a switch to propagate VLAN information to other switches
    - Involves a VTP domain and VTP mode (server, client, or transparent/forwarding)
    - vtp mode [client | server | transparent]
    - vtp domain DOMAIN
    - vtp version 2
    - vtp password PASSWORD

**Static and Default Routes:**

* ip route 0.0.0.0 0.0.0.0 192.168.1.1
* ip route 192.168.2.0 255.255.255.0 192.168.2.1

**RIP Routing:**

* Distance vector routing protocol
* router rip
* version 2
* network 192.168.1.0
* default-information originate
* passive-interface Fa0/1

**OSPF Routing:**

* Link state routing protocol based on SPF algorithm, uses **cost** (bandwidth) for metric
* Router ID is determined by “router-id”, or highest IP address of all loopback interfaces
* Is divided into subdomains called **areas**, where each area has its own collection of networks
* Commands:
  + - router ospf 1 — enables OSPF on the router with a 16-bit process ID
    - network 192.168.1.0 0.0.0.255 area 0 — add network to be routed, single area OSPF
    - router-id — sets the IP of the router (IP address used for OSPF)
    - bandwidth 100 — modify the bandwidth field used for routing calculations
    - ip ospf cost 100 — directly specify the cost for a single interface
    - ip ospf priority [0-255] — directly specify the priority for a single interface
    - default-information originate — redistribute default and static routes in OSPF updates
    - **redistribute eigrp 1 metric 1000 subnets**

**Multi-Area OSPF:**

* Area 0 is the backbone area, all traffic moving from one area to another must go through it
* Each area is a different collection of networks, different areas don’t know about each other
* Using multiple areas helps reduce the database size and improves performance of OSPF
* Route redistribution: (somewhat unrelated)
  + - If you have network A running one routing protocol and network B running another routing protocol, but you want them both to be able to communicate with one another, then you need to use route redistribution.
    - Suppose router C lies between networks A and B, and that A is running OSPF and B is running EIGRP.
    - Then for C, in the OSPF router configuration, set it to redistribute EIGRP.
    - And in the EIGRP router configuration, set it to redistribute OSPF.

**Spanning Tree:**

* Prevents loops and broadcast storms by electing a root bridge (switch with lowest BID)
* Loops happen when there are multiple paths to the same destination
* Prevent loops by disabling redundant paths to the same destination, send BPDUs
* Elect a root bridge, elect root ports on non-root nodes, and elect designated ports on root
* Progression: Blocking -> Listening -> Learning -> Forwarding  (or Disabled, set manually)
* Commands:
  + - spanning-tree cost 50
    - spanning-tree portfast — enter the forwarding state immediately
    - spanning-tree mode pvst — use per-vlan spanning tree mode
    - spanning-tree vlan 1 root [primary | secondary] diameter 4
    - spanning-tree vlan 1-5,6 priority 24576
    - no spanning-tree

**EIGRP:**

* Distance vector routing protocol, uses **bandwidth** and **delay** as the metric
* Unlike RIP, there can be multiple instances of EIGRP running
* Commands:
  + - router eigrp 1 — enables EIGRP on the router with 16-bit process ID
    - network 192.168.1.0 — no subnet mask
    - network 192.168.1.0 0.0.0.255 — with subnet mask
    - no auto-summary — disables auto summarization
    - redistribute static — include static routes when sending routing information
    - ip bandwidth-percent eigrp 50 — sets bandwidth limit percentage
    - **redistribute ospf 1 metric 1500 10 10 1 1500**

**PPP:  (open standard)**

* Used for point-to-point serial connections between two nodes (layer 2 protocol)
* For authentication, we can use either PAP or CHAP (first is unsecure, second is secure)
* Commands:
  + - encapsulation ppp
    - compression [predict | stac]
    - ppp quality 50 — sets quality level, number between 1 - 100
    - ppp multilink — enables load balancing for PPP
    - ppp authentication [pap | chap] — sets authentication mode
    - username USER password PASS — sets login credentials in global configuration

**HDLC:  (Cisco proprietary)**

* High-Level Data Link Control (Layer 2 communication protocol)
* Similar to PPP but provides no support for authentication or compression
* It is the default setting for serial interfaces on Cisco routers
* Commands:
  + - encapsulation hdlc

**HSRP:  (Cisco proprietary)**

* Redundancy protocol for setting up a fault-tolerant gateway in a LAN environment
* One HSRP group (of multiple routers) for each subnet and gateway, based on GID
* The active router performs forwarding, while the standby router waits until it goes down
* A “virtual router” with a virtual IP and MAC address serves as the gateway
  + - There is one virtual router per gateway/subnet
* Set the IP of the real routers in the group normally (via the interface -> “ip add”)
* Commands: (where GID is between 0 - 4095)
  + - standby version 2
    - standby GID ip 192.168.2.1 — enable HSRP and set IP of virtual gateway router
    - standby GID priority 120 — sets the priority (higher = better)
    - standby GID preempt

