CCNA

4 - Build a Physical Cisco Lab Environment

register **0x2142** = ignore startup configuration

7 - Describe Network Functions and Equipment

Switch
- each port has own collision domain
- learn MAC addresses (CAM table, removes MAC after 5 minutes of silence)
- forward or floods data
- receive frames and directs them to their dest

Wireless AP
- communicates packets (frames) wirelessly
- wifi standard 802.11
- channels

Router
- control broadcast and forward directed data packets
- relies on ip-based routing table
- gateway between network types

MDF - main distribution facility

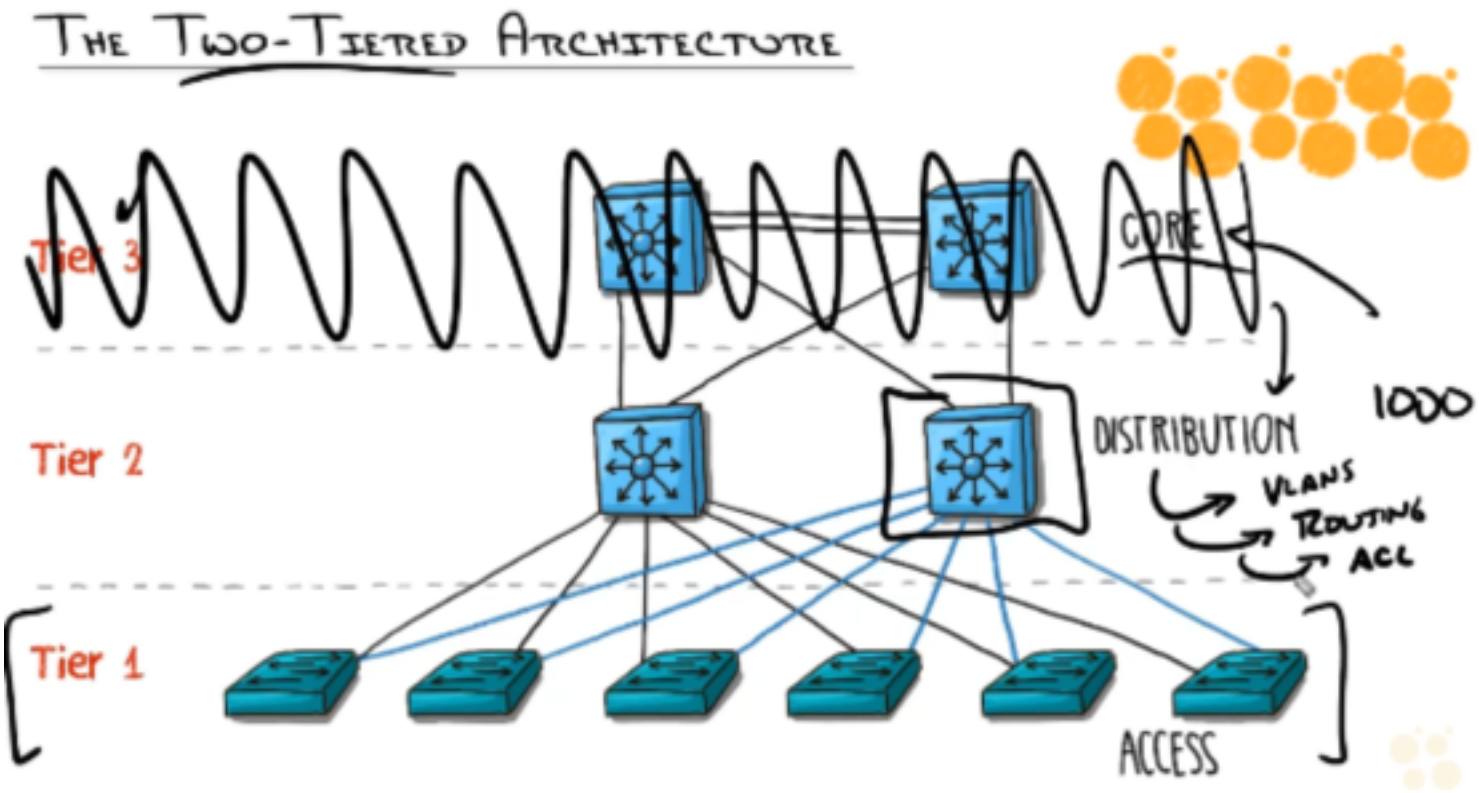
LAG/Etherchannel - link aggregarion

8 - Explain Network Communication Using the OSI and TCP-IP Model

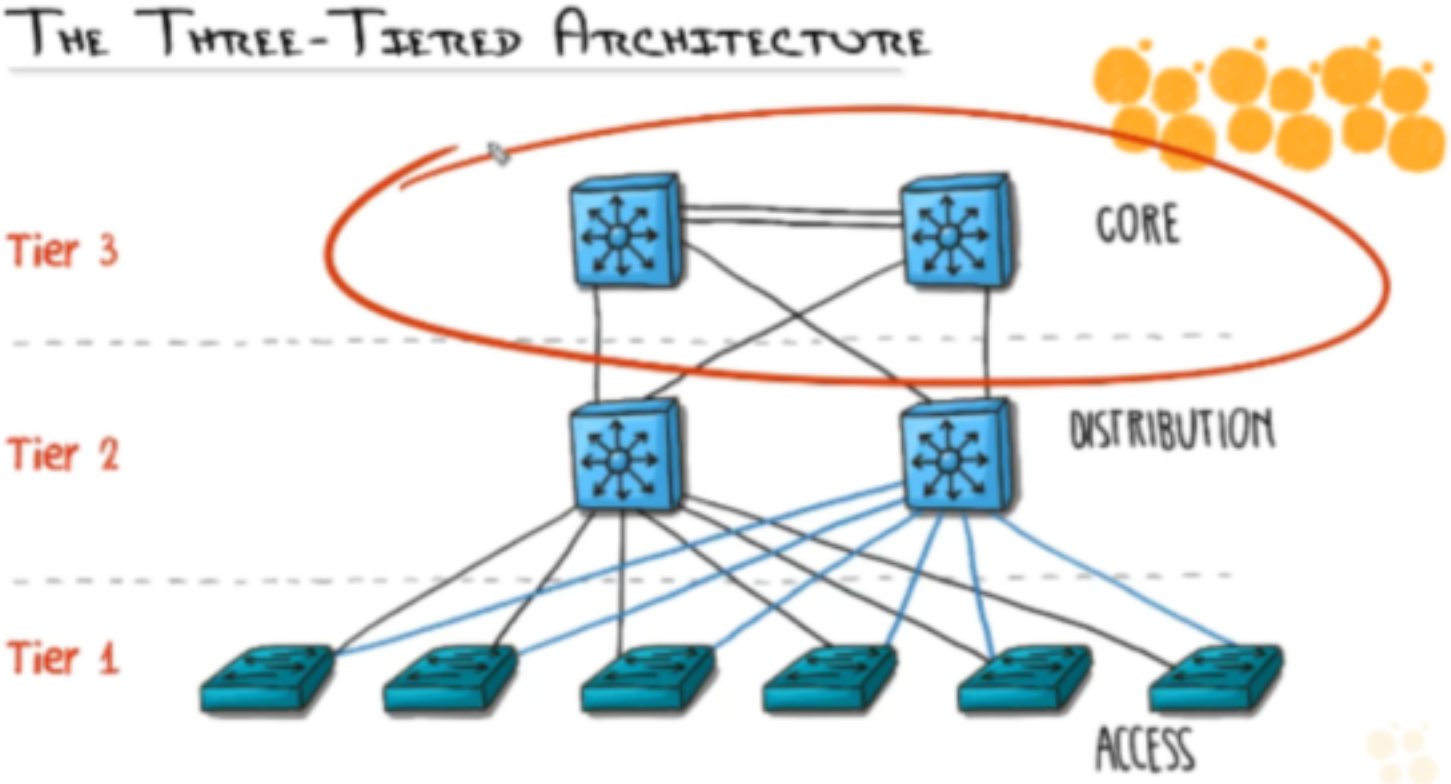
aplication - layer for network aware apps
presentation - formats, codings, encryption
session - start, stop, maintenance of session
transportation - choice of protocol by app, usually TCP, UDP (can be TCP, IP, UDP, ICMP...), ports
network - ip, routers, packet
data link - mac, switch, frame
physical - bits

9 - Describe Common Network Architectures

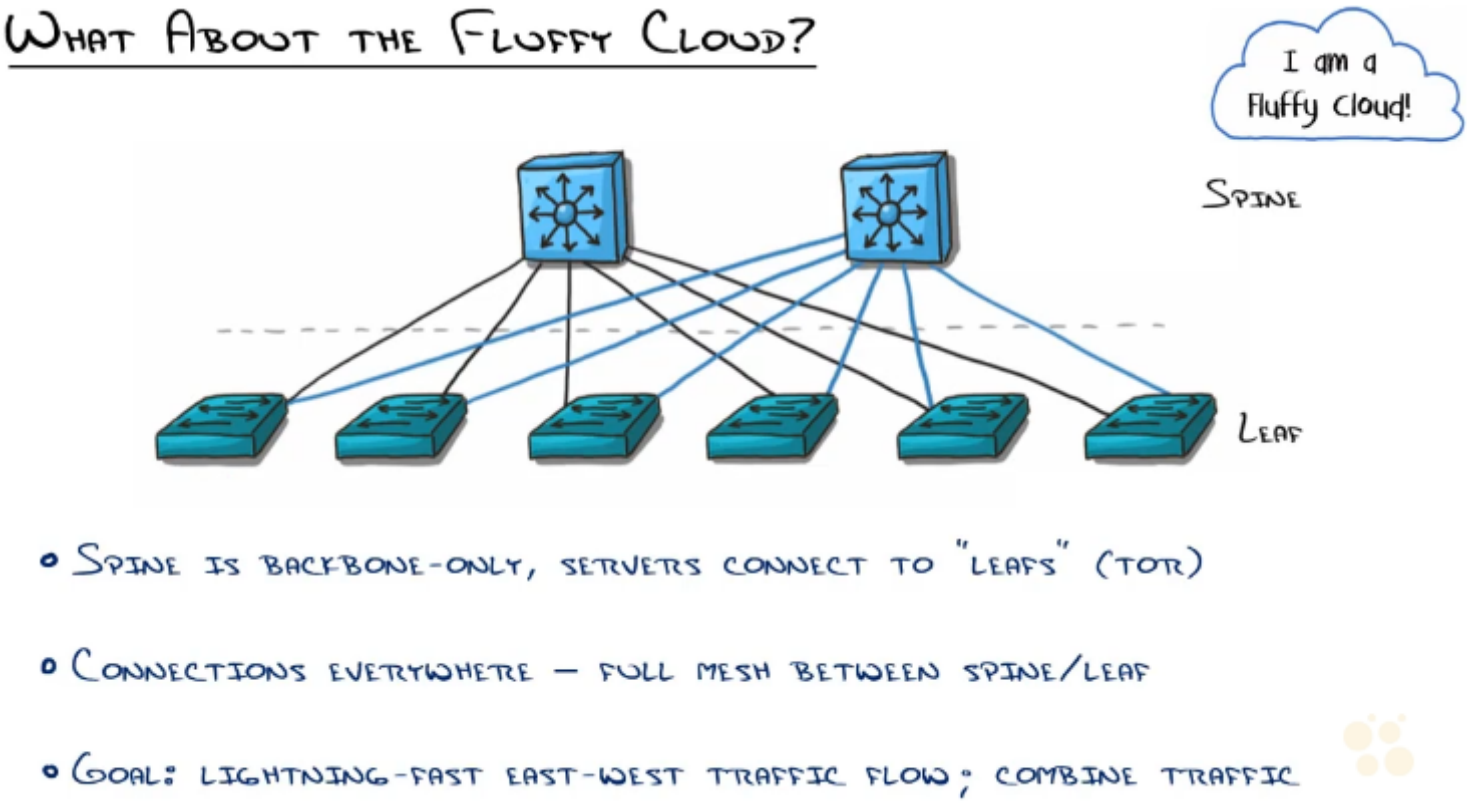
Two-tie architecture (collapsed core)
- core switch - critical stuff (router), should be good switch
- add second core switch for redundancy
- access switch - pc, printer etc.



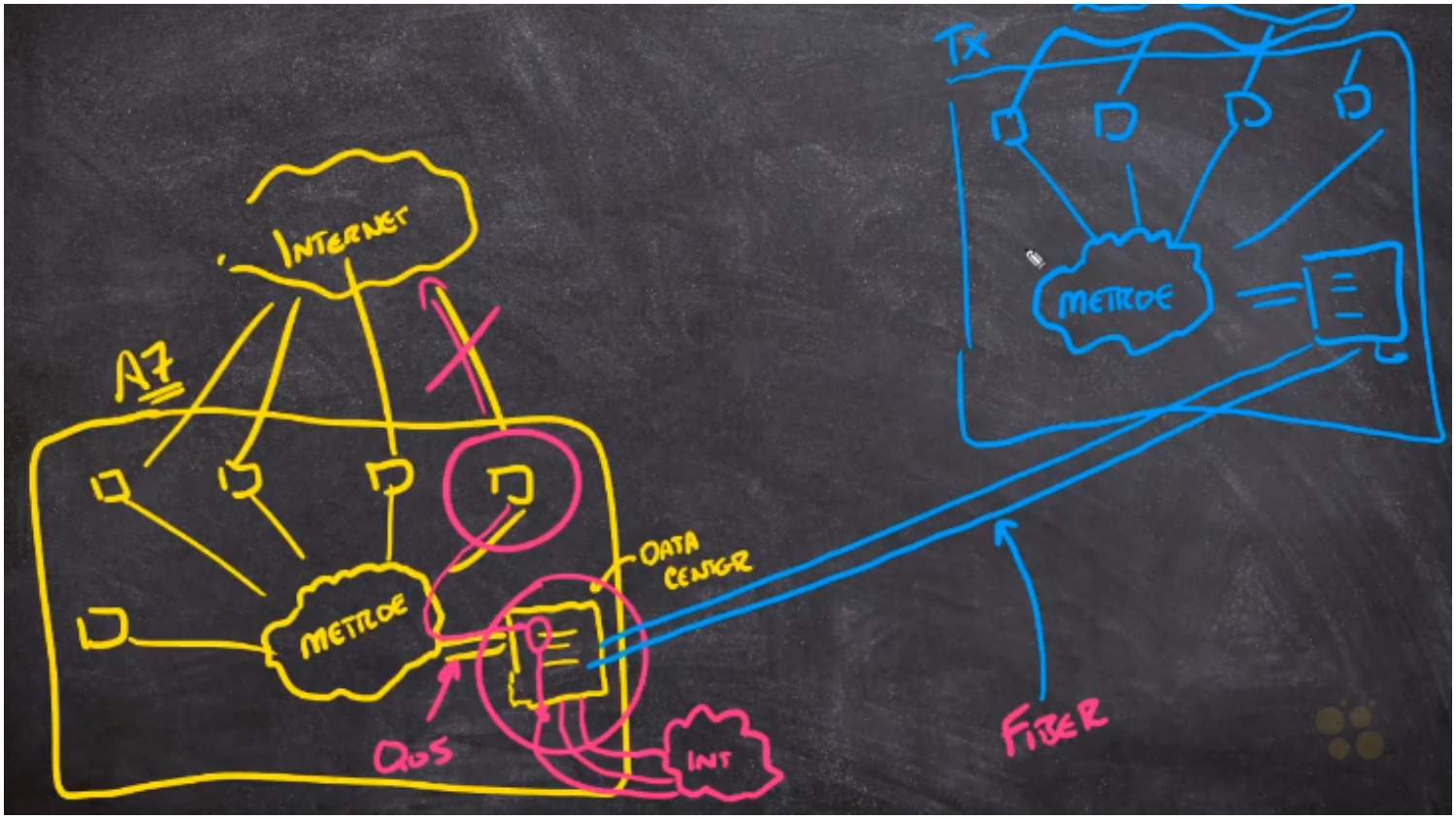
Three-tier
- campus network - spans multiple buildings
- adds core layer to interconnect buildings
- fast connect point, really good switches
- design for north-south traffic flow (up/down)



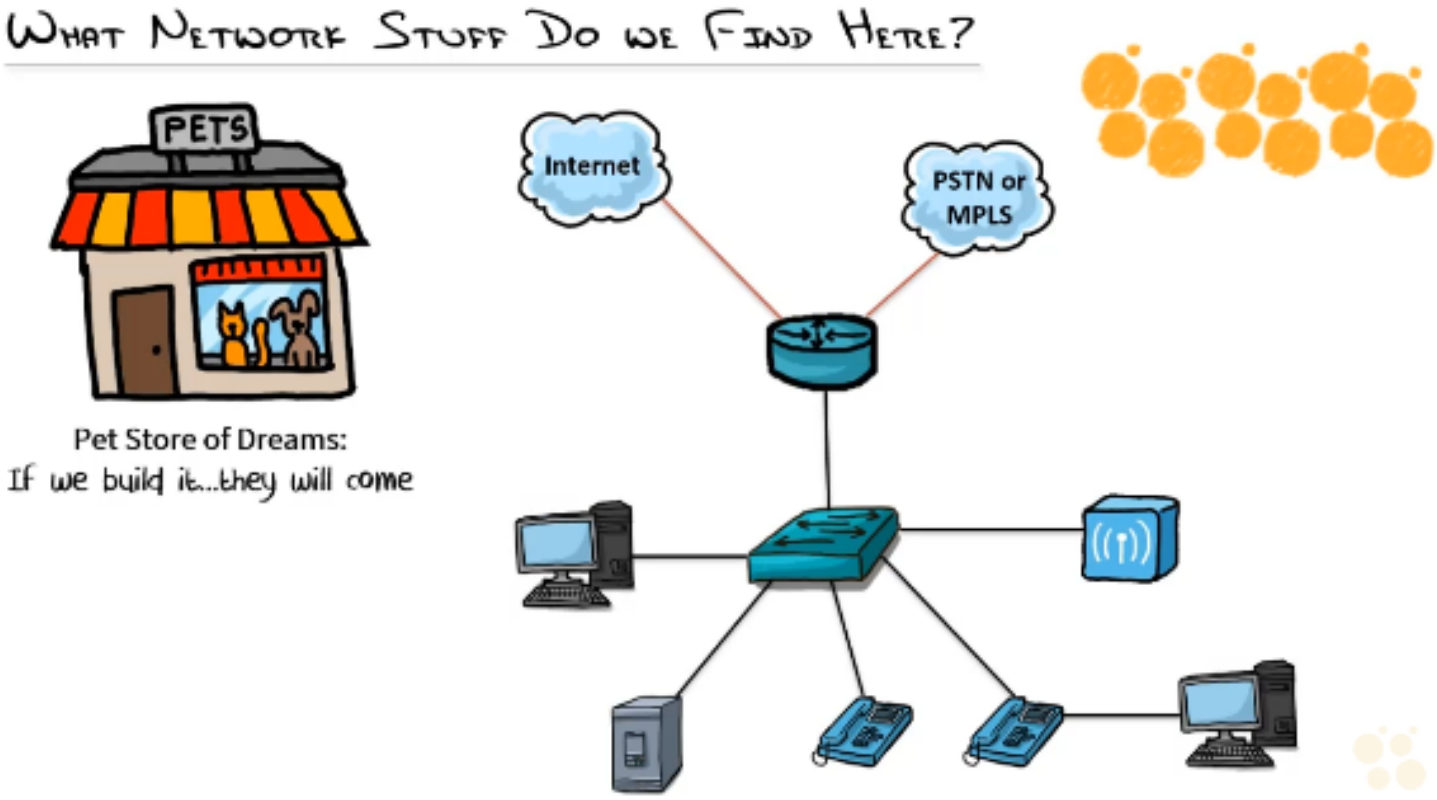
Spine-leaf
- design for west-east (side way)
- it is DATACENTER DESIGN (could be 3-T business design and Spine-leaf datacenter design)
- TOR=top of rack switch
- spine switches are not directly connected, leafs are one hop from each other
- leaf switch can use loadbalancing and use use multiple links to eliminate bottlenecks



WAN
- usually leased lines or MPLS or MetroE
- you get QoS (SLA for the line)
- leased lines - private lines rented to connect 2 sites, your bandwith, pretty expensive (T1, E1)
- MPLS - packet switched, virtual circuits
- MetroE - bunch of optical cord which can be subleased
- nowadays even just VPN and internet (cheapest)



SOHO



Modern business
- mix public cloud, private cloud, on-prem

10 - Discern Copper and Fiber Optic Network Cable Characteristics

Fiber

Single mode (SMF)
- usually yellow
- small glass core
- long distance
- more expensive, more bandwidth

Multi mode (MMF)
- usually aqua or orange
- thicker, plastic core
- shorter distance
- cheaper, lower bandwidth

11 - Connect and Navigate Cisco Internetwork Operating System (IOS)

`enable`
- move from user mode to priviledge mode

configure terminal
- global config

no
- negate command

>
- user mode, limited, used for viewing and testing

#
- priviledge mode, full access,

show ip interface brief
list interfaces

factory reset:
enable
write erase (erase nv ram)
reload (reload running config)

12 - Create a Base Configuration for Cisco Devices

hostname XXXX
- set name of the device

banner motd
- set logon banner

how to set up static ip address
enable
configure terminal
interface g0/0
ip address 192.168.10.1 255.255.255.0
no shutdown

Set up for telnet and SSH
- telnet is turned on by default

-setup using local db for telnet and ssh (so it asks for username)
line vty 0 4
login local

crypto key generate rsa
- generate keys for ssh
- to use it, you have assign hostname and ip domain-name first

transport input ssh
- limit connection just to ssh, disable telnet

base config:
enable
configure terminal
hostname AZ-RT01 (set name of router)
do show ip int brief (do... to run priviledged command in any other mode)
int g0/0
    ip address 192.168.1.1 255.255.255.0 (set static ip)
    exit

do show ip int brief
banner motd % motd message blabla %
line console 0
    logging synchronous (keep status msgs from interupting typing)
    password cisco
    login
    exit
line vty 0 4 (configure 5 telnet ports)
    login
    password cisco
    exit
do show run | b vty (show running cfg, beginning with vty)
enable secret cisco (password for enable mode)
int g0/0
    no shutdown
    exit
no ip domain-lookup
service password-encryption (encrypt not encrypted passwords)
exit
wr (copy running to startup config)
- ip shouldn't be given to g0/0 but to vlan1 of shwitch
interface g0/0
    no ip address
    shutdown
interface vlan 1
    ip address 192.168.1.1 255.255.255.0
    exit

switch:
enable
configure terminal
interface vlan 1
    no shutdown
    ip address 192.168.1.2 255.255.255.0
    exit
wr

13 - Create a Base Configuration for Cisco Devices Hands-On Lab

14 - Wireshark Fundamentals Capturing, Viewing, and Filtering Data

15 - Describe and Analyze TCP and UDP

Communication

16 - Configure Windows, MAC, or Linux for Network Access

17 - Describe Network Switch Functions and How to Locate Network Devices

show mac

show mac address-table aging-time
- time in seconds to remember mac

clear mac address-table dynamic
- clear table and relearn (floods network temporarly)

show spanning-tree
- see SPT network, ports which are disabled

broadcast storm - STP prevents it

show mac address-trable | include fd1a (grep fd1a mac from output)

if you need to find out mac address, ping IP first and then check arp table
arp -a

show mac address-table interface fa0/13 (just check if there is only 1 mac under the port

show cdp neighbors
- show other cisco devices

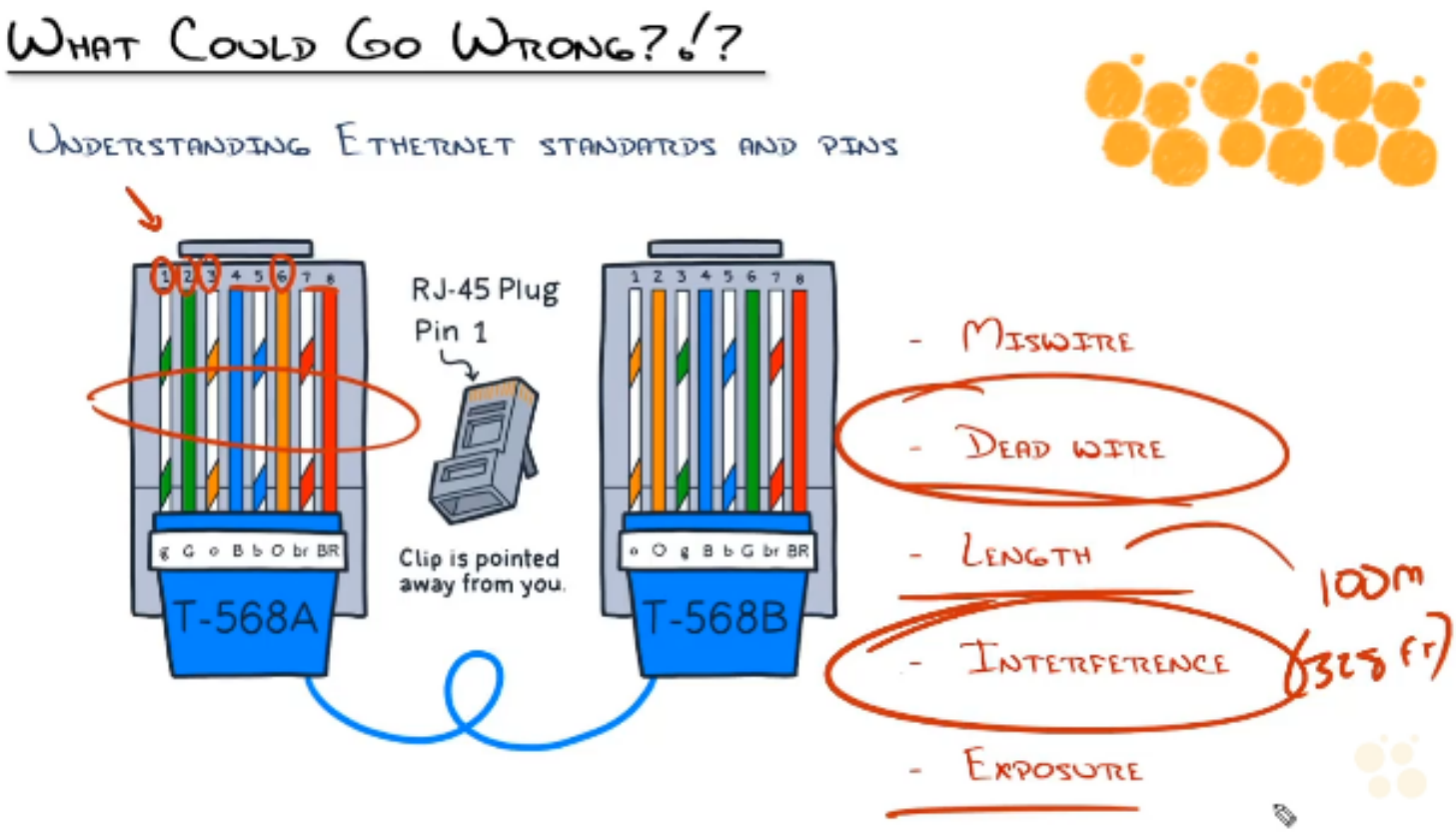
show cdp neighbors detail
- to find out IP or the remote device

18 - Diagnose Interface Status, Errors, and Cabling Issues on a Cisco Switch

T-568A
T-568B

A->A or B->B is straight through
A->B or B->A is crossover

MDI-X = auto find out if crossover or straight



show interfaces status
- similar as brief
show ip interface brief

- best practice on 10/100 switches hardcode speed on both sides, and *disable AUTO*
- because if 1 hardcoded the other would auto go auto to **100MB HALFduplex!**

show interfaces fa0/1
- info about port, mac, MTU, txload (transmit load, how utilised it is), rxload (receive load), ...
- 5 minute input rate (5 min avg)
- 5 minute output rate (5 min avg)
- statistics under it
- broadcast cnt (a lot? maybe broadcast storm?)
- statistics
  → runts - too small (<64B)
  → giants - too large (over MTU)
  → collision/late collision - shouldn't happen on switch (if happens, you have duplex mismatch)
  → frame - frame has illegal format

19 - Describe Power over Ethernet (PoE) Capabilities and Standards

PoE process:
1. Power source identified
2. Power device detected using Fast Link Pulse (FLP)
3. power negotiation occurs to negotiate power needed (3 classes)
   - LLDP can negotiate power class as well
4. Power delivered (orig over non-data wires)

PoE standards:
• Cisco Inline Power - not compatible with others
• 802.3AF
  ◇ 2003, POE, pulse to determine power class
  ◇ most common stadard, 15.4W
  ◇ Mode A, Mode B (sometimes not compatible)
• 802.3AT
  ◇ 2009, POE+, LLDP to determine power class
  ◇ 30W
• 802.3BT Type 3
  ◇ 2018, 4PPOE, LLDP to determine power class
  ◇ 60W, use all 4 pairs
• 802.3BT Type 4
  ◇ 100W, LLDP to determine power class, use all 4 pairs

- back compatible except for Cisco CIP
- standard PoE is 48V

show power inline
- verify PoE

20 - Explain IP Addressing and Subnetting Concepts

- A, B, C classes mainly
- classless change default mask, not classes itself
- rfc **1918** ip address allocation
- APIPA 169.254.x.x

small office 10-100
- start using VLAN for wifi and voip
- mix of static and dynamic addresses

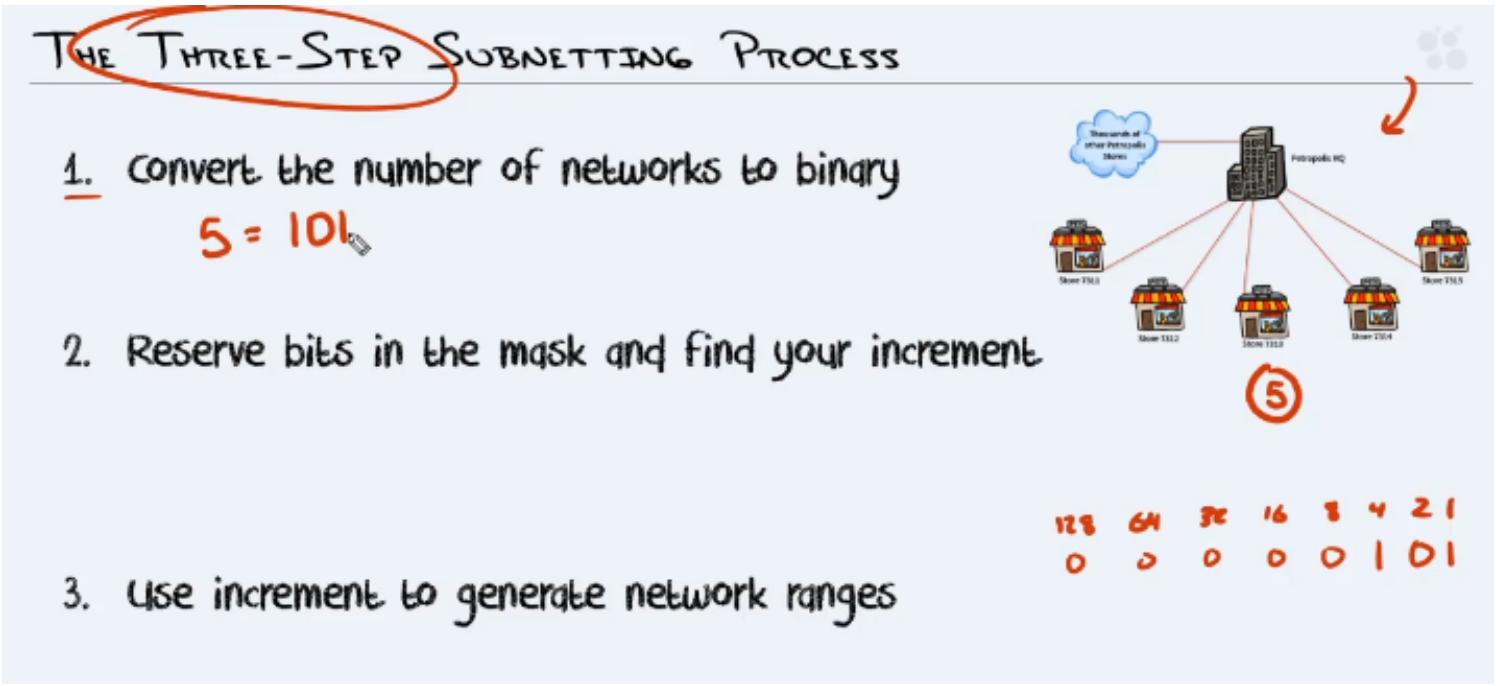
medium/multiple offices
- each office is one or more subnets
- WAN is a subnet
- internet is subnet

- every interface of router is subnet
- every office is at least 1 VLAN

21 - Convert Decimal to Binary and Back

22 - Perform Subnetting Based on Network Requirements

CIDR - classless inter domain routing



C CLASS: You get 192.168.5.0/24, you need 5 subnetworks

Three step subnetting process
1. Convert the number of networks to binary
1) 5=101= 3 bits
2. Reserve bits in the mask and find your increment
1) original is /24 so 11111111.11111111.11111111.00000000
2) need 3 so /27 so 11111111.11111111.11111111.11100000 so 255.255.255.224
3. Use increment to generate new network ranges
1) check last mask bit 1 to find increment, here it is 32
1- 192.168.5.0 - 192.168.5.31 -> 30 usable addresses each subnet
2- 192.168.5.32 - 192.168.5.63
3- 192.168.5.64 - 192.168.5.95
4- 192.168.5.96 - 192.168.5.127
5- 192.168.5.128 - 192.168.5.159
6- 192.168.5.160 just need 5 networks so doesn't care here

B CLASS: You get 172.16.0.0/16, you need 60 subnetworks

- you create 60 subnets and hand them over, they can subnet them again as needed

1. Convert the number of networks to binary
1) 60, I don't need to calc exact binary number, it will be 001xxxxx so need 6 bits
2. Reserve bits in the mask and find your increment
1) original /16 so new one /22 so 255.255.252.0
3. Use increment to generate new network ranges
1) check last 1 in mask position -> 4 so increment is 4
1- 172.16.0.0 - 172.16.3.255 -> usable is (2^10)-2 so 1022 hosts
2- 172.16.4.0 - 172.16.7.255
3- 172.16.8.0 - 172.16.11.255
4- etc...

A CLASS: You get 10.0.0.0/8, you need 1000 subnets

1. Convert the number of networks to binary
1) 1000, so 10 bits (2^10=1024, 2^9=512, so 9 not enough so 10)
2. Reserve bits in the mask and find your increment
1) original /8 so new /18 so 255.255.192.0, so 11111111.11111111.11000000.00000000
3. Use increment to generate new network ranges
1) check last 1 in mask position -> 64 so increment 64
1- 10.0.0.0 - 10.0.63.255 -> usable is (2^14)-2 so 16382

2- 10.0.64.0 - 10.0.127.255
3- 10.0.128.0 - 10.0.191.255
4- 10.0.192.0
5- 10.1.0.0

- be careful re specific values, eg 2 networks (4,8,16...):
→ 2 doesn't need 2 bits but just 1 so careful with first step
→ 4 networks need 2 bits not 3
→ you can use 2^y to find out correct (2^2=4 networks so 2 bits)
→ or just subtract 1 from networks you need

128 64 32 16 8 4 2 1

23 - Perform Subnetting Based on Host Requirements

128 64 32 16 8 4 2 1

C CLASS: 192.168.5.0/24, need 25 hosts per network

1. Convert hosts to binary
   1) 25 hosts = 5 bits HOST
2. Reserve bits in the mask and find your increment
   1) mask was /24 so 8 host (32-24), I just need 5 host so new mask will be /27
   2) new 255.255.255.224
3. Use increment to generate new network ranges
   1) last bit in mask is 32 so increment is 32
      1- 192.168.5.0 - 192.168.5.31 so (2^5)-2 30 hosts per network
      2- 192.168.5.32
      3- 192.168.5.64
      4- 192.168.5.96 ...

B CLASS: 172.30.0.0/16, need 80 hosts per network

1. 80 hosts, need 7 bits
2. mask was 16 so 16 hosts, need just 7 host so mask /25
   1) new 255.255.255.128
3. last bit in mask is 128 so increment is 128
   1) 172.30.0.0 - 172.30.0.127 -> 2^7-2=126 hosts
   2) 172.30.0.128 - 172.30.0.255
   3) 172.30.1.0 - 172.30.1.127
   4) 172.30.1.128

A CLASS: 10.0.0.0/8, need 1000 hosts per subnet

1024 512 256 128 64 32 16 8 4 2 1
1. 1000 hosts, 01x.xxxxxxxx so you need 10 bits
2. mask is /8 so 24 host so new is /22 so 255.255.11111100.0
   1) new mask 255.255.252.0
3. last bit in mask is 4 so increment is 4
   1) 10.0.0.0 - 10.0.3.255 -> 2^10-2=1022 hosts
   2) 10.0.4.0 - 10.0.7.255
   3) 10.0.8.0

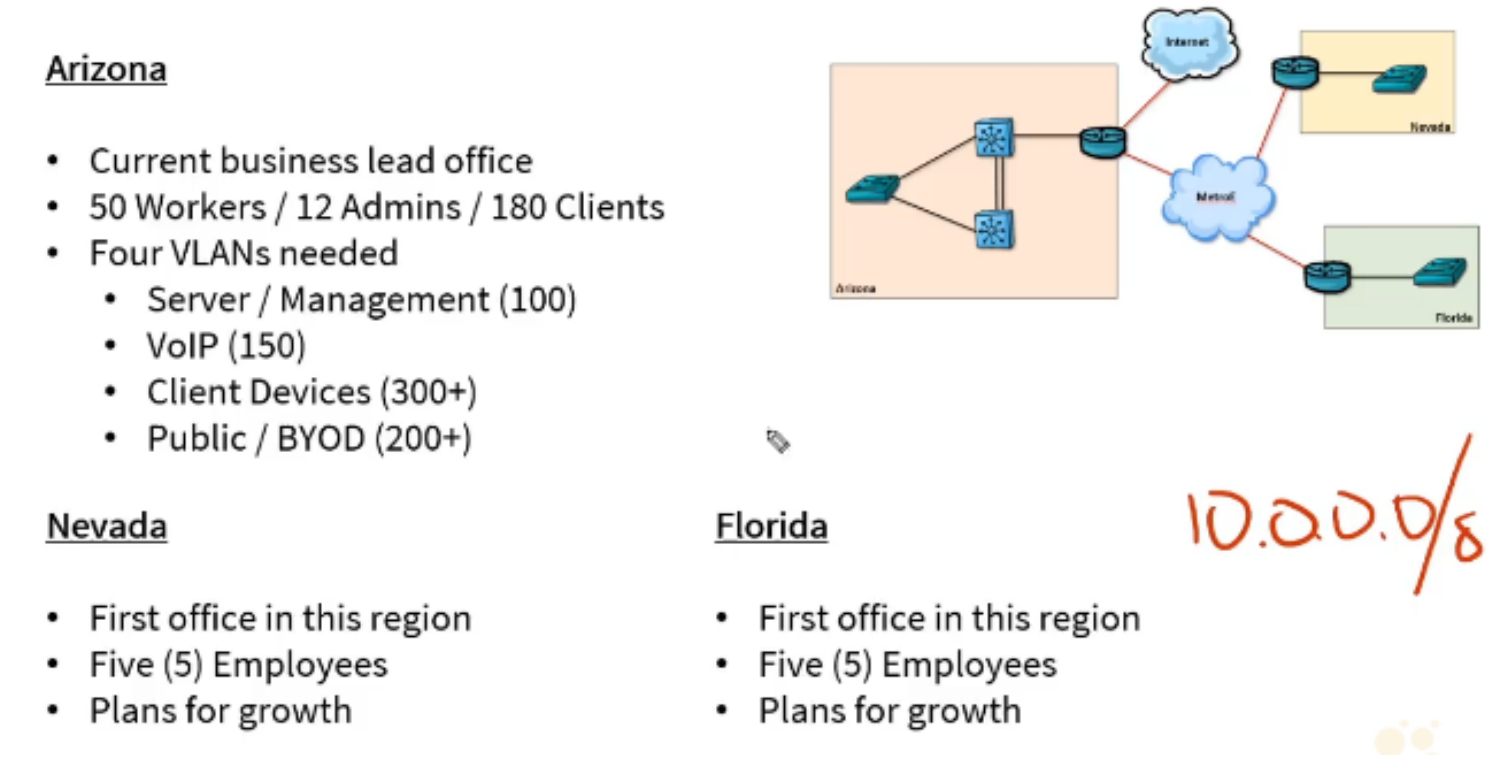
- if it is boundary 2,4,8,16...
  → for host is it 15,31,127 that is 1 short of boundary you end up 1 short
  → so if you need 31 hosts, figure for +1 so 32

24 - Reverse Engineering Subnets and Using VLSM

- you can do same process but you have mask already so just get increment and ranges to determine in what range IP is

VLSM - variable length subnet masking
192.168.10.0/24, we need divide into 1x100,2x20,1x10 devices (3 sub networks)
- it is C so to accommodate 100 we would divide to 2 subnetworks of 128 but it is not enough
- we divide based on the largest so 100 so we need actually 128
  → **192.168.10.0 - 192.168.10.127/25** -> 126 devices
  → 192.168.10.128 - 192.168.10.255/25 -> 126 devices
    ⇒ I divide further to get 2 subnets of 20
    ⇒ 20=5 bits so /27, increment 32
    ⇒ **192.168.10.128 - 192.168.10.159/27 -> 30 devices**
    ⇒ **192.168.10.160 - 192.168.10.191/27**
    ⇒ 192.168.10.192 …
      • I divide further to get subnet for 10 devices
      • 10=4 bits so /28, increment 16
      • **192.168.10.192 -192.168.10.207/28 -> 14 devices**
      • 192.168.10.208 ...

