VLANs (Part-3)

Native VLAN, Wireshark, and Inter-VLAN Routing Method

- Topics covered:
  - Using the concept of a native VLAN on a router when using router-on-a-stick for inter-VLAN routing.
  - Looking at Wireshark captures to see the dot1q tag.
  - The final method of inter-VLAN routing, which is Layer 3 switching (multilayer switching)

Native VLAN feature on a router.

In the previous, best practice was explained as setting the native VLAN to an unused VLAN, because the native VLAN feature can cause security issues.
Network security will be discussed later in the course.

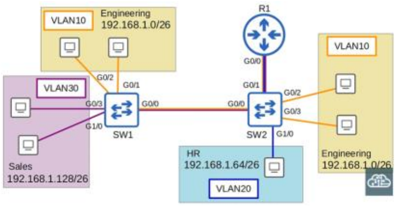
However, if you want to use the native VLAN feature, it is important to know how to configure it on a router.

◆ The native VLAN feature does have one benefit.

Frames that belong to the native VLAN are not tagged.
Because of this:

- Each frame is smaller
- The device can send more frames per second
- This makes it more efficient

◆ Network Topology



- **Native VLAN setup used for this demonstration.**

In the previous video:

- Native VLAN was set to **VLAN 1001** on:
  - SW1 G0/0
  - SW2 G0/0
  - SW2 G0/1

For this demonstration:

- The native VLAN is set back to a used VLAN
- **VLAN 10** is configured as the native VLAN on all trunk links

```
SW1(config)#int g0/0
SW1(config-if)#switchport trunk native vlan 10
SW1(config-if)#
```

```
SW2(config)#int g0/0
SW2(config-if)#switchport trunk native vlan 10
SW2(config-if)#int g0/1
SW2(config-if)#switchport trunk native vlan 10
SW2(config-if)#
```

- **There are 2 methods of configuring the native VLAN on a router.**

1. Using a subinterface with "encapsulation dot1q vlan-id native"
2. Using the physical interface without any subinterface

Both methods achieve the same result.

- **First method:** Using a subinterface with the native VLAN.

Command used on the router:

#interface g0/0.10

#encapsulation dot1q 10 native

```
R1(config)#int g0/0.10
R1(config-subif)#encapsulation dot1q 10 native
R1(config-subif)#
```

What this means:

- This subinterface belongs to VLAN 10

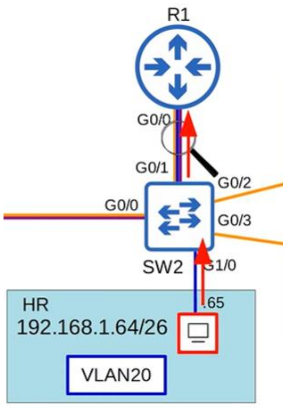
- VLAN 10 is treated as the native VLAN
- Untagged frames are assumed to belong to VLAN 10
- Frames sent in VLAN 10 are not tagged

The IP address was already configured from the previous lecture.
The only change is adding the word NATIVE.

◆ **Now let's look at a Wireshark capture to demonstrate the native VLAN.**

PC details:

- PC in **VLAN 20** → IP address **192.168.1.65**
- PC in **VLAN 10** → IP address **192.168.1.1**



Wireshark is used to monitor the link between R1 and SW2.
It captures all frames in both directions.

◆ **ICMP Echo Request from VLAN 20 (SW2 → R1).**

This packet:

- Comes from **192.168.1.65**
- Is sent to **192.168.1.1**
- Belongs to **VLAN 20**
- Needs **inter-VLAN routing**