**GLBP:  (Cisco proprietary)**

* Load balancing protocol for choosing a router to serve as the gateway router
* More efficient compared to HSRP, which actively uses one link but leaves the other unused
* For each group, one router is elected to be the Active Virtual Gateway (AVG)
* Election is based on priority (highest = best) first, then based on highest IP
* Roles in each group:
  + - Active Virtual Gateway (AVG) — highest weight, then highest IP, is elected
      * + assigns virtual MAC addresses to each member of the group
    - Standby AVG (SVG) — second highest weight or IP
      * + takes over for the AVG in case it goes down
    - Active Virtual Forwarder (AVF) — four highest weight values
    - Standby AVF (SVF) — after the first four, take over for AVFs if they go down
* Note that an AVG can also serve as an AVF
* When the PC sends an ARP (IP to MAC) request to the AVG, it will respond with a different MAC address each time in such a way that all of the AVFs are fully utilized
* Commands:  (where GID is an integer)
  + - glbp 10 ip 192.168.1.1 — enable GLBP on interface and set IP of the virtual gateway
    - glbp 10 weight 100 — sets the priority level (weight) of a gateway in the GLBP group
    - glbp 10 load-balancing X — sets the load balancing method to be used
    - glbp 10 name mygroup — assigns a name to the group
    - glbp 10 authentication text mypassword — sets the plaintext password (non-encrypted)
    - glbp 10 authentication md5 key-string MD5\_HASH — sets the password for a GLBP group

**Frame Relay Routing:**

* On the frame relay switch (FRS router):
  + - no ip routing
    - frame-relay switching
    - int S1/2
      * + no ip address
        + no ip route-cache
        + encapsulation frame-relay
        + clock rate 64000
        + frame-relay inft-type dce
        + frame-relay route 205 int S1/5 502
        + no shut
* On router 2:  (S0/0/0 <-> S1/2)
  + - int S0/0/0
      * + no ip address
        + encapsulation frame-relay
    - int S0/0/0.205 point-to-point
      * + ip address 172.32.1.1 255.255.255.252
        + frame-relay interface-dlci 205
* On router 5:  (S0/0/0 <-> S1/5)
  + - int S0/0/0
      * + no ip address
        + encapsulation frame-relay
    - int S0/0/0.502 point-to-point
      * + ip address 172.32.1.2 255.255.255.252
        + frame-relay interface-dlci 502

**DHCP:**

* service dhcp
* ip dhcp excluded-address 192.168.1.1 192.168.1.4
* ip dhcp pool VLAN-POOL
  + - network 192.168.1.0 255.255.255.0
    - default-router 192.168.1.1
    - dns-server 192.168.1.1
* int Fa 0/1
  + - ip add 192.168.1.1 255.255.255.0
    - (no need to run “ip add dhcp”)

**NAT:**

* Setting up access lists and NAT address pools:
  + - access-list 1 permit 192.168.5.0 0.0.0.255
    - ip nat pool POOL 201.201.201.1 201.201.201.1 netmask 255.255.255.252
    - ip nat pool POOL 192.168.1.1 192.168.1.255 netmask 255.255.255.0 — dynamic NAT pool
* Actual NAT translations:
  + - ip nat inside source list 1 pool POOL — bind NAT pool to private address list
      * + add “overload” at the end if you want to enable PAT
    - ip nat inside source static 192.168.1.1 1.2.3.4 — static NAT
* Defining inside and outside interfaces:
  + - int Fa 0/1  =>  ip nat inside
    - int S 0/0/0  =>  ip nat outside
* Definitions:
  + - Inside local = your host IP
    - Inside global = your public IP
    - Outside local = remote host IP
    - Outside global = remote public IP
    - Local = private addresses (as in private IP address blocks like 192.168.0.0)
    - Global = public IP addresses (as in globally routable addresses)

**ACL:**

* Types of ACLs:  (also inbound vs. outbound)
  + - Standard — 1-99, 1300-1999, filter by source IP only
    - Extended — 100-199, 2000-2699, filter by everything
* If packet matches ANY rule, it’s forwarded. If it matches NONE, then it’s dropped.
* access-list 1 permit XXX
* no access-list 1
* ip access-list [standard | extended] MYACL  — create an ACL entry with a name
  + - afterwards you are in config-ext-nacl and can apply multiple rules one by one
* ip access-group 1 [in | out]  — apply an ACL to a specific interface
* no ip access-group 1  — remove ACL from an interface
* Rule format:
  + - [permit | deny] tcp source\_ip source\_wildcard dest\_ip dest\_wildcard eq [protocol\_name]
    - [permit | deny] tcp source\_ip source\_wildcard host host\_ip
    - \*\* You might need an implicit “deny ip any any” or “permit ip any any” at the end of the rule
    - You want to apply the ACL as close to the source as possible

**IPv6:**

* 128 bit addresses, 8x 16-bit fields of 4 characters each (fe80), each character is 4 bits
* Every valid IPv6 address always starts with 2001 and :: represents fields of all 0s
* Eg: 2001:fe::1 becomes 2001:00fe:0000:0000:0000:0000:0000:0001
* 2001:db8::/32 means “all addresses where the first 32 bits are 2001:0db8"
* int Fa0/0
  + - ipv6 add 2001::1/64 eui-64
    - ipv6 ospf 1 area 0
* ipv6 unicast-routing  —  used to ENABLE ipv6 mode
* router ospf 1
  + - router-id 2.2.2.2
* ipv6 router ospf 1
  + - router-id 1.1.1.1

**Debugging commands:**

* ping 192.168.1.1
* ping 192.168.1.1 source 192.168.1.5
* ping ipv6 2001::2d0:d3ff:febb:3001
* traceroute 192.168.1.1
* show ip route
* show ip int brief
* show int trunk
* show run
* show ip dhcp binding
* show ip nat translations
* show frame-relay pvc
* show frame-relay map
* debug ip nat
* show ipv6 interface
* show ipv6 route
* show vlan
* show vlan brief
* show vtp status
* show spanning-tree
* show ip ospf
* show ip ospf interface
* show access-lists
* show ip eigrp neighbor
* show ip eigrp topology
* show standby
* show standby brief
* show glbp
* show glbp GID
* show ip access-lists