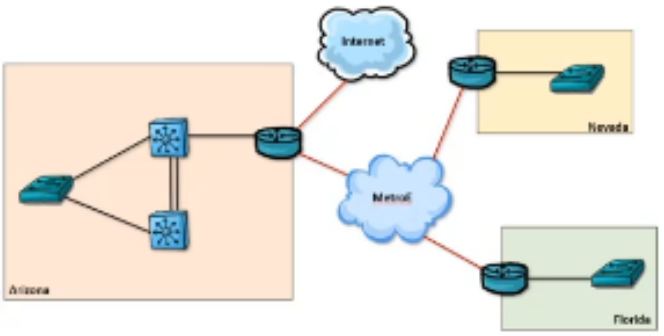
Real work example:



VLANs
• servers 100+30%=130 -> /24
  ◇ 10.0.0.0/24 (V10)
  ◇ 10.0.1.0/24 reserved for expansion
• voip 150+30% -> /24
  ◇ 10.0.2.0/24 (V20)
  ◇ 10.0.3.0/24 reserved for expansion
• cliend devices, use VLSM from 10.0.4.0/24
  ◇ 300 h=9 bits = /23, increment 2

◇ 10.0.4.0 - 10.0.5.255/23 (V30)
• public devices, just use next one same size
◇ 10.0.6.0/23 (V40)

Arizona



- Current business lead office
- 50 Workers / 12 Admins / 180 Clients
- Four VLANs needed
- V10 • Server / Management (100) – 10.0.0.0 -255/24
- V20 • VoIP (150) – 10.0.2.0-255/24
- V30 • Client Devices (300+) – 10.0.4.0 - 5.255/23
- V40 • Public / BYOD (200+) – 10.0.6.0 - 7.255/23

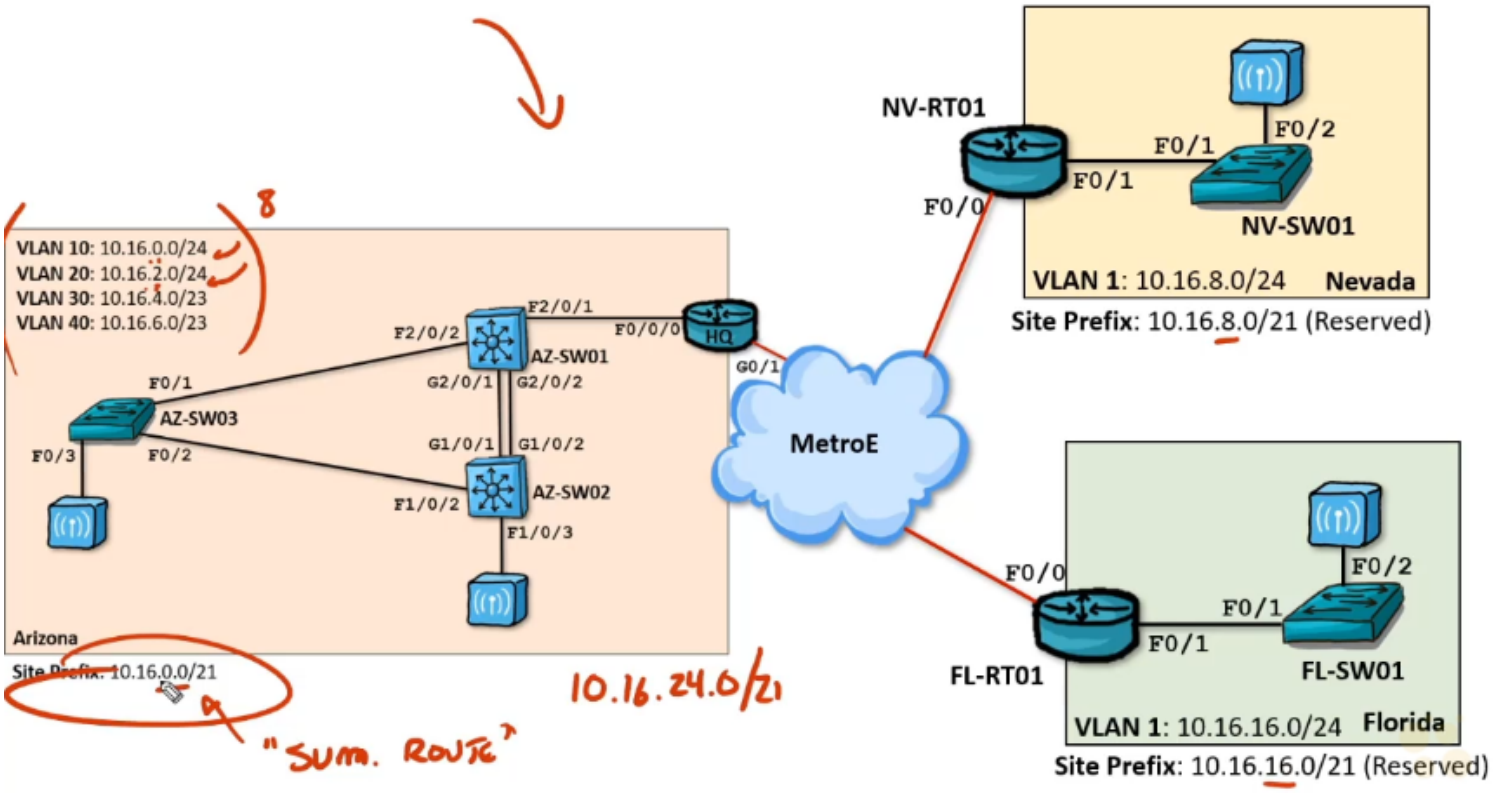
Nevada

- First office in this region
- Five (5) Employees
- Plans for growth

Florida

- First office in this region
- Five (5) Employees
- Plans for growth

Arizona: 10.0.0.0/21
Nevada: 10.0.8.0/21
Florida: 10.0.16.0/21



Level 3 Diagram

25 - Describe IPv6 Core Addressing Concepts

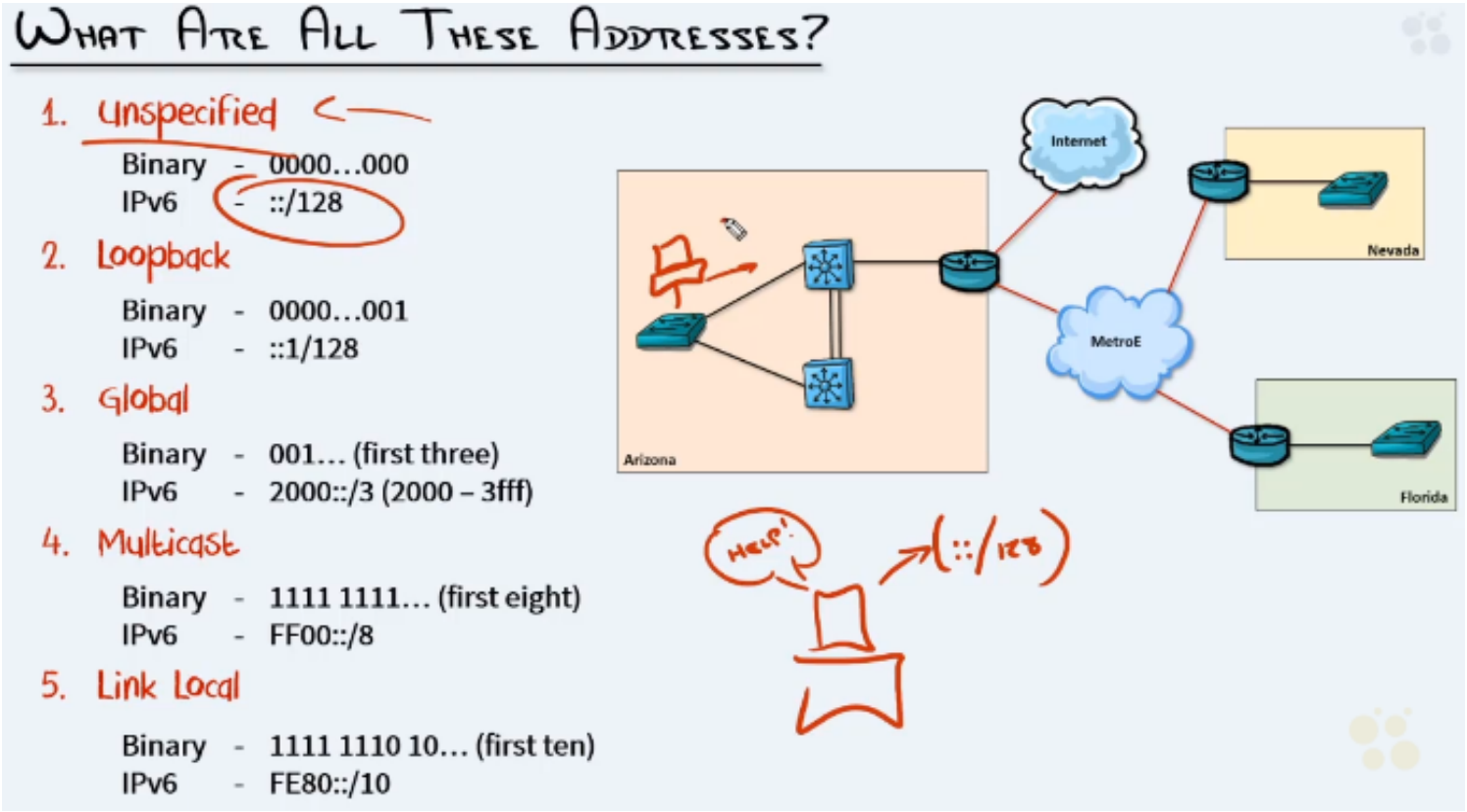
Same:
- L3 addresses
- network and host portion of address
- mask for decision what is network part
- link to L4 protocols

Different:
- IPv4=32bit address, dotted decimal, decimal mask or CIDR, IPv4=address
- IPv6=128bit address, colon hexadecimal, just CIDR notation, IPv6=global address

- overlay = leave IPv4 and put IPv6 over it and slowly remove IPv4 if it works

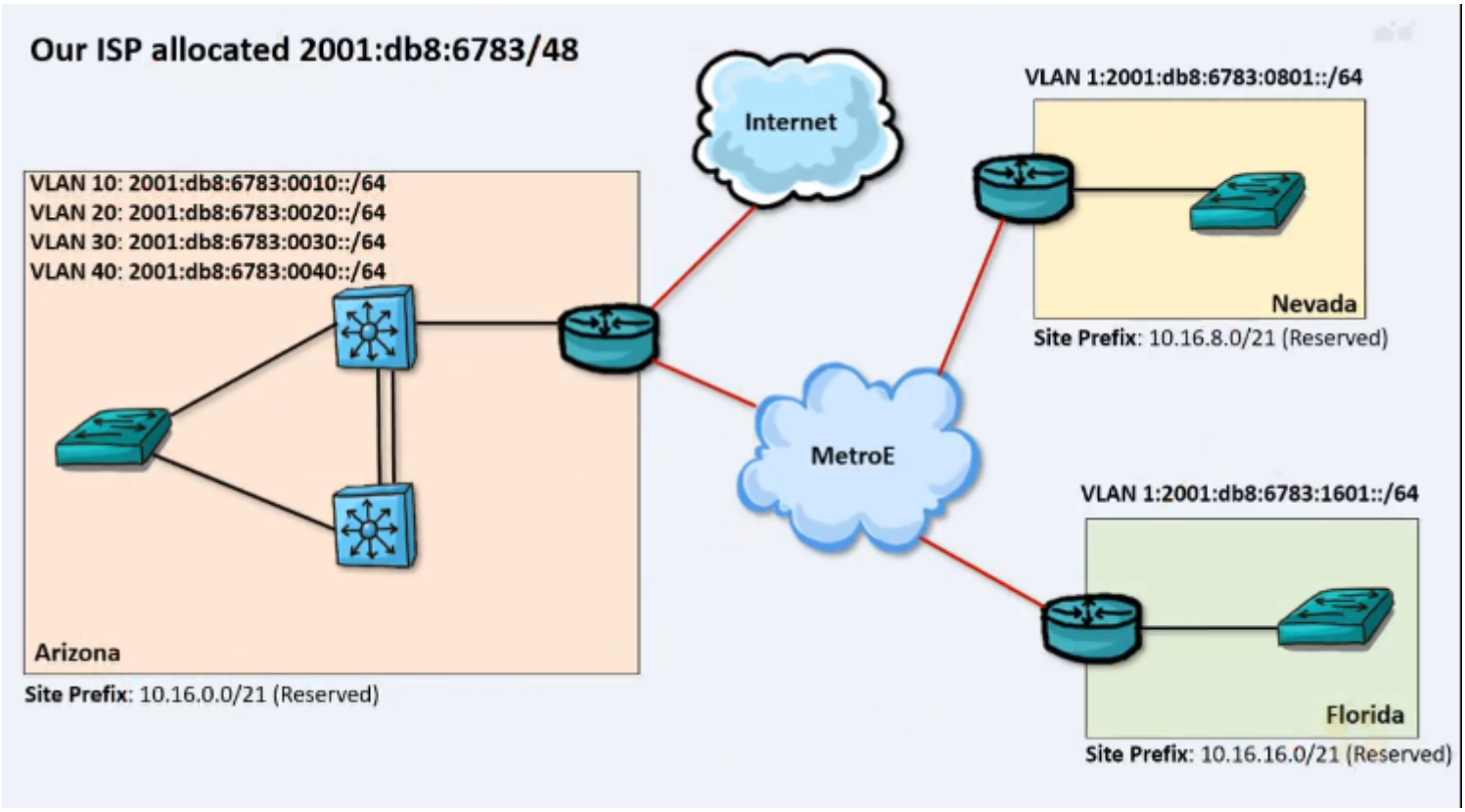
Shortening:
- leading zeros can be dropped
- you can replace running groups of zeros with :: once

- no broadcast addresses in IPv6



2xxx - global
FFxx - multicast
FE80 - link local, get last 64b: EUI-64=take mac address, split in the middle, put in the middle FFFE, invert 7th bit
  - link local address is used for neighbour protocol as it is more stable
FC00::/7 - unique local, private addressing

ISP allocated 2001:db8:6783/48



- usually split 128bit address in the middle to get net and host
  → 2001:db8:6783:XXXX::/64 for network
- you can mirror you IPv4 and vlan number for example
  → 10.16.16.0/21 IPv4 Vlan1 to XXXX=(16)(01) so 2001:db8:6783:1601::/64
- you don't need really DHCP, you can use EUI-64 for hosts

How to set ipv6 address:

```
configure terminal
ipv6 unicast-routing (to enable ipv6 routing)
int fa0/1
    ipv6 address xxxx/xx
show ipv6 interface
```

You can auto config (without global address)

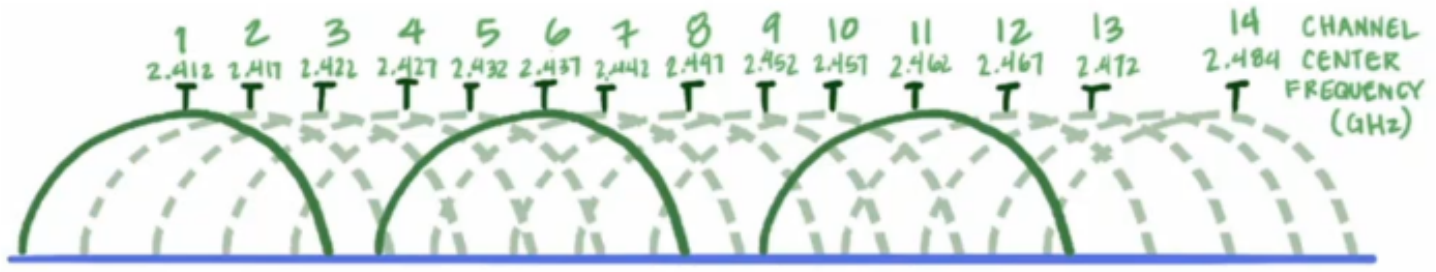
```
ipv6 address autoconfig
```

26 - Design a Basic Wireless Network

- signal sent in distinct frequencies
- final strength measure in RSSI (dBm), -1dBm means you doubled badness of signal
  - ◇ -20,-30dBm very good
  - ◇ -40,-50dBm good
  - ◇ -60,-70(dead)
- SNR = signal to noise ration
  - ◇ you want low SNR

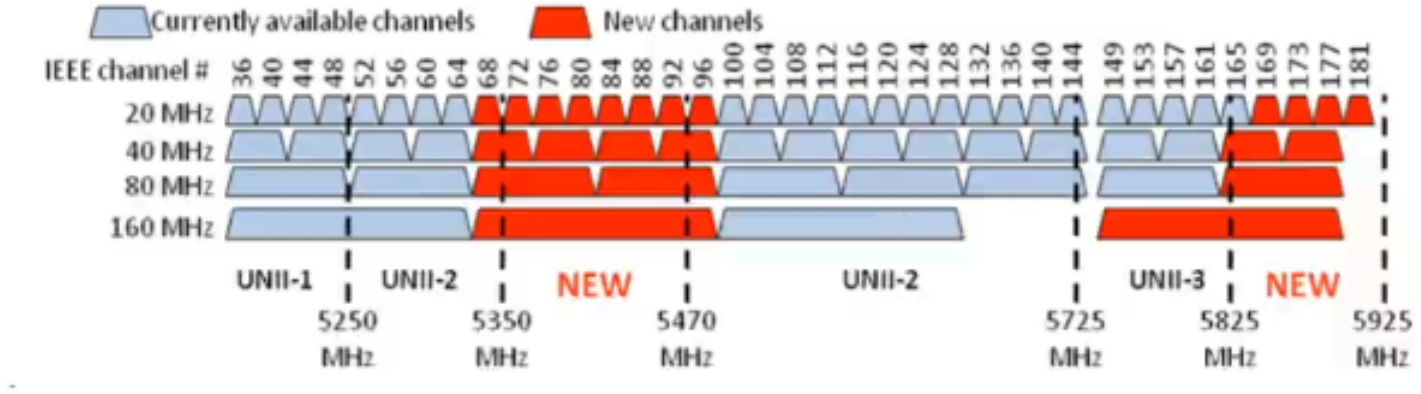
2.4GHz:

- channel=specific frequency range
- non-overlapping are just 1,6,11



Channel	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Center Frequency (MHz)	2412	2417	2422	2427	2432	2437	2442	2447	2452	2457	2462	2467	2472	2484

5GHz:



Basics:
- multiple AP = have non-overlapping channels

WAP
- Basic Service Area (BSA) - signal reach, area covered by WAP
- Basic Service Set (BSS) - parameters of network so client can use it, one WAP broadcast one SSID
- Extended Service Set (ESS) - multiple WAPs broadcasting one SSID
- Service Set Identifier (SSID) - name of network, not unique
- Basic Service Set Identifier (BSSID) - unique name of SSID(can look like MAC address)
- Distribution network - whcih switch/WAP it is connected to distribute wireless
- Roaming - when device needs to change WAP when moving for example

27 - Explain VLANs and Configure VLANs on a Single Switch

- L2 has no QoS
- L2 has no security
- L2 has no segmentation

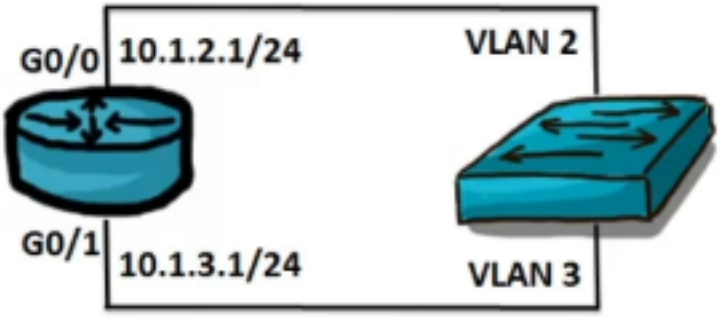
Solution: VLANs
- before that, routers everywhere, servers on every network

VLAN:
• create multiple broadcast domains, subnets, networks
→ PCs can't communicate across VLANs on switch
→ by default all ports are in VLAN1
→ TRUNK connection between switches so they can send all VLANs
⇒ adds 802.1Q tag to header of packet
• extend L2 fabric (stops at router)

• segment and isolate traffic

How to talk between VLANs:

1. Old way: VLAN=subnet, not practical for more VLANs



2. Using Router on a stick (ROAS)
- connect TRUNK of switch to router (single interface)
- create subinterface on router

```
G0/0.2 (vlan 2)
  ip address 10.1.2.1/24
  encapsulation dot1Q 2
G0/0.3 (vlan 3)
  ip address 10.1.3.1/24
  encapsulation dot1Q 3
```

3. Using L3 switch
- L3 switch can route between VLANs at wire speed (ASIC-based: application specific integrated circuit)
- but classic router has more functionality

How to set up VLANs:

Create vlan 10 with name STATIC:
en
show vlan (to see vlan already configured)
conf term
vlan 10 (go to configure vlan10)
name STATIC
exit

Add port fa0/2 to vlan10:
conf term
interface fa0/2
switchport mode access (here you can set up access/auto/trunk)
switchport access vlan 10 (this way you add to specific vlan)

How to configure router on a stick:
- create 2 subinterfaces for vlan 10 and 20
on router:

```
en
config term
inter fa0/1
    no ip address (remove old ip addr, the interface itself has no ip)
    exit
int fa0/1.10
```

```
     encapsulation dot1Q 10 (set vlan 10)
     ip address 10.16.8.1 255.255.255.0
     exit
 int fa0/1.20
     encap dot1Q 20
     ip address 10.16.10.1 255.255.255.0
     exit
```

on switch:

```
 en
 conf term
 int fa0/1
 switchport mode trunk
```

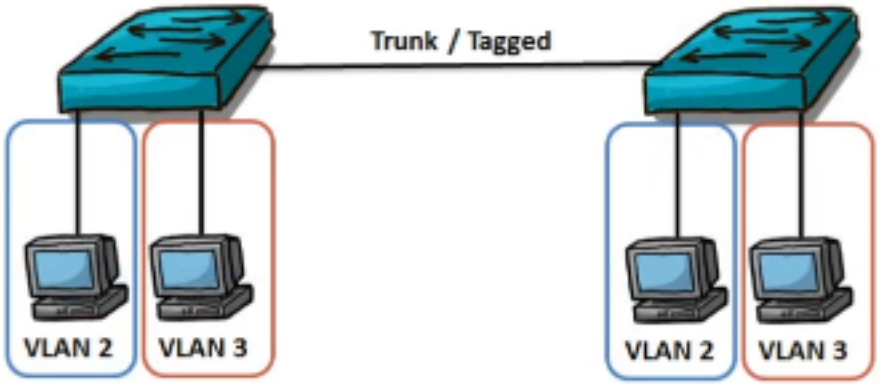
28 - Configure Trunking Between Switches, WAPs, and Servers

- trunking = tagging done between switches
  → manual configuration (turn on trunk and set to nonegotiate - it is faster, no DTP)
    ⇒ even it is manual it sends DTP unless nonegotiate set
  → dynamic configuration (dynamic auto / dynamic desirable)
    ⇒ not good because switch is trying to switch to trunk so I can just plug another switch and they establish trunk - security problem

- Dynamic auto - hoping something happens, not sending any DTP (passive)
- Dynamic desirable - actively sending DTP (lets be trunk, lets be trunk) (active)

- **Dynamic auto -> Dynamic auto**
  ◇ nothing, they stay ACCESS
- **Dynamic auto -> Dynamic desirable**
  ◇ they become TRUNK
- **Dynamic desirable -> Dynamic desirable**
  ◇ they become TRUNK
- **Trunk -> Dynamic auto/Dynamic desirable/Trunk**
  ◇ they become TRUNK
- **Trunk -> Access**
  ◇ broken configuration

Overview of Trunking



- Tagging done between switches with 802.1Q protocol
- Tag removed on exiting access port
- Trunk ports configured on each interface

Best trunk configuration:
```
conf term
interface fa0/10
    switchport mode trunk
    switchport nonegotiate
```

Native VLAN
- trunk ports should receive traffic with tags
- NATIVE VLAN number has to match on both sides!
- if no vlan, they are tagged by **Native VLAN** (usually VLAN1) on the interface they come
- how would that happen?
  → switch originated traffic (CDP, console to switch and telnet elsewhere)
  → pass-through devices
  → virtualized servers

How to set up native vlan:
```
conf term
interface fa0/10
    switchport trunk native vlan 10
```
(you have to change on the other side of trunk as well!)

VTP - Vlan Trunking Protocol
- proprietary, propagation of vlans from 1 switch to others
- it is not a trunking protocol !!!, just works over trunk
- it can go wrong when you connect switch which has not been used for a while and potentially overwrite used VLANs

```
show vtp status
vtp mode transparent
```
(disable vtp on the device)

How to limit VLANs on trunk:
```
show interface trunk
```
(see what interfaces are trunk etc)

```
conf t
int fa2/0/2
    switchport trunk allowed vlan remove 45
```
(this will remove 45 from being sent)
or
```
    switchport trunk allowed vlan 1,10,20,30,40
```

SVI - switch virtual interface on L3 switch
- vlan X - just create vlan X
- interface vlan x - create SVI for vlan x

```
config term
interface vlan 10
    ip address 10.16.0.1 255.255.255.0
    exit
interface vlan 20
    ip address 10.16.2.1 255.255.255.0
    exit
```

29 - Create a Network Diagram with Cisco CDP and LLDP

CDP
- cisco discovery protocol
- layer 2

LLDP
- link layer discovery protocol
- is expendable
- used for PoE negotiation for example
- layer 2

```
show cdp neighbors
show lldp neighbors
show lldp neighbors detail
```

How to set up:
```
lldp run
interface fa2/0/3
    no lldp transmit
```
(will not transmit or receive lldp on specific port)

Creating network diagram:

show ip interface brief (see interfaces, IPs, vlans)
show cdp neighbors (see connected devices on specific local and remote ports)

```
AZ-RT01#show cdp neighbors
Capability Codes: R - Router, T - Trans Bridge, B - Source Route Bridge
                  S - Switch, H - Host, I - IGMP, r - Repeater, P - Phone,
                  D - Remote, C - CVTA, M - Two-port Mac Relay

Device ID        Local Intrfce     Holdtme    Capability  Platform  Port ID
AZ-SW02          Fas 0/0/1          164              S I   WS-C3750- Fas 1/0/1
AZ-SW01          Fas 0/0/0          163              S I   WS-C3750- Fas 2/0/1
```

show cdp entry AZ-SW01 (details to specific device and its IP)
telnet to AZ-SW01 and again same
show cdp neighbors

30 - Design and Configure a Rapid Spanning Tree Protocol (STP) Network

STP - spanning tree protocol

- all about redundancy
- solving loops in network
- sending BPDU -bridge protocol data unit
  → if it fails, unblock blocked port
  → sends every 2s

1. Elect the root
- should be center of network but since it runs by default the oldest (manufactured date) is root for stability
- not always, it still goes through process of ROOT election

2. Find best path to the root
  1) Lowest cost
    1- 10Mbps=100
    2- 100Mbps=19
    3- 1Gbps=4
    4- 2xGbps in LAG=3
  3) Lowest bridge ID
    5- Name, combination of priority and mac
  5) Lowest port number

3. Block whatever is left over

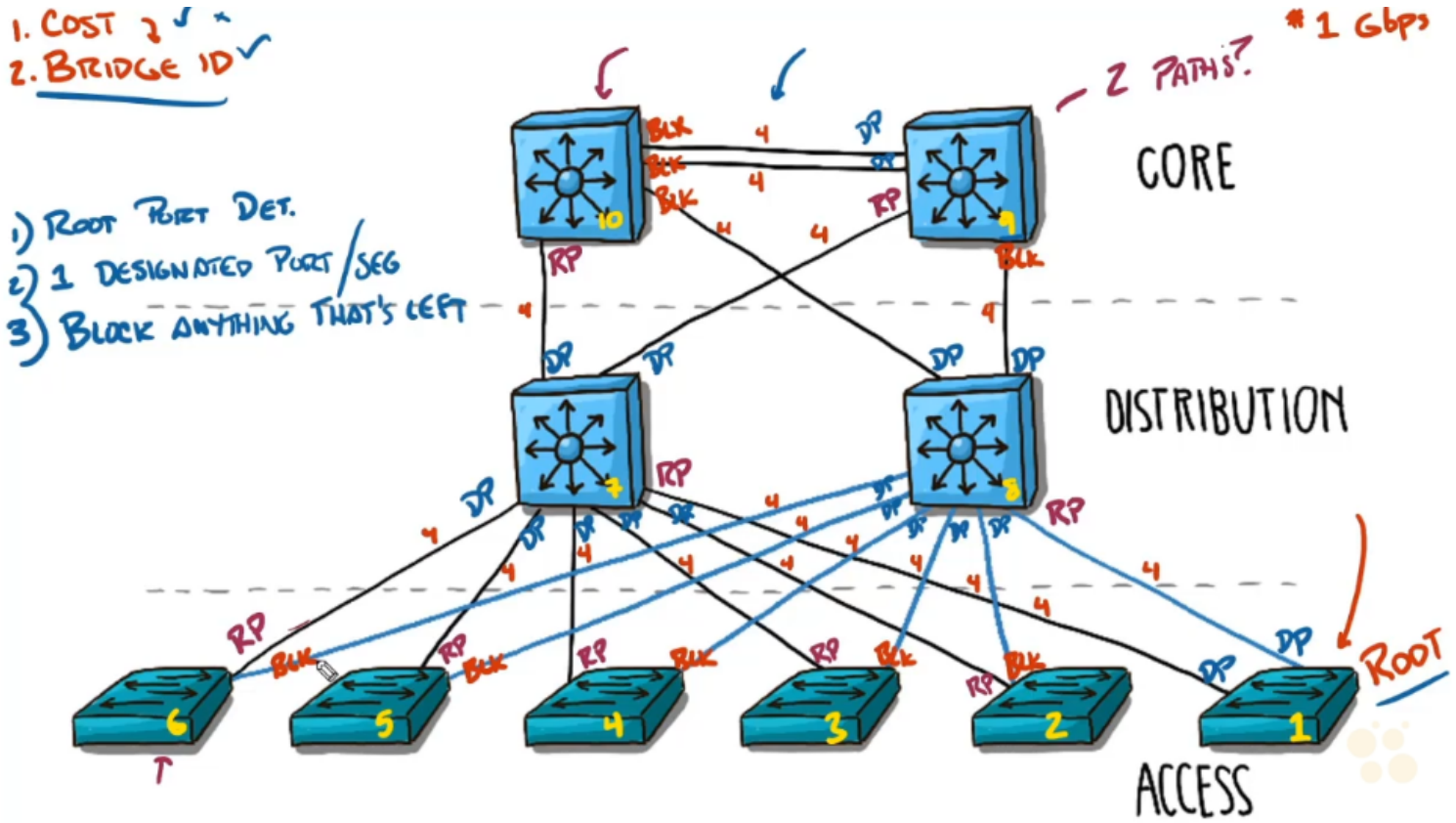
Types:
- Common spanning tree (CST or STP - 802.1D)
  → take long time, 30-50s
- Per-VLAN spanning tree (PVST+)
  → proprietary version by Cisco
  → Root bridges per VLAN, can be different devices
  → overhead, runs STP for every VLAN
- Rapid Spanning Tree (RSTP - 802.1W)
- Per-VLAN rapid spanning tree (PVRST)
- Multiple spanning tree protocol (MSTP - 802.1S)
  → similar like per-VLAN but groups VLANs so more effiicient

How ROOT is elected:
- Bridge priority
  → you can change it to help being selected as Root

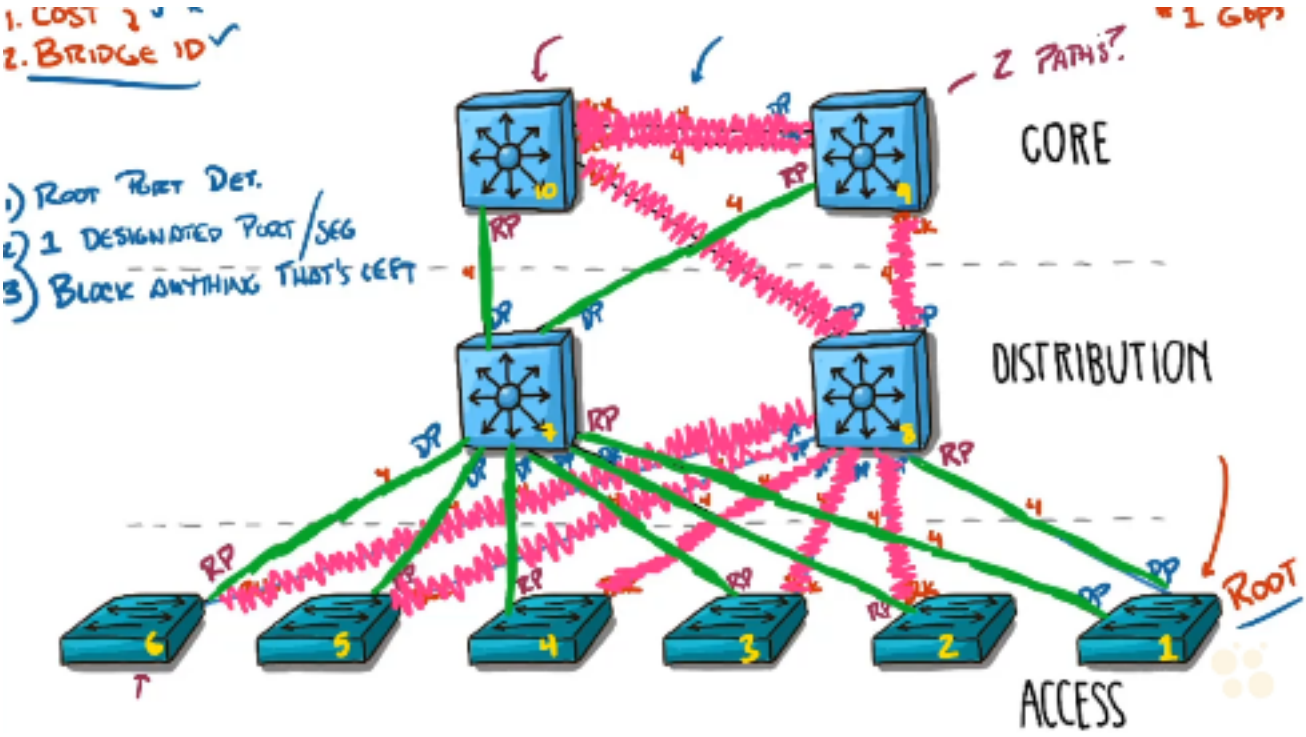
→ default priority out of box is 32768
- Bridge MAC
→ not port mac, but base mac of switch (show version)
- Combined to get BRIDGE ID
→ eg: {32768}priority.{00A0.1101.B011}base_mac
→ Cisco adds VLAN number to priority so default priority for vlan1 is 32769

show spanning-tree (to see who is bridge and info)
- Rood ID = info about root
- Bridge ID = info about me, current switch
- Interface = list of interfaces running STP

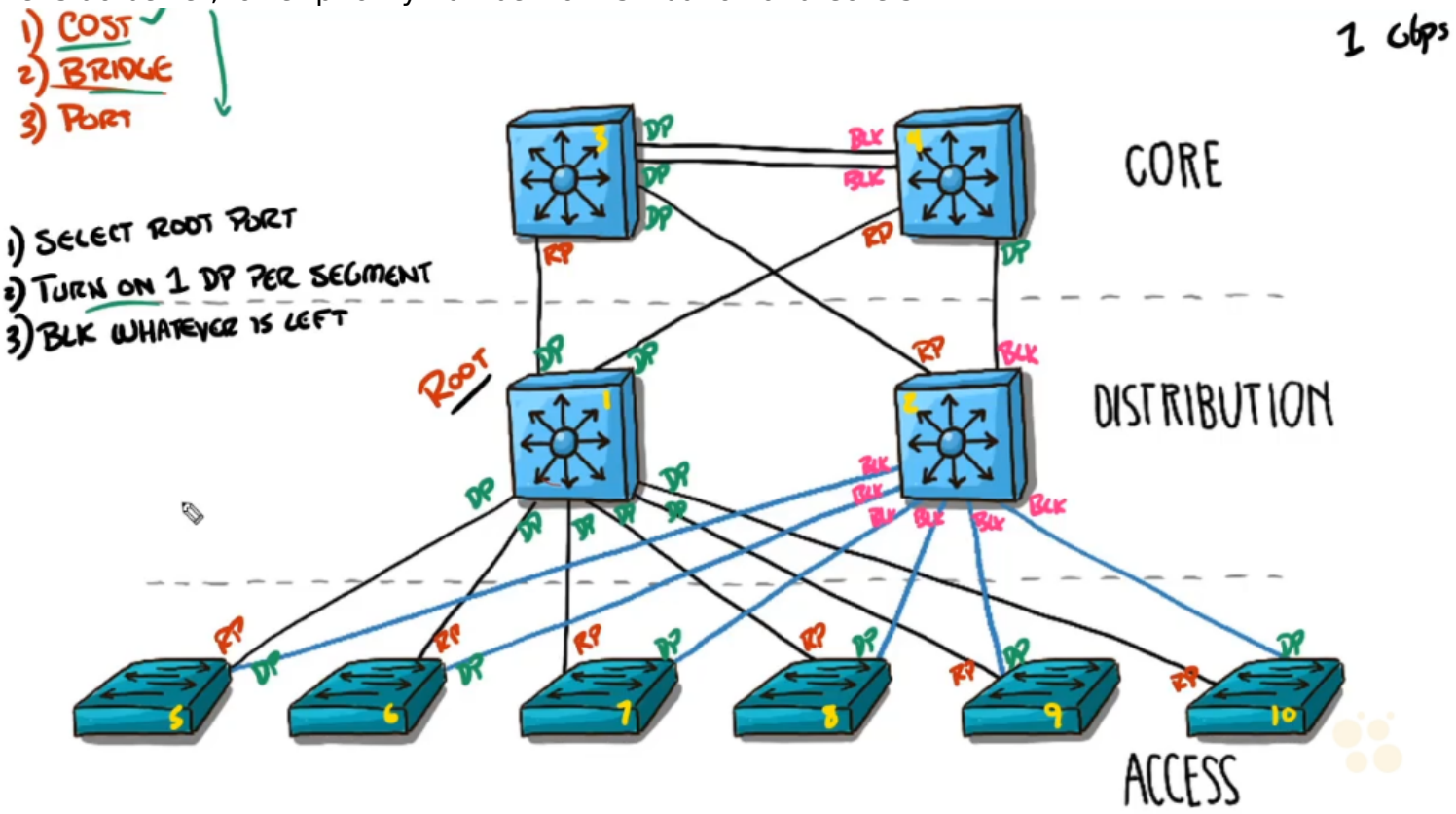
How to determine your STP topology:
1. Find root
1) one is ROOT, the oldest one
2. Root port determination on each switch
1) everyone else start calc to determine path to ROOT to get ROOT PORT on them:
1- cost to get to ROOT
2- if multiple same cost, lowest Bridge ID wins, if same then lowest port number
3. Light up one designated port per segment (segment=link,cable)
1) opposite side of link where RP was determined so you have RP----DP
2) links without RP have nothing so same process: cost to root, bridge id, port # to determine DP
1- So eg for link between SW9 and SW8, SW8 gets DP because its cost to Root is 4
4. Block anything that's left



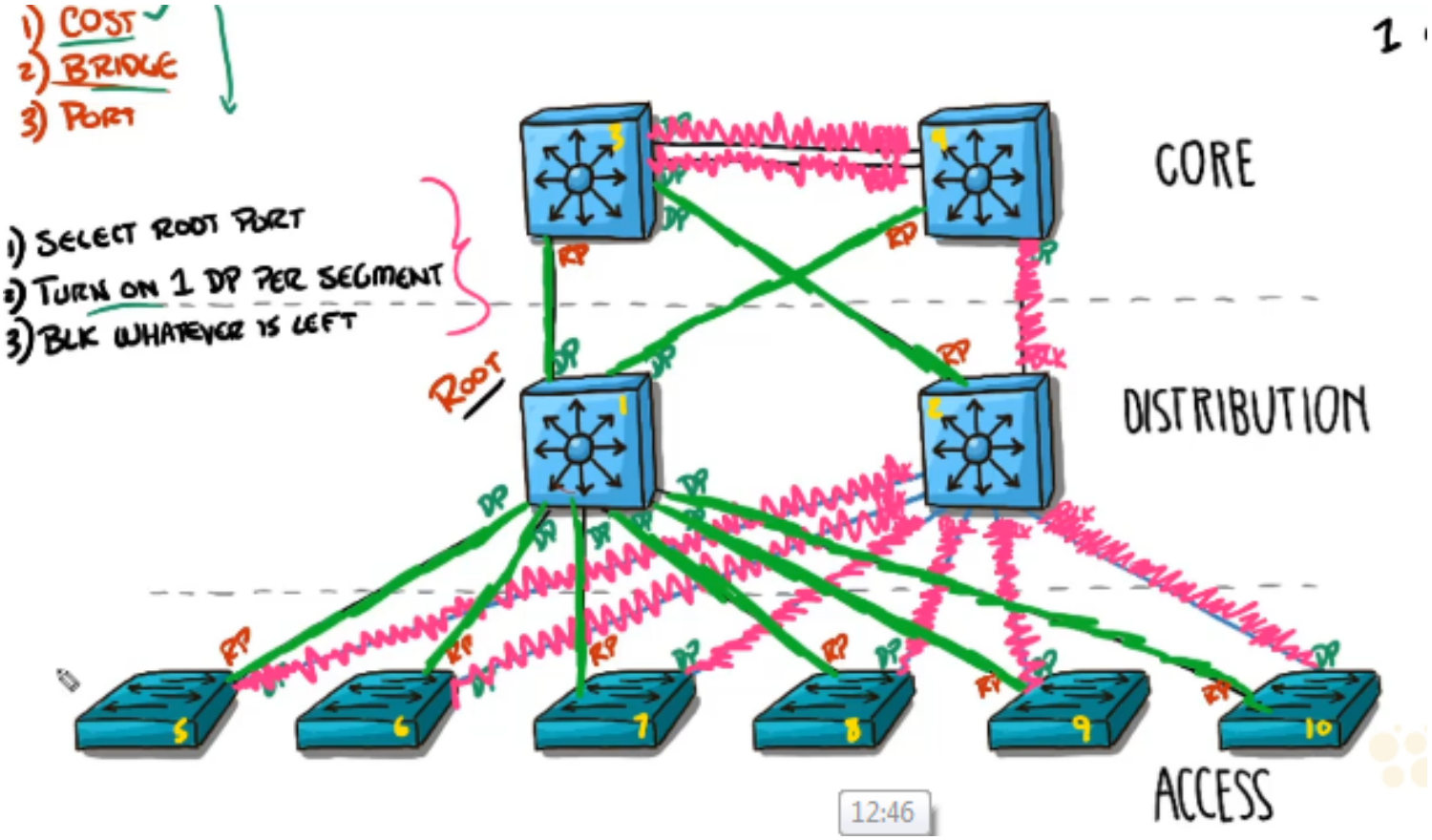
Effectively it will become:
- sw8 is separated from rest and goes through sw1 always (distribution sw through access sw - bottleneck)



Let's do better, lower priority number to Distribution and Core sw:



Effectively become:



conf term
spanning-tree mode rapid-pvst (to enable RSTP)
spanning-tree vlan 1,10,20,30,40 priority <0-61440> (cisco use last 4096 for adding vlan number, allowed just increments of 4096)
spanning-tree vlan 1,10,20,30,40 root primary (auto figure priority so it become root or backup root with secondary param)

31 - Increase Network Capacity Using EtherChannel

EtherChannel

- you load balance over 2,4,8 connections
- symbol=circle over connections
- EtherChannel is Cisco, rest of world is LAG

- port grouper
- load balancer
- super challenging
  → right channel group and protocol
  → right config
    ⇒ speed, duplex, access/trunk
- good idea is to have template and copy paste