Ethernet Header Analysis (Tagged Frame)

```
> Frame 104: 118 bytes on wire (944 bits), 118 bytes captured (944 bits) on interface 0
v Ethernet II, Src: 0c:bd:ad:00:70:00 (0c:bd:ad:00:70:00), Dst: 0c:bd:ad:c5:08:00 (0c:bd:ad:c5:08:00)
  > Destination: 0c:bd:ad:c5:08:00 (0c:bd:ad:c5:08:00)
  > Source: 0c:bd:ad:00:70:00 (0c:bd:ad:00:70:00)
    Type: 802.1Q Virtual LAN (0x8100)
v 802.1Q Virtual LAN, PRI: 0, DEI: 0, ID: 20
    000. .... .... .... = Priority: Best Effort (default) (0)
    ...0 .... .... .... = DEI: Ineligible
    .... 0000 0001 0100 = ID: 20
    Type: IPv4 (0x0800)
> Internet Protocol Version 4, Src: 192.168.1.65, Dst: 192.168.1.1
> Internet Control Message Protocol
```

Explanation:

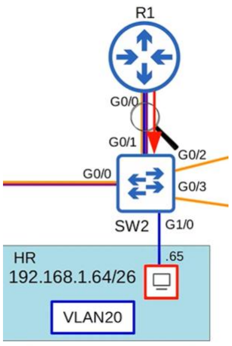
- 0x8100 is the TPID, indicating an 802.1Q tag
- The dot1q tag is inserted after the source MAC address

Fields inside the dot1q tag:

- **PCP (Priority Code Point)** = 0 → Best effort
- **DEI (Drop Eligible Indicator)** = 0 → Not dropped during congestion
- **VLAN ID** = 20 → Matches the source VLAN
- **Type** = IPv4 → IP packet inside

Because VLAN 20 is not the native VLAN, the frame is tagged.

◆ ICMP Echo Request from R1 to SW2 (VLAN 10).



Now the same ICMP packet:

- Is routed by R1
- Sent toward VLAN 10
- VLAN 10 is configured as the native VLAN

Ethernet Header Analysis (Untagged Frame)

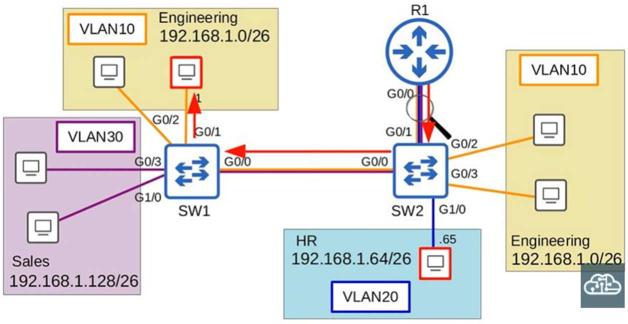
```
> Frame 105: 114 bytes on wire (912 bits), 114 bytes captured (912 bits) on interface 0
v Ethernet II, Src: 0c:bd:ad:c5:08:00 (0c:bd:ad:c5:08:00), Dst: 0c:bd:ad:84:0a:00 (0c:bd:ad:84:0a:00)
  > Destination: 0c:bd:ad:84:0a:00 (0c:bd:ad:84:0a:00)
  > Source: 0c:bd:ad:c5:08:00 (0c:bd:ad:c5:08:00)
    Type: IPv4 (0x0800)
> Internet Protocol Version 4, Src: 192.168.1.65, Dst: 192.168.1.1
> Internet Control Message Protocol
```

Explanation:

- The frame has no dot1q tag
- Both R1 and SW2 understand that:
  - Untagged frames belong to VLAN 10
- This is the native VLAN function

The packet continues untagged all the way to the destination PC in VLAN 10.

◆ **ICMP Echo Reply behaviour.**



When the PC in VLAN 10 sends the reply:

- The frame is untagged until it reaches R1
- R1 then:
  - Tags it as VLAN 20
  - Sends it back to 192.168.1.65

- **Second method: Using the physical interface for the native VLAN.**

In this method:

- No sub-interface is used for the native VLAN
- No encapsulation dot1q command is required

Steps used:

#no interface g0/0.10

#interface g0/0

#ip address 192.168.1.62 255.255.255.192

```
R1(config)#no interface g0/0.10
R1(config)#interface g0/0
R1(config-if)#ip address 192.168.1.62 255.255.255.192
R1(config-if)#
```

Explanation:

- The subinterface is deleted
- The IP address for VLAN 10 is configured directly on G0/0
- This physical interface now represents the native VLAN

- **Show running-config explanation.**

```
!
interface GigabitEthernet0/0
 ip address 192.168.1.62 255.255.255.192
 duplex auto
 speed auto
 media-type rj45
!
interface GigabitEthernet0/0.20
 encapsulation dot1Q 20
 ip address 192.168.1.126 255.255.255.192
!
interface GigabitEthernet0/0.30
 encapsulation dot1Q 30
 ip address 192.168.1.190 255.255.255.192
!
```

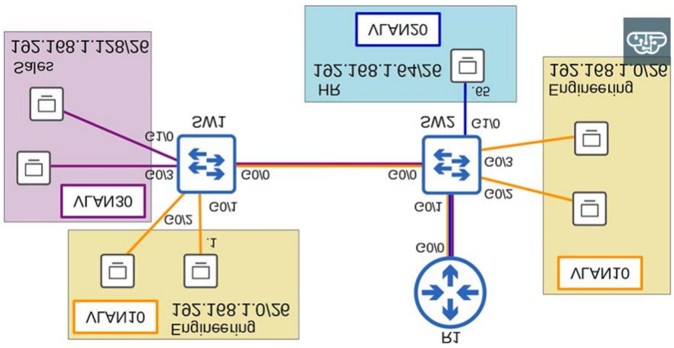
- Physical interface G0/0: used for VLAN 10 (native VLAN)
- Sub-interfaces: VLAN 20 and VLAN 30 are tagged
- Behaviour is the same as the first method

◆ **Final note on native VLAN usage.**

- Native VLAN should ideally be unused for security reasons
- However, if you choose to use it:
  - These are the two correct methods on a router
- Understanding this is important for inter-VLAN routing concepts

Introducing the Layer 3 switch (Multilayer switch).

- **Here's the network diagram.**



We have:

- One router (R1)
- Two switches

At first, both switches are Layer 2 switches.
This is the icon used for a regular Layer 2 switch.

Now, there is another type of switch.

- **The Layer 3 switch (Multilayer switch).**

This is the icon used for a Layer 3 switch, also called a multilayer switch.
From now on:

- Layer 3 switch
- Multilayer switch

Both terms mean the same thing, and you should know both.

These are the official Cisco icons, but cleaner icons are used here for simplicity.

◆ What exactly does a multilayer switch do?

A multilayer switch can do:

- Switching
- Routing

It is Layer 3 aware.

A regular Layer 2 switch:

- Is NOT Layer 3 aware
- Does not think about IP addresses
- Only works with Layer 2 information, such as MAC addresses

◆ IP addresses on switches.

Previously:

- IP addresses were configured only on routers
- Switches did not have IP addresses

With a **Layer 3 switch**:

- You can assign IP addresses
- You can configure routed ports
- These routed ports work like router interfaces

◆ Switch Virtual Interfaces (SVIs).

A multilayer switch can create virtual interfaces, one for each VLAN.

These are called: SVIs (Switch Virtual Interfaces)

Important points:

- They are not physical interfaces
- They exist in the software of the switch
- They can be assigned IP addresses
- They can be used to route traffic at Layer 3

You can also:

- Configure static routes
- Configure default routes

Just like on a router.

◆ Inter-VLAN routing so far.

So far, we have seen two methods of inter-VLAN routing.

First method: One interface per VLAN.

- One router interface for each VLAN
- Works fine
- But routers usually don't have enough interfaces for many VLANs

Second method: Router on a stick.

- One trunk link
- Carries traffic for all VLANs
- Efficient in terms of interfaces
- But all traffic goes to the router and back
- In busy networks, this can cause congestion

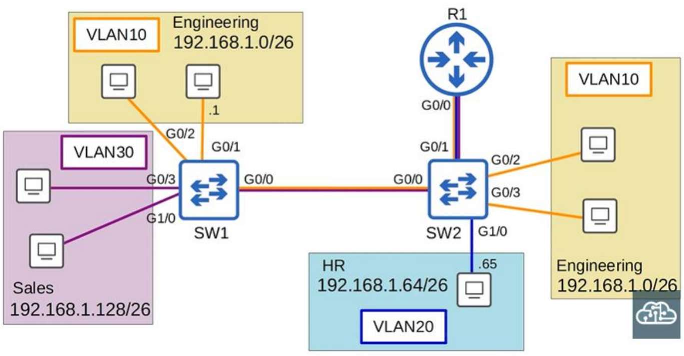
Preferred method in large networks.

In large networks:

- A multilayer switch is preferred
- Inter-VLAN routing is done on the switch itself

  Now let's see how that works.

- **Old topology.**

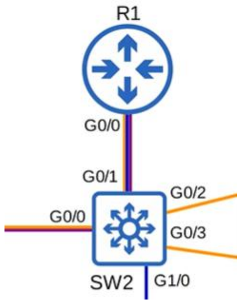


- **Replacing SW2 with a multilayer switch.**

Now:

- SW2 is a multilayer switch
- SW1 remains a Layer 2 switch

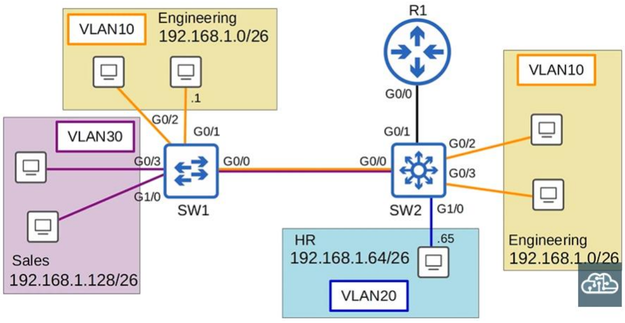
One more change is made.



- **Replacing the trunk link between SW2 and R1.**

Previously:

- SW2 and R1 were connected using a **trunk**
- VLANs were running across this link



Now:

- The trunk is removed
- A point-to-point Layer 3 link is used
- No VLANs run across this link

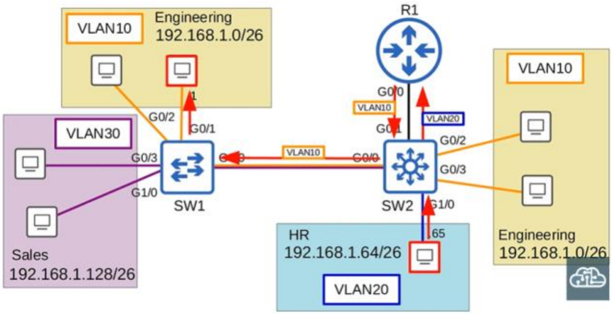
IP addresses will be assigned later.
For now, focus on inter-VLAN routing on SW2.

◆ Review: Router-on-a-stick traffic flow.

Example:

- PC in **VLAN 20** pings PC in **VLAN 10**

Traffic path before:



1. PC → SW2
2. SW2 → R1 (tagged VLAN 20)
3. R1 → SW2 (tagged VLAN 10)
4. SW2 → SW1 (tagged VLAN 10)
5. SW1 → destination PC

All inter-VLAN routing happened on **R1**.

◆ Inter-VLAN routing with a multilayer switch.

Now:

- SW2 is Layer 3 aware
- It does not need to send traffic to R1
- It uses SVIs to route traffic

- **Gateway configuration change.**

  Previously:

  - PCs used the router as their gateway

  Now:

  - PCs use the SVI on SW2 as their gateway
  - The router is not the gateway anymore

  PCs send traffic to:

  - SW2
  - SW2 routes the traffic between VLANs

- **SVIs configured on SW2.**

  The SVIs on SW2 use:

  - The **same IP addresses**
  - That were previously configured on R1

  They are the last usable IP address in each subnet Because of this, PCs already have the correct gateway addresses & No PC configuration changes are needed