Modes
- on (manual)
  → manually turn on
  → for 2 cables you have to configure same all 4 ports

→ best practice is use PAGP or LACP, it is more secure for misconfiguration

- Desirable / auto (PAGP = cisco)

→ Desirable actively trying, Auto is passive and waiting

→ so best is D->A, D->D takes more traffic but ok, A->A never forms LAG

- Active / passive (LACP = industry standard, non cisco)

→ same logic as above

EtherChannel manually:

SW2:

en
conf term
interface range g1/0/1-2
channel-group 1 mode on (unique on the switch and both ports in it)
exit
show etherchannel summary

```
AZ-SW02#show etherchannel summary
Flags:  D - down        P - bundled in port-channel
        I - stand-alone s - suspended
        H - Hot-standby (LACP only)
        R - Layer3      S - Layer2
        U - in use      f - failed to allocate aggregator

        M - not in use, minimum links not met
        u - unsuitable for bundling
        w - waiting to be aggregated
        d - default port


Number of channel-groups in use: 1
Number of aggregators:           1

Group  Port-channel  Protocol    Ports
------+-------------+-----------+-----------------------
1      Po1(SU)           -      Gi1/0/1(P)  Gi1/0/2(P)
```

SW1:

en
conf term
interface range g2/0/1-2
channel-group 1 mode on (unique on the switch and both ports in it, doesn't need to be same on both SW but it is prob better)
exit
show etherchannel summary

```
Group  Port-channel  Protocol    Ports
------+-------------+-----------+-----------------------
1      Po1(SD)           -      Gi2/0/1(D)  Gi2/0/2(D)
```

- interface is down because we didn't shutdown interfaces first before setting it up - one side had LAG and the other not so ERROR

SW2:

conf term
interface range g1/0/1-2
shutdown

no shutdown
interface port-channel 1 (virtual interface for etherchannel, all changes should be here now and not on physical interfaces)
shutdown
no shutdown
exit

- 2xGbit is cost 3 instead of 4

EtherChannel automatically:
SW1:
conf term
interface range g2/0/1-2
channel-group 10 mode active (you can do active everywhere just in case and not care where Act and where Pas)

SW2:
conf term
interface range g1/0/1-2
channel-group 10 mode active (or passive here)

EtherChannel load distribution
- src + stuff
→ based on some src parameters of device which communicate so it will always go on same link for that device
→ not perfectly balanced if dev1 use lot of traffic on link1 and dev2 little on link2
- dst + stuff
→ based on some dst parameters
- src-dst + stuff
→ combination so probably best balanced over both links

conf term
port-channel load-balance src-dst-mac (depends on device what supports and what network needs)

32 - Configure a Basic Cisco Wireless Network using the WLC GUI

CAPWAP - tunnel between WAP and controller
- usually you want EtherChannel between switch and controller for all the traffic

PSK - pre shared key=password
Split-MAC design=client can roam over several APs as it tunnels over CAPWAP

33 - Explain End-To-End IP Communications

MAC never leaves network so in packet Dst.mac=router, not the destination!, but Dst.ip=destination

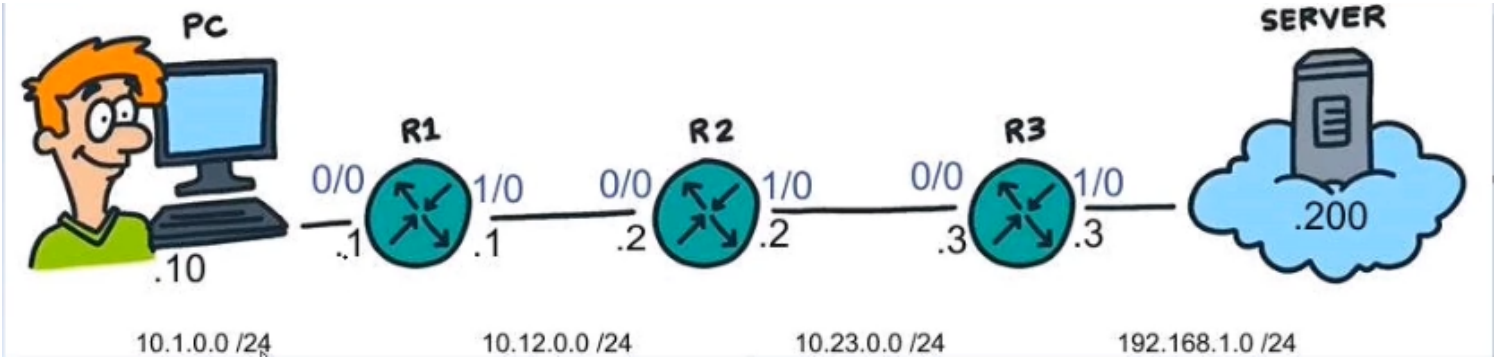
`show spanning-tree vlan 10` (run on all devices to see)

34 - Configure and Verify Cisco IPv4 Static Routes

1. Directly connected network
  1) you know how to get within same subnet
2. Static Routes
3. Dynamic Routing
  1) routers talk to exchange information about routes

Directly connected network

- R3 will be able to ping server but no one else because there are no routes (except for directly connected)



R1:
`show ip route` (shows routes, empty on not configured router)
`show ip int brief` (overview of interfaces, up/down and status)

```
conf term
int gig0/0
    no shut
    ip address 10.1.0.1 255.255.255.0
    exit
int gig1/0
    no shut
    ip address 10.12.0.1 255.255.255.0
```

R2:

```
conf term
int gig0/0
    no shut
    ip address 10.12.0.2 255.255.255.0
    exit
int gig1/0
    no shut
    ip address 10.23.0.2 255.255.255.0
```

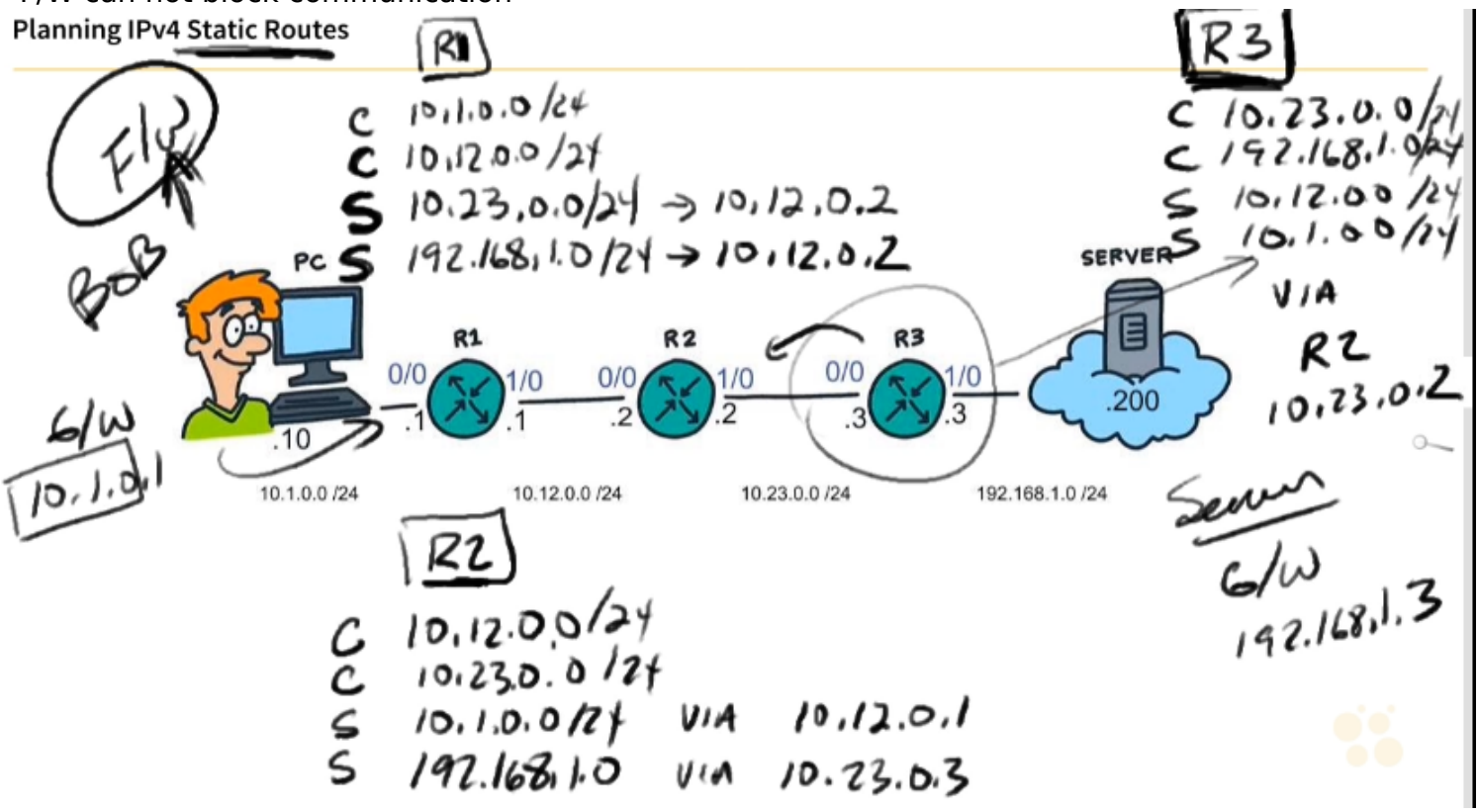
R3:

```
conf term
int gig0/0
    no shut
    ip address 10.23.0.3 255.255.255.0
    exit
int gig1/0
```

```
    no shut
    ip address 192.168.1.3 255.255.255.0
```

Static Routes - network

- set static routes (both directions!)
- both endpoints has to set up correct default GW
- F/W can not block communication



R1:

show ip interface brief (before we start, just to verify IPs)
conf term
do show ip route (to see current routes)

```
    ip route 10.23.0.0 255.255.255.0 10.12.0.2
    ip route 192.168.1.0 255.255.255.0 10.12.0.2
    do show ip route
```

```
     10.0.0.0/8 is variably subnetted, 5 subnets, 2 masks
C       10.1.0.0/24 is directly connected, GigabitEthernet0/0
L       10.1.0.1/32 is directly connected, GigabitEthernet0/0
C       10.12.0.0/24 is directly connected, GigabitEthernet1/0
L       10.12.0.1/32 is directly connected, GigabitEthernet1/0
S       10.23.0.0/24 [1/0] via 10.12.0.2
S    192.168.1.0/24 [1/0] via 10.12.0.2
```

R2:

show ip interface brief (before we start, just to verify IPs)
conf term
do show ip route (to see current routes)

```
    ip route 10.1.0.0 255.255.255.0 10.12.0.1
    ip route 192.168.1.0 255.255.255.0 10.23.0.3
    do show ip route
```

```
      10.0.0.0/8 is variably subnetted, 5 subnets, 2 masks
S        10.1.0.0/24 [1/0] via 10.12.0.1
C        10.12.0.0/24 is directly connected, GigabitEthernet0/0
L        10.12.0.2/32 is directly connected, GigabitEthernet0/0
C        10.23.0.0/24 is directly connected, GigabitEthernet1/0
L        10.23.0.2/32 is directly connected, GigabitEthernet1/0
S     192.168.1.0/24 [1/0] via 10.23.0.3
```

R3:
show ip interface brief (before we start, just to verify IPs)
conf term
do show ip route (to see current routes)
ip route 10.12.0.0 255.255.255.0 10.23.0.2
ip route 10.1.0.0 255.255.255.0 10.23.0.2
do show ip route

```
      10.0.0.0/8 is variably subnetted, 4 subnets, 2 masks
S        10.1.0.0/24 [1/0] via 10.23.0.2
S        10.12.0.0/24 [1/0] via 10.23.0.2
C        10.23.0.0/24 is directly connected, GigabitEthernet0/0
L        10.23.0.3/32 is directly connected, GigabitEthernet0/0
      192.168.1.0/24 is variably subnetted, 2 subnets, 2 masks
C        192.168.1.0/24 is directly connected, GigabitEthernet1/0
L        192.168.1.3/32 is directly connected, GigabitEthernet1/0
```

- to test it you can ping R3 to R1 using extended ping so you pink farther interfaces
ping ip 10.1.0.1 source 192.168.1.3

Static Routes - host

- point to specific host rather than network
- we can test it with loopback interface

R3:
conf term
int loopback 0
ip address 3.3.3.3 255.255.255.255

R2:
conf term
ip route 3.3.3.3 255.255.255.255 10.23.0.3
do show ip route

```
      3.0.0.0/32 is subnetted, 1 subnets
S        3.3.3.3 [1/0] via 10.23.0.3
      10.0.0.0/8 is variably subnetted, 5 subnets, 2 masks
S        10.1.0.0/24 [1/0] via 10.12.0.1
C        10.12.0.0/24 is directly connected, GigabitEthernet0/0
L        10.12.0.2/32 is directly connected, GigabitEthernet0/0
C        10.23.0.0/24 is directly connected, GigabitEthernet1/0
L        10.23.0.2/32 is directly connected, GigabitEthernet1/0
S     192.168.1.0/24 [1/0] via 10.23.0.3
```

R1:

```
conf term
ip route 3.3.3.3 255.255.255.255 10.12.0.2
```

Static Routes - default route

```
ip route 0.0.0.0 0.0.0.0 <next-hop>
```

R1:

```
conf term
ip route 0.0.0.0 0.0.0.0 10.12.0.2
```

Floating static routes

- if you have more routes to destination it load balance
- so you can't add slow link same as regular route
- you use ADMINISTRATIVE DISTANCE, lower=better
- static routes has AD=1, load balancing between 2 routes with same AD
- just set AD to higher and it will not be used, only as back up when lower AD link goes down

R1:

```
conf term
int ser 3/1
    no shutdown
    ip address 10.13.0.1 255.255.255.0
    clock rate 64000
```

R3:

```
conf term
int ser 3/3
    no shut
    ip address 10.13.0.3 255.255.255.0
```

R1:

```
config term
ip route 192.168.1.0 255.255.255.0 10.13.0.3
do show ip route
```

```
      10.0.0.0/8 is variably subnetted, 7 subnets, 2 masks
C        10.1.0.0/24 is directly connected, GigabitEthernet0/0
L        10.1.0.1/32 is directly connected, GigabitEthernet0/0
C        10.12.0.0/24 is directly connected, GigabitEthernet1/0
L        10.12.0.1/32 is directly connected, GigabitEthernet1/0
C        10.13.0.0/24 is directly connected, Serial3/1
L        10.13.0.1/32 is directly connected, Serial3/1
S        10.23.0.0/24 [1/0] via 10.12.0.2
S     192.168.1.0/24 [1/0] via 10.13.0.3
                     [1/0] via 10.12.0.2
```

```
ip route 192.168.1.0 255.255.255.0 10.13.0.3 55
```
(static route with AD=55)

```
do show ip route
```

```
      10.0.0.0/8 is variably subnetted, 7 subnets, 2 masks
C        10.1.0.0/24 is directly connected, GigabitEthernet0/0
L        10.1.0.1/32 is directly connected, GigabitEthernet0/0
C        10.12.0.0/24 is directly connected, GigabitEthernet1/0
L        10.12.0.1/32 is directly connected, GigabitEthernet1/0
C        10.13.0.0/24 is directly connected, Serial3/1
L        10.13.0.1/32 is directly connected, Serial3/1
S        10.23.0.0/24 [1/0] via 10.12.0.2
S     192.168.1.0/24 [1/0] via 10.12.0.2
```

- the second is not in routing table, it appears only when the one with lower AD goes down

Static routes using Outbound Interface

- good to use only when you have p2p link
- over ethernet you should always include next hop address!

R1:

```
conf term
ip route 192.168.1.3 255.255.255.255 serial 3/1
do show ip route
```

```
      10.0.0.0/8 is variably subnetted, 7 subnets, 2 masks
C        10.1.0.0/24 is directly connected, GigabitEthernet0/0
L        10.1.0.1/32 is directly connected, GigabitEthernet0/0
C        10.12.0.0/24 is directly connected, GigabitEthernet1/0
L        10.12.0.1/32 is directly connected, GigabitEthernet1/0
C        10.13.0.0/24 is directly connected, Serial3/1
L        10.13.0.1/32 is directly connected, Serial3/1
S        10.23.0.0/24 [1/0] via 10.12.0.2
      192.168.1.0/24 is variably subnetted, 2 subnets, 2 masks
S        192.168.1.0/24 [1/0] via 10.12.0.2
S        192.168.1.3/32 is directly connected, Serial3/1
```

Connecting Arizona,Florida,Nevada using Static routes

R1:

```
show ip route
conf term
ip route 10.16.8.0 255.255.248.0 10.16.7.6
ip route 10.16.16.0 255.255.248.0 10.16.7.7
```

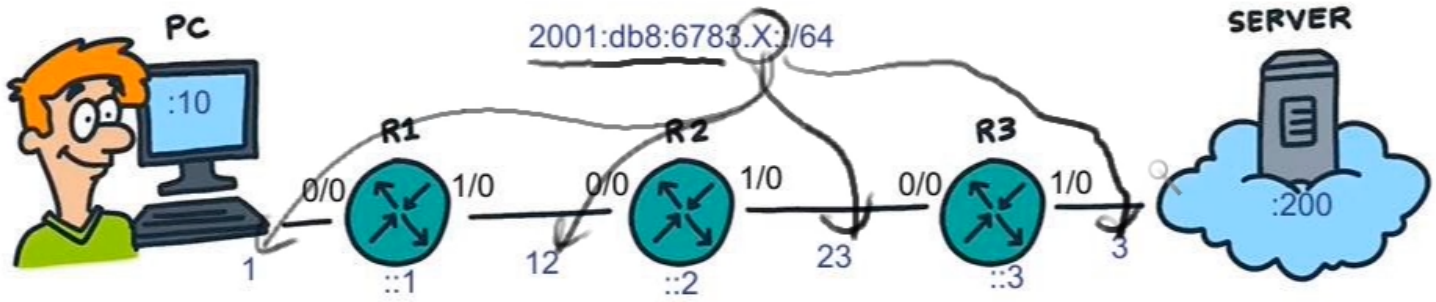
R2:

```
conf term
ip route 0.0.0.0 0.0.0.0 10.16.7.5
```

R3:

```
conf term
ip route 0.0.0.0 0.0.0.0 10.16.7.5
```

35 - Configure and Verify Cisco IPv6 Static Routes



R1:

```
config term
int gig 0/0
    no shut
    ipv6 address fe80::1 link-local
    ipv6 address 2001:db8:6783:1::1/64
int gig 1/0
    no shut
    ipv6 address fe80::1 link-local
    ipv6 address 2001:db8:6783:12::1/64
    do show ipv6 interface brief
```

```
Ethernet0/0                [administratively down/down]
    unassigned
GigabitEthernet0/0         [up/up]
    FE80::1
    2001:DB8:6783:1::1
GigabitEthernet1/0         [up/up]
    FE80::1
    2001:DB8:6783:12::1
```

R2:

```
config term
int gig 0/0
    no shut
    ipv6 address fe80::2 link-local
    ipv6 address 2001:db8:6783:12::2/64
int gig 1/0
    no shut
    ipv6 address fe80::2 link-local
    ipv6 address 2001:db8:6783:23::2/64
    do show ipv6 interface brief
```

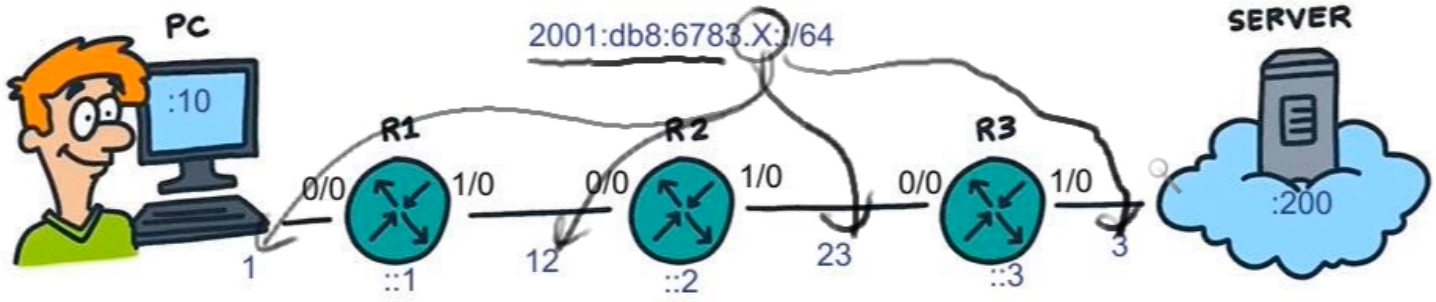
```
Ethernet0/0                [administratively down/down]
    unassigned
GigabitEthernet0/0         [up/up]
    FE80::2
    2001:DB8:6783:12::2
GigabitEthernet1/0         [up/up]
    FE80::2
    2001:DB8:6783:23::2
```

R3:

```
config term
int gig 0/0
    no shut
    ipv6 address fe80::3 link-local
    ipv6 address 2001:db8:6783:23::2/64
int gig 1/0
    no shut
    ipv6 address fe80::3 link-local
    ipv6 address 2001:db8:6783:3::3/64
do show ipv6 interface brief
```

```
Ethernet0/0                [administratively down/down]
    unassigned
GigabitEthernet0/0         [up/up]
    FE80::3
    2001:DB8:6783:23::3
GigabitEthernet1/0         [up/up]
    FE80::3
    2001:DB8:6783:3::3
```

Planning static ipv6 network and default routes



- we can specify 6 static routes total or we can use default
  → R1: default to R2
  → R2: static to ::1, ::3
  → R3: default to R2

R1:

conf term
ipv6 route ::/0 2001:db8:6783:12::2 (default route)
ipv6 unicast-routing (enable ipv6 routing !!!!)

R2:
conf term
ipv6 unicast-routing
ipv6 route 2001:db8:6783:1::/64 2001:db8:6783:12::1
ipv6 route 2001:db8:6783:3::/64 2001:db8:6783:23::3

R3:
conf term
ipv6 route ::/0 2001:db8:6783:23::2
ipv6 unicast-routing

- to test from R1:
ping 2001:db8:6783:3::3 source 2001:db8:6783:1::1

Floating static ipv6 routes

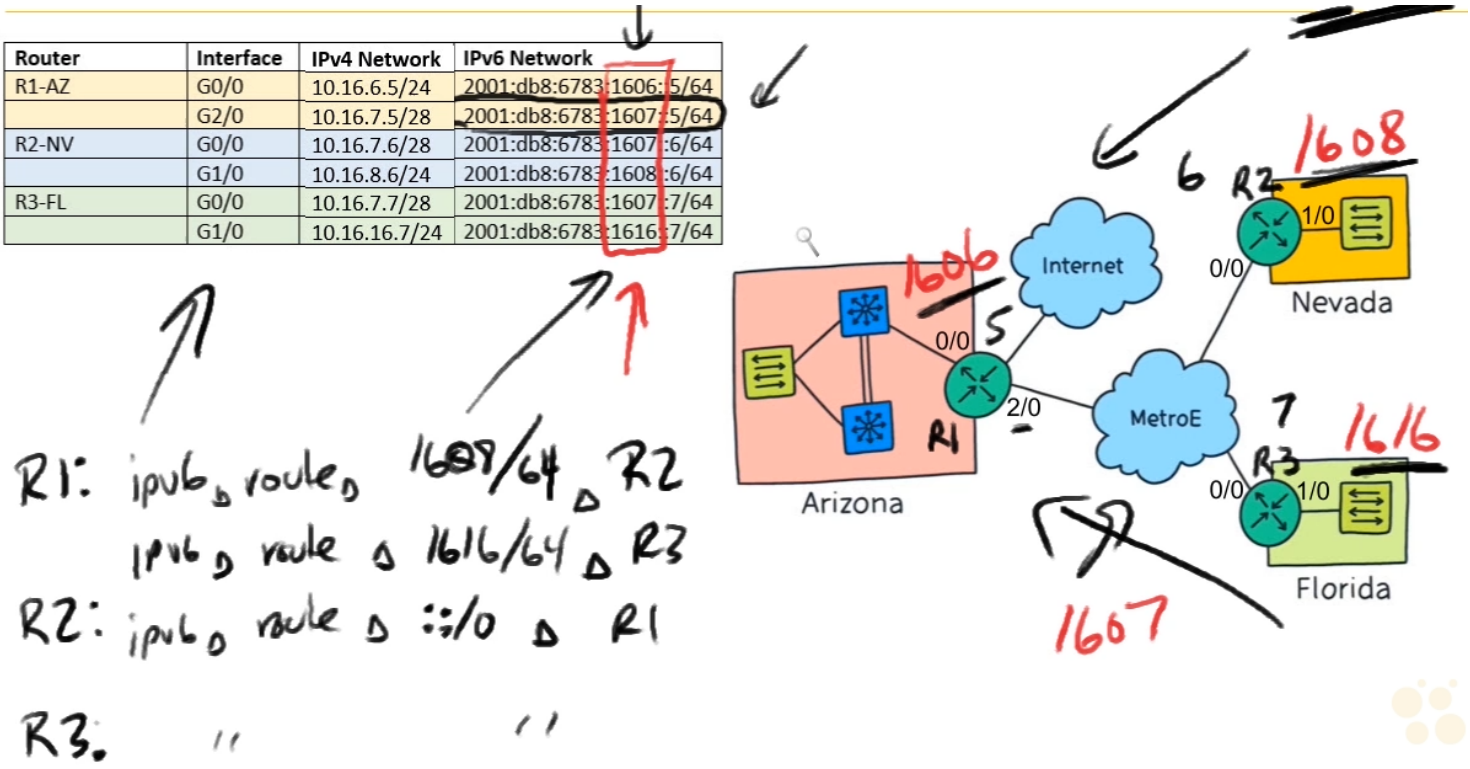
- setup route to lo0 of R1 and floating to l0 on R2, both same ipv6
R1:
conf t
int loopback 0
**   ipv6 address 2001:db8:6783:99:99/128**

R3:
conf t
int loopback 0
**   ipv6 address 2001:db8:6783:99:99/128**

R2:
conf t
ipv6 route 2001:db8:6783:99:99/128 2001:db8:6783:12::1 5 (AD=5 so it is higher than default)
ipv6 route 2001:db8:6783:99:99/128 2001:db8:6783:23::3 6 (AD=6, higher then previous so this one is floating)

Real world case study

Router	Interface	IPv4 Network	IPv6 Network
R1-AZ	G0/0	10.16.6.5/24	2001:db8:6783:1606::5/64
	G2/0	10.16.7.5/28	2001:db8:6783:1607::5/64
R2-NV	G0/0	10.16.7.6/28	2001:db8:6783:1607::6/64
	G1/0	10.16.8.6/24	2001:db8:6783:1608::6/64
R3-FL	G0/0	10.16.7.7/28	2001:db8:6783:1607::7/64
	G1/0	10.16.16.7/24	2001:db8:6783:1616::7/64



36 - Describe Cisco Dynamic IPv4 Routing with OSPF

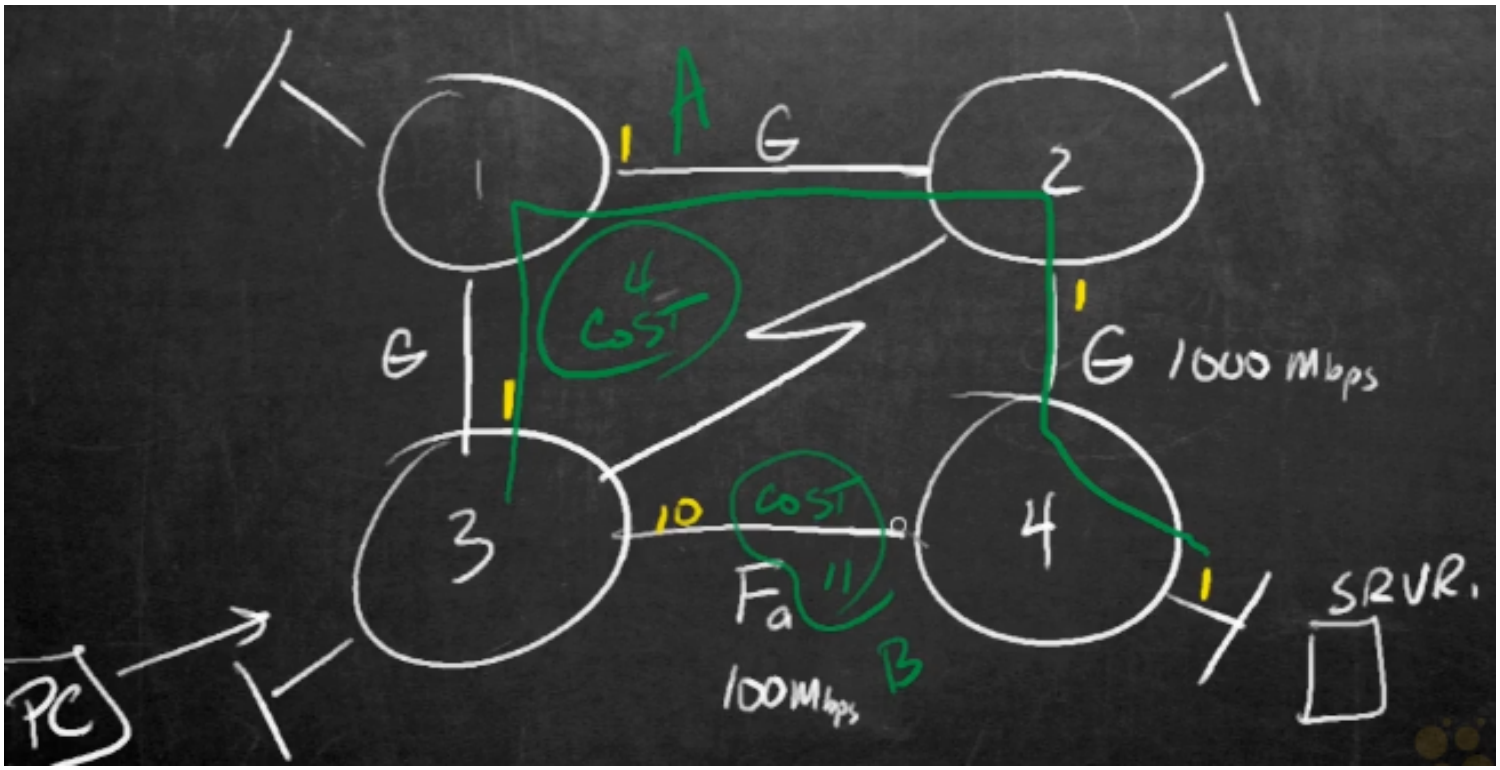
Dynamic routing protocols

- Distance Vector

→ how far it is and what direction to go
→ not that smart, doesn't take cost into consideration, just HOP count to destination
→ RIP, RIPv2, EIGRP

- Link State

→ take into account cost
→ OSPFv2 (for ipv4), OSPFv3(for ipv6)
→ LSA=link state advertisement, each router generates it
→ that means all Routers know entire topology and know entire path
⇒ this way they can choose Shortest Path First



- A is betters, more hops but better cost

OSPF RID (Router ID)

- looks like ip, 32b

Process to determine RID:

1. RID manually configured --> use it
2. if not, use Loopback Address (highest IP) --> use highest IP of all loopbacks
3. if no loopback addrs, IP on active interface (highest IP) --> use highest IP of active interfaces

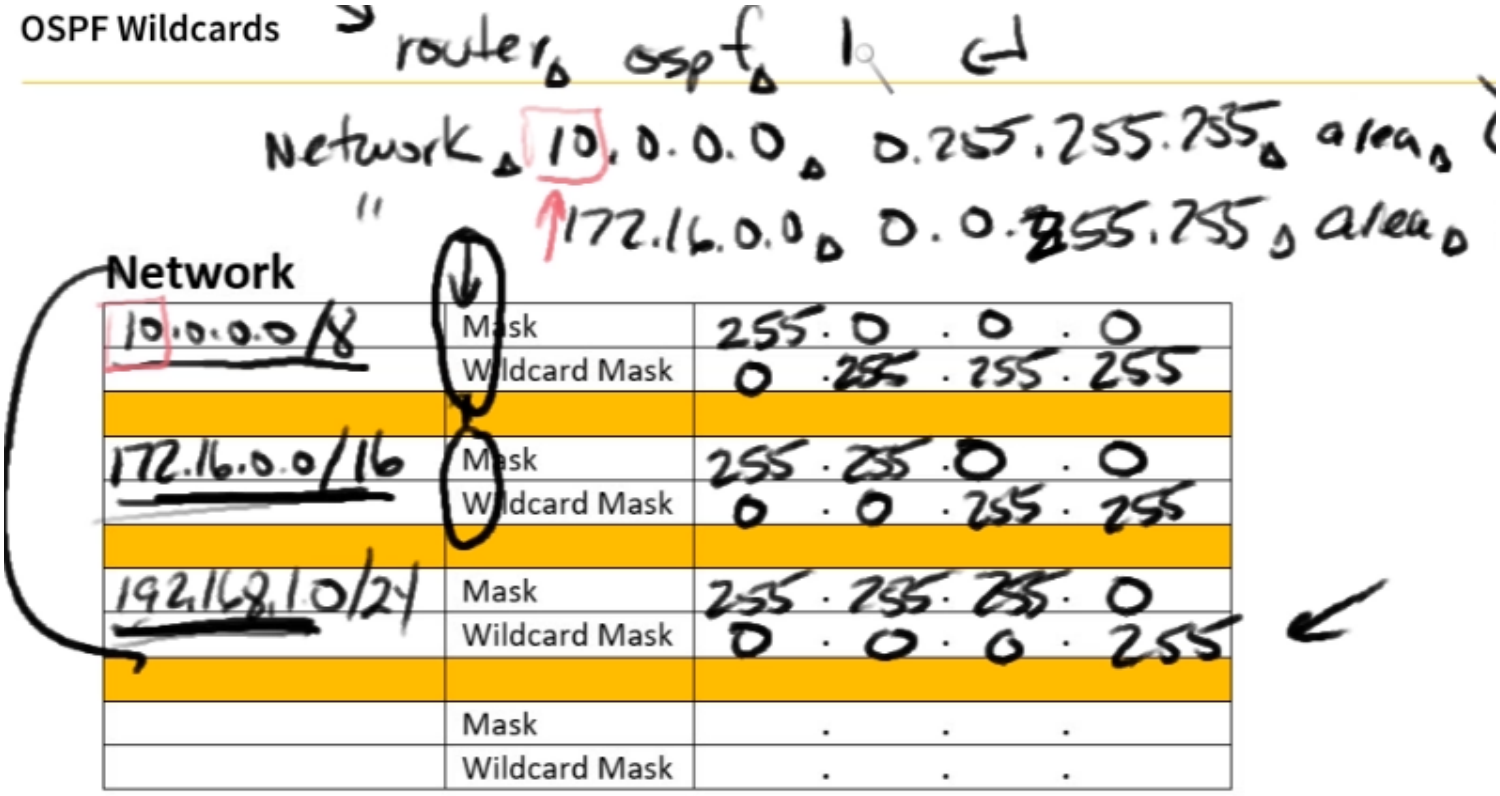
`router ospf 1` (process 1, if you run multiple ospf)
`router-id 7.8.9.0`

OSPF Areas

- we can divide network to areas
- CCNA single area OSPF to understand concept

OSPF Wildcards

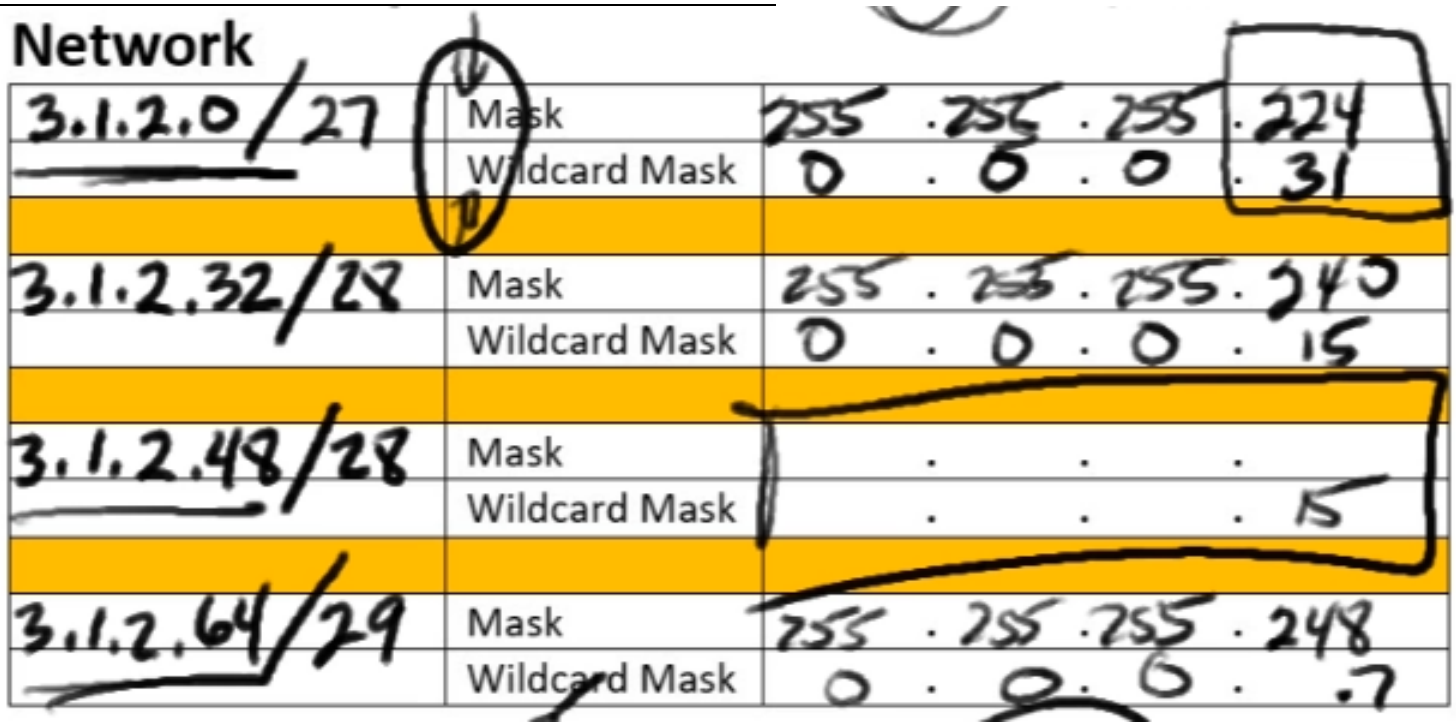
OSPF Wildcards




```
conf t
router ospf 1
    network 10.0.0.0 0.255.255.255 area 0 (match 10.x.x.x)
    network 172.16.0.0 0.0.255.255 area 0
    network 192.168.1.0 0.0.0.255 area 0
    do show ip ospf int brief
```

```
R1(config-router)#do show ip ospf int brief
Interface    PID   Area            IP Address/Mask    Cost  State Nbrs F/C
Lo10         1     0               10.0.0.1/8         1     LOOP  0/0
Lo172        1     0               172.16.0.1/16      1     LOOP  0/0
Lo192        1     0               192.168.1.1/24     1     LOOP  0/0
```

OSPF Wildcards mask for subnets



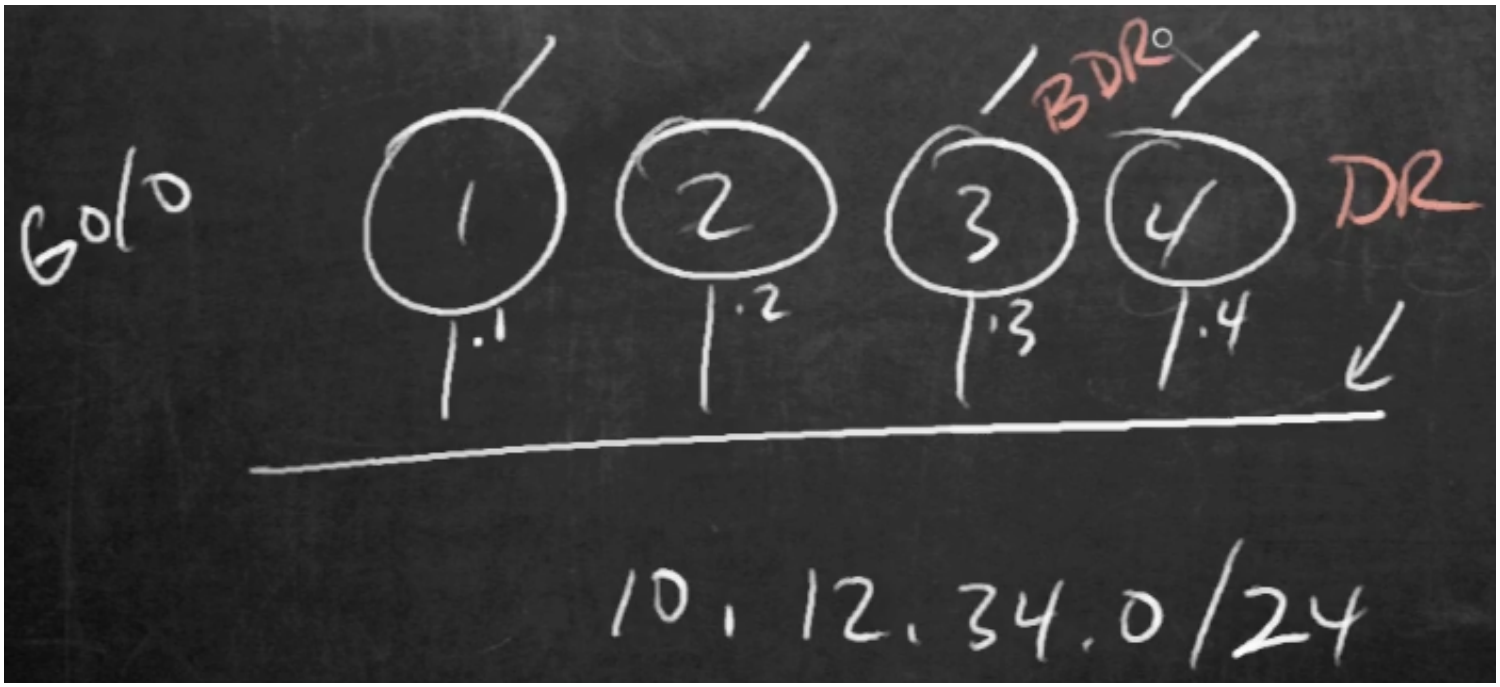

```
conf t
router ospf 1
```

```
   network 3.1.2.0 0.0.0.31 area 0
   network 3.1.2.32 0.0.0.15 area 0
   network 3.1.2.48 0.0.0.15 area 0
   network 3.1.2.64 0.0.0.7 area 0
   do show ip ospf int brief
R1#show ip ospf int brief
Interface    PID   Area            IP Address/Mask    Cost  State Nbrs F/C
Lo0          1     0               3.1.2.1/27         1     LOOP  0/0
Lo32         1     0               3.1.2.33/28        1     LOOP  0/0
Lo48         1     0               3.1.2.49/28        1     LOOP  0/0
Lo64         1     0               3.1.2.65/29        1     LOOP  0/0
```

OSPF DR and BDRs on Broadcast Network

- DR=designated router, BDR=backup designated router
- DR is elected based on priority RID
- BDR is elected based on 2nd highest priority RID
- routers create adjacency to DR and BDR, not full mesh
- all other router are "DROTHER"

show ip ospf int
show ip ospf int brief
show ip ospf neighbor



R1:
show ip ospf int brief

```
R1#show ip ospf int brief
Interface    PID   Area            IP Address/Mask    Cost  State Nbrs
Lo0          1     0               1.1.1.1/32         1     LOOP  0/0
Gi0/0        1     0               10.12.34.1/24      1     DROTH 2/3
```

show ip ospf neighbor

```
R1#show ip ospf neighbor

Neighbor ID     Pri   State           Dead Time   Address         Interface
2.2.2.2           1   2WAY/DROTHER    00:00:37    10.12.34.2      GigabitEthernet0/0
3.3.3.3           1   FULL/BDR        00:00:35    10.12.34.3      GigabitEthernet0/0
4.4.4.4           1   FULL/DR         00:00:35    10.12.34.4      GigabitEthernet0/0
```

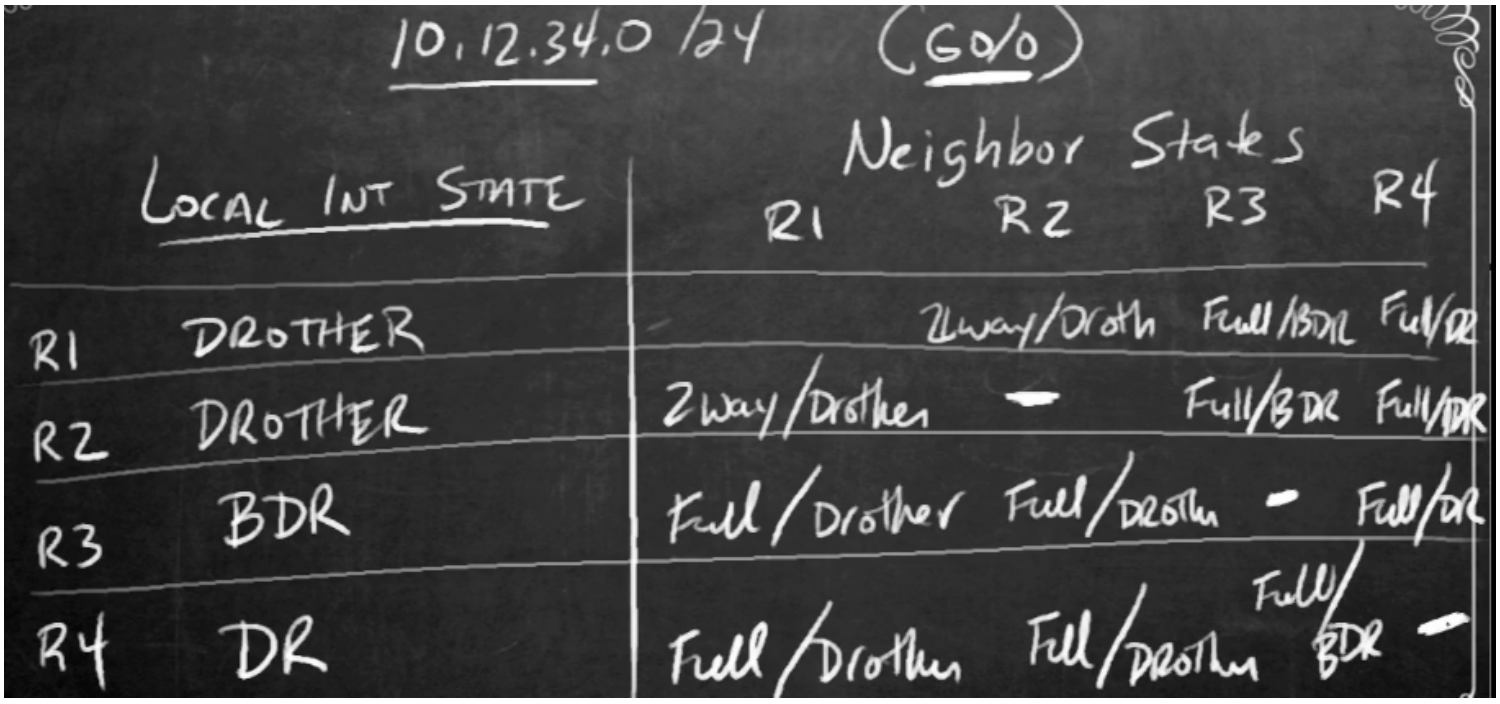
R2:

```
R2#show ip ospf int brief
Interface    PID   Area            IP Address/Mask    Cost  State Nbrs F/C
Lo0          1     0               2.2.2.2/32         1     LOOP  0/0
Gi0/0        1     0               10.12.34.2/24      1     DROTH 2/3
R2#
R2#
R2#
R2#show ip ospf ne
R2#show ip ospf neighbor

Neighbor ID     Pri   State           Dead Time   Address         Interface
1.1.1.1           1   2WAY/DROTHER    00:00:37    10.12.34.1      GigabitEthernet0/0
3.3.3.3           1   FULL/BDR        00:00:33    10.12.34.3      GigabitEthernet0/0
4.4.4.4           1   FULL/DR         00:00:32    10.12.34.4      GigabitEthernet0/0
```

R3:

```
R3#show ip ospf int brief
Interface    PID   Area            IP Address/Mask    Cost  State Nbrs F/C
Lo0          1     0               3.3.3.3/32         1     LOOP  0/0
Gi0/0        1     0               10.12.34.3/24      1     BDR   3/3
R3#
R3#
R3#show ip ospf ne
R3#show ip ospf neighbor

Neighbor ID     Pri   State           Dead Time   Address         Interface
1.1.1.1           1   FULL/DROTHER    00:00:34    10.12.34.1      GigabitEthernet0/0
2.2.2.2           1   FULL/DROTHER    00:00:30    10.12.34.2      GigabitEthernet0/0
4.4.4.4           1   FULL/DR         00:00:39    10.12.34.4      GigabitEthernet0/0
```

Result:



OSPF LSAs and DRs

- LSDB = Link State Database
- LSA 1
  → every router generates LSA1 on its own to advertise what they have going on
  → generated by all
- LSA 2

→ describing specific network and identifying routers connected to the network
→ generated just by DR

Type 1

```
Open Shortest Path First
  OSPF Header
  LS Update Packet
    Number of LSAs: 1
    LSA-type 1 (Router-LSA), len 48
      .000 0000 0000 0001 = LS Age (seconds): 1
      0... .... .... .... = Do Not Age Flag: 0
      Options: 0x22 ((DC) Demand Circuits, (E) External Routing)
      LS Type: Router-LSA (1)
      Link State ID: 3.3.3.3
      Advertising Router: 3.3.3.3
      Sequence Number: 0x80000002
      Checksum: 0x6d18
      Length: 48
      Flags: 0x00
      Number of Links: 2
      Type: Stub     ID: 3.3.3.3        Data: 255.255.255.255 Metric: 1
      Type: Transit  ID: 10.12.34.4     Data: 10.12.34.3      Metric: 1
```

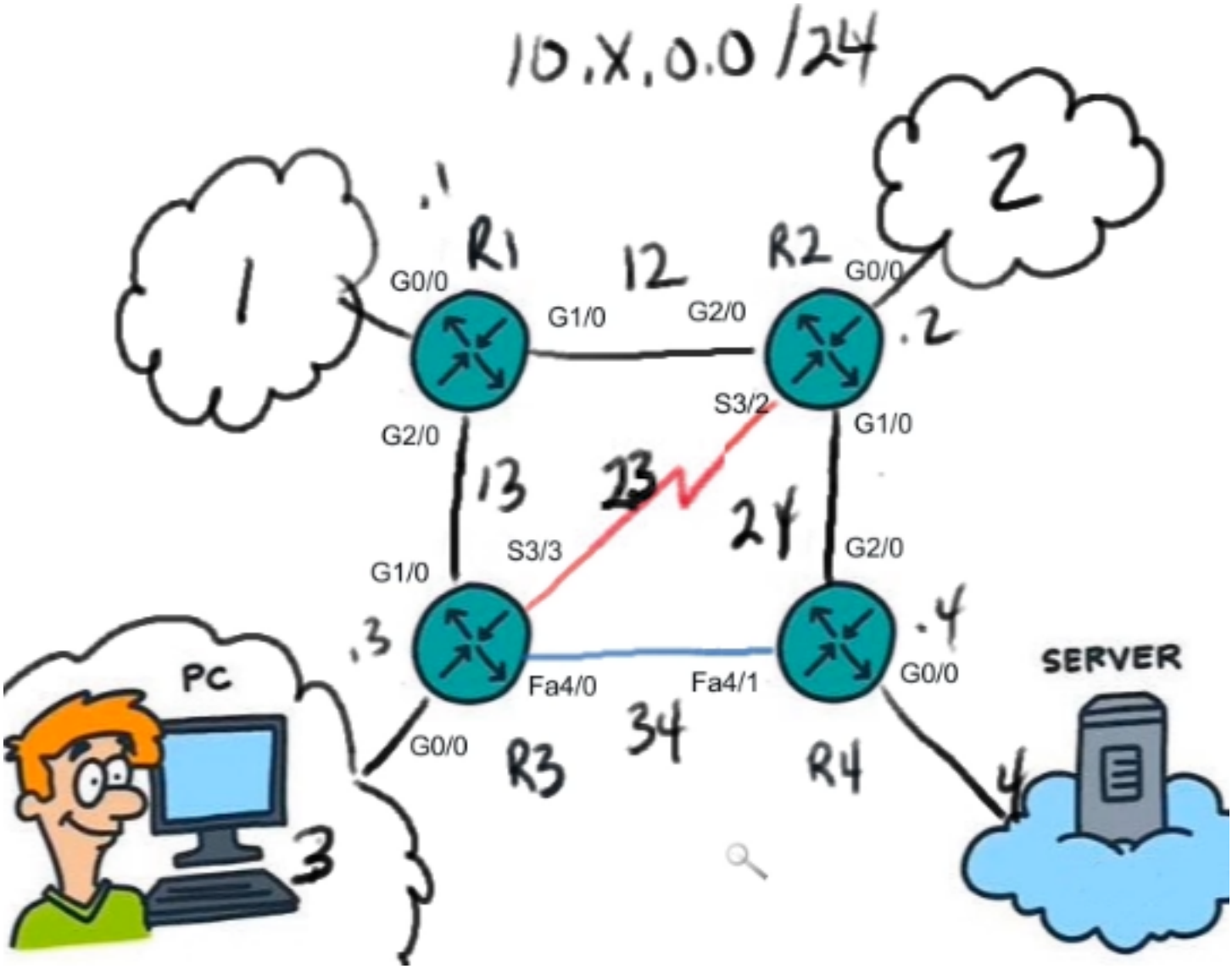
Type 2

```
Open Shortest Path First
  OSPF Header
  LS Update Packet
    Number of LSAs: 1
    LSA-type 2 (Network-LSA), len 40
      .000 1110 0001 0000 = LS Age (seconds): 3600
      0... .... .... .... = Do Not Age Flag: 0
      Options: 0x22 ((DC) Demand Circuits, (E) External Routing)
      LS Type: Network-LSA (2)
      Link State ID: 10.12.34.4
      Advertising Router: 4.4.4.4
      Sequence Number: 0x80000002
      Checksum: 0xb804
      Length: 40
      Netmask: 255.255.255.0
      Attached Router: 4.4.4.4
      Attached Router: 1.1.1.1
      Attached Router: 2.2.2.2
      Attached Router: 3.3.3.3
```

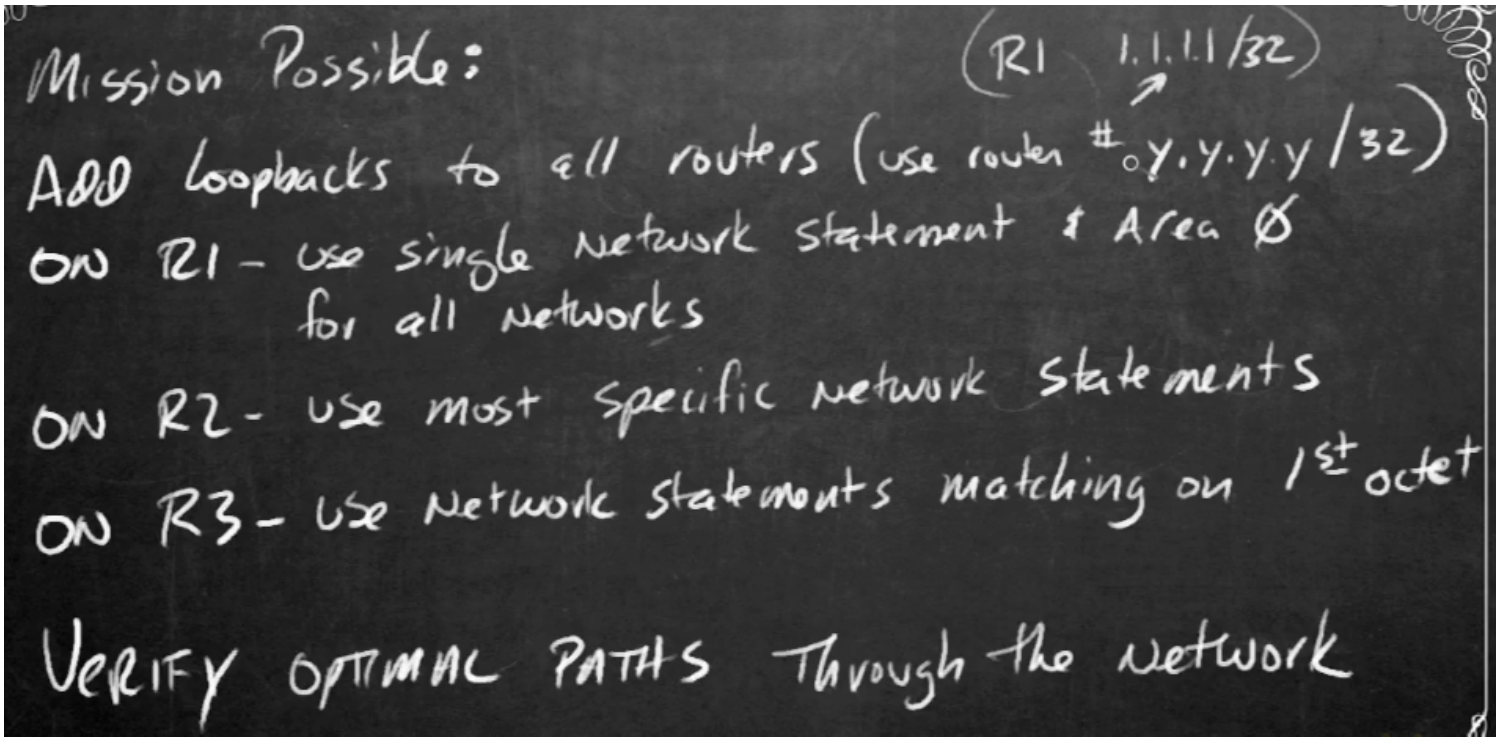
OSPF and P2P Network Types

```
R4#show ip ospf int brief
Interface    PID   Area            IP Address/Mask    Cost  State Nbrs F/C
Lo0          1     0               4.4.4.4/32         1     LOOP  0/0
Se3/0        1     0               10.45.0.4/29       647   P2P   1/1
Gi0/0        1     0               10.12.34.4/24      1     DR    3/3
R4#
R4#
R4#show ip ospf ne
R4#show ip ospf neighbor

Neighbor ID     Pri   State           Dead Time   Address         Interface
5.5.5.5           0   FULL/  -        00:00:31    10.45.0.5       Serial3/0
1.1.1.1           1   FULL/DROTHER    00:00:38    10.12.34.1      GigabitEthernet0/0
2.2.2.2           1   FULL/DROTHER    00:00:36    10.12.34.2      GigabitEthernet0/0
3.3.3.3           1   FULL/BDR        00:00:32    10.12.34.3      GigabitEthernet0/0
```

37 - Implement Cisco Dynamic IPv4 Routing with OSPF



Exercise:



R1:
conf term
int lo 0
ip address 1.1.1.1 255.255.255.255
end
show ip int brief (just to check we configured what we wanted)

R2:
conf term
int lo 0
ip address 2.2.2.2 255.255.255.255
end
show ip int brief (just to check we configured what we wanted)

R3:
conf term
int lo 0
ip address 3.3.3.3 255.255.255.255
end
show ip int brief (just to check we configured what we wanted)

R4:
conf term
int lo 0
ip address 4.4.4.4 255.255.255.255
end
show ip int brief (just to check we configured what we wanted)

~~~~~~

R1:
conf term
router ospf 1
network 0.0.0.0 255.255.255.255 area 0 (any interface with any ip address, their directly connected networks will participate in area 0)

To see if any dynamic routing protocol is running:
~~~~~~

```
R1#show ip protocols
*** IP Routing is NSF aware ***

Routing Protocol is "ospf 1"
  Outgoing update filter list for all interfaces is not set
  Incoming update filter list for all interfaces is not set
  Router ID 1.1.1.1
  Number of areas in this router is 1. 1 normal 0 stub 0 nssa
  Maximum path: 4
  Routing for Networks:
    0.0.0.0 255.255.255.255 area 0
  Routing Information Sources:
    Gateway         Distance      Last Update
  Distance: (default is 110)
```

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

R2:

```
R2#show ip interface brief
Interface              IP-Address      OK? Method Status                Protocol
Ethernet0/0            unassigned      YES NVRAM  administratively down down
GigabitEthernet0/0     10.2.0.2        YES NVRAM  up                    up
GigabitEthernet1/0     10.24.0.2       YES NVRAM  up                    up
GigabitEthernet2/0     10.12.0.2       YES NVRAM  up                    up
Serial3/0              unassigned      YES NVRAM  administratively down down
Serial3/1              unassigned      YES NVRAM  administratively down down
Serial3/2              10.23.0.2       YES NVRAM  up                    up
Serial3/3              unassigned      YES NVRAM  administratively down down
FastEthernet4/0        unassigned      YES NVRAM  administratively down down
FastEthernet4/1        unassigned      YES NVRAM  administratively down down
Loopback0              2.2.2.2         YES NVRAM  up                    up
```

show ip ospf (to see that ospf is not running)

```
config term
router ospf 1
    network 10.2.0.2 0.0.0.0 area 0
    network 10.24.0.2 0.0.0.0 area 0
    network 10.23.0.2 0.0.0.0 area 0
    network 2.2.2.2 0.0.0.0 area 0
end
```

```
R2(config-router)#do show ip ospf int brief
Interface    PID   Area            IP Address/Mask    Cost  State Nbrs
Lo0          1     0               2.2.2.2/32         1     LOOP  0/0
Se3/2        1     0               10.23.0.2/24       64    P2P   0/0
Gi2/0        1     0               10.12.0.2/24       1     BDR   1/1
Gi1/0        1     0               10.24.0.2/24       1     WAIT  0/0
Gi0/0        1     0               10.2.0.2/24        1     DR    0/0
```

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

R3:

```
R3#show ip int brief
Interface              IP-Address      OK? Method Status                Protocol
Ethernet0/0            unassigned      YES NVRAM  administratively down down
GigabitEthernet0/0     10.3.0.3        YES NVRAM  up                    up
GigabitEthernet1/0     10.13.0.3       YES NVRAM  up                    up
GigabitEthernet2/0     unassigned      YES NVRAM  administratively down down
Serial3/0              unassigned      YES NVRAM  administratively down down
Serial3/1              unassigned      YES NVRAM  administratively down down
Serial3/2              unassigned      YES NVRAM  administratively down down
Serial3/3              10.23.0.3       YES NVRAM  up                    up
FastEthernet4/0        10.34.0.3       YES NVRAM  up                    up
FastEthernet4/1        unassigned      YES NVRAM  administratively down down
Loopback0              3.3.3.3         YES NVRAM  up                    up
```

config term
router ospf 1
network 10.0.0.0 0.255.255.255 area 0 (task is to match network on 1st octet)
network 3.0.0.0 0.255.255.255 area 0 (task is to match network on 1st octet, we have just 10.x and not 3.x networks)
end

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

R4:

```
R4(config-router)#do show ip int brief
Interface              IP-Address      OK? Method Status                Protocol
Ethernet0/0            unassigned      YES NVRAM  administratively down down
GigabitEthernet0/0     10.4.0.4        YES NVRAM  up                    up
GigabitEthernet1/0     unassigned      YES NVRAM  administratively down down
GigabitEthernet2/0     10.24.0.4       YES NVRAM  up                    up
Serial3/0              unassigned      YES NVRAM  administratively down down
Serial3/1              unassigned      YES NVRAM  administratively down down
Serial3/2              unassigned      YES NVRAM  administratively down down
Serial3/3              unassigned      YES NVRAM  administratively down down
FastEthernet4/0        unassigned      YES NVRAM  administratively down down
FastEthernet4/1        10.34.0.4       YES NVRAM  up                    up
Loopback0              4.4.4.4         YES NVRAM  up                    up
```

conf term
router ospf 1
network 0.0.0.0 255.255.255.255 area 0 (not specified so we just add everything)

OSPF Link Cost Calculation

- it is old from when we had 10Mbps
- Reference bandwidth 100Mbps divided by link speed
  → 10Mbps = 100/10=10
  → 100Mbps = 100/100=1
  → 1000Mbps = 100/1000=0.1 => round to 1
  → so 100 and 1000 has both cost 1
- how to fix it??
  → set reference B/W to 1000Mbps instead of 100Mbps

R4:
config term
router ospf 1
do show ip ospf (to verify reference B/W)

```
  IETF NSF helper support enabled
  Cisco NSF helper support enabled
  Reference bandwidth unit is 100 mbps
    Area BACKBONE(0)
        Number of interfaces in this a
```
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

auto-cost reference-bandwidth 1000 (you have to put it on all routers)

R1, R2, R3:
config term
router ospf 1
auto-cost reference-bandwidth 1000

R4:
show ip route ospf

```
      1.0.0.0/32 is subnetted, 1 subnets
O        1.1.1.1 [110/3] via 10.24.0.2, 00:17:28, GigabitEthernet2/0
      2.0.0.0/32 is subnetted, 1 subnets
O        2.2.2.2 [110/2] via 10.24.0.2, 00:17:38, GigabitEthernet2/0
      3.0.0.0/32 is subnetted, 1 subnets
O        3.3.3.3 [110/4] via 10.24.0.2, 00:01:19, GigabitEthernet2/0
      10.0.0.0/8 is variably subnetted, 12 subnets, 2 masks
O        10.1.0.0/24 [110/3] via 10.24.0.2, 00:17:28, GigabitEthernet2/0
O        10.2.0.0/24 [110/2] via 10.24.0.2, 00:17:38, GigabitEthernet2/0
O        10.3.0.0/24 [110/4] via 10.24.0.2, 00:01:19, GigabitEthernet2/0
O        10.12.0.0/24 [110/2] via 10.24.0.2, 00:17:38, GigabitEthernet2/0
O        10.13.0.0/24 [110/3] via 10.24.0.2, 00:01:19, GigabitEthernet2/0
O        10.23.0.0/24 [110/648] via 10.24.0.2, 00:00:43, GigabitEthernet2/0
```

- you can see that it send all traffic to gi2/0 rather than fa4/1 and you can see cost to get there

To become neighbours:
- same subnet
- same area
- same HELLO interval (default HELLO 10)
- same authentication

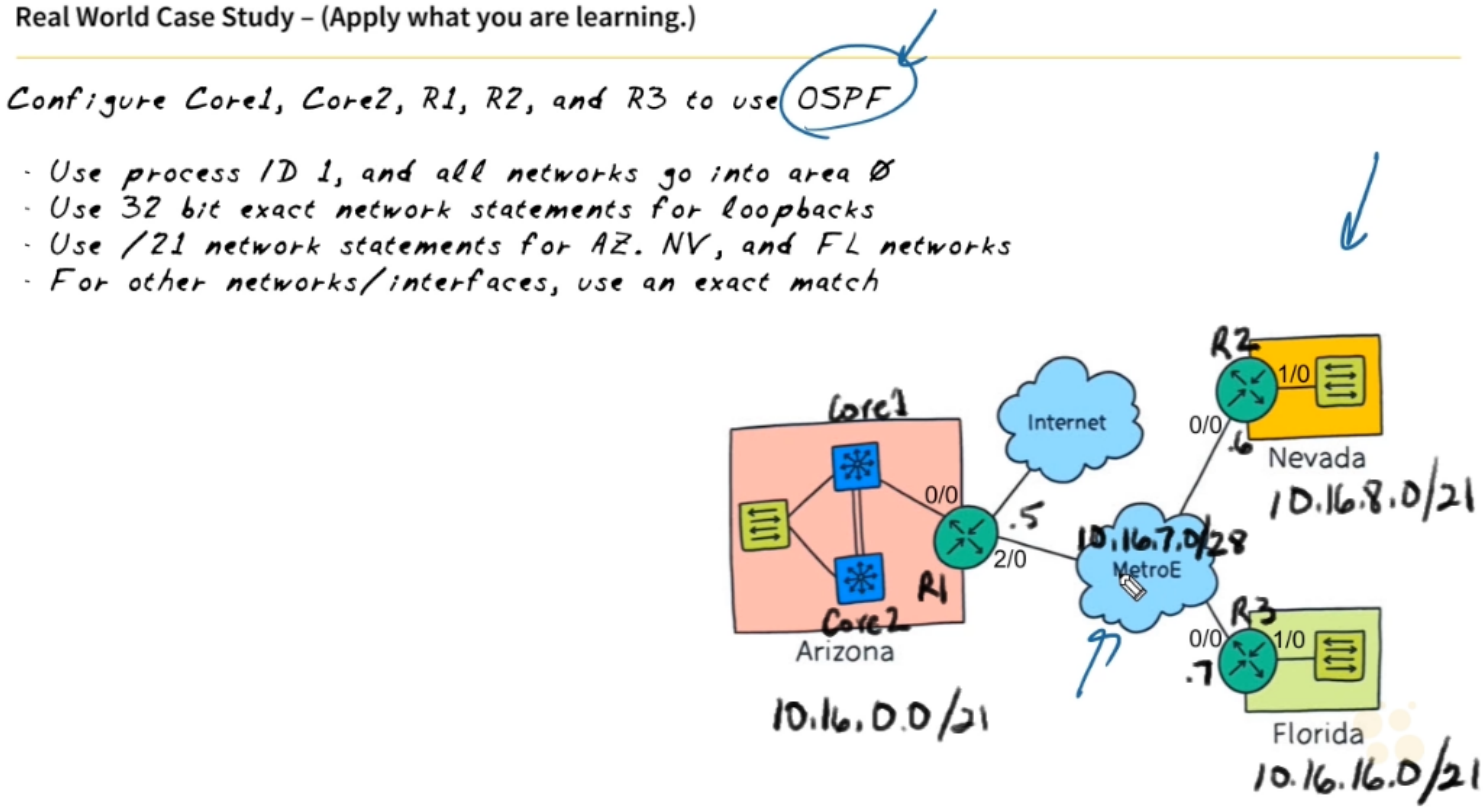
Default routes in OSPF

- no matter how R1 got default route, it needs to share it with rest of the routers
- inject to ospf

router ospf 1
default-information originate always (inject default route always - even if you don't have it)

R1:
show ip ospf (is ospf even running on the router?)
show ip protocols (is any dynamic routing protocol running, incl ospf?)
show ip ospf int brief (to check interfaces etc)
show ip ospf neighbor (to see neighbors)
show ip route (to check if default route is set)
conf term
ip route 0.0.0.0 0.0.0.0 gig 0/0 10.1.0.99 (default route)
router ospf 1
default-information originate always (advertise default route)

Real world case study



/21=255.255.248.0

Core1:
conf term
router ospf 1
do show ip interface brief (to see loopbacks)
network 11.11.11.11 0.0.0.0 area 0 (use 32b exact network statement for loopbacks)

```
Core1(config-router)#do show ip ospf int brief
Interface    PID   Area            IP Address/Mask    Cost  State Nbrs F/C
Lo0          1     0               11.11.11.11/32     1     LOOP  0/0
```

network 10.16.0.0 0.0.7.255 area 0

```
Core1(config-router)#do show ip ospf int brief
Interface    PID   Area            IP Address/Mask    Cost  State Nbrs F/C
Lo0          1     0               11.11.11.11/32     1     LOOP  0/0
Vl140        1     0               10.16.6.1/24       1     WAIT  0/0
Vl130        1     0               10.16.4.1/24       1     WAIT  0/0
Vl120        1     0               10.16.2.1/24       1     WAIT  0/0
Vl110        1     0               10.16.0.1/24       1     WAIT  0/0
```

Core2:
conf term
router ospf 1
do show ip interface brief (to see loopbacks)
network 22.22.22.22 0.0.0.0 area 0 (use 32b exact network statement for loopbacks)
network 10.16.0.0 0.0.7.255 area 0
end

```
Core2#show ip ospf int brief
Interface    PID   Area            IP Address/Mask    Cost  State Nbrs F/C
Lo0          1     0               22.22.22.22/32     1     LOOP  0/0
Vl40         1     0               10.16.6.2/24       1     BDR   1/1
Vl30         1     0               10.16.4.2/24       1     BDR   1/1
Vl20         1     0               10.16.2.2/24       1     BDR   1/1
Vl10         1     0               10.16.0.2/24       1     BDR   1/1
```

R1:

conf term
router ospf 1
network 1.1.1.1 0.0.0.0 area 0 (loopback)
network 10.16.0.0 0.0.7.255 area 0

```
R1-AZ(config-router)#do show ip ospf int br
%OSPF-5-ADJCHG: Process 1, Nbr 22.22.22.22 on GigabitEthernet0/0 from LOADI
R1-AZ(config-router)#do show ip ospf int brief
Interface    PID   Area            IP Address/Mask    Cost  State Nbrs F/C
Lo0          1     0               1.1.1.1/32         1     LOOP  0/0
Gi2/0        1     0               10.16.7.5/28       1     WAIT  0/0
Gi0/0        1     0               10.16.6.5/24       1     DROTH 2/2
```

R2:

conf term
router ospf 1
network 2.2.2.2 0.0.0.0 area 0 (loopback)
network 10.16.8.0 0.0.7.255 area 0
network 10.16.7.0 0.0.0.15 area 0 (for /28 mask)

```
R2-NV(config-router)#do show ip ospf int brief
Interface    PID   Area            IP Address/Mask    Cost  State Nbrs
Lo0          1     0               2.2.2.2/32         1     LOOP  0/0
Gi0/0        1     0               10.16.7.6/28       1     BDR   1/1
Gi1/0        1     0               10.16.8.6/24       1     DR    0/0
```

R3:

conf term
router osp 1
network 3.3.3.3 0.0.0.0 area 0 (loopback)
network 10.16.7.0 0.0.0.15 area 0 (for /28 mask) (metro ethernet)
network 10.16.16.0 0.0.7.255

```
R3-FL#show ip ospf int brief
Interface    PID   Area            IP Address/Mask    Cost  State Nbrs
Lo0          1     0               3.3.3.3/32         1     LOOP  0/0
Gi1/0        1     0               10.16.16.7/24      1     WAIT  0/0
Gi0/0        1     0               10.16.7.7/28       1     DROTH 2/2
```

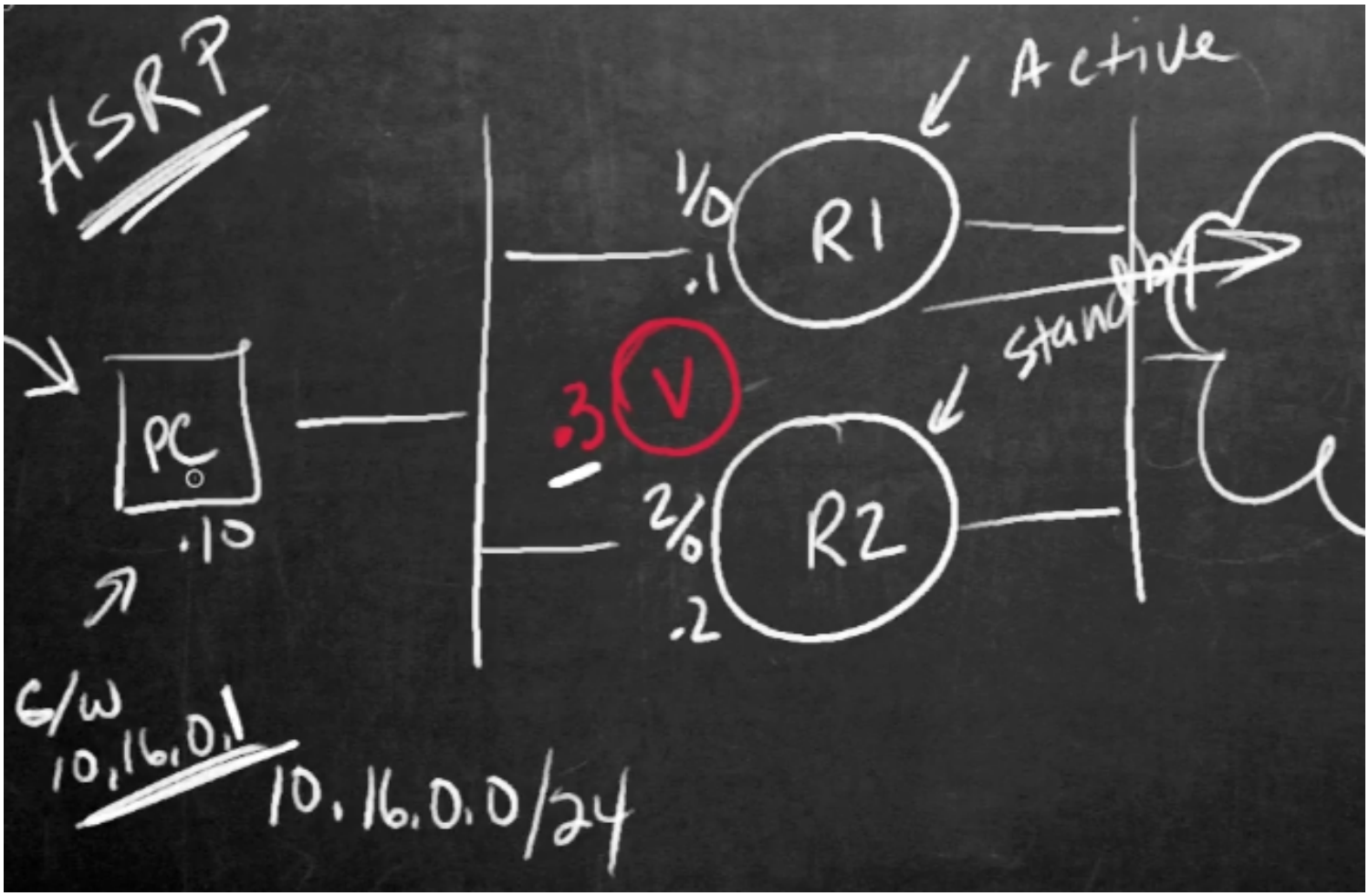
38.0 - Configure and Verify First Hop Redundancy Protocols (FHRP)

First Hop Redundancy Protocols

concept:
- 2 routers, create virtual router interface so it looks like 1 routers for clients

HSRP - Hot Standby Router Protocol

- from Cisco
- 1 active R and 1 standby R
- fault tolerance



R1:

cont term
int gig 1/0
standby 10 ip 10.16.0.3 (group 10, ip)

```
R1#show standby
GigabitEthernet1/0 - Group 10
  State is Active
    2 state changes, last state change 00:00:21
  Virtual IP address is 10.16.0.3
  Active virtual MAC address is 0000.0c07.ac0a (MAC In Use)
    Local virtual MAC address is 0000.0c07.ac0a (v1 default)
  Hello time 3 sec, hold time 10 sec
    Next hello sent in 0.080 secs
  Preemption disabled
  Active router is local
  Standby router is unknown
  Priority 100 (default 100)
  Group name is "hsrp-Gi1/0-10" (default)
```

R2:

conf term
int gig 2/0
standby 10 ip 10.16.0.3

```
R2#show standby brief
                     P indicates configured to preempt.
                     |
Interface   Grp  Pri P State   Active          Standby         Virtual IP
Gi2/0       10   100   Standby 10.16.0.1       local           10.16.0.3
```

VRRP - Virtual Router Redundancy Protocol

- open standard, similar to HSRP
- 1 master R, 1 backup R
- fault tolerance
- MAC is `00-00-5E-00-01-<VRID_hex>`

R1:

```
conf term
int gig 1/0
   vrrp 10 ip 10.16.0.3
```

(group 10, ip of new virtual int)

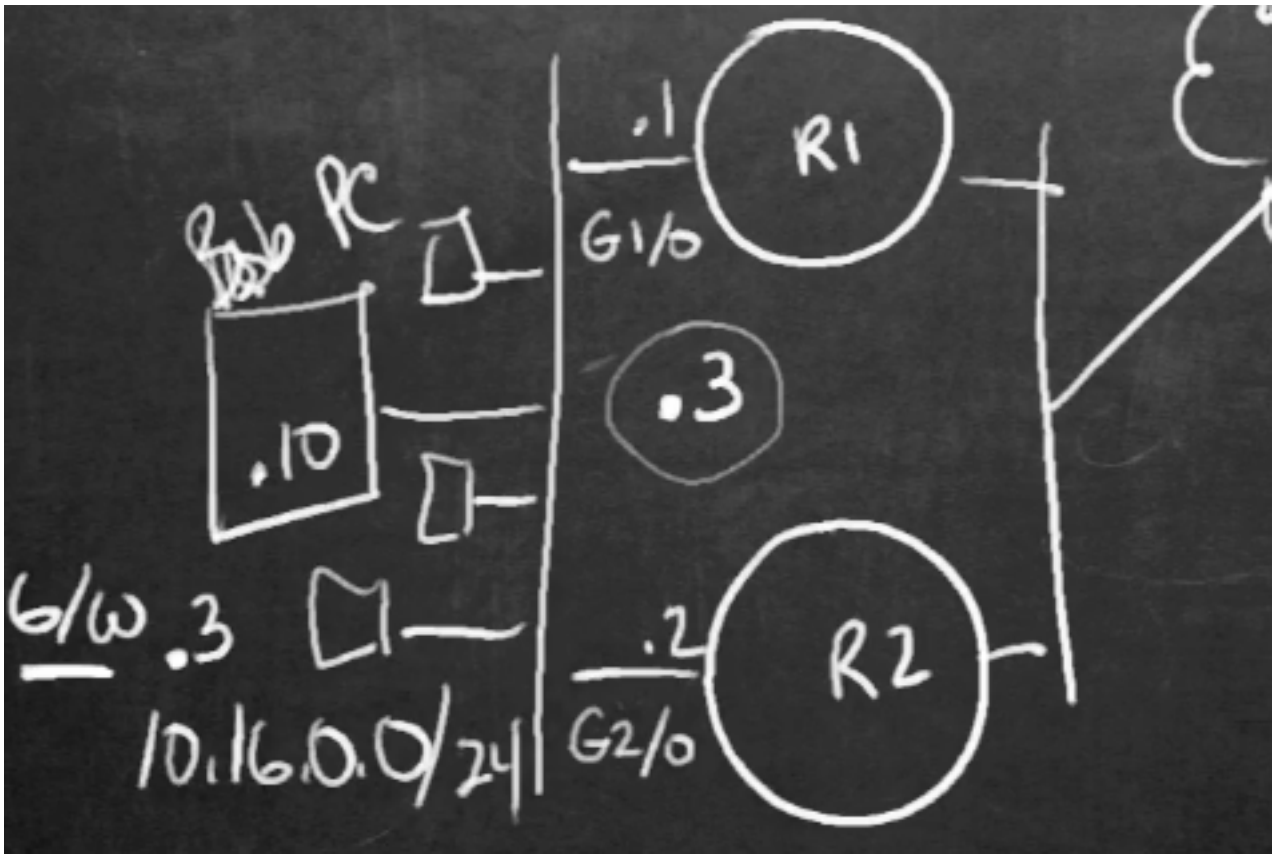
R2:

```
conf term
int gig 2/0
   vrrp 10 ip 10.16.0.3
```

```
R2#show vrrp
GigabitEthernet2/0 - Group 10
  State is Master
  Virtual IP address is 10.16.0.3
  Virtual MAC address is 0000.5e00.010a
  Advertisement interval is 1.000 sec
  Preemption enabled
  Priority is 100
  Master Router is 10.16.0.2 (local), priority is 100
  Master Advertisement interval is 1.000 sec
  Master Down interval is 3.609 sec
```

GLBP - Gateway Load Balancing Protocol

- previous 2 use just one R at a time
- performance wise it would be better load balance over both
- fault tolerance AND load balancing
- how?
  → when client sends ARP, one router (round robin) acts as AVG=active virtual gateway and responds

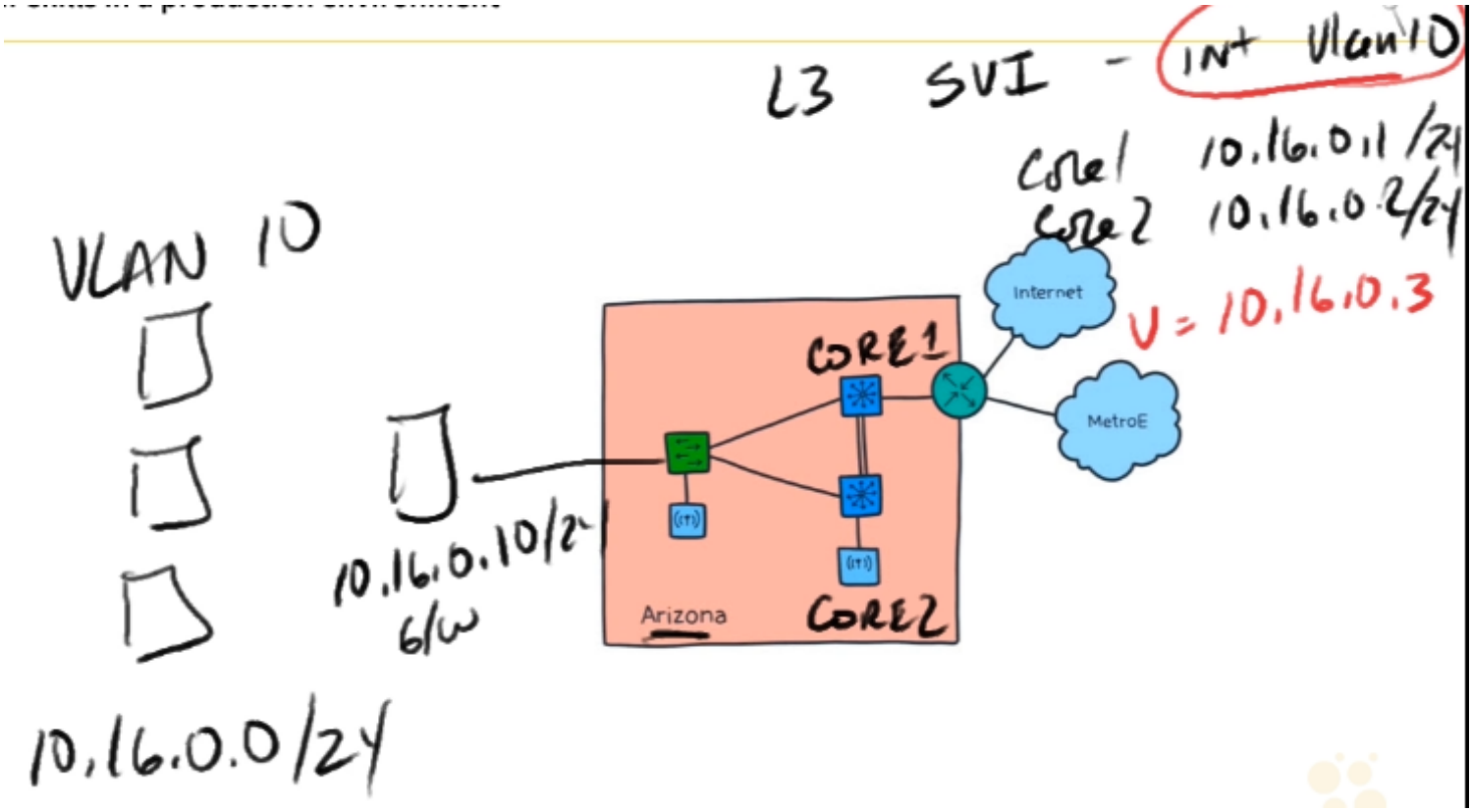


R1:
conf term
int gig 1/0
    glbp 10 ip 10.16.0.3 (group 10, new virtual ip)

R2:
conf term
int gig 2/0
    glbp 10 ip 10.16.0.3
    do show glbp

```
Redirect time 600 sec, forwarder time-out 14400 sec
Preemption disabled
Active is 10.16.0.1, priority 100 (expires in 10.720 sec)
Standby is local
Priority 100 (default)
Weighting 100 (default 100), thresholds: lower 1, upper 100
Load balancing: round-robin
Group members:
  ca01.127c.001c (10.16.0.1)
  ca02.12e4.0038 (10.16.0.2) local
There are 2 forwarders (1 active)
Forwarder 1
  State is Listen
  MAC address is 0007.b400.0a01 (learnt)
  Owner ID is ca01.127c.001c
  Time to live: 14399.872 sec (maximum 14400 sec)
  Preemption enabled, min delay 30 sec
  Active is 10.16.0.1 (primary), weighting 100 (expires in 10.688 sec)
Forwarder 2
  State is Active
    1 state change, last state change 00:00:23
  MAC address is 0007.b400.0a02 (default)
  Owner ID is ca02.12e4.0038
  Preemption enabled, min delay 30 sec
  Active is local, weighting 100
```

Applying it in a production network



Core1:

```
Core1#show ip int brief | exclude unassig
Interface              IP-Address      OK? Method Status                Protocol
Loopback0              11.11.11.11     YES NVRAM  up                    up
Vlan10                 10.16.0.1       YES NVRAM  up                    up
Vlan20                 10.16.2.1       YES NVRAM  up                    up
Vlan30                 10.16.4.1       YES NVRAM  up                    up
Vlan40                 10.16.6.1       YES NVRAM  up                    up
```

conf term
int vlan10
    standby use-bia (HSRP uses interface's burned in address)
    standby 10 ip 10.16.0.3

```
Core1(config-if)#do show standby
Vlan10 - Group 10
  State is Listen
  Virtual IP address is 10.16.0.3
  Active virtual MAC address is unknown (MAC Not In Use)
    Local virtual MAC address is 00dc.d221.800a (bia)
  Hello time 3 sec, hold time 10 sec
  Preemption disabled
  Active router is unknown
  Standby router is unknown
  Priority 100 (default 100)
  Group name is "hsrp-Vl10-10" (default)
```

Core2:
conf term

```
Core2(config)#do show ip int brief | ex unassi
Interface              IP-Address      OK? Method Status                Protocol
Loopback0              22.22.22.22     YES NVRAM  up                    up
Vlan10                 10.16.0.2       YES NVRAM  up                    up
Vlan20                 10.16.2.2       YES NVRAM  up                    up
Vlan30                 10.16.4.2       YES NVRAM  up                    up
Vlan40                 10.16.6.2       YES NVRAM  up                    up
```

```
int vlan 10
    standby use-bia
    standby 10 ip 10.16.0.3
```

38 - Interpret and Describe a Cisco IP Routing Table

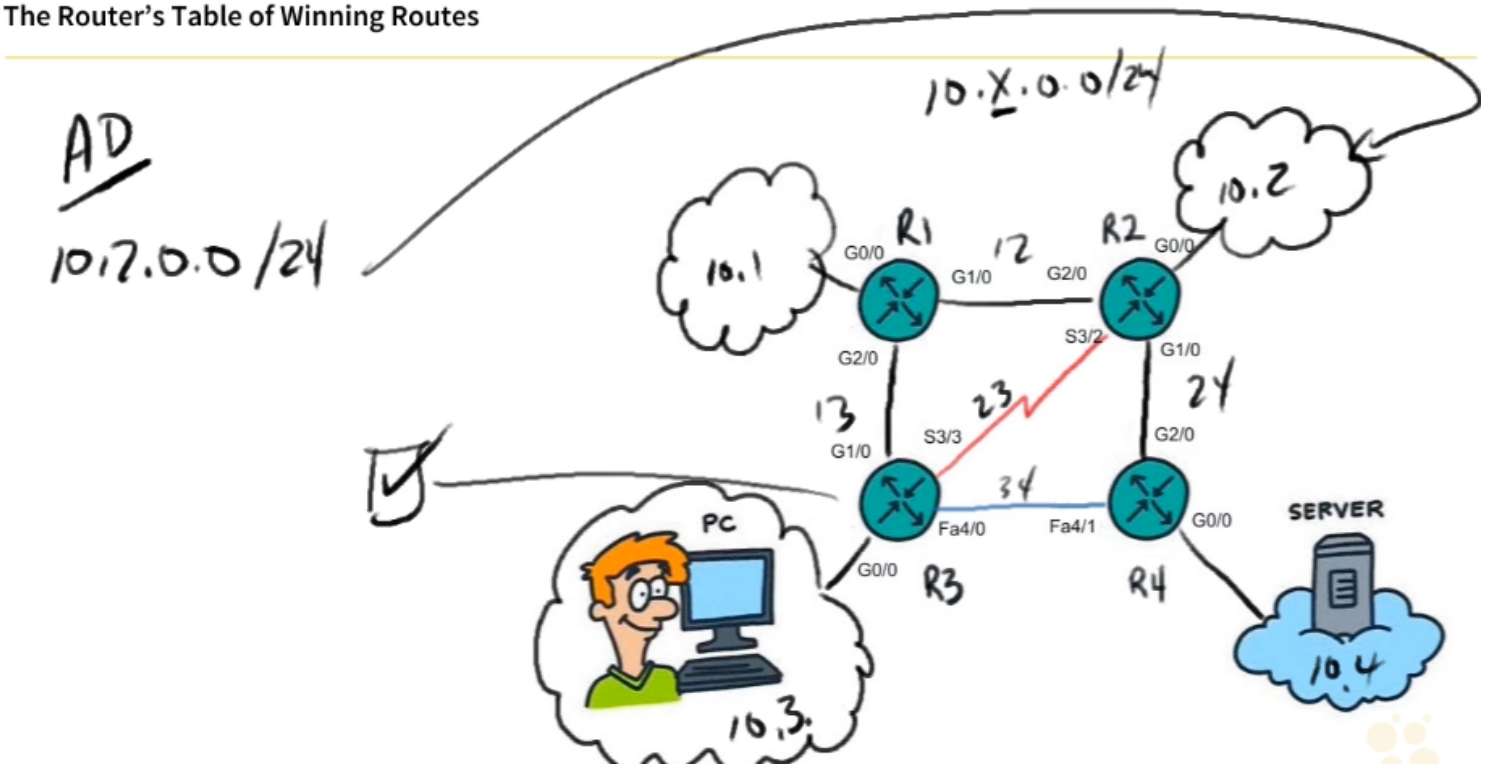
C - connected, when you set up ip on interface
S - static route
R - RIP
O - OSPF

- routing is 2 way street, you need routes there and back
- if router has 2 paths and both have same AD and cost, it is round robin and can use both
- default Administrative Distance
  → OSPF=**110**
  → EIGRP=**90**
  → RIP=**120**
  → static=**1**
  → directly connected=**0**
- [110/648] = AD is 110, metric cost 648, in routing table you see if same, if different just to lower would be there
  → in RIP, metric=hop counts [120/2]=RIP,2 hops

39 - Predict a Cisco Router's IP Forwarding Decisions

"You have to wake up early in the morning to figure what route to put in routing table"
• AM= Administrative distance, Metric(cost) - order of how route is chosen
• when equal, both in routing table and load balance

Choosing route due to lower AD



R3:

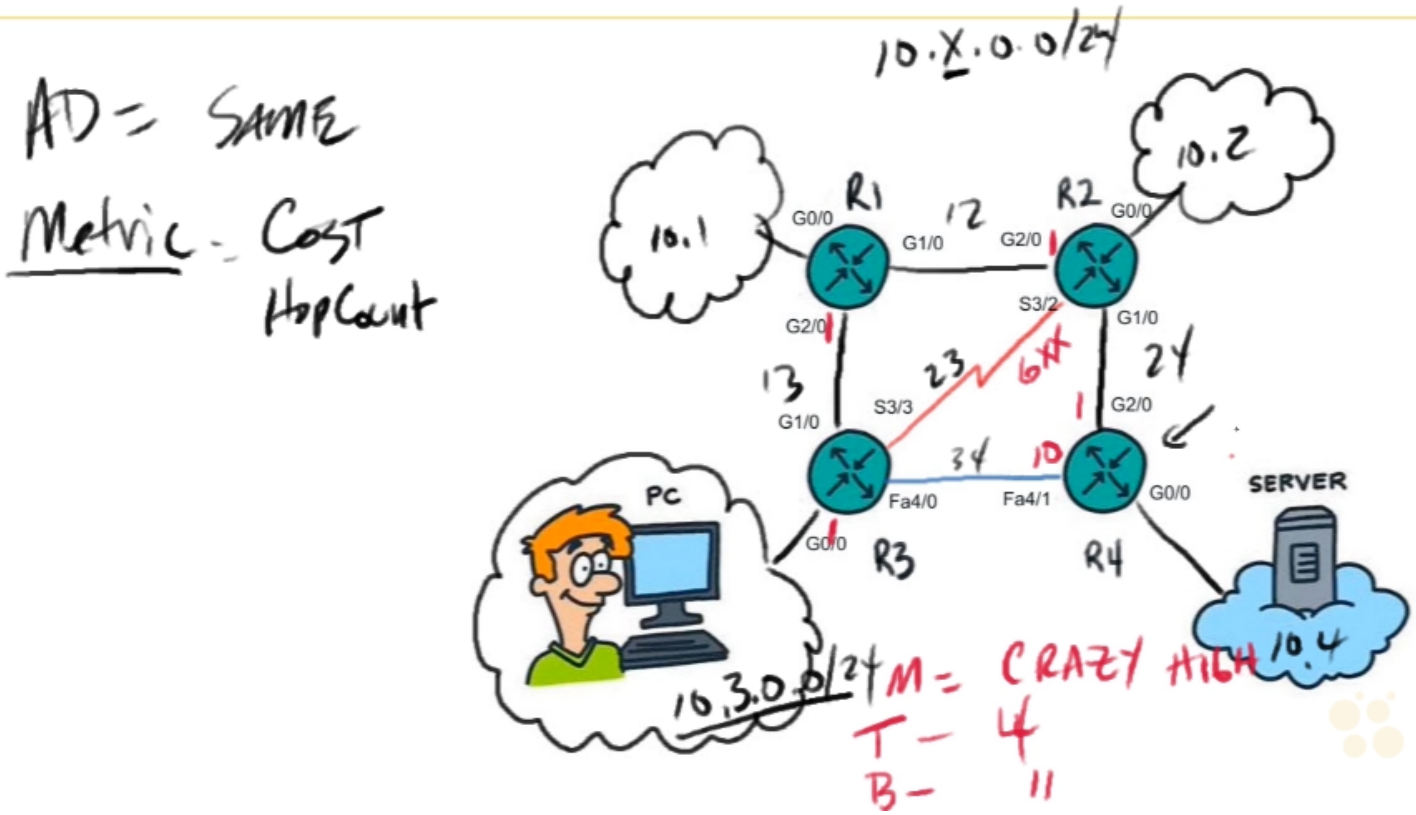
```
R3#show ip route 10.0.0.0
Routing entry for 10.0.0.0/8, 14 known subnets
  Attached (9 connections)
  Variably subnetted with 2 masks
  Redistributing via rip
O        10.1.0.0/24 [110/2] via 10.13.0.1, 00:03:49, GigabitEthernet1/0
O        10.2.0.0/24 [110/3] via 10.13.0.1, 00:03:49, GigabitEthernet1/0
C        10.3.0.0/24 is directly connected, GigabitEthernet0/0
L        10.3.0.3/32 is directly connected, GigabitEthernet0/0
O        10.4.0.0/24 [110/4] via 10.13.0.1, 00:03:44, GigabitEthernet1/0
O        10.12.0.0/24 [110/2] via 10.13.0.1, 00:03:49, GigabitEthernet1/0
C        10.13.0.0/24 is directly connected, GigabitEthernet1/0
L        10.13.0.3/32 is directly connected, GigabitEthernet1/0
C        10.23.0.0/24 is directly connected, Serial3/3
C        10.23.0.2/32 is directly connected, Serial3/3
L        10.23.0.3/32 is directly connected, Serial3/3
O        10.24.0.0/24 [110/3] via 10.13.0.1, 00:03:44, GigabitEthernet1/0
C        10.34.0.0/24 is directly connected, FastEthernet4/0
L        10.34.0.3/32 is directly connected, FastEthernet4/0
```

```
R3#show ip route 10.2.0.2
Routing entry for 10.2.0.0/24
  Known via "ospf 1", distance 110, metric 3, type intra area
  Last update from 10.13.0.1 on GigabitEthernet1/0, 00:09:40 ago
  Routing Descriptor Blocks:
  * 10.13.0.1, from 2.2.2.2, 00:09:40 ago, via GigabitEthernet1/0
      Route metric is 3, traffic share count is 1
```

conf term
ip route 10.2.0.0 255.255.255.0 10.23.0.2 109 (AD=109)
do show ip route 10.0.0.0

```
R3(config)#do show ip route 10.0.0.0
Routing entry for 10.0.0.0/8, 14 known subnets
  Attached (9 connections)
  Variably subnetted with 2 masks
  Redistributing via rip
O        10.1.0.0/24 [110/2] via 10.13.0.1, 00:11:00, GigabitEthernet1/0
S        10.2.0.0/24 [109/0] via 10.23.0.2
C        10.3.0.0/24 is directly connected, GigabitEthernet0/0
L        10.3.0.3/32 is directly connected, GigabitEthernet0/0
O        10.4.0.0/24 [110/4] via 10.13.0.1, 00:10:55, GigabitEthernet1/0
O        10.12.0.0/24 [110/2] via 10.13.0.1, 00:11:00, GigabitEthernet1/0
C        10.13.0.0/24 is directly connected, GigabitEthernet1/0
L        10.13.0.3/32 is directly connected, GigabitEthernet1/0
C        10.23.0.0/24 is directly connected, Serial3/3
C        10.23.0.2/32 is directly connected, Serial3/3
L        10.23.0.3/32 is directly connected, Serial3/3
O        10.24.0.0/24 [110/3] via 10.13.0.1, 00:10:55, GigabitEthernet1/0
C        10.34.0.0/24 is directly connected, FastEthernet4/0
L        10.34.0.3/32 is directly connected, FastEthernet4/0
R3(config)#do show ip route 10.2.0.2
Routing entry for 10.2.0.0/24
  Known via "static", distance 109, metric 0
  Routing Descriptor Blocks:
  * 10.23.0.2
      Route metric is 0, traffic share count is 1
```

Choosing route based on the Metric(Cost, hop cnt)



R4:

```
R4#show ip route 10.0.0.0
Routing entry for 10.0.0.0/8, 14 known subnets
  Attached (6 connections)
  Variably subnetted with 2 masks
  Redistributing via rip
O        10.1.0.0/24 [110/3] via 10.24.0.2, 00:24:07, GigabitEthernet2/0
O        10.2.0.0/24 [110/2] via 10.24.0.2, 00:24:07, GigabitEthernet2/0
O        10.3.0.0/24 [110/4] via 10.24.0.2, 00:24:07, GigabitEthernet2/0
C        10.4.0.0/24 is directly connected, GigabitEthernet0/0
L        10.4.0.4/32 is directly connected, GigabitEthernet0/0
O        10.12.0.0/24 [110/2] via 10.24.0.2, 00:24:07, GigabitEthernet2/0
O        10.13.0.0/24 [110/3] via 10.24.0.2, 00:24:07, GigabitEthernet2/0
O        10.23.0.0/24 [110/648] via 10.24.0.2, 00:24:07, GigabitEthernet2/0
R        10.23.0.2/32 [120/1] via 10.34.0.3, 00:00:01, FastEthernet4/1
R        10.23.0.3/32 [120/1] via 10.24.0.2, 00:00:18, GigabitEthernet2/0
C        10.24.0.0/24 is directly connected, GigabitEthernet2/0
L        10.24.0.4/32 is directly connected, GigabitEthernet2/0
C        10.34.0.0/24 is directly connected, FastEthernet4/1
L        10.34.0.4/32 is directly connected, FastEthernet4/1
```

```
R4#show ip route 10.3.0.3
Routing entry for 10.3.0.0/24
  Known via "ospf 1", distance 110, metric 4, type intra area
  Last update from 10.24.0.2 on GigabitEthernet2/0, 00:24:23 ago
  Routing Descriptor Blocks:
  * 10.24.0.2, from 3.3.3.3, 00:24:23 ago, via GigabitEthernet2/0
      Route metric is 4, traffic share count is 1
```

```
conf term
int gig 2/0
    shutdown
    end
```

```
R4#show ip route 10.0.0.0
Routing entry for 10.0.0.0/8, 11 known subnets
  Attached (4 connections)
  Variably subnetted with 2 masks
  Redistributing via rip
O        10.1.0.0/24 [110/12] via 10.34.0.3, 00:00:03, FastEthernet4/1
O        10.2.0.0/24 [110/13] via 10.34.0.3, 00:00:03, FastEthernet4/1
O        10.3.0.0/24 [110/11] via 10.34.0.3, 00:00:03, FastEthernet4/1
C        10.4.0.0/24 is directly connected, GigabitEthernet0/0
L        10.4.0.4/32 is directly connected, GigabitEthernet0/0
O        10.12.0.0/24 [110/12] via 10.34.0.3, 00:00:03, FastEthernet4/1
O        10.13.0.0/24 [110/11] via 10.34.0.3, 00:00:03, FastEthernet4/1
O        10.23.0.0/24 [110/657] via 10.34.0.3, 00:00:03, FastEthernet4/1
R        10.23.0.2/32 [120/1] via 10.34.0.3, 00:00:21, FastEthernet4/1
C        10.34.0.0/24 is directly connected, FastEthernet4/1
L        10.34.0.4/32 is directly connected, FastEthernet4/1
```

```
conf t
int gig 2/0
    no shut
```

```
R4(config-if)#do show ip route 10.0.0.0
Routing entry for 10.0.0.0/8, 13 known subnets
  Attached (6 connections)
  Variably subnetted with 2 masks
  Redistributing via rip
O        10.1.0.0/24 [110/3] via 10.24.0.2, 00:00:05, GigabitEthernet2/0
O        10.2.0.0/24 [110/2] via 10.24.0.2, 00:00:05, GigabitEthernet2/0
O        10.3.0.0/24 [110/4] via 10.24.0.2, 00:00:05, GigabitEthernet2/0
C        10.4.0.0/24 is directly connected, GigabitEthernet0/0
L        10.4.0.4/32 is directly connected, GigabitEthernet0/0
O        10.12.0.0/24 [110/2] via 10.24.0.2, 00:00:05, GigabitEthernet2/0
O        10.13.0.0/24 [110/3] via 10.24.0.2, 00:00:05, GigabitEthernet2/0
O        10.23.0.0/24 [110/648] via 10.24.0.2, 00:00:05, GigabitEthernet2/0
R        10.23.0.2/32 [120/1] via 10.34.0.3, 00:00:16, FastEthernet4/1
C        10.24.0.0/24 is directly connected, GigabitEthernet2/0
L        10.24.0.4/32 is directly connected, GigabitEthernet2/0
C        10.34.0.0/24 is directly connected, FastEthernet4/1
L        10.34.0.4/32 is directly connected, FastEthernet4/1
```

Concept of using Longest Match in the routing
- router finds longest match in the routing table

R4:

```
conf term
int loop 777
    ip address 192.168.1.1 255.255.255.0
```

R1:

```
conf term
ip route 192.168.0.0 255.255.0.0 10.12.0.2
do show ip route
  S     192.168.0.0/16 [1/0] via 10.12.0.2
  R     192.168.1.0/24 [120/2] via 10.13.0.3, 00:00:21, GigabitEthernet2/0
                       [120/2] via 10.12.0.2, 00:00:09, GigabitEthernet1/0
```

(192.168.1.1 could match both but will match second because it uses longest match)

```
R1(config)#do show ip route 192.168.1.1
Routing entry for 192.168.1.0/24
  Known via "rip", distance 120, metric 2
  Redistributing via rip
  Last update from 10.13.0.3 on GigabitEthernet2/0, 00:00:00 ago
  Routing Descriptor Blocks:
    10.13.0.3, from 10.13.0.3, 00:00:00 ago, via GigabitEthernet2/0
      Route metric is 2, traffic share count is 1
  * 10.12.0.2, from 10.12.0.2, 00:00:15 ago, via GigabitEthernet1/0
      Route metric is 2, traffic share count is 1
```

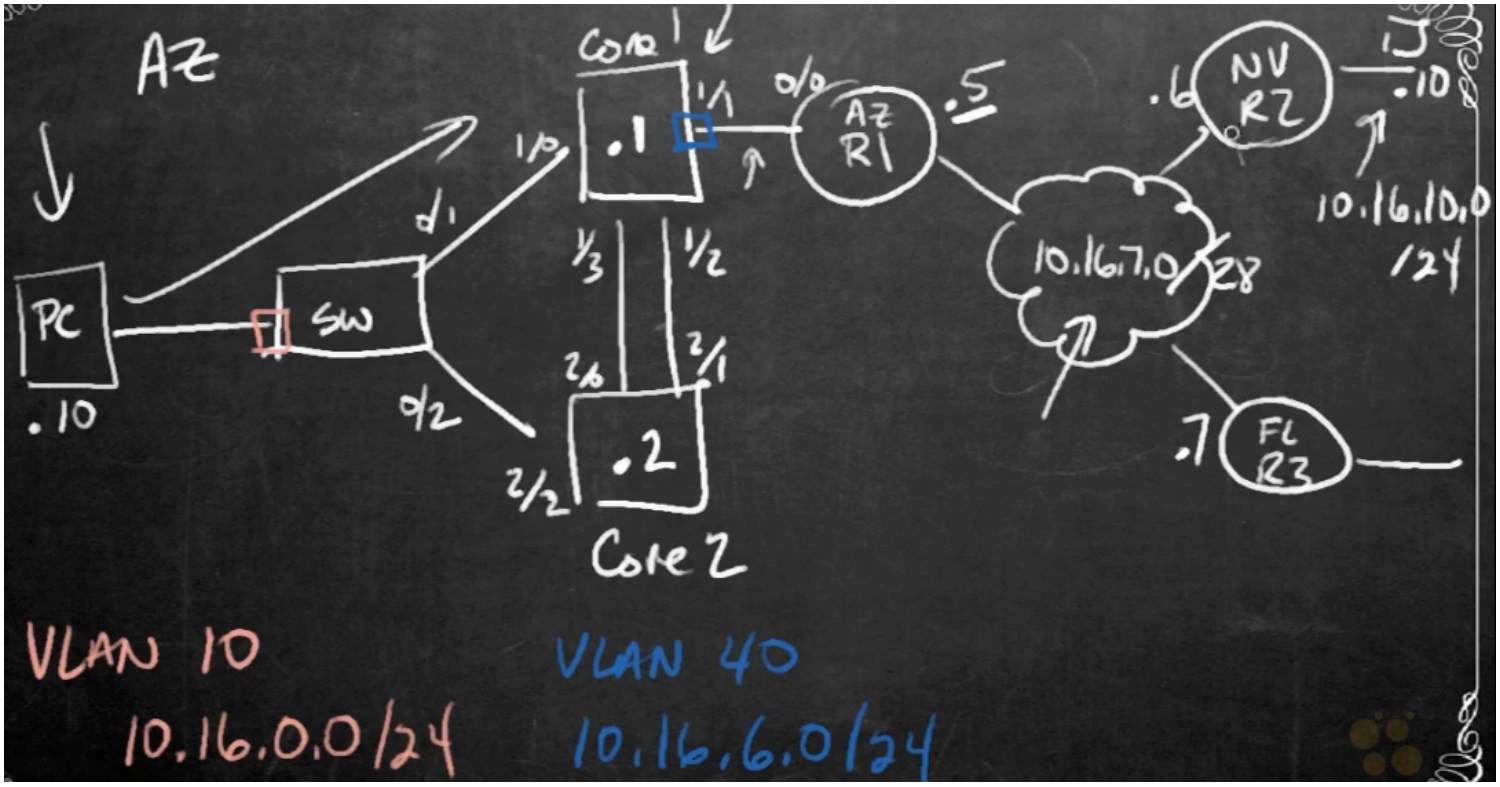
Which route will be matched:

```
O*E1  0.0.0.0/0 [110/2] via 10.13.0.1, 00:16:18, GigabitEthernet1/0
      1.0.0.0/32 is subnetted, 1 subnets
R        1.1.1.1 [120/1] via 10.13.0.1, 00:00:19, GigabitEthernet1/0
      2.0.0.0/32 is subnetted, 1 subnets
R        2.2.2.2 [120/1] via 10.23.0.2, 00:00:11, Serial3/3
      3.0.0.0/32 is subnetted, 1 subnets
C        3.3.3.3 is directly connected, Loopback0
      4.0.0.0/32 is subnetted, 1 subnets
R        4.4.4.4 [120/1] via 10.34.0.4, 00:00:14, FastEthernet4/0
      10.0.0.0/8 is variably subnetted, 21 subnets, 9 masks
O E1     10.0.0.0/8 [110/4] via 10.13.0.1, 00:16:18, GigabitEthernet1/0
O        10.1.0.0/24 [110/2] via 10.13.0.1, 00:16:18, GigabitEthernet1/0
O        10.2.0.0/24 [110/3] via 10.13.0.1, 00:16:13, GigabitEthernet1/0
C        10.3.0.0/24 is directly connected, GigabitEthernet0/0
L        10.3.0.3/32 is directly connected, GigabitEthernet0/0
O E1     10.4.0.0/17 [110/4] via 10.13.0.1, 00:16:18, GigabitEthernet1/0
O E1     10.4.0.0/20 [110/4] via 10.13.0.1, 00:16:18, GigabitEthernet1/0
O E1     10.4.0.0/23 [110/4] via 10.13.0.1, 00:16:18, GigabitEthernet1/0
O        10.4.0.0/24 [110/4] via 10.13.0.1, 00:16:03, GigabitEthernet1/0
O E1     10.4.0.0/26 [110/4] via 10.13.0.1, 00:16:18, GigabitEthernet1/0
O E1     10.4.0.0/29 [110/4] via 10.13.0.1, 00:16:18, GigabitEthernet1/0
O E1     10.4.0.0/30 [110/4] via 10.13.0.1, 00:16:18, GigabitEthernet1/0
O        10.12.0.0/24 [110/2] via 10.13.0.1, 00:16:13, GigabitEthernet1/0
C        10.13.0.0/24 is directly connected, GigabitEthernet1/0
L        10.13.0.3/32 is directly connected, GigabitEthernet1/0

O        172.16.2.0/24 [110/4] via 10.13.0.1, 00:54:14, GigabitEthernet1/0
O        172.16.4.0/25 [110/4] via 10.13.0.1, 00:54:14, GigabitEthernet1/0
O        172.16.6.0/26 [110/4] via 10.13.0.1, 00:54:14, GigabitEthernet1/0
O        172.16.8.0/27 [110/4] via 10.13.0.1, 00:54:14, GigabitEthernet1/0
O        172.16.10.0/28 [110/4] via 10.13.0.1, 00:54:14, GigabitEthernet1/0
O        172.16.12.0/29 [110/4] via 10.13.0.1, 00:54:14, GigabitEthernet1/0
```

10.4.0.1: 10.4.0.0/30
10.4.0.4: 10.4.0.0/29
10.4.0.25: 10.4.0.0/26 (32 16 8 4 2 1 -> need 5b so 27)
10.9.1.6: (10.0.0.0/8 ?)
23.1.4.5: 0.0.0.0/0
172.16.8.25: 172.16.8.0/27
- /27: .8.0, .8.32, .8.64

172.16.8.50: doesn't fit to .8.1-.8.31 so it will go to default 0.0.0.0
- 50 needs 6b so /26 but there is not /26 so default

Case study



Can we reach 10.16.10.10 from PC

Core1:

show ip route

```
O        10.16.8.0/24 [110/3] via 10.16.6.5, 00:05:04, Vlan40
S        10.16.10.8/29 [55/0] via 10.16.6.5, Vlan40
S        10.16.10.8/30 [77/0] via 10.16.6.5, Vlan40
O        10.16.16.0/24 [110/3] via 10.16.6.5, 00:05:04, Vlan40
```

- /29 10.16.10.8: 10.9-10.14 (increment 8)
- /30 10.16.10.8: 10.9-10.10 (increment 4)
  → /30 is longest match, use vlan 40 via 10.16.6.5

R1:

show ip route

```
L        10.16.7.5/32 is directly connected, GigabitEthernet2/0
S        10.16.8.0/21 [125/0] via 10.16.7.6
O        10.16.8.0/24 [110/2] via 10.16.7.6, 00:18:13, GigabitEthernet2/0
S        10.16.16.0/21 [125/0] via 10.16.7.7
```

- 10.16.8.0/21: 10.16.8.0 8.1-15.254
  → /21 is longest match

R2:

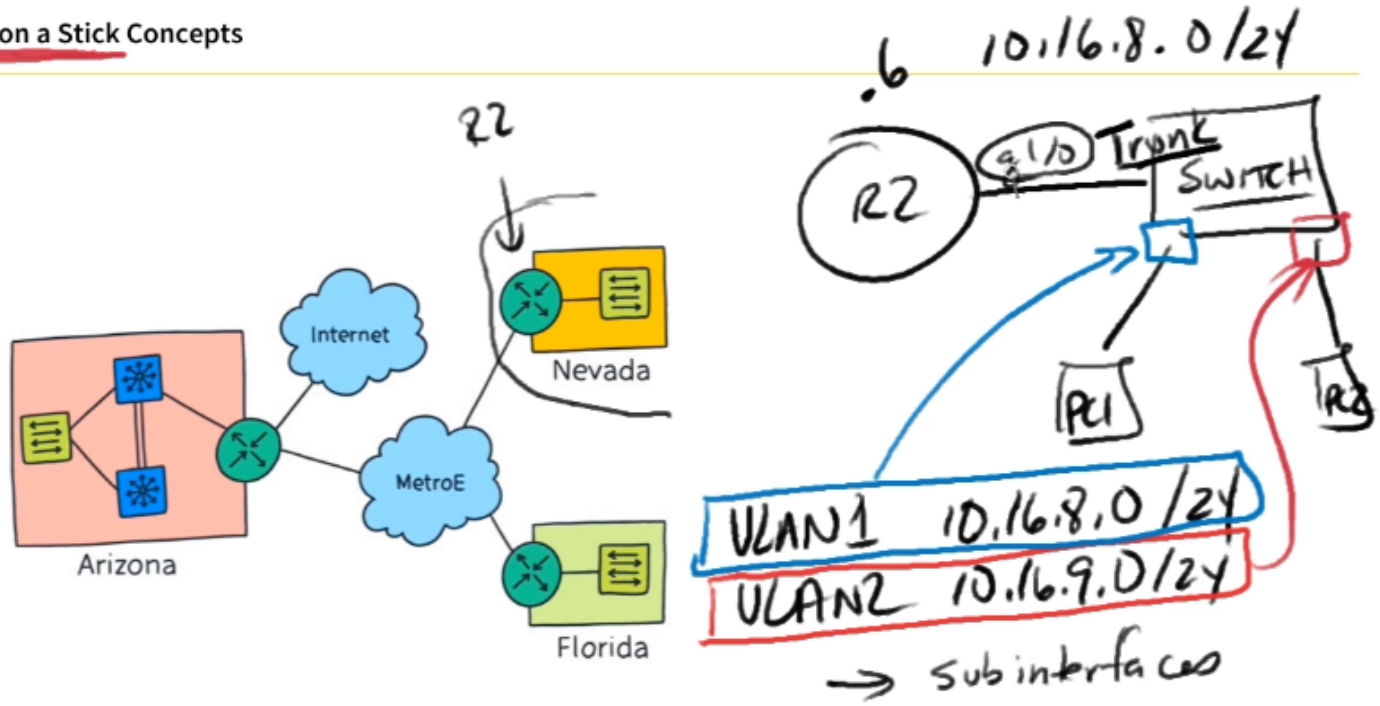
show ip route

```
L        10.16.8.6/32 is directly connected, GigabitEthernet1/0
C        10.16.10.0/24 is directly connected, Loopback10
L        10.16.10.10/32 is directly connected, Loopback10
S        10.16.16.0/24 [125/0] via 10.16.7.7
```

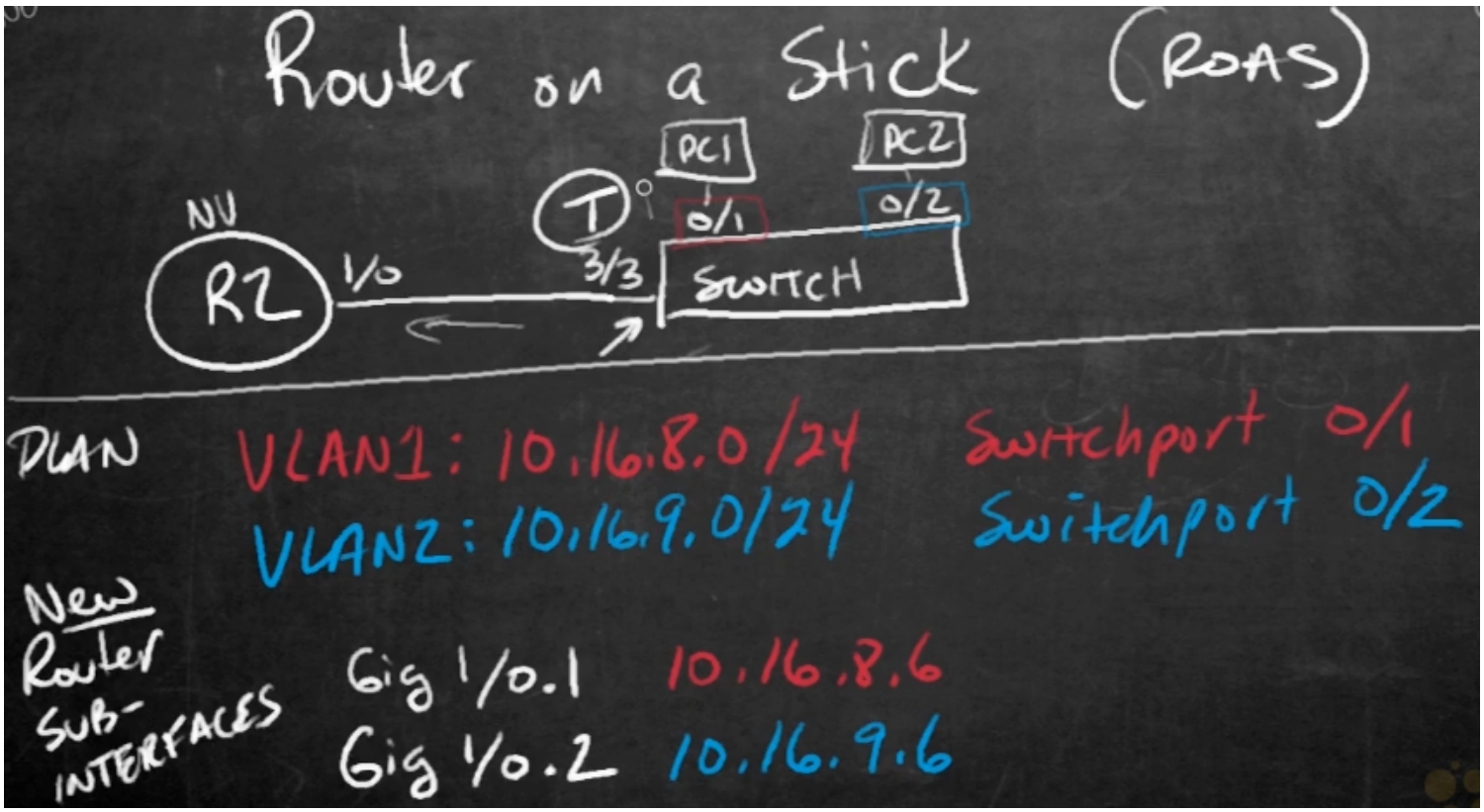
- 10.16.10.10/32 is best match we can get

40 - Configure and Verify Cisco’s Router on a Stick

Router on a Stick Concepts



- subinterface on the router g1/0.1, g1/0.2 etc...



Switch:
conf term
int gig 3/3
switchport trunk encap dot1q
switchport mode trunk

```
NV-Switch#show interfaces trunk

Port        Mode             Encapsulation  Status        Native vlan
Gi3/3       on               802.1q         trunking      1

Port        Vlans allowed on trunk
Gi3/3       1-4094

Port        Vlans allowed and active in management domain
Gi3/3       1-2

Port        Vlans in spanning tree forwarding state and not pruned
Gi3/3       none
```

R2:
conf term
int gig 1/0
no ip address
int gig 1/0.1
encap dot1Q 1 native (vlan 1)
ip address 10.16.8.6 255.255.255.0
int gig 1/0.2
encap dot1Q 2 (vlan 2)
ip adress 10.16.9.6 255.255.255.0

```
R2-NV#show ip int brief
Interface              IP-Address      OK? Method Status                Protocol
Ethernet0/0            unassigned      YES NVRAM  administratively down down
GigabitEthernet0/0     10.16.7.6       YES NVRAM  up                    up
GigabitEthernet1/0     unassigned      YES manual up                    up
GigabitEthernet1/0.1   10.16.8.6       YES manual up                    up
GigabitEthernet1/0.2   10.16.9.6       YES manual up                    up
GigabitEthernet2/0     unassigned      YES NVRAM  administratively down down
Serial3/0              unassigned      YES NVRAM  administratively down down
Serial3/1              unassigned      YES NVRAM  administratively down down
Serial3/2              unassigned      YES NVRAM  administratively down down
Serial3/3              unassigned      YES NVRAM  administratively down down
FastEthernet4/0        unassigned      YES NVRAM  administratively down down
FastEthernet4/1        unassigned      YES NVRAM  administratively down down
Loopback0              2.2.2.2         YES NVRAM  up                    up
```

```
R2-NV#show ip route
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
       + - replicated route, % - next hop override

Gateway of last resort is 10.16.7.5 to network 0.0.0.0

S*    0.0.0.0/0 [200/0] via 10.16.7.5
      2.0.0.0/32 is subnetted, 1 subnets
C        2.2.2.2 is directly connected, Loopback0
      10.0.0.0/8 is variably subnetted, 8 subnets, 4 masks
S        10.16.0.0/21 [125/0] via 10.16.7.5
C        10.16.7.0/28 is directly connected, GigabitEthernet0/0
L        10.16.7.6/32 is directly connected, GigabitEthernet0/0
C        10.16.8.0/24 is directly connected, GigabitEthernet1/0.1
L        10.16.8.6/32 is directly connected, GigabitEthernet1/0.1
C        10.16.9.0/24 is directly connected, GigabitEthernet1/0.2
L        10.16.9.6/32 is directly connected, GigabitEthernet1/0.2
S        10.16.16.0/21 [125/0] via 10.16.7.7
```

41 - Troubleshoot an IP Network

- verify what should normally work and how it normally work

42 - Cisco NAT Concepts and Base Configurations Static and Overload

NAT overload = PAT
- NAT sharing, share 1 public IP to internal devices
- using ports to distinguish

Static NAT
- both ways but usually translated from outside-in
- so email server has private IP but can receive emails
- you don't have to dedicate entire public IP, but just IP:port you want

Dynamic NAT
- translate pool of addresses to pool of addresses

→ maybe when you need merge 2 networks and they have same IP range
→ you could you Dynamic NAT to translate them so they appear like they are different
- translate public pool of addresses to private randomly

Terminology

- inside
→ who owns that address - YOU
- outside
→ who owns the address - SOMEONE ELSE
- local
→ private
- global
→ public

- inside local
→ my computer
- outside global
→ servers public ip
- inside global
→ my routers public ip
- outside local
→ I can set up private IP to be translated to some public so it looks like it is on my network but it is not

Setup

1. ACL - standard (ip src)
2. identify interfaces - in/out
3. NAT global

RT01:
conf term
ip access-list standard NAT-SOURCES (create named acl)
permit 10.16.0.0 0.0.7.255 (/21 wildcard mask)
permit 10.16.8.0 0.0.7.255
permit 10.16.16.0 0.0.7.255 (all 3 sites)

```
AZ-RT01#show ip access-lists
Standard IP access list NAT-SOURCES
    10 permit 10.16.0.0, wildcard bits 0.0.7.255
    20 permit 10.16.8.0, wildcard bits 0.0.7.255
    30 permit 10.16.16.0, wildcard bits 0.0.7.255
```

```
AZ-RT01#show ip int brief
Interface                  IP-Address      OK? Method Sta
GigabitEthernet0/0         203.0.113.2     YES DHCP   up
GigabitEthernet0/1         unassigned      YES NVRAM  adm
FastEthernet0/0/0          unassigned      YES unset  up
FastEthernet0/0/1          unassigned      YES unset  dow
FastEthernet0/0/2          unassigned      YES unset  adm
FastEthernet0/0/3          unassigned      YES unset  adm
Vlan1                      unassigned      YES manual up
Vlan10                     10.16.0.1       YES manual up
Vlan20                     10.16.2.1       YES manual up
Vlan30                     10.16.4.1       YES manual up
Vlan40                     10.16.6.1       YES manual up
AZ-RT01#conf t
```

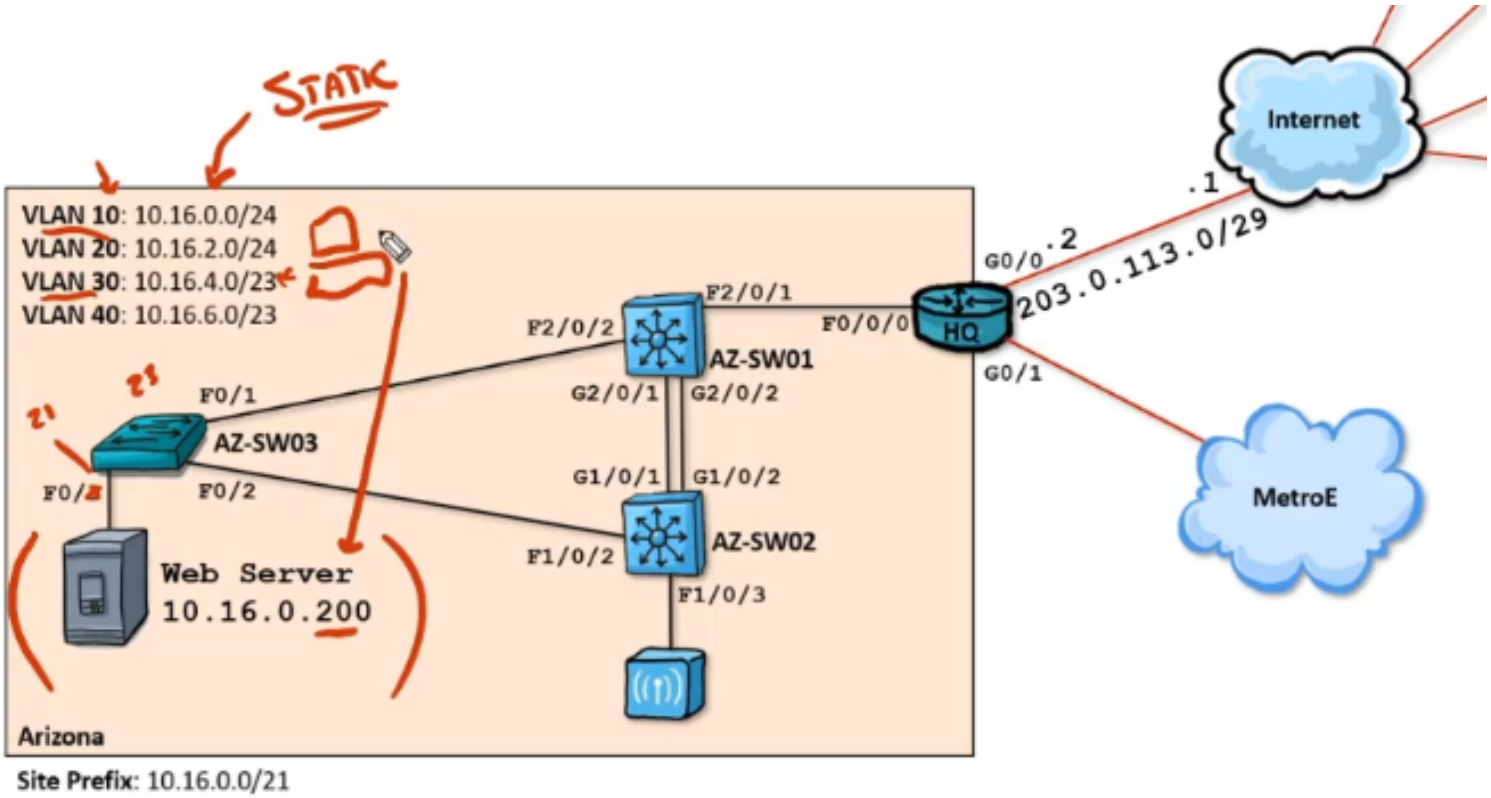
```
int gi0/0
    ip nat outside
    exit
int vlan 10
    ip nat inside
int vlan 20
    ip nat inside
int vlan 30
    ip nat inside
int vlan 40
    ip nat inside
    end
```

ip nat inside source list NAT-SOURCES interface gi0/0 overload
(I wanna nat from inside to interface gi0/0 and use overload)

```
AZ-RT01#show ip nat translations
Pro Inside global      Inside local       Outside local      Outside global
tcp 203.0.113.2:6471   10.16.5.50:6471    185.188.32.1:5938  185.188.32.1:5938
tcp 203.0.113.2:6473   10.16.5.50:6473    52.117.209.71:5938 52.117.209.71:5938
tcp 203.0.113.2:6475   10.16.5.50:6475    172.217.11.164:80  172.217.11.164:80
tcp 203.0.113.2:6476   10.16.5.50:6476    172.217.11.164:80  172.217.11.164:80
```

How to change to pool of public IPs instead of just 1?
conf term
ip nat pool PUBLIC-IPS 203.0.113.2 203.0.113.4 prefix-length 29 (/29)
no ip nat inside source list NAT-SOURCES interface gi0/0 overload (copy paste from before)
ip nat inside source list NAT-SOURCES pool PUBLIC-IPS overload

Static NAT



- web server will be accessible from internet on 203.0.113.3

1. Identify interfaces - which are inside and which outside
2. globally build static nat mapping

RT01:
conf term
ip nat inside source static 10.16.0.200 203.0.113.3

```
AZ-RT01#show ip nat translations
Pro Inside global      Inside local       Outside local      Outside global
tcp 203.0.113.3:49757  10.16.0.200:49757  52.242.211.89:443  52.242.211.89:443
tcp 203.0.113.3:51051  10.16.0.200:51051  68.231.24.76:6690  68.231.24.76:6690
tcp 203.0.113.3:51053  10.16.0.200:51053  68.231.24.76:6690  68.231.24.76:6690
tcp 203.0.113.3:51059  10.16.0.200:51059  68.231.24.76:6690  68.231.24.76:6690
tcp 203.0.113.3:51061  10.16.0.200:51061  68.231.24.76:6690  68.231.24.76:6690
--- 203.0.113.3        10.16.0.200        ---                ---
AZ-RT01#show run | i ip nat
 ip nat outside
 ip nat inside
 ip nat inside
 ip nat inside
 ip nat inside
ip nat pool PUBLIC-IPS 203.0.113.2 203.0.113.4 prefix-length 29
ip nat inside source list NAT-SOURCES pool PUBLIC-IPS overload
ip nat inside source static 10.16.0.200 203.0.113.3
```

Static PAT:
conf term
no ip nat inside source static 10.16.0.200 203.0.113.3
no ip nat inside source static tcp 10.16.0.200 80 203.0.113.3 80 (this translates just port 80 tcp to webserver)
(works both ways so except for port 80 the server will use dynamic overload anyway)
no ip nat inside source static tcp 10.16.0.201 25 203.0.113.3 25 (this translates just port 25 tcp to mailserver)

```
AZ-RT01#show running-config | i ip nat
 ip nat outside
 ip nat inside
 ip nat inside
 ip nat inside
 ip nat inside
ip nat pool PUBLIC-IPS 203.0.113.2 203.0.113.4 prefix-length 29
ip nat inside source list NAT-SOURCES pool PUBLIC-IPS overload
ip nat inside source static tcp 10.16.0.201 25 203.0.113.3 25 extendable
ip nat inside source static tcp 10.16.0.200 80 203.0.113.3 80 extendable
```

43 - Configure and Verify Cisco NTP

Types:
- client/server
- multicast
- broadcast

conf term
clock timezone SYDNEY -10 (name sydney and time diff)
ntp server 203.0.113.1
ntp master 5 (stratum=5, set server as a master)

```
AZ-RT01#show ntp associations

  address         ref clock       st   when   poll reach  delay  offset   disp
 ~203.0.113.1     .INIT.          16      -     64     0  0.000   0.000 16000.
 * sys.peer, # selected, + candidate, - outlyer, x falseticker, ~ configured
```

```
AZ-RT01#show clock
13:38:07.139 ARIZONA Thu Nov 21 2019
```

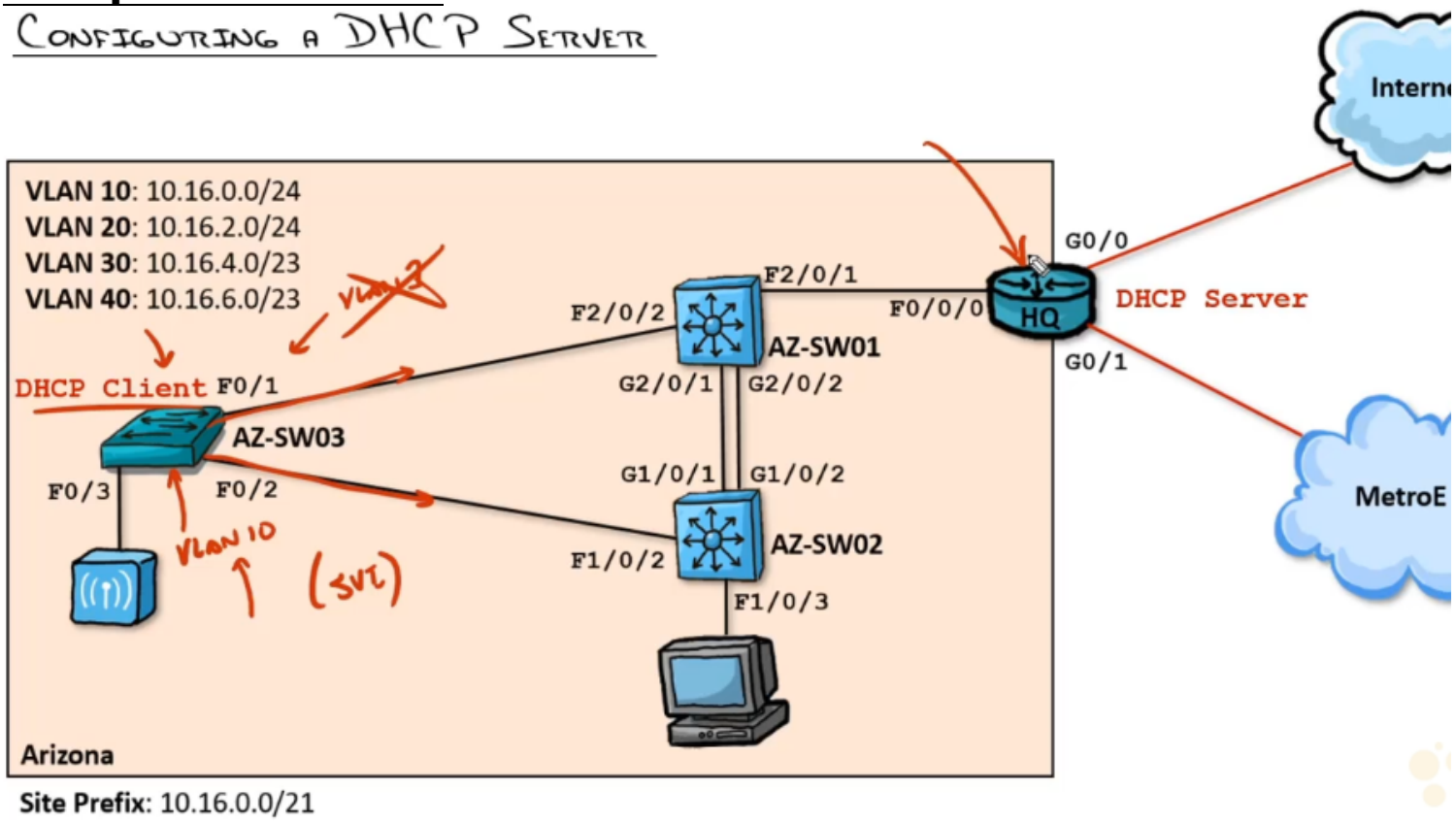
On Switch1:
conf term
ntp server 10.16.0.1

```
AZ-SW01#show ntp associations

     address         ref clock      st  when  poll reach  delay  offset    disp
*~10.16.0.1        203.0.113.1       4    16    64  377     1.1    0.66     0.3
 * master (synced), # master (unsynced), + selected, - candidate, ~ configured
```

44 - Configure DHCP Server and Relay Functions

DHCP client
DHCP Server
DHCP Relay - relay dhcp request broadcast to dhcp server even if it is in different vlan etc

Setup DHCP server



Site Prefix: 10.16.0.0/21

RT01:

```
AZ-RT01#show ip int brief
Interface                  IP-Address      OK? Method Status                Protocol
GigabitEthernet0/0         203.0.113.2     YES DHCP   up                    up
GigabitEthernet0/1         172.16.0.1      YES NVRAM  up                    up
FastEthernet0/0/0          unassigned      YES unset  up                    up
FastEthernet0/0/1          unassigned      YES unset  down                  down
FastEthernet0/0/2          unassigned      YES unset  administratively down down
FastEthernet0/0/3          unassigned      YES unset  administratively down down
NVI0                       unassigned      YES unset  administratively down down
Vlan1                      unassigned      YES NVRAM  up                    up
Vlan10                     10.16.0.1       YES NVRAM  up                    up
Vlan20                     10.16.2.1       YES NVRAM  up                    up
Vlan30                     10.16.4.1       YES NVRAM  up                    up
Vlan40                     10.16.6.1       YES NVRAM  up                    up
```

conf term
ip dhcp excluded-address 10.16.0.1 10.16.0.3 (exclude specific address range)
ip dhcp pool SERVERS(V10) (create named pool)
network 10.16.0.0 /24
default-router 10.16.0.1
dns-server 8.8.8.8 4.2.2.2

```
AZ-RT01(dhcp-config)#?
DHCP pool configuration commands:
  accounting            Send Accounting Start/Stop messages
  bootfile              Boot file name
  class                 Specify a DHCP class
  client-identifier     Client identifier
  client-name           Client name
  default-router        Default routers
  dns-server            DNS servers
  domain-name           Domain name
  exit                  Exit from DHCP pool configuration mode
  hardware-address      Client hardware address
  host                  Client IP address and mask
  import                Programatically importing DHCP option parameters
  lease                 Address lease time
  netbios-name-server   NetBIOS (WINS) name servers
  netbios-node-type     NetBIOS node type
  network               Network number and mask
  next-server           Next server in boot process
  no                    Negate a command or set its defaults
  option                Raw DHCP options
  origin                Configure the origin of the pool
  relay                 Function as a DHCP relay
  remember              Remember released bindings
  renew                 Configure renewal policy
  server                Configure the server ID option value
  subnet                Subnet allocation commands
  update                Dynamic updates
  utilization           Configure various utilization parameters
  vrf                   Associate this pool with a VRF
```

```
AZ-RT01#show run | s dhcp
ip dhcp excluded-address 10.16.0.1 10.16.0.3
ip dhcp pool SERVER(V10)
   network 10.16.0.0 255.255.255.0
   default-router 10.16.0.1
   dns-server 8.8.8.8 4.2.2.2
   domain-name cbtnuggets.com
 ip address dhcp
```

```
AZ-RT01#show ip dhcp binding
Bindings from all pools not associated with VRF:
IP address          Client-ID/              Lease expiration        Type
                    Hardware address/
                    User name
10.16.0.4           0063.6973.636f.2d30.    Nov 23 2019 01:49 PM    Automatic
                    3030.632e.3835.3462.
                    2e65.6538.302d.566c.
                    3130
10.16.0.5           0100.e04c.6801.21       Nov 23 2019 01:09 PM    Automatic
```

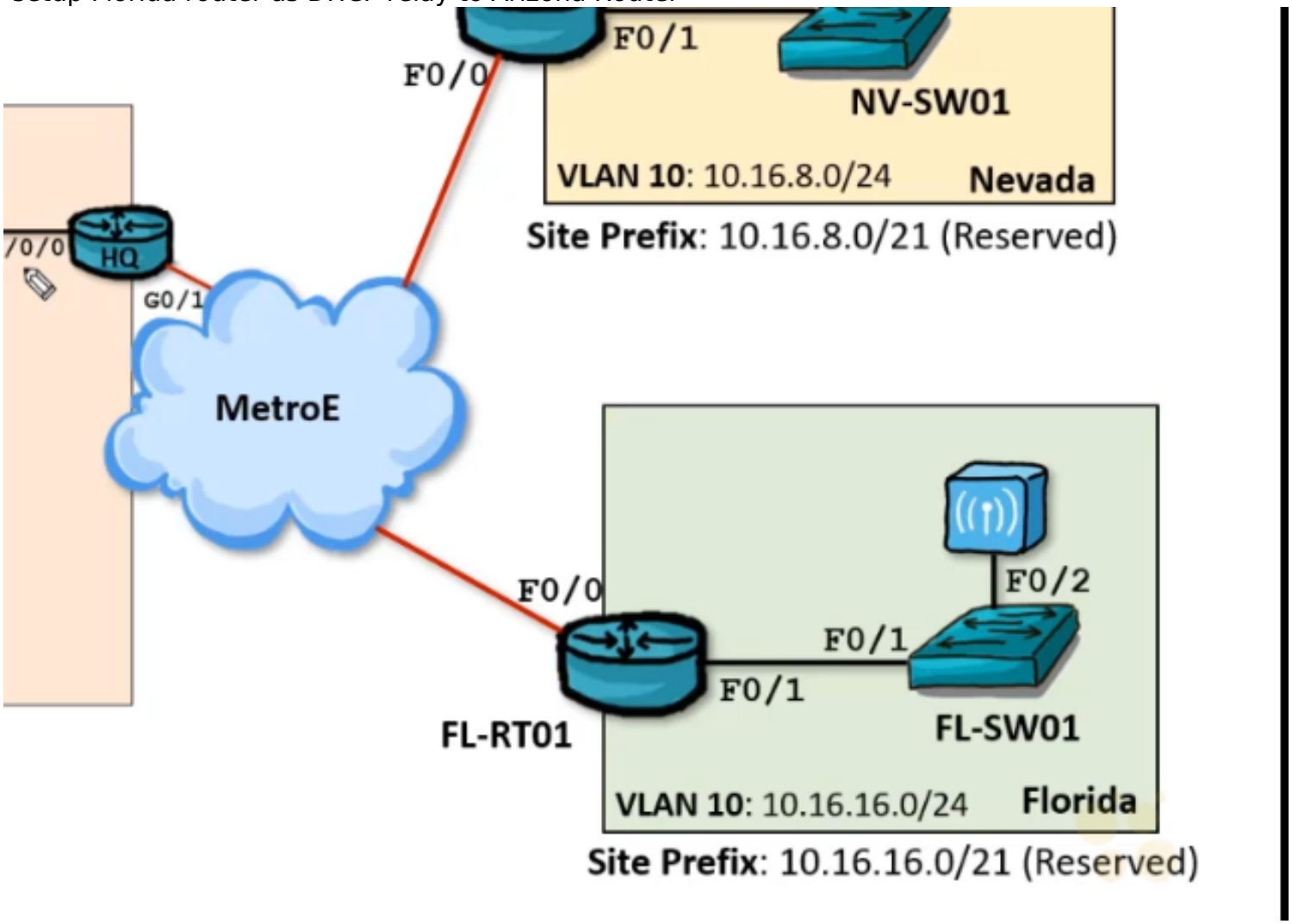
```
conf term
clear ip dhcp binding 10.16.0.5
ip dhcp pool JEREMYS-APPLE
    hardware-address 00e0.4c68.0121
    host 10.16.0.5 /24
    exit
```

```
AZ-RT01#show running-config | s dhcp
ip dhcp excluded-address 10.16.0.1 10.16.0.3
ip dhcp pool SERVER(V10)
   network 10.16.0.0 255.255.255.0
   default-router 10.16.0.1
   dns-server 8.8.8.8 4.2.2.2
   domain-name cbtnuggets.com
ip dhcp pool JEREMYS-APPLE
   host 10.16.0.5 255.255.255.0
   hardware-address 00e0.4c68.0121
 ip address dhcp
```

```
AZ-RT01#show ip dhcp binding
Bindings from all pools not associated with VRF:
IP address          Client-ID/                  Lease expiration        Type
                    Hardware address/
                    User name
10.16.0.4           0063.6973.636f.2d30.        Nov 23 2019 01:49 PM    Automatic
                    3030.632e.3835.3462.
                    2e65.6538.302d.566c.
                    3130
10.16.0.5           00e0.4c68.0121              Infinite                Manual
```

Setup DHCP relay

-setup Florida router as DHCP relay to Arizona Router



AZ-RT01:

```
conf term
ip dhcp pool FLORIDA(V10)
    exit
ip dhcp excluded-address 10.16.16.1 10.16.16.5
ip dhcp pool FLORIDA(V10)
    network 10.16.16.0 /24
    default-router 10.16.16.1
    dns-server 8.8.8.8 4.2.2.2
    domain-name florida.cbtnuggets.com
    exit
```

FL-RT01:

```
FL-RT01#show ip int brief
Interface                  IP-Address      OK? Method Status                Protocol
FastEthernet0/0            172.16.0.3      YES NVRAM  up                    up
Serial0/0                  unassigned      YES NVRAM  administratively down down
FastEthernet0/1            unassigned      YES manual up                    up
FastEthernet0/1.10         10.16.16.1      YES manual up                    up
FastEthernet0/1.20         10.16.18.1      YES manual up                    up
FastEthernet0/1.30         10.16.20.1      YES manual up                    up
FastEthernet0/1.40         10.16.22.1      YES manual up                    up
Serial0/1                  unassigned      YES NVRAM  administratively down down
```

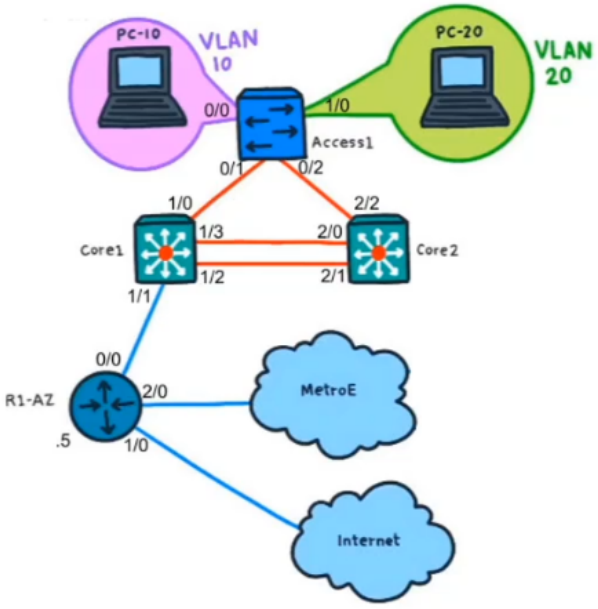
```
conf term
int fa0/1.10 (you go to interface which will receive DHCP request)
    ip helper-address 172.16.0.1
```

AZ-RT01:

```
AZ-RT01#show ip dhcp binding
Bindings from all pools not associated with VRF:
IP address          Client-ID/                Lease expiration        Type
                    Hardware address/
                    User name
10.16.0.4           0063.6973.636f.2d30.      Nov 23 2019 01:49 PM    Automatic
                    3030.632e.3835.3462.
                    2e65.6538.302d.566c.
                    3130
10.16.0.6           0100.e04c.6801.21         Nov 23 2019 01:19 PM    Automatic
10.16.16.6          0063.6973.636f.2d30.      Nov 23 2019 05:56 PM    Automatic
                    3031.612e.6131.6438.
                    2e36.3634.312d.566c.
                    3130
```

Case study

Lab Topology – Configure a Cisco DHCP Server



Lab Objectives:

A network auditor recently found that the devices of your network were all statically assigned. Due to the recent **#DHCPLove** social movement, all devices are demanding dynamic addressing.

1. Configure the NIC of **PC-10** to use **dynamic IP addressing**. Due to the lack of DHCP services, it should soon auto-generate a **169.254.x.x** IP address.

2. Set up a DHCP server on the **Core1** L3 switch with the following parameters:
   - Network: **10.16.0.0/24**
   - Gateway: **10.16.0.1**
   - DNS: **8.8.8.8** (primary) and **4.2.2.2** (secondary)
   - Excluded: **10.16.0.1 – 10.16.0.25**

3. Verify **PC-10** receives an IP address from the pool as expected (might need an **ipconfig /renew** to speed things along). Verify the **IP to MAC** DHCP mapping on **Core1**.

PC-10:
- change ip to dynamic -> will assign APIPA

Core1:

```
conf term
ip dhcp excluded-address 10.16.0.1 10.16.0.25
ip dhcp pool DHCPLOVE
    network 10.16.0.0 /24
    default-router 10.16.0.1
    dns-server 8.8.8.8 4.2.2.2
    exit
```

```
Core1#show ip dhcp binding
Bindings from all pools not associated with VRF:
IP address      Client-ID/              Lease expiration        Type       State      Interface
                Hardware address/
                User name
10.16.0.26      0100.155d.7788.99       Sep 22 2019 07:50 PM    Automatic  Active     Vlan10
```

45 - Explain Common Network Services SNMP, Syslog, QoS, and TFTP-FTP

SNMP

- simple network management protocol
- poll data from devices
- common pool interval is 60s

- programs like PRTG

- OID/MIB
  → object identifier, looks like long ip address
  → management information book, db of all OID you can have on a device

Versions
- version 1 - old
- version 2c - very popular, very limited auth (community string), you should use ACL
- version 3 - increasing security, MD5/SHA with DES
  → noAuthNoPriv - no encryption, username match
  → authNoPriv - no encryption, hashed auth code
  → authPriv - encypted MD5/SHA

SNMPv3
- SNMP View - what you can see
- SNMP Group
- SNMP User

Syslog
- date, time, message

`logging 10.1.1.1` (send syslog to ip address)

Severity levels

VALUE	SEVERITY	KEYWORD	DESCRIPTION	EXAMPLES
0	Emergency	emerg	System is unusable	This level should not be used by applications.
1	Alert	alert	Should be corrected immediately	Loss of the primary ISP connection.
2	Critical	crit	Critical conditions	A failure in the system's primary application.
3	Error	err	Error conditions	An application has exceeded its file storage limit and attempts to write are failing.
4	Warning	warning	May indicate that an error will occur if action is not taken.	A non-root file system has only 2GB remaining.
5	Notice	notice	Events that are unusual, but not error conditions.	
6	Informational	info	Normal operational messages that require no action.	An application has started, paused or ended successfully.
7	Debug	debug	Information useful to developers for debugging the application.	

QoS

- quality of service
- qos works only when there is congestion on the network, no need for qos without congestion
- traffic discrimination
  → classify - classify packets to class
  → mark - mark packets, Layer2: CoS = Class Of Service = level of priority 0-7, Layer3: ToS = Type Of Service = priority (DSCP is implementation)
  → queue - queue packets in priority queueing, without QoS it would be FIFO queue
    ⇒ cisco use LLQ(low latency queueing): capped priority queue (most important first) with class based (still important but not most) and weighted fair queue
    ⇒ simply means low bandwidth stream has more priority over high bandwidth stream
  → shoot (WRED) - Weighted random early detection
    ⇒ congestion avoidance, stop TCP windows to build up too big
  → shape - shape traffic down but not throttle
  → police - putting anti-quality control on some traffic, throttle specific traffic

TFTP and FTP

FTP - file transfer protocol
- tcp, port 21
- simple security (username and passwd)

TFTP - trivial file transfer protocol
- udp, port 69
- no security
- confirmation of every packet on top of udp

46 - Define Key Concepts Regarding Network Security

C I A
- confidentiality - data are private
- integrity - data are not manipulated or changed
- availability - data are available when needed

Vulnerability
- *weakness*, that has been taken advantage of to get unauthorised access
- bug or flaw
- bad or lacking configuration
- legacy system
- humans!!!
- unpatched system

Exploits
- way to compromise system
- tool/method/system to take advantage of vulnerability
  → trust ARP
  → believing every source mac is real
  → processing all CDP messages
  → allowing all DHCP server messages

Threats and risk
- threat
  → anything that can exploit or take advantage of a vulnerability (weakness)
  → actual use of exploit against a vulnerability
- risk
  → possibility of thread, likelihood
  → how likely the exploit is used against a vulnerability

Mitigation Techniques

- awareness
  → apps security (OWASP)
  → network infrastructure
  → humans (training, educate etc., testing our users)

47 - Describe Security Program Elements

AUP - acceptable use policy

User awareness
- train and test them

48 - Describe Elements of Secure Password Policies

Password element with examples
- length: 7-8 minimum
- complexity
- age: max age to change passwd
- history: not able to change to same passwd
- minimum age: change max 1 a day

MFA - multi factor auth

2FA choose 2 from:
A - user knows
  - password, pin, answer to question
B- user has
  - card, key-fob, one time passwd, security token, certificate
C - user is
  - biometric

49 - Configure Cisco Device Access Control Using Local Passwords

CONSOLE

```
Router>show users
    Line       User       Host(s)              Idle       Location
*   0 con 0                idle                 00:00:00
```

- star shows we are connected to con0= console port

```
Router>show privilege
Current privilege level is 1
Router>enable
Router#show privilege
Current privilege level is 15
```

- no login is required for access
- enable → default without passwd

conf term
line con 0
    password consolepw (set password)
    login (require password)

AUX

- console port active while booting
- aux only after it booted up
- you could connect it to modem for backup access

```
Router>show users
    Line       User       Host(s)              Idle       Location
*  1 aux 0                idle                 00:00:00

  Interface    User               Mode         Idle     Peer Address
```

- you can not go to privileged mode unless password is set up

over Console:
conf term
enable secret secretpw (encrypted)
enable password BadIdea (plain text)
do show run
(the enable password is secretpw, it ignores plain password even though it was entered later)

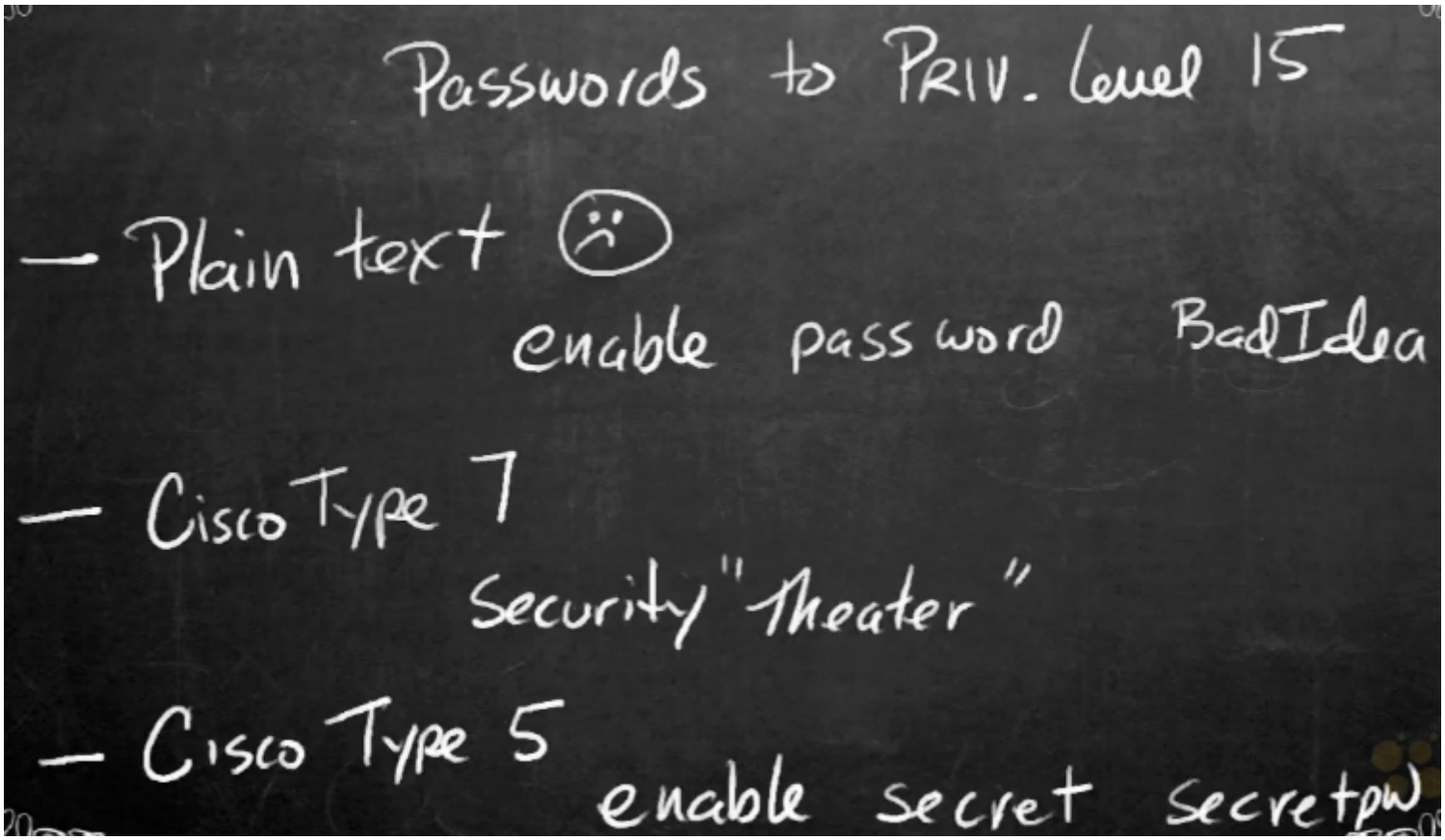
```
!
!
enable secret 5 $1$mERr$MS3KwxTgPes.naJ9WWsKb0
enable password BadIdea
!
```

line aux 0
    password auxpw
    login

VTY

- telnet, ssh
- you can't connect without password

conft term
line vty 0 4
    password vtypw
(login is here by default thats why you cant connect without passwd)



conf term
service password-encryption (encrypt all plaintext passwords on the device with type 7)

most secure:
enable algorithm-type scrypt secret Type-9-PW

```
R-6783(config)#do show run | include enable secret 9
enable secret 9 $9$B0nWRNF1juQ03Y$D/YXyrUCh7.UrFlgErZFOXF5HZh5iVVkjfCvuvrm6Z.
```

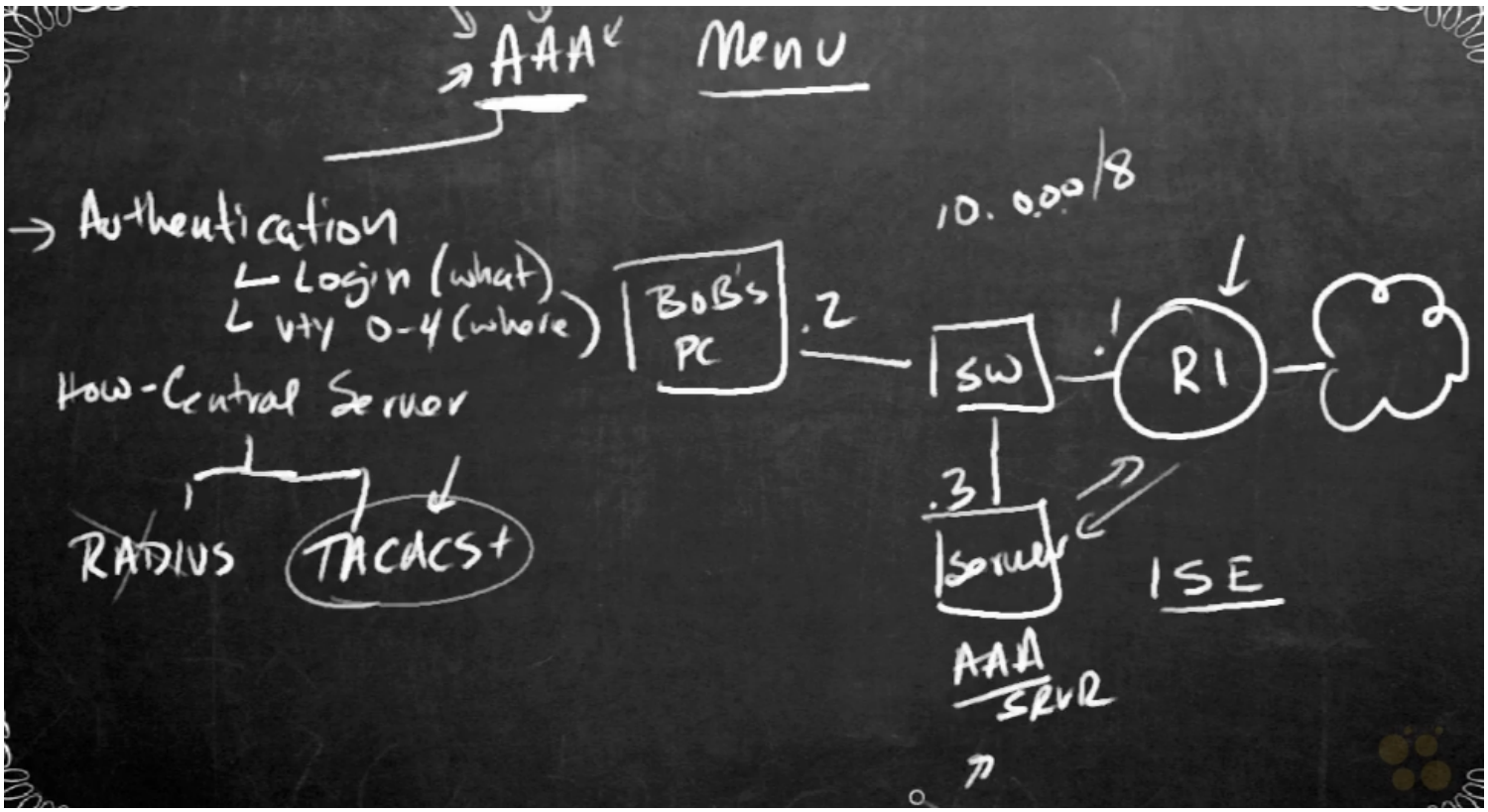
Usernames and passwords
conf term
username bob secret bobsecret
line vty 0 4
login local (use local name and passwd for auth)

50 - Summarize and Differentiate AAA Concepts

AAA

A - authentication (who are you?)
A - authorization (what you can do?)
A - accounting, auditing (record what they did)

1. AAA Server
   → centralized server
2. AAA client
   → any client, device etc
3. Protocol
   → TACACS (sends auth and accounting in different requests)
   → RADIUS



AAA Server at 10.0.0.3:

TACACS client: R1 10.0.0.1 , p/w aaa-cs-pw
User account: bob, p/w aaa-user-pw

R1 as AAA Client using AAA Server:

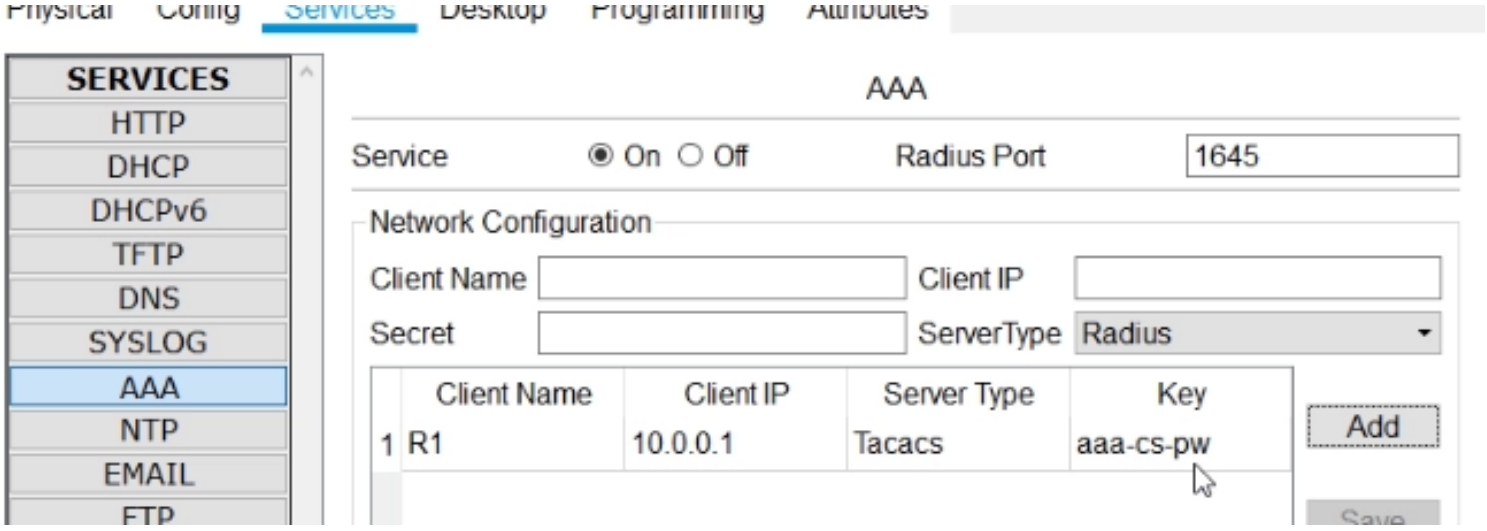
```
aaa new-model
tacacs-server host 10.0.0.3 key aaa-cs-pw

aaa authentication login Use-AAA group tacacs+

line vty 0 4
 login authentication Use-AAA
```

R1:

```
en
conf term
int gig 0/0
    no shut
    ip address 10.0.0.1 255.0.0.0
```

PC:
ip: 10.0.0.2 255.0.0.0

Srvr:
ip: 10.0.0.3 255.0.0.0
Services>AAA:



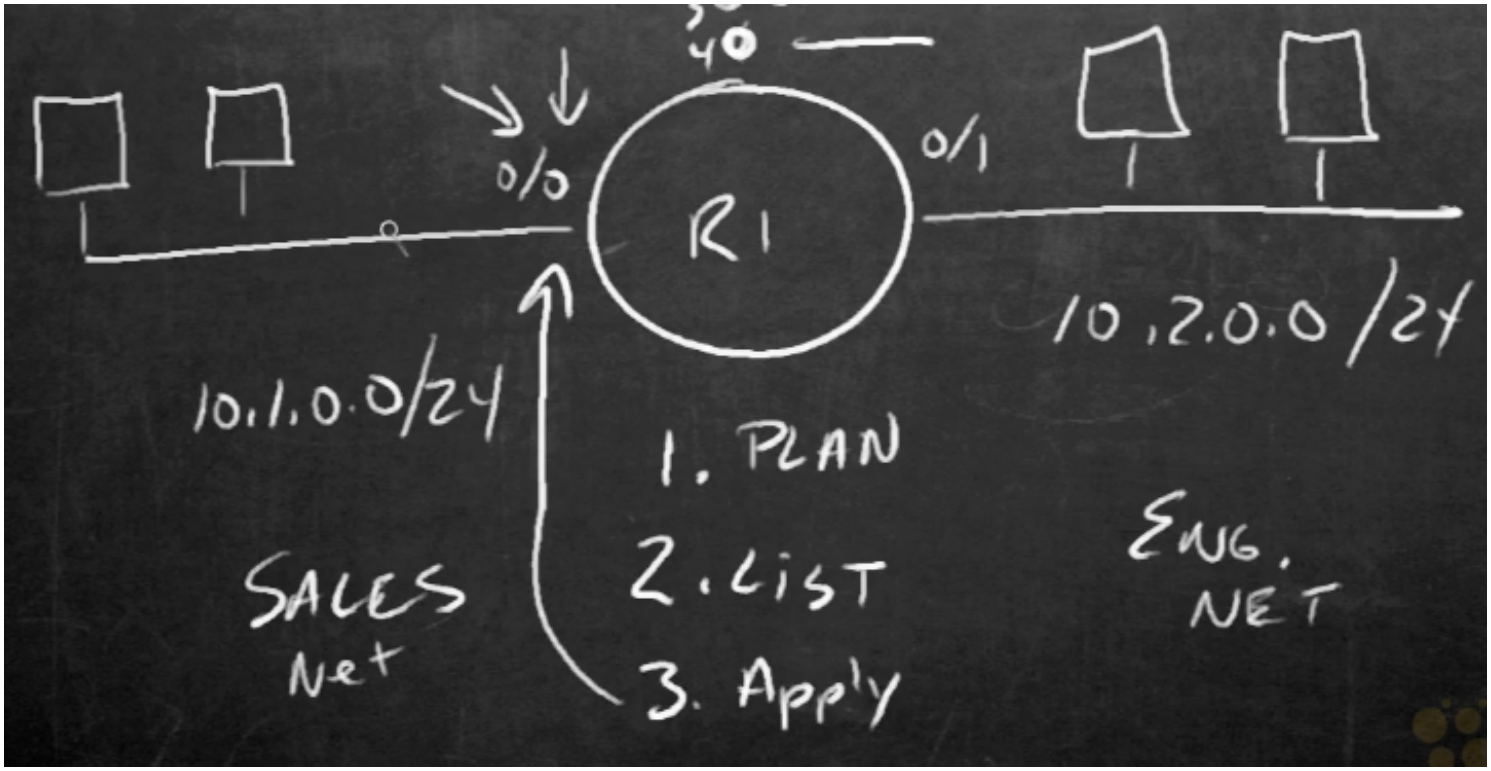
R1:
conf term
hostname R1
aaa new-model (changes some defaults and you can use advanced commands)
tacacs-server host 10.0.0.3 key aaa-cs-pw
aaa autherntication login Use-AAA group tacacs+ (create method list named Use-AAA)

```
R1(config)#aaa authentication login ?
  WORD     Named authentication list.
  default  The default authentication list.
R1(config)#aaa authentication login Use-AAA ?
  enable      Use enable password for authentication.
  group       Use Server-group.
  local       Use local username authentication.
  local-case  Use case-sensitive local username authentication.
  none        NO authentication.
R1(config)#aaa authentication login Use-AAA gro
R1(config)#aaa authentication login Use-AAA group tac
R1(config)#aaa authentication login Use-AAA group tacacs+ ?
  enable      Use enable password for authentication.
  group       Use Server-group.
  local       Use local username authentication.
  local-case  Use case-sensitive local username authentication.
  none        NO authentication.
  <cr>
R1(config)#aaa authentication login Use-AAA group tacacs+
```

line vty 0 4
login authentication Use-AAA

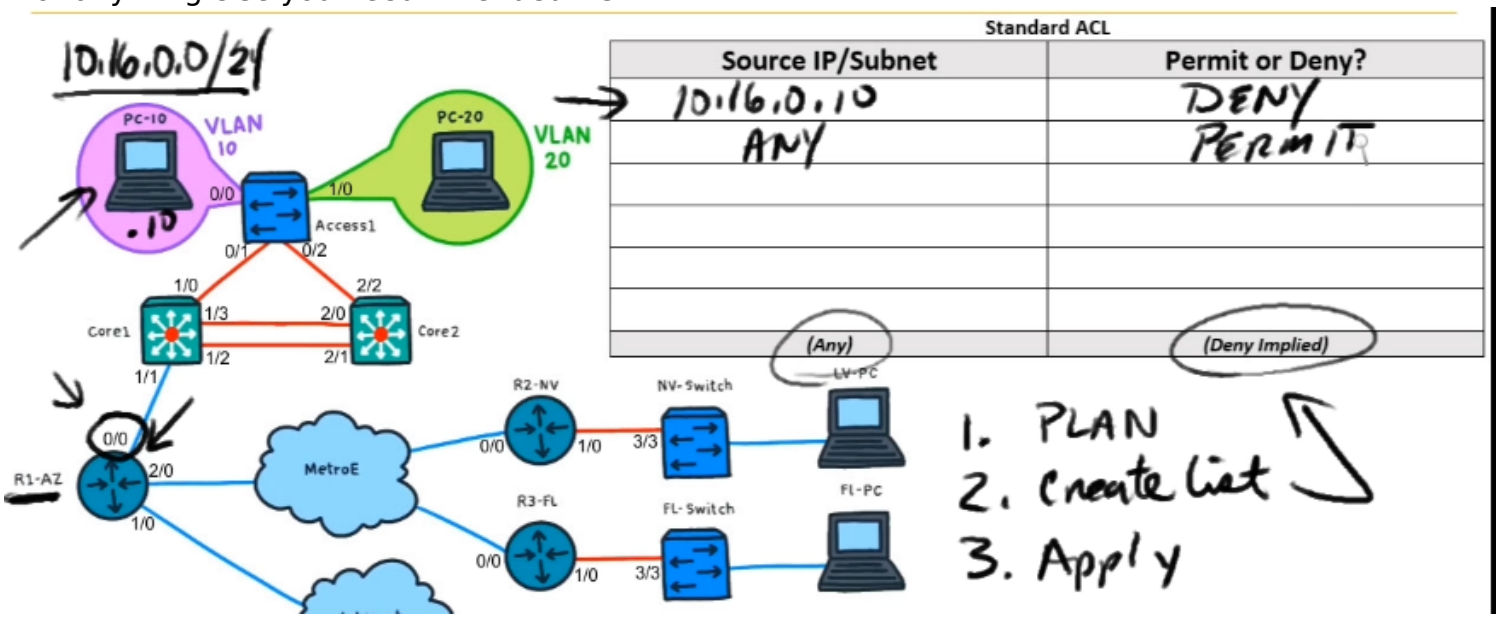
51 - Apply and Verify Cisco Access Control Lists

default DENY



Standard ACL

- you can match on just source IP -> allow/deny
- number 0-99
- for anything else you need Extended ACL



R1-AZ:

show access-lists (verify ACLs if any)
show ip int gig 0/0 (check if any ACLs applied to specific interface)
conf term
access-list 1 deny host 10.16.0.10 (create list #1)
access-list 1 permit any
int gig 0/0
ip access-group 1 in (use list #1)

```
R1-AZ#show access-lists
Standard IP access list 1
    10 deny   10.16.0.10 log (80 matches)
    20 permit any (25 matches)
```

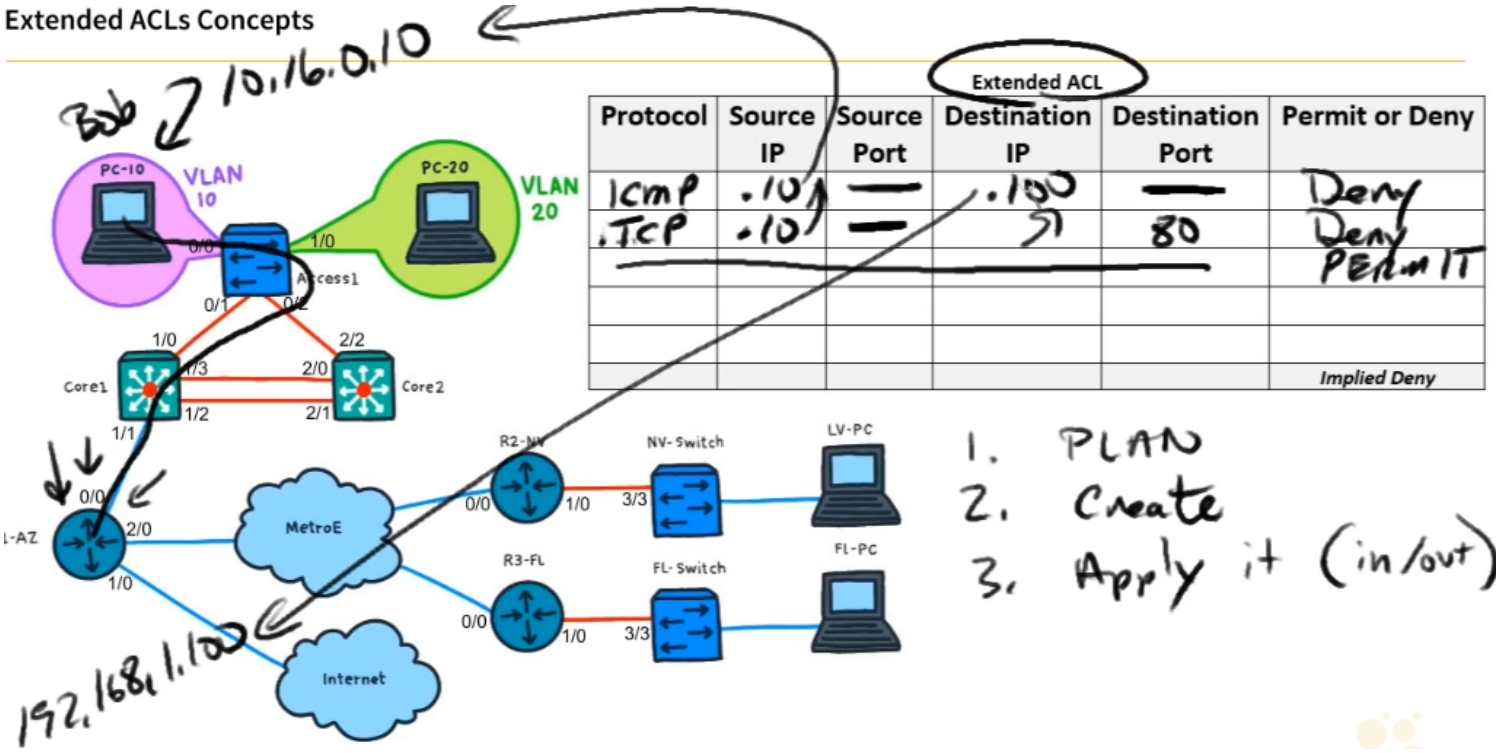
Wildcard masks concepts

- deny entire /24 network but permit everything else

```
conf term
access-list 2 deny 10.16.0.0 0.0.0.255
access-list 2 permit any
int gig 0/0
    ip access-group 2 in
```
(use list #2)

Extended ACL

- can match on anything L3, L4
- list number **100-199**



R1-AZ:

```
show access-list
conf term
access-list 100 deny icmp host 10.16.0.10 host 192.168.1.100 log
access-list 100 deny tcp host 10.16.0.10 host 192.168.1.100 eq 80
```
(deny traffic src.ip, dst.ip and dst.port 80)

```
access-list 100 permit ip any any
int gig 0/0
    ip access-group 100 in
```

Named Extended ACL

Protocol	Source IP	Source Port	Destination IP	Destination Port	Permit or Deny
TCP	PC-10	—	ANY	21	Deny
IP	VLAN10	—	FL	—	Deny
ICMP	Vlan10		VLAN 20	—	Deny
IP	ANY	—	ANY	—	Permit
					Implied Deny

R1-AZ:

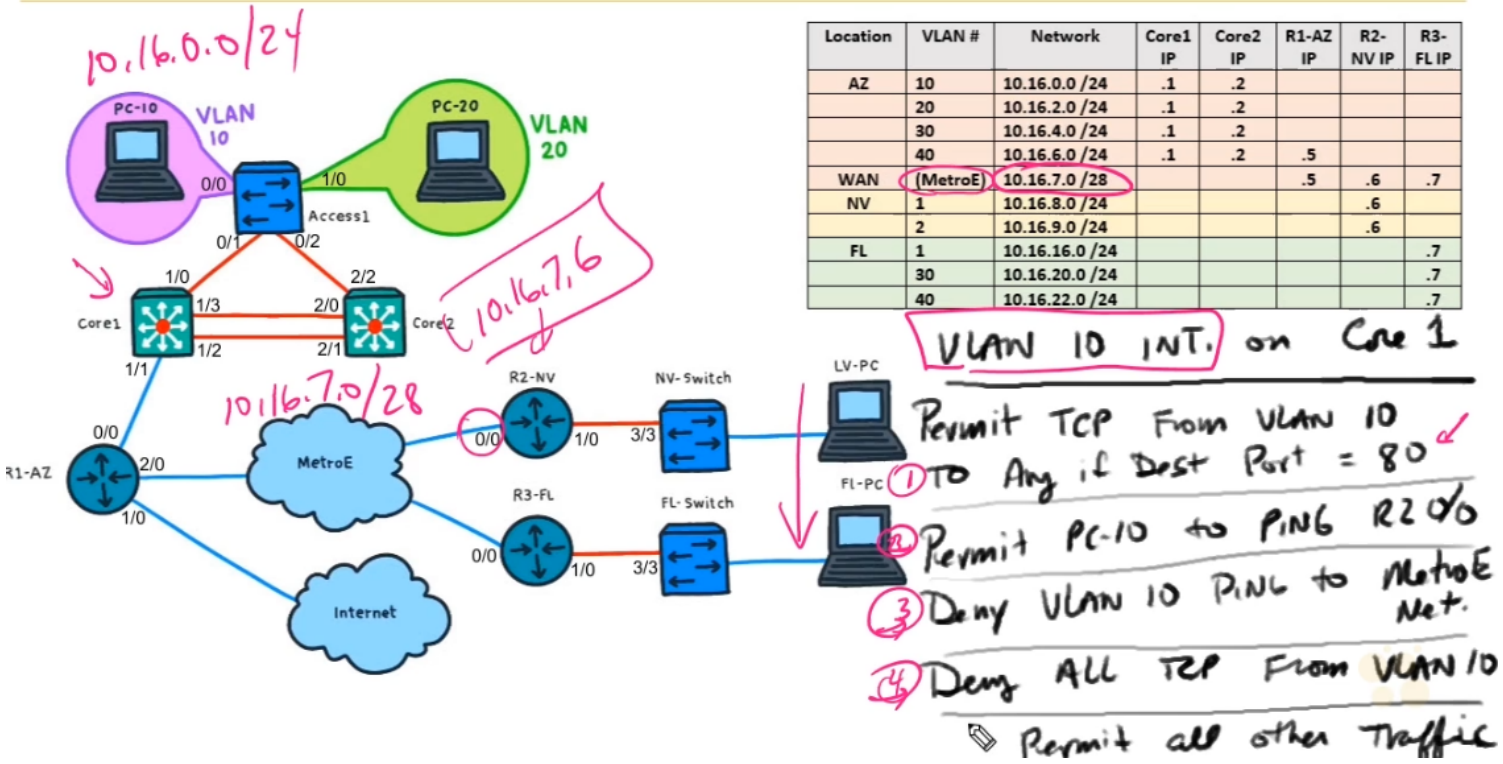
```
show access-list
conf term
ip access-list extended Our-ACL
    deny tcp host 10.16.0.10 any eq 21
    deny ip 10.16.0.0 0.0.0.255 10.16.16.0 0.0.7.255
    deny icmp 10.16.0.0 0.0.0.255 10.16.2.0 0.0.0.255
    permit ip any any
int gig 0/0
    ip access-group Our-ACL in
    end
```

```
R1-AZ#show access-lists
Extended IP access list Our-ACL
    10 deny tcp host 10.16.0.10 any eq ftp (3 matches)
    20 deny ip 10.16.0.0 0.0.0.255 10.16.16.0 0.0.7.255 (4 matches)
    30 deny icmp 10.16.0.0 0.0.0.255 10.16.2.0 0.0.0.255
    40 permit ip any any (155 matches)
```

- icmp still works because R1 in this scenario is not involved in routing vlan10 to vlan20 - it is L3 switch

Case study

Named ACLs



Location	VLAN #	Network	Core1 IP	Core2 IP	R1-AZ IP	R2-NV IP	R3-FL IP
AZ	10	10.16.0.0 /24	.1	.2			
	20	10.16.2.0 /24	.1	.2			
	30	10.16.4.0 /24	.1	.2			
	40	10.16.6.0 /24	.1	.2	.5		
WAN	(MetroE)	10.16.7.0 /28			.5	.6	.7
NV	1	10.16.8.0 /24				.6	
	2	10.16.9.0 /24				.6	
FL	1	10.16.16.0 /24					.7
	30	10.16.20.0 /24					.7
	40	10.16.22.0 /24					.7

Core1:

```
conf term
ip access-list extended The-List
    permit tcp 10.16.0.0 0.0.0.255 any eq 80
    permit icmp host 10.16.0.10 host 10.16.7.6
    deny icmp  10.16.0.0 0.0.0.255 10.16.7.0 0.0.0.15 (permit is first so it will match permit first for specific host)
    deny  tcp 10.16.0.0 0.0.0.255 any
    permit ip any any
```

```
Core1(config-ext-nacl)#do show access-list The-List
Extended IP access list The-List
    10 permit tcp 10.16.0.0 0.0.0.255 any eq www
    20 permit icmp host 10.16.0.10 host 10.16.7.6
    30 deny icmp 10.16.0.0 0.0.0.255 10.16.7.0 0.0.0.15
    40 deny tcp 10.16.0.0 0.0.0.255 any
    50 permit ip any any
```

```
int vlan 10
    ip access-group The-List in
```

52 - Configure and Verify Cisco Port Security

switchport mode access
switchport host
- you don't wanna dynamic port, you should set if it is access or trunk

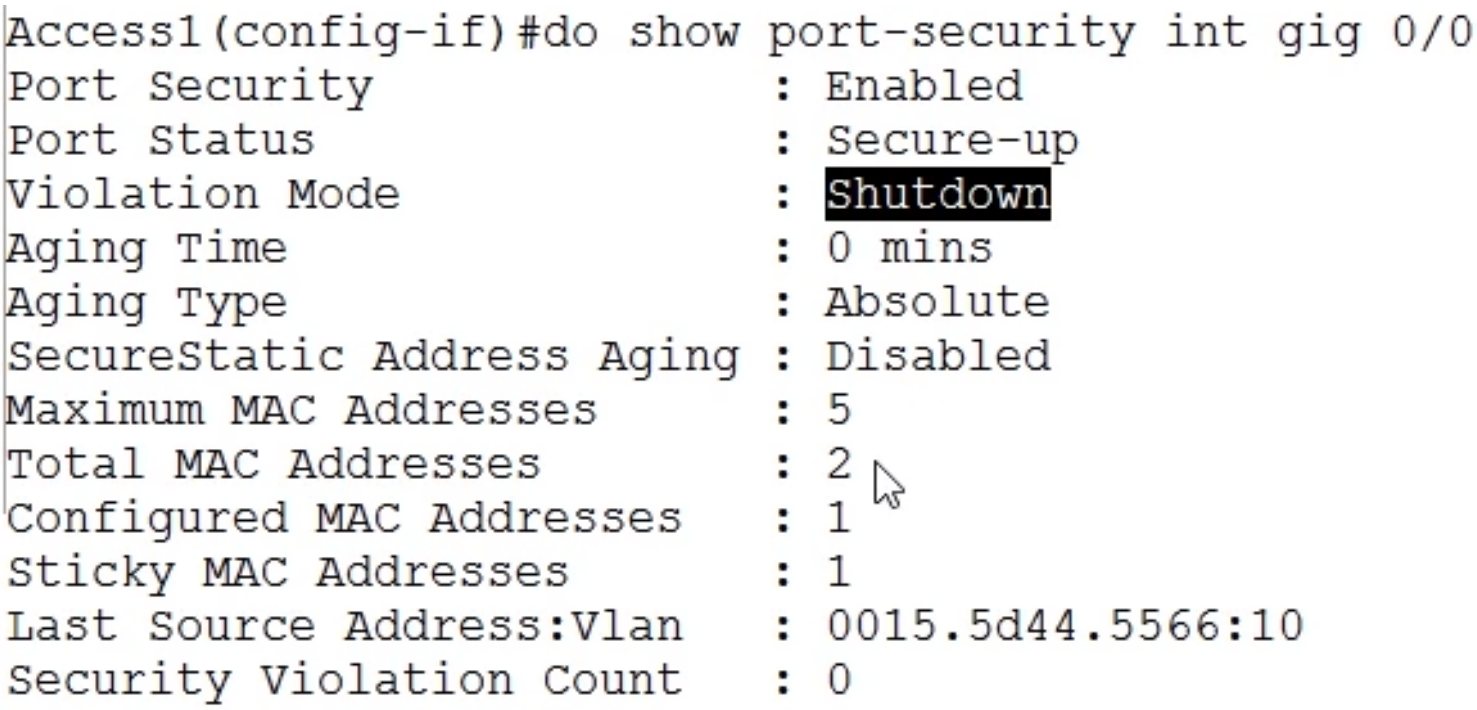
switchport port-security
- port can learn max 1 mac
- violation = shutdown port

Switch:
conf term
int g 0/0
    switchport mode access
    switchport access vlan 10
    switchport host (STP portfast enabled)

```
Access1(config-if)#do show port-security int gig 0/0
Port Security              : Disabled
Port Status                : Secure-down
Violation Mode             : Shutdown
Aging Time                 : 0 mins
Aging Type                 : Absolute
SecureStatic Address Aging : Disabled
Maximum MAC Addresses      : 1
Total MAC Addresses        : 0
Configured MAC Addresses   : 0
Sticky MAC Addresses       : 0
Last Source Address:Vlan   : 0000.0000.0000:0
Security Violation Count   : 0
```

Customizing port security

conf term
int g 0/0
    switchport mode access
    switchport access vlan 10
    switchport host (STP portfast enabled)
    switchport port-security maximum 5 (max 5 mac on the port)
    switchport port-security mac-address 0022.3344.5566 (specific mac connected, guaranteed it is one of the 5)
    switchport port-security mac-address sticky (dynamically learnt mac will be added to running config)
    switchport port-security violation shutdown (shutdown port if more than 5 mac = shutdown is default)
(set up but not enabled yet)
    switchport port-security (this will enable port security)



show interfaces status err-disabled (see if any ports are shut because of violation or something)

Error recovery

- if port is shutdown because of violation of policy then turn it on after 30s

```
conf term
int g 0/0
    switchport host
    switchport port-security maximum 5
    switchport port-security (to enable port sec)
    end
```

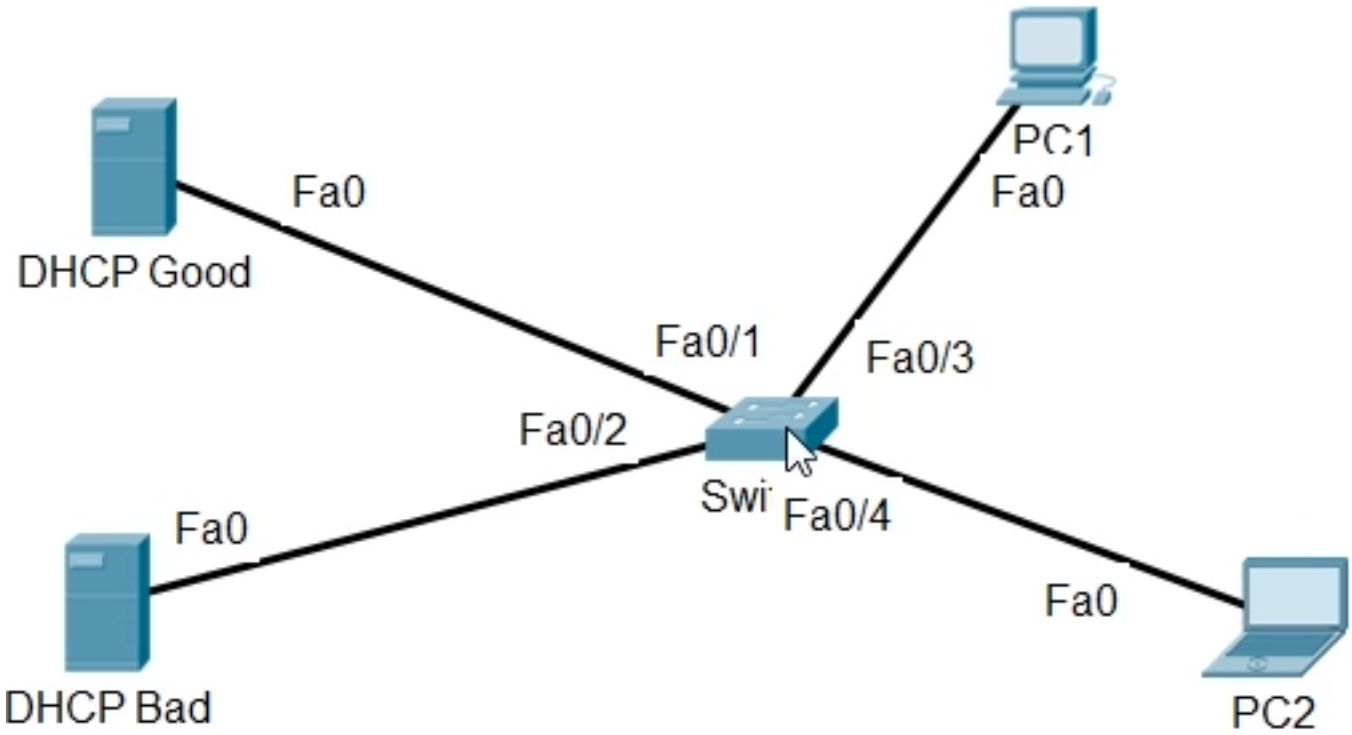
conf term
errdisable recovery cause psecure-violation
errdisable recovery interval 30 (30 seconds)

53 - Configure and Verify Cisco DHCP Snooping

DHCP Snooping

- pay attention to DHCP traffic
- say NO to all dhcp traffic except for port where your trusted DHCP server is plugged

- when ON, all ports are untrusted by default
- you have to set trunk ports as trusted
- option 82 includes info about DHCP relay agents
  → we should not include option 82 in dhcp messages as it could cause some problems



Switch:
conf term
ip dhcp snooping (enable)
ip dhcp snooping vlan 1 (configure it for vlan 1)
no ip dhcp snooping information option (disable insertion option 82)

```
Switch(config)#do show ip dhcp snooping
Switch DHCP snooping is enabled
DHCP snooping is configured on following VLANs:
1
Insertion of option 82 is disabled
Option 82 on untrusted port is not allowed
Verification of hwaddr field is enabled
Interface                  Trusted    Rate limit (pps)
------------------------   -------    ----------------
```

int fa 0/1
    ip dhcp snooping trust (trust this port - you have to as DHCP server is connected there)

```
Switch#show ip dhcp snooping
Switch DHCP snooping is enabled
DHCP snooping is configured on following VLANs:
1
Insertion of option 82 is disabled
Option 82 on untrusted port is not allowed
Verification of hwaddr field is enabled
Interface                    Trusted    Rate limit (pps)
-----------------------      -------    ----------------
FastEthernet0/1              yes        unlimited
FastEthernet0/2              no         unlimited
FastEthernet0/3              no         unlimited
FastEthernet0/4              no         unlimited
```

Source guard

- protect from spoofing addresses, verify IP or IP and MAC

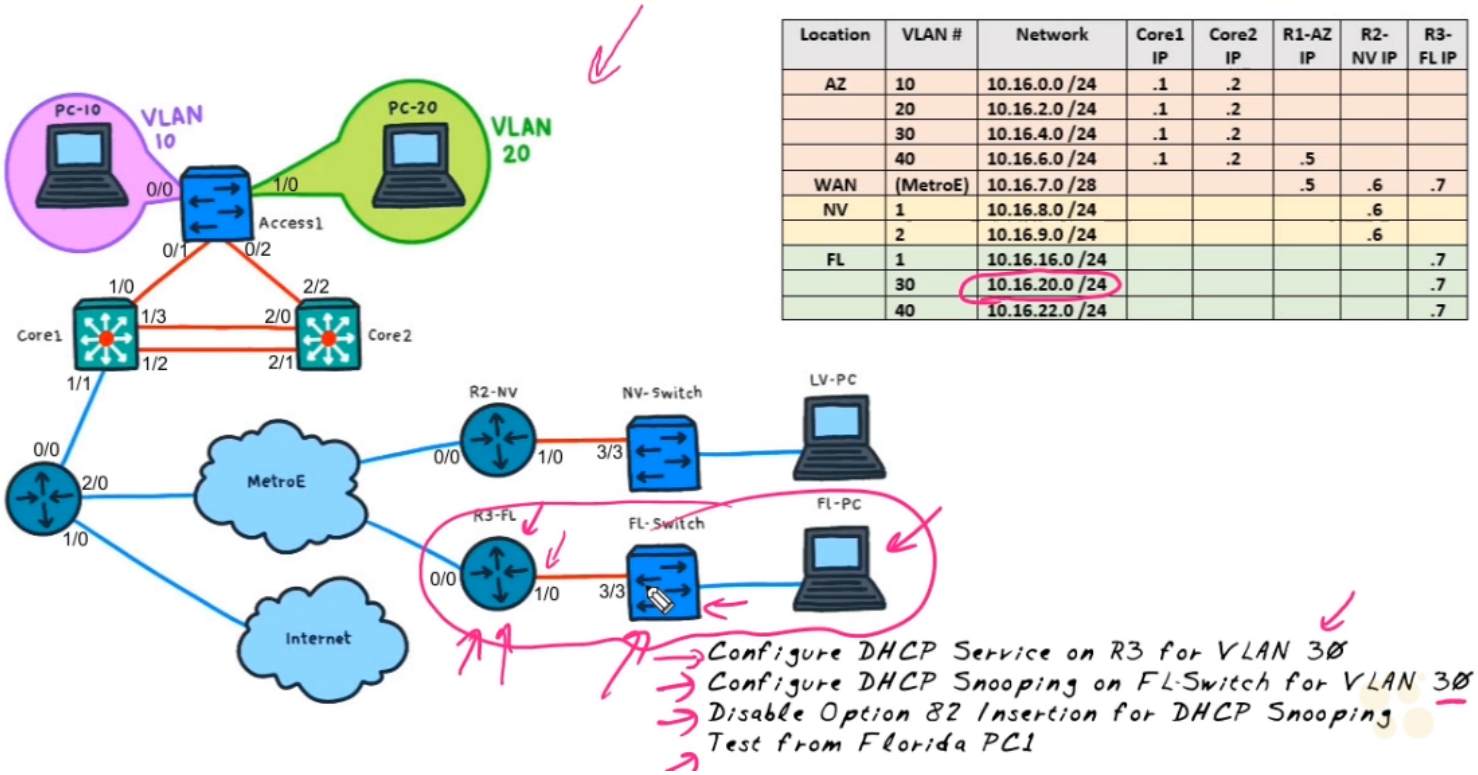
Switch:

```
Access-Switch#show ip dhcp snooping binding
MacAddress          IpAddress        Lease(sec)  Type           VLAN  Interface
------------------  ---------------  ----------  -------------  ----  --------------------
00:50:79:66:68:04   10.16.20.101     85538       dhcp-snooping  30    GigabitEthernet0/1
Total number of bindings: 1
```

conf term
int g 0/1
ip verify source port-security (verify MAC as well, without port-security it would verify just IP)

```
Access-Switch#show ip verify source
Interface  Filter-type  Filter-mode  IP-address       Mac-address        Vlan
---------  -----------  -----------  ---------------  -----------------  ----
Gi0/1      ip-mac       active       10.16.20.101     permit-all         30
```

Case study



Location	VLAN #	Network	Core1 IP	Core2 IP	R1-AZ IP	R2-NV IP	R3-FL IP
AZ	10	10.16.0.0 /24	.1	.2			
	20	10.16.2.0 /24	.1	.2			
	30	10.16.4.0 /24	.1	.2			
	40	10.16.6.0 /24	.1	.2	.5		
WAN	(MetroE)	10.16.7.0 /28			.5	.6	.7
NV	1	10.16.8.0 /24				.6	
	2	10.16.9.0 /24				.6	
FL	1	10.16.16.0 /24					.7
	30	10.16.20.0 /24					.7
	40	10.16.22.0 /24					.7

R3-FL:

```
conf term
ip dhcp pool OURPOOL
    network 10.16.20.0 /24
    default-gateway 10.16.20.7
    dns-server 8.8.8.8
    exit
ip dhcp excluded-address 10.16.20.1 10.16.20.10
end
```

```
R3-FL#show ip dhcp pool

Pool OURPOOL :
 Utilization mark (high/low)    : 100 / 0
 Subnet size (first/next)       : 0 / 0
 Total addresses                : 254
 Leased addresses               : 0
 Excluded addresses             : 10
 Pending event                  : none
 1 subnet is currently in the pool :
 Current index        IP address range                    Leased/Excluded/Total
 10.16.20.1           10.16.20.1       - 10.16.20.254      0     / 10    / 254
```

FL-Switch:

```
conf term
ip dhcp snooping
ip dhcp snooping vlan 1
no ip dhcp snooping information option
```

```
FL-Switch(config)#do show ip dhcp snooping
Switch DHCP snooping is enabled
Switch DHCP gleaning is disabled
DHCP snooping is configured on following VLANs:
1
DHCP snooping is operational on following VLANs:
1
DHCP snooping is configured on the following L3 Interfaces:

Insertion of option 82 is disabled
   circuit-id default format: vlan-mod-port
   remote-id: 00dc.d2be.ff00 (MAC)
Option 82 on untrusted port is not allowed
Verification of hwaddr field is enabled
Verification of giaddr field is enabled
DHCP snooping trust/rate is configured on the following Interfaces:

Interface                  Trusted    Allow option    Rate limit (pps)
-----------------------    -------    ------------    ----------------
```

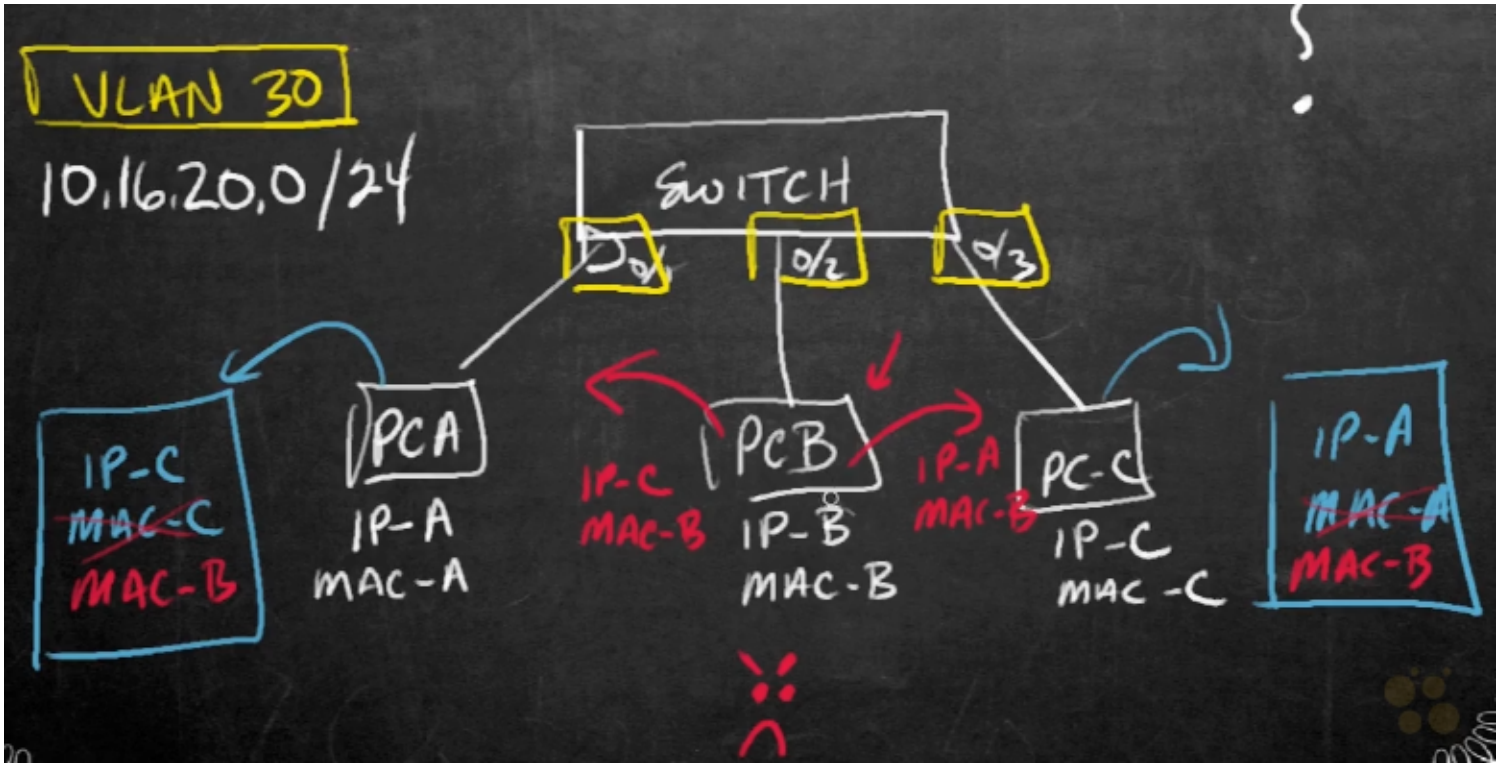
int g 3/3
ip dhcp snooping trust
end
ip dhcp snooping vlan 30 (client is in vlan 30)

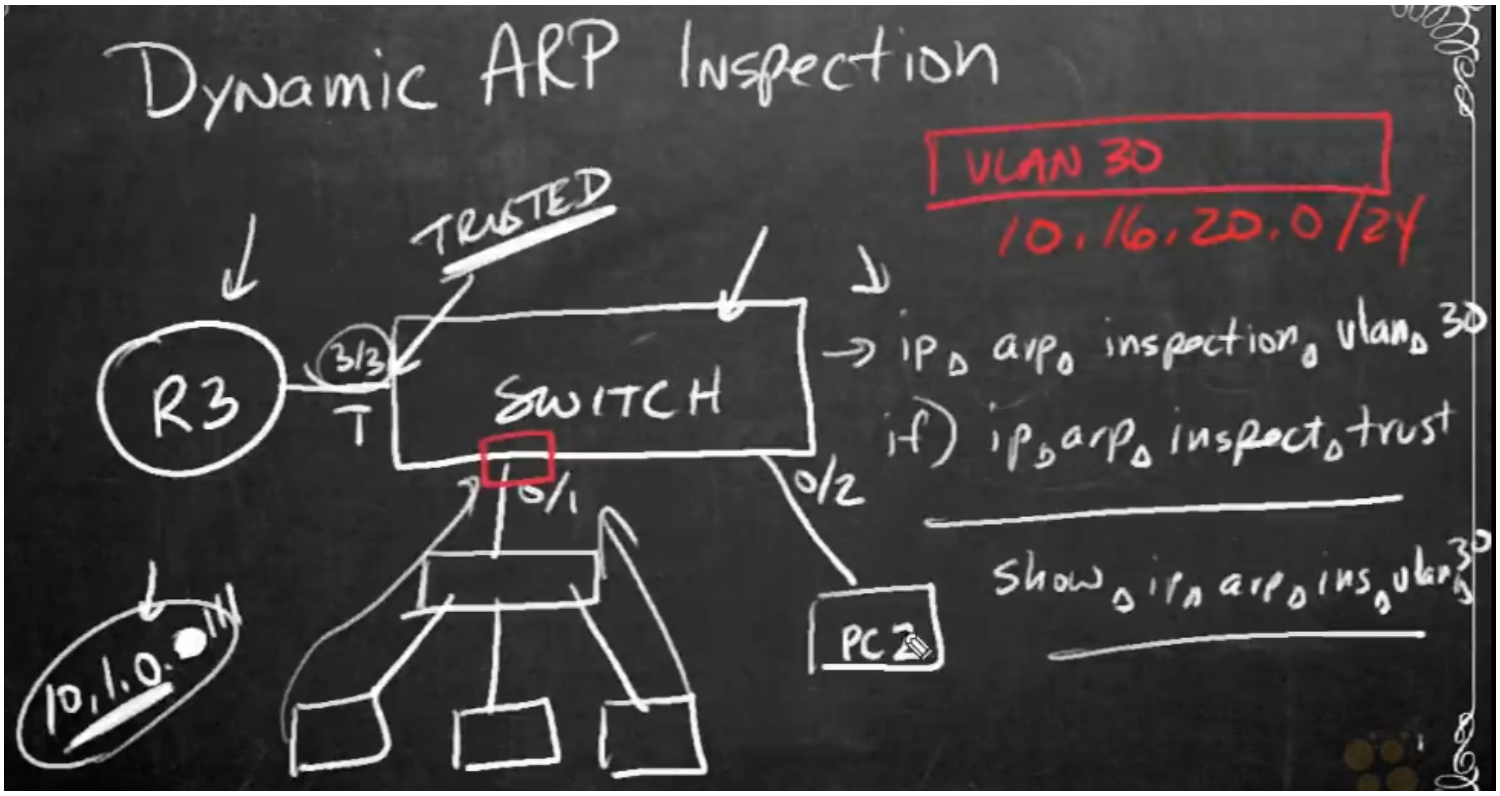
```
FL-Switch#show ip dhcp snooping
Switch DHCP snooping is enabled
Switch DHCP gleaning is disabled
DHCP snooping is configured on following VLANs:
1,30
DHCP snooping is operational on following VLANs:
1,30
DHCP snooping is configured on the following L3 Interfaces:

Insertion of option 82 is disabled
   circuit-id default format: vlan-mod-port
   remote-id: 00dc.d2be.ff00 (MAC)
Option 82 on untrusted port is not allowed
Verification of hwaddr field is enabled
Verification of giaddr field is enabled
DHCP snooping trust/rate is configured on the following Interfaces:

Interface                  Trusted    Allow option    Rate limit (pps)
-----------------------    -------    ------------    ----------------
GigabitEthernet3/3         yes        yes             unlimited
  Custom circuit-ids:
```

54 - Configure and Verify Cisco Dynamic ARP Inspection

Dynamic ARP inspection - DAI

- it can use DHCP snooping table to capture all mappings
- or config manually





FL-Switch:

```
FL-Switch#show ip dhcp snooping binding
MacAddress          IpAddress        Lease(sec)  Type           VLAN  Interface
------------------  ---------------  ----------  -------------  ----  --------------------
00:50:79:66:68:00   10.16.22.101     85604       dhcp-snooping   40    GigabitEthernet0/2
00:15:5D:77:77:01   10.16.20.102     85689       dhcp-snooping   30    GigabitEthernet0/1
Total number of bindings: 2
```

```
FL-Switch#show ip arp inspection vlan 30

Source Mac Validation      : Disabled
Destination Mac Validation : Disabled
IP Address Validation      : Disabled

 Vlan     Configuration    Operation   ACL Match          Static ACL
 ----     -------------    ---------   ---------          ----------
   30     Disabled         Inactive

 Vlan     ACL Logging      DHCP Logging      Probe Logging
 ----     -----------      ------------      -------------
   30     Deny             Deny              Off
FL-Switch#
```

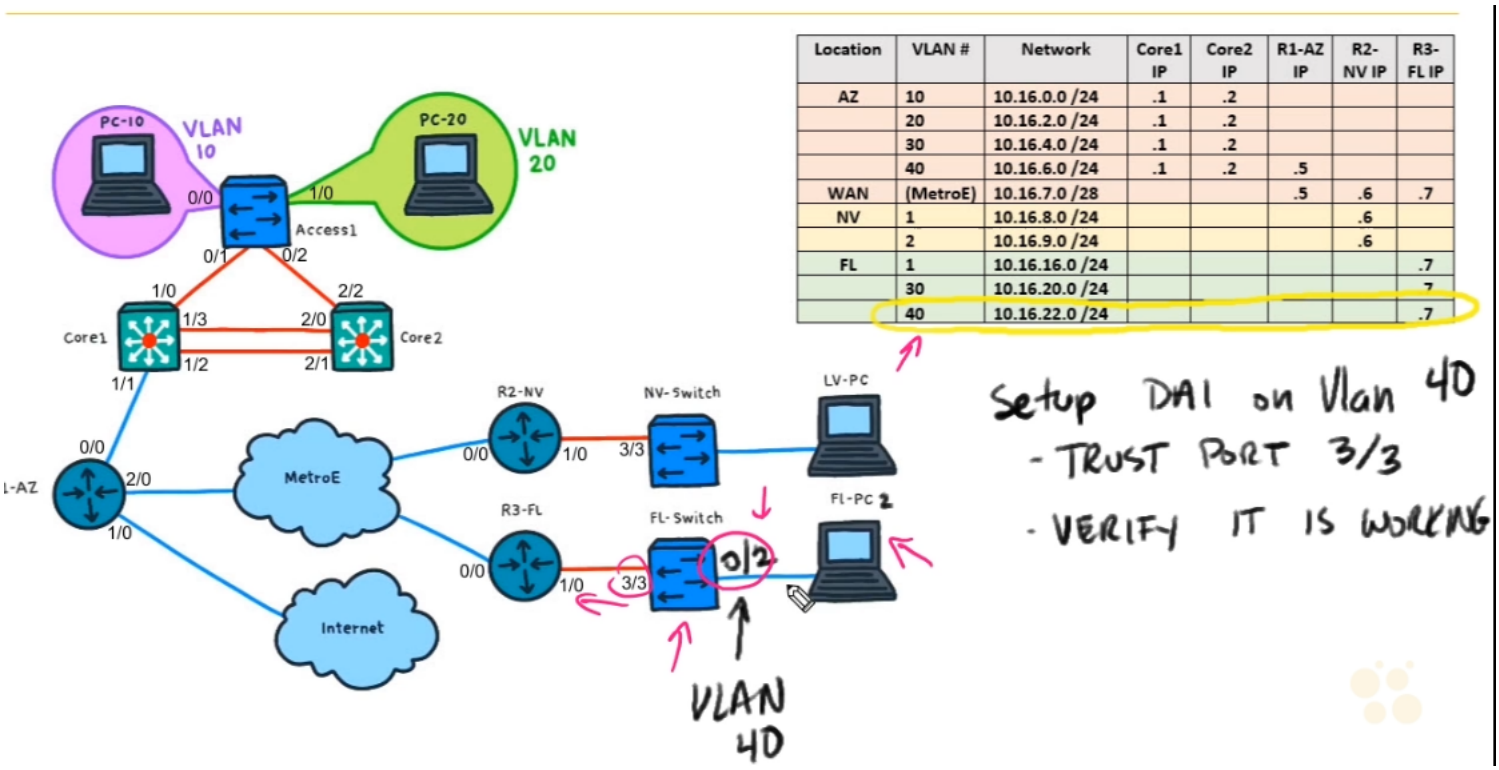
conf term
inf gig 3/3
    ip arp inspection trust (made the port trusted)
    exit
ip arp inspection vlan 30 (apply it for vlan 30)
arp access-list OUR-LIST
    permit ip host 10.1.0.111 mac host 0015.5d67.8322 (manually add 2 hosts)
    permit ip host 10.1.0.120 mac host 0015.5d44.5566
    exit
ip arp inspection filter OUT-LIST vlan 30 (apply list)

```
FL-Switch#show arp access-list
ARP access list OUR-LIST
    permit ip host 10.1.0.111 mac host 0015.5d67.8322
    permit ip host 10.1.0.120 mac host 0015.5d44.5566
```

- by default dynamic ARP inspection limited to 15 per port
ip arp inspection limit rate 100 (change rate limit to 100)

- you can enable (disabled by default)
  → source mac validation
  → destination mac validation
  → ip address validation
ip arp inspection validate dst-mac ip src-mac

Case study



Location	VLAN #	Network	Core1 IP	Core2 IP	R1-AZ IP	R2-NV IP	R3-FL IP
AZ	10	10.16.0.0 /24	.1	.2			
	20	10.16.2.0 /24	.1	.2			
	30	10.16.4.0 /24	.1	.2			
	40	10.16.6.0 /24	.1	.2	.5		
WAN	(MetroE)	10.16.7.0 /28			.5	.6	.7
NV	1	10.16.8.0 /24				.6	
	2	10.16.9.0 /24				.6	
FL	1	10.16.16.0 /24					.7
	30	10.16.20.0 /24					.7
	40	10.16.22.0 /24					.7

FL-Switch:

```
FL-Switch#show vlan brief

VLAN Name                             Status    Ports
---- -------------------------------- --------- -------------------------------
1    default                          active    Gi0/0, Gi0/3, Gi1/0, Gi1/1
                                                Gi1/2, Gi1/3, Gi2/0, Gi2/1
                                                Gi2/2, Gi2/3, Gi3/0, Gi3/1
                                                Gi3/2
30   Engineering                      active    Gi0/1
40   Sales                            active    Gi0/2
1002 fddi-default                     act/unsup
1003 token-ring-default               act/unsup
1004 fddinet-default                  act/unsup
1005 trnet-default                    act/unsup
```

conf term
int gig 3/3
 ip arp inspection trust
 exit
ip arp inspection vlan 40

```
FL-Switch#show ip dhcp snooping binding
MacAddress          IpAddress        Lease(sec)  Type           VLAN  Interface
------------------  ---------------  ----------  -------------  ----  --------------------
00:50:79:66:68:00   10.16.22.101     85681       dhcp-snooping   40   GigabitEthernet0/2
00:15:5D:77:77:01   10.16.20.102     85766       dhcp-snooping   30   GigabitEthernet0/1
Total number of bindings: 2
```

FL-PC2:

```
FL-PC2> show ip

NAME        : FL-PC2[1]
IP/MASK     : 10.16.22.101/24
GATEWAY     : 10.16.22.7
DNS         :
DHCP SERVER : 10.16.22.7
DHCP LEASE  : 85105, 86400/43200/75600
MAC         : 00:50:79:66:68:00
LPORT       : 10008
RHOST:PORT  : 127.0.0.1:10009
MTU:        : 1500
```

ping 10.16.22.7 (works)
ip 10.16.22.200 (set static ip)
ping 10.16.22.7 (dosn't work, because switch killed arp request)
ip dhcp (back to dhcp address)
ping 10.16.22.7 (works)

55 - Describe Remote Access and Site-to-Site VPNs

Virtual Private Network - VPN

1. **site to site VPN**
2. **remote access VPN**

Cryptography

- confidentiality
→ the communication is confidential and third party can't read it
→ encryption
⇒ AES
• advanced encryption standard
• 128, 192, 256... bigger=better
- integrity
→ verify that nothing has been modified
→ hashing
⇒ md5, sha

Site to site VPN

- using IPsec
→ collection of protocols etc, it is umbrella
- two routers establish SA=Security Association between each other

- phase 1 tunnel
→ ISAKMP SA - Internet Security Association Key Management Protocol Security Association
⇒ negotiation what encryption, hast, key

- phase 2 tunnel
  → IPsec SA
    ⇒ tunnel for user data

- you have to create Crypto Map and apply it to the interfaces

R1:

```
R1-AZ#show crypto isakmp sa
IPv4 Crypto ISAKMP SA
dst             src             state          conn-id status

IPv6 Crypto ISAKMP SA

R1-AZ#show crypto ipsec sa
No SAs found
R1-AZ#
R1-AZ#
R1-AZ#show cry
R1-AZ#show crypto map
Crypto Map IPv4 "OUR-MAP" 10 ipsec-isakmp
        Peer = 10.16.7.6
        Extended IP access list 100
            access-list 100 permit icmp 10.16.0.0 0.0.0.255 10.16.8.0 0.0.0.255
        Security association lifetime: 4608000 kilobytes/3600 seconds
        Responder-Only (Y/N): N
        PFS (Y/N): Y
        DH group:  group5
        Transform sets={
                OUR-SET:  { esp-256-aes esp-sha256-hmac  } ,
        }
        Interfaces using crypto map OUR-MAP:
```

```
conf term
int gig 2/0
    crypto map OUR-MAP
    end
```

R2:

```
conf term
int gig 0/0
    crypto map OUR-MAP
    end
```

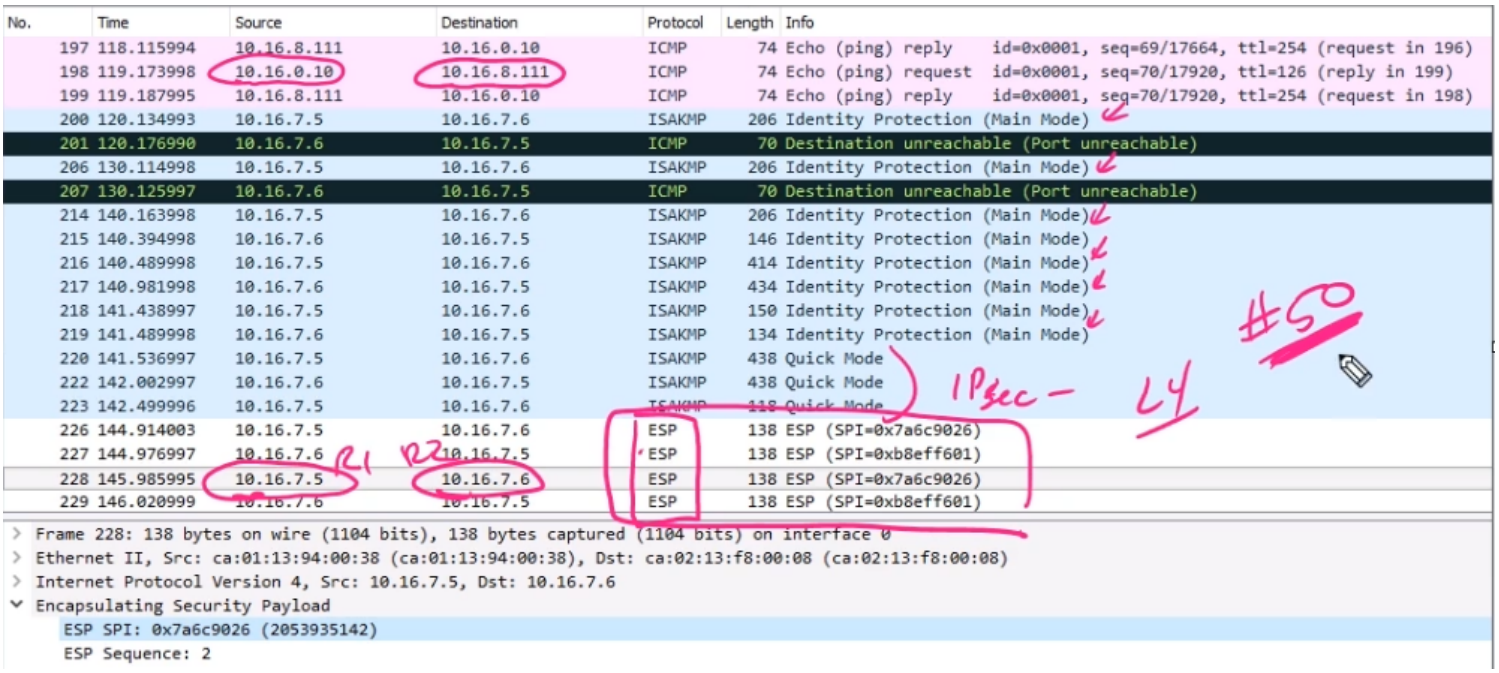
R1:

```
R1-AZ#show crypto isakmp sa
IPv4 Crypto ISAKMP SA
dst             src             state          conn-id status
10.16.7.6       10.16.7.5       QM_IDLE           1001 ACTIVE

IPv6 Crypto ISAKMP SA
```

```
R1-AZ#show crypto ipsec sa

interface: GigabitEthernet2/0
    Crypto map tag: OUR-MAP, local addr 10.16.7.5

   protected vrf: (none)
   local  ident (addr/mask/prot/port): (10.16.0.0/255.255.255.0/1/0)
   remote ident (addr/mask/prot/port): (10.16.8.0/255.255.255.0/1/0)
   current_peer 10.16.7.6 port 500
     PERMIT, flags={origin_is_acl,}
    #pkts encaps: 34, #pkts encrypt: 34, #pkts digest: 34
    #pkts decaps: 33, #pkts decrypt: 33, #pkts verify: 33
    #pkts compressed: 0, #pkts decompressed: 0
    #pkts not compressed: 0, #pkts compr. failed: 0
    #pkts not decompressed: 0, #pkts decompress failed: 0
    #send errors 0, #recv errors 0

     local crypto endpt.: 10.16.7.5, remote crypto endpt.: 10.16.7.6
     path mtu 1500, ip mtu 1500, ip mtu idb GigabitEthernet2/0
     current outbound spi: 0x1ACF41C6(449790406)
     PFS (Y/N): Y, DH group: group5

     inbound esp sas:
      spi: 0x1026D8E2(270981346)
 --More--
```

```
R1-AZ#show crypto engine connections active
Crypto Engine Connections

   ID  Type    Algorithm           Encrypt  Decrypt LastSeqN IP-Address
    1  IPsec   AES256+SHA256             0        0        0 10.16.7.5
    2  IPsec   AES256+SHA256             0        0        0 10.16.7.5
    3  IPsec   AES256+SHA256             0       33       33 10.16.7.5
    4  IPsec   AES256+SHA256            34        0        0 10.16.7.5
 1001  IKE     SHA256+AES                0        0        0 10.16.7.5
```

Packet capture:



No.	Time	Source	Destination	Protocol	Length	Info
197	118.115994	10.16.8.111	10.16.0.10	ICMP	74	Echo (ping) reply id=0x0001, seq=69/17664, ttl=254 (request in 196)
198	119.173998	10.16.0.10	10.16.8.111	ICMP	74	Echo (ping) request id=0x0001, seq=70/17920, ttl=126 (reply in 199)
199	119.187995	10.16.8.111	10.16.0.10	ICMP	74	Echo (ping) reply id=0x0001, seq=70/17920, ttl=254 (request in 198)
200	120.134993	10.16.7.5	10.16.7.6	ISAKMP	206	Identity Protection (Main Mode)
201	120.176990	10.16.7.6	10.16.7.5	ICMP	70	Destination unreachable (Port unreachable)
206	130.114998	10.16.7.5	10.16.7.6	ISAKMP	206	Identity Protection (Main Mode)
207	130.125997	10.16.7.6	10.16.7.5	ICMP	70	Destination unreachable (Port unreachable)
214	140.163998	10.16.7.5	10.16.7.6	ISAKMP	206	Identity Protection (Main Mode)
215	140.394998	10.16.7.6	10.16.7.5	ISAKMP	146	Identity Protection (Main Mode)
216	140.489998	10.16.7.5	10.16.7.6	ISAKMP	414	Identity Protection (Main Mode)
217	140.981998	10.16.7.6	10.16.7.5	ISAKMP	434	Identity Protection (Main Mode)
218	141.438997	10.16.7.5	10.16.7.6	ISAKMP	150	Identity Protection (Main Mode)
219	141.489998	10.16.7.6	10.16.7.5	ISAKMP	134	Identity Protection (Main Mode)
220	141.536997	10.16.7.5	10.16.7.6	ISAKMP	438	Quick Mode
222	142.002997	10.16.7.6	10.16.7.5	ISAKMP	438	Quick Mode
223	142.499996	10.16.7.5	10.16.7.6	ISAKMP	118	Quick Mode
226	144.914003	10.16.7.5	10.16.7.6	ESP	138	ESP (SPI=0x7a6c9026)
227	144.976997	10.16.7.6	10.16.7.5	ESP	138	ESP (SPI=0xb8eff601)
228	145.985995	10.16.7.5	10.16.7.6	ESP	138	ESP (SPI=0x7a6c9026)
229	146.020999	10.16.7.6	10.16.7.5	ESP	138	ESP (SPI=0xb8eff601)


```
> Frame 228: 138 bytes on wire (1104 bits), 138 bytes captured (1104 bits) on interface 0
> Ethernet II, Src: ca:01:13:94:00:38 (ca:01:13:94:00:38), Dst: ca:02:13:f8:00:08 (ca:02:13:f8:00:08)
> Internet Protocol Version 4, Src: 10.16.7.5, Dst: 10.16.7.6
v Encapsulating Security Payload
    ESP SPI: 0x7a6c9026 (2053935142)
    ESP Sequence: 2
```

Remote Access VPN

- SSL/TLS - overhead, it is TCP based so lot of acks
- DTLS - Datagram transport layer security, UDP based
- could be IPsec IKEv1/v2
- PPTP, L2TP

Split tunneling

- just traffic which has to go to VPN tunnel (10.16.0.0/21), rest outside

56 - Describe, Configure, and Verify Wireless Security protocols

Security options:

- disabled/none
- WEP - broken, easy to crack
- WPA - cracked
- WPA2
- WPA3

- WPA/WPA2/WPA3
  → Personal
    ⇒ PSK=pre-shared key=password (aes-256)
  → Enterprise

⇒ AAA server (RADIUS)

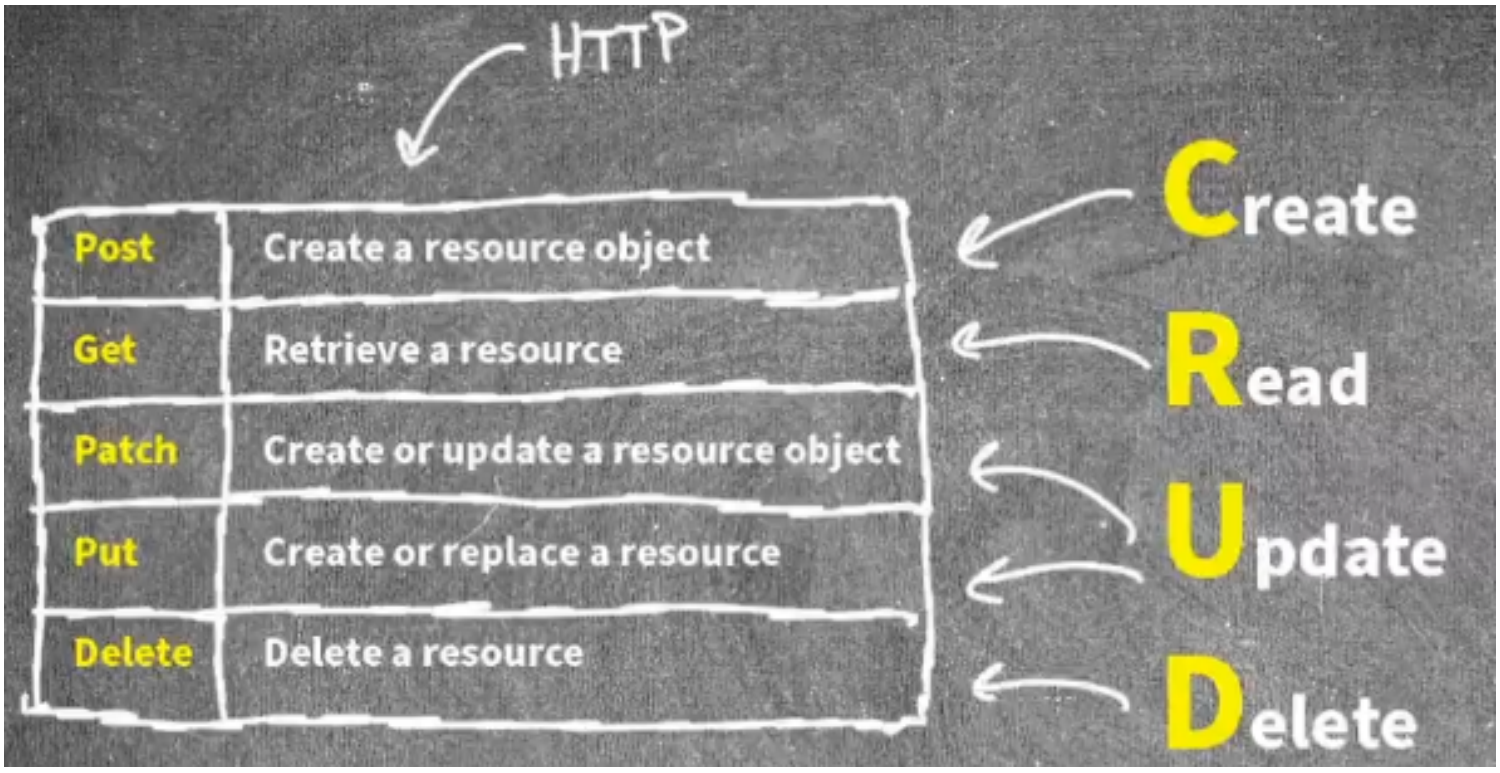
Options
- SOHO - router/switch/ap all in one
- Small - autonomous AP
- Medium/Enterprise - controller and lightweight AP

57 - What is Network Automation

- process of automating the configuring, managing, testing, deploying and operating of physical and virtual devices within a network

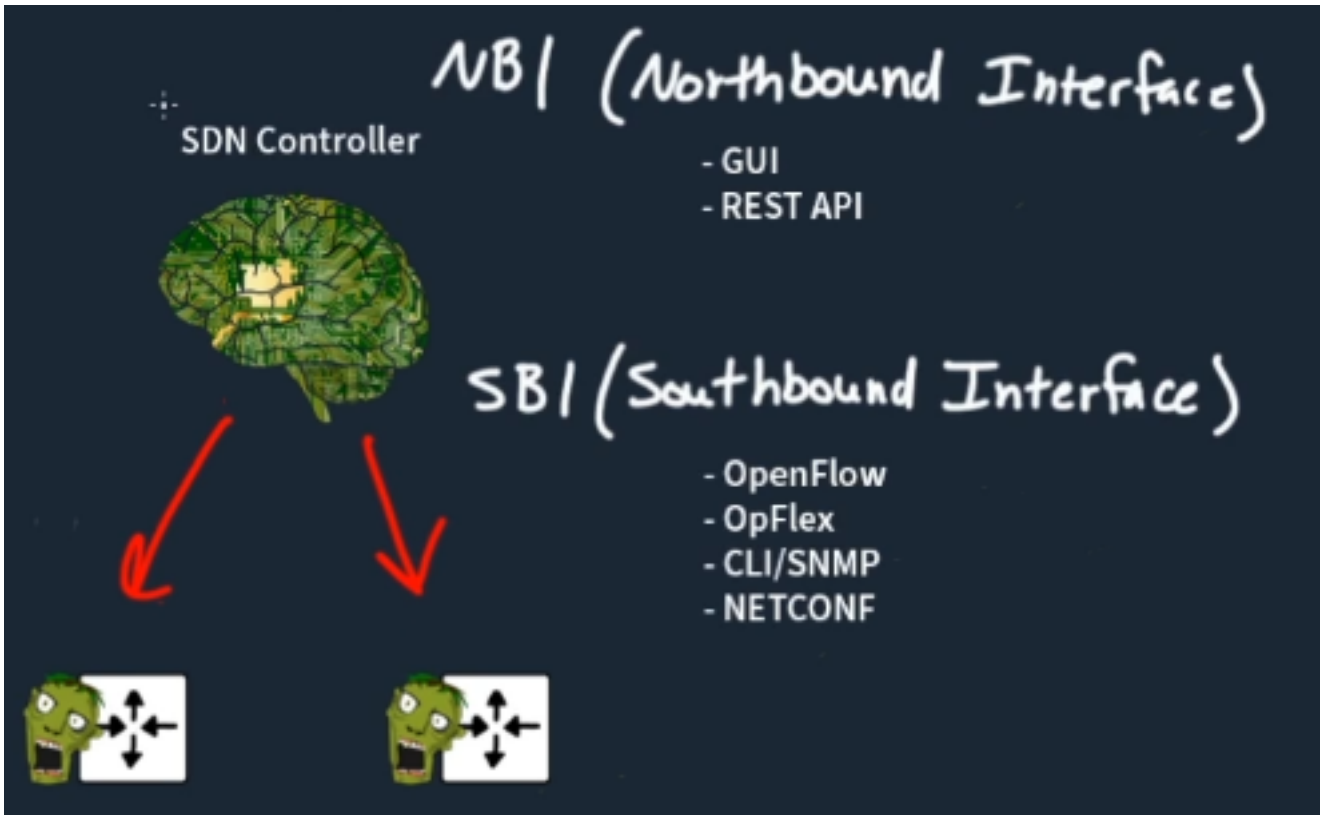
58 - Use REST APIs and JSON

REST=REpresentation State Transfer
- client-server
  → you have server providing API and client sending calls
- state-less
  → doesn't rely on previous interactions, you have to provide everything in a call, you have to give context
- cacheable
  → is data cacheable or not, some can be cached and some not because they change a lot
- uniform interface
- layered
- code on-demand



59 - Controller-Based Networking

SDN - software defined networking

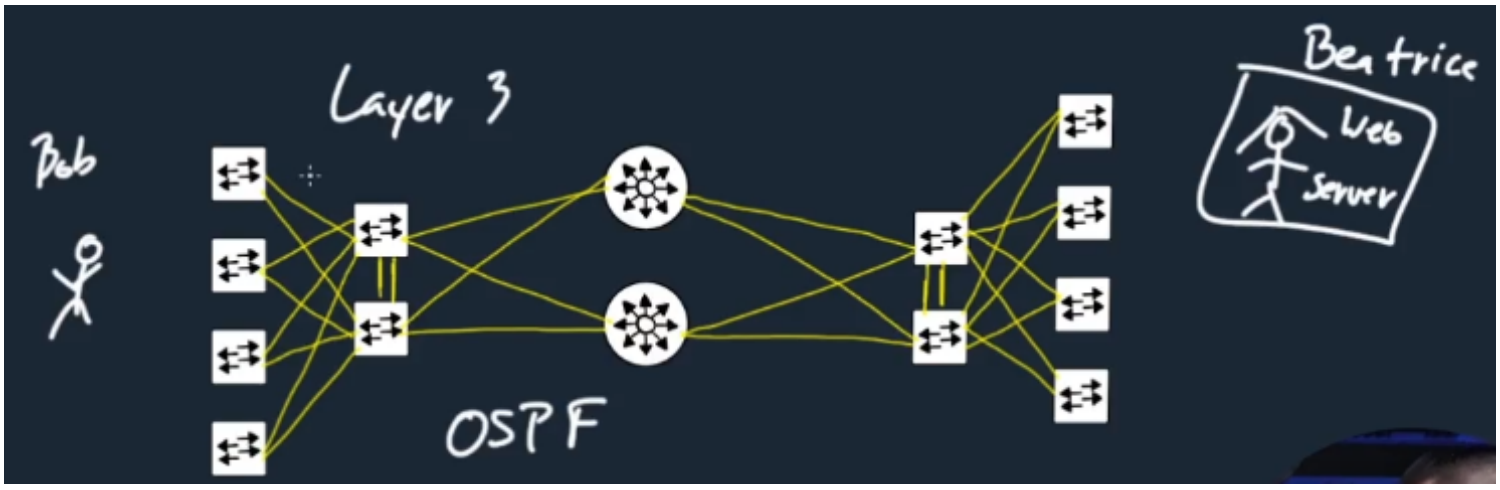


SBI - Southbound Interface
- ways how controller can control devices
  → OpenFlow - opensource
  → OpFlex - used with Cisco ACI
  → CLI/SNMP - DNA Center uses it
  → NETCONF - DNA Center uses it as well

NBI - Northbound Interface
- can be GUI
- REST API

IBN - Intent Based Networking

SDN architectures



- fabric

→ everything you see on the picture

- overlay

→ virtual network tunnelled over your underlay devices

→ tunnelled using **VXLAN**

- underlay

→ physical network that provides connectivity for the overlay

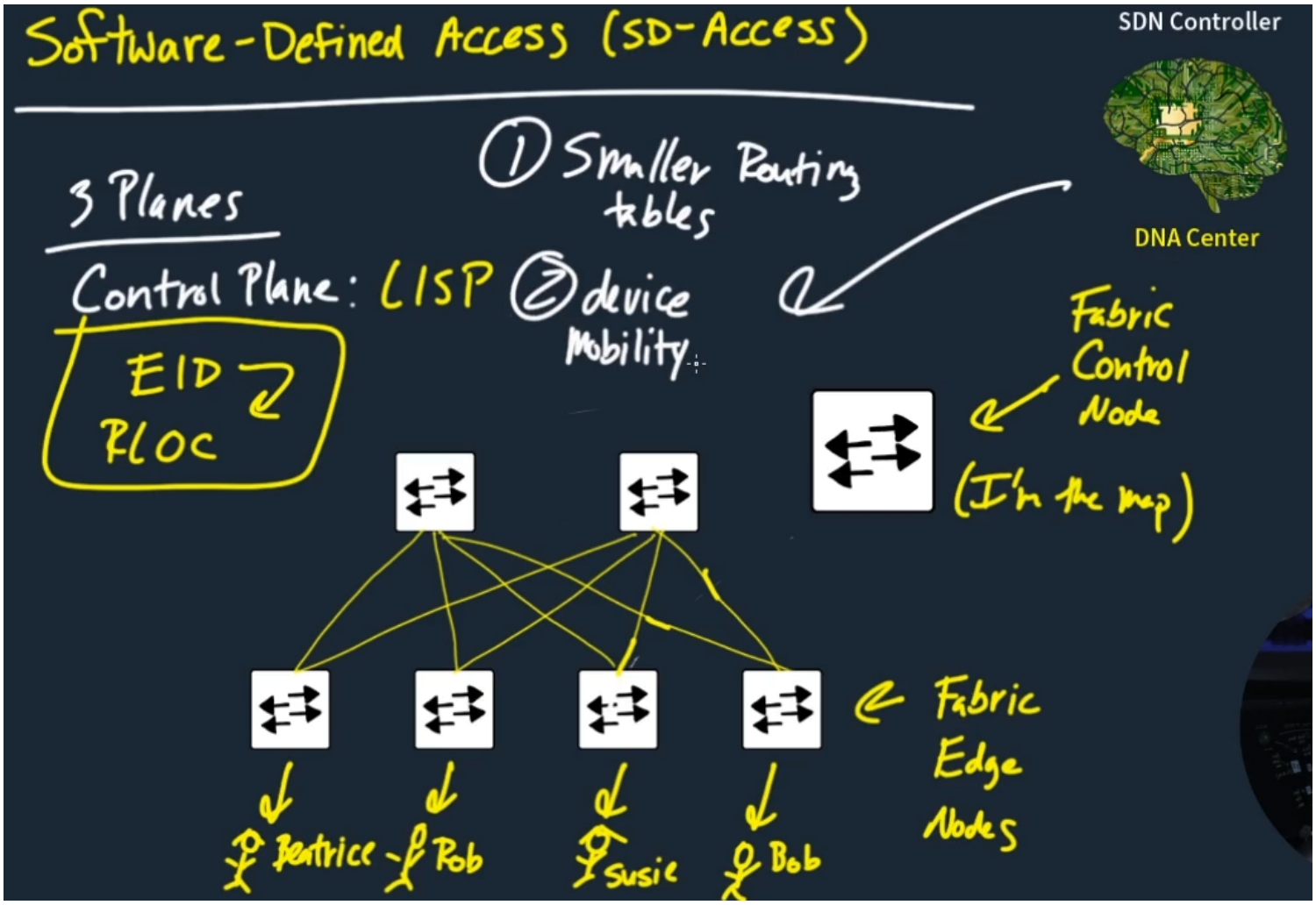
ACI - Application Centric Infrastructure

- for data centers
- ACI
  → Southbound int - OpFlex
  → Northbound - GUI
- Spine/Leaf aka CLOS
  → endpoints connected to leafs only
  → leafs connected to all spines (distribution layer - leafs, core layer - spines)
  → everything is 3 hops away
- SDN Controller = in ACI world it is APIC (application policy infrastructure controller)
- you say I wanna these devices talk to these devices

SD-Access

- for campus environment
- DNA Center handles the Underlay
- 3 planes of operation
  → Control Plane
    ⇒ LISP - Location Identity Separation Protocol
      • like DNS for your network
      • identify each host and devices with EID (Endpoint ID) and matches it with RLOC (Routing Locator)
  → Data Plane
    ⇒ VXLAN (Virtual Extensible LAN)
      • allows us to do L2 over L3, visualised
  → Policy Plane
    ⇒ CTS (Cisco Trust Sec)
      • use SGT (Scalable Group Tags) to apply policy

- Fabric control node
  → keep the map with all EID/RLOC
- Fabric edge node
  → hosts connected to it



DNA Center

- has history so you can see issues back when they happened
- other solutions
  → Solar Winds
  → Prime Infrastructure
- shared capabilities
  - network discovery
  - topology
  - network mapping
  - network automation

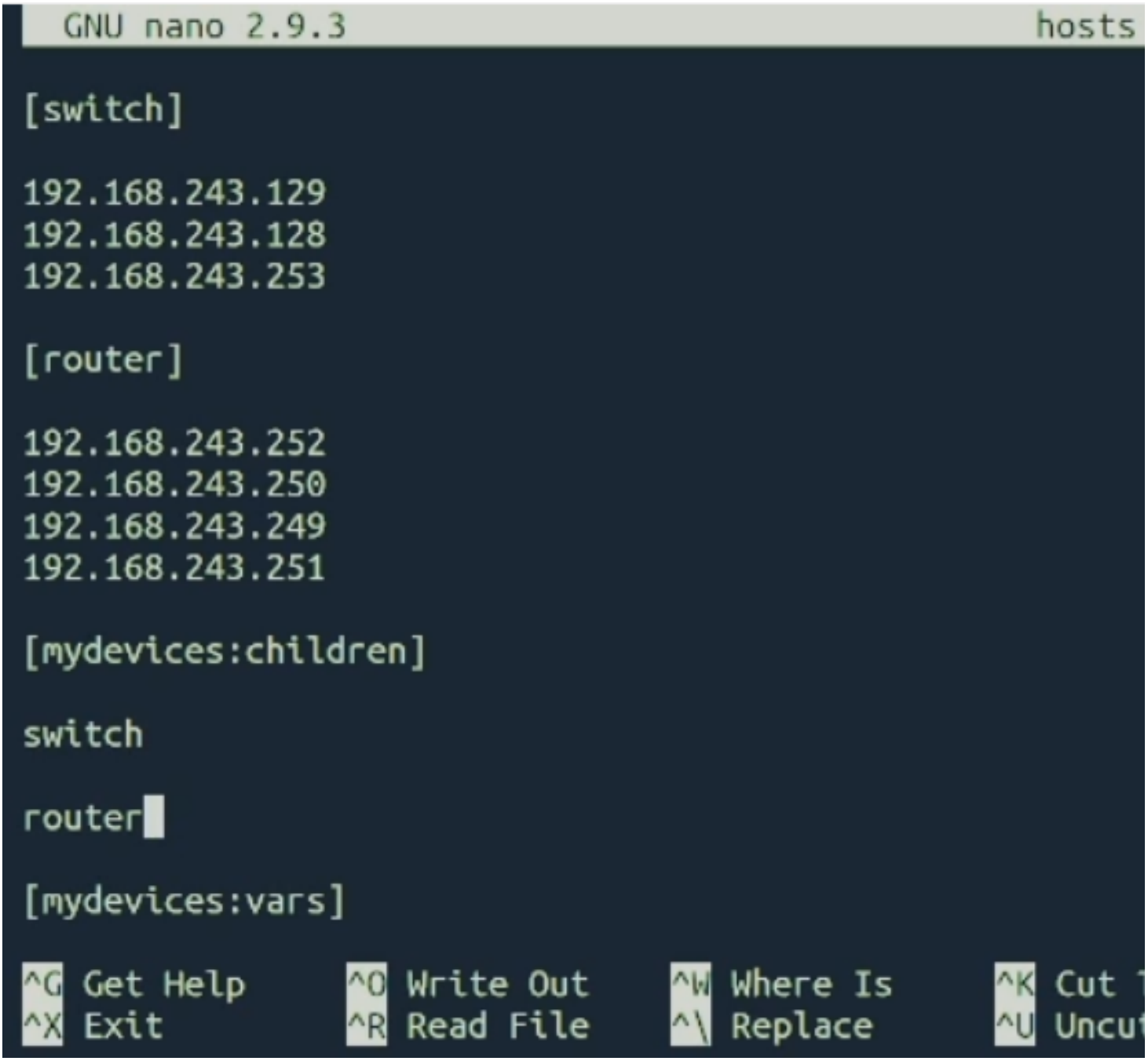
60 - Network Automation Tools Ansible, Puppet, and Chef

DevOps
- automating processes

IaaC
- Infrastructure as a Code
- solving
  → Configuration Drift
  → Idempotence - config will be applied only if it results in a change (don't apply things which are already there)

Ansible
- agent-less
  → you don't have to install anything on a client
- Python based
- connect via SSH and run commands
- usually running on Linux based systems
- uses YAML for serialization
- opensource and free
- pushy model → “take it take it”

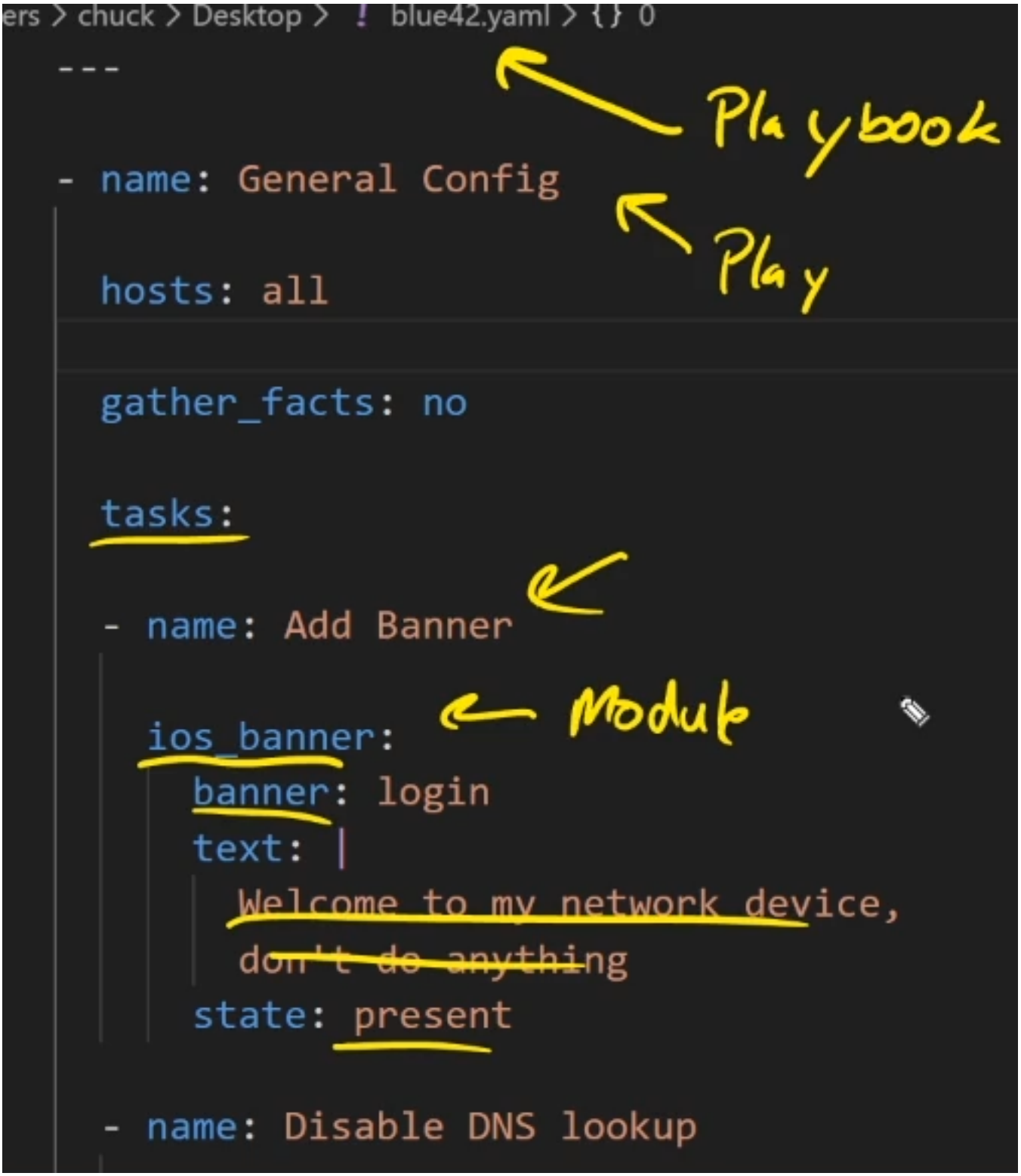
1. Inventory
  ◇ create file with all devices and variables with user/pass
  ◇ /etc/ansible/hosts



2. Playbook

◇ YAML

◇ Playbook → Plays → Tasks



3. Templates
◇ Jinja template, Python to template

Puppet

- requires agent
- recently they have agent-less version for cisco ios
- Ruby based
- pull model → "you can have it if you want it"
- port 8140 TCP

- puppet MASTER
  → linux based server
  → DSL=declarative structure language, domain specific language
  → use DSL to create manifest = set of instructions to be executed

- module → manifest → classes → resources

Chef

- requires agent
- it doesn't integrate with cisco right now
- good option for sysadmin and developers
- Ruby
- ports 10000,10002,10003

- central Chef server
  → Bookshelf → Cookbook → Recipes
- workstation node
  → Chef installed
  → Recipes - collection of resources that determine the config policy of a node
- node
  → chef agent installed