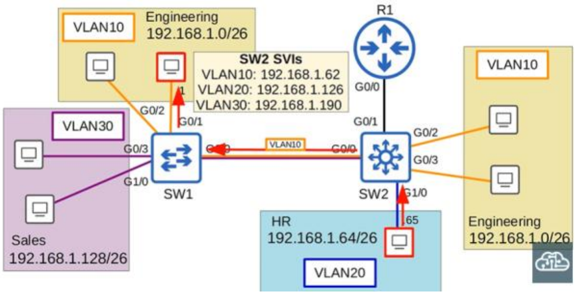
  SW2 SVIs
  VLAN10: 192.168.1.62
  VLAN20: 192.168.1.126
  VLAN30: 192.168.1.190

- **Traffic flow with SVIs.**

  Example:

  - PC in VLAN 20 sends traffic to VLAN 10

  Steps:



1. Frame arrives at SW2
2. Destination is in 192.168.1.0/26
3. SW2 checks its routing table
4. It sees the destination is connected to VLAN 10 SVI
5. Traffic is routed to VLAN 10
6. Frame is sent to SW1 tagged as VLAN 10
7. SW1 forwards it to the destination PC

Routing is done entirely on SW2.

◆ **What about destinations outside the LAN?**

A cloud is added to represent the **Internet**.

Because:

- SW2 is the **default gateway**
- All traffic destined outside the subnet is sent to **SW2**

But:

- Router-on-a-stick configurations no longer work
- The link between SW2 and R1 is now Layer 3

◆ **Layer 3 link between SW2 and R1.**

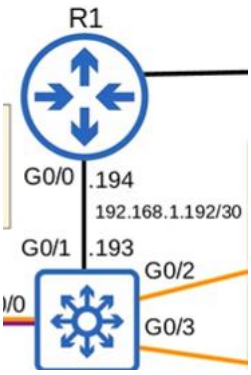
We configure:

- A point-to-point subnet
- 192.168.1.192/30

IP addresses:

- SW2 G0/1 → **192.168.1.193**
- R1 G0/0 → **192.168.1.194**

This link is not a trunk.



◆ **Default route on SW2.**

A default route is configured on SW2:

- Points toward R1
- Sends all non-local traffic to the router

This allows:

- LAN traffic → routed by SW2
- Internet traffic → forwarded to R1

◆ **Summary before configuration.**

- SW2 is now a multilayer switch
- Inter-VLAN routing is done using SVIs
- R1 is used only for external connectivity
- A default route on SW2 sends traffic to R1

Configure Point-To-Point b/w R1→SW2(Layer 3)

- **Remove R1's (ROSA) configurations and configure that new IP address on G0/0.**

    1. The router-on-a-stick configuration on R1 must be removed
    2. A new IP address will be configured on G0/0

STEP 1] Delete each sub-interface:

The sub-interfaces are removed using:

```
R1(config)#no interface g0/0.10
R1(config)#no interface g0/0.20
R1(config)#no interface g0/0.30
```

This removes all VLAN sub-interfaces that were used for router on a stick.

Resetting the physical interface.

Next:

```
R1(config)#default interface g0/0
Interface GigabitEthernet0/0 set to default configuration
R1(config)#do show ip interface brief
Interface              IP-Address      OK? Method Status                Protocol
GigabitEthernet0/0     unassigned      YES NVRAM  up                    up
GigabitEthernet0/0.10  unassigned      YES manual deleted               down
GigabitEthernet0/0.20  unassigned      YES manual deleted               down
GigabitEthernet0/0.30  unassigned      YES manual deleted               down
GigabitEthernet0/1     unassigned      YES NVRAM  administratively down down
GigabitEthernet0/2     unassigned      YES NVRAM  administratively down down
GigabitEthernet0/3     unassigned      YES NVRAM  administratively down down
R1(config)#
```

This resets G0/0 back to its default settings, removing any leftover configuration.

Verifying the interfaces, I use "SHOW IP INTERFACE BRIEF"

What you should notice:

- The sub-interfaces still appear
- Their status shows DELETED
- The sub-interfaces are removed
- They remain listed as deleted until the router is reloaded
- This is not a problem, so we leave them as they are

STEP 2] Configuring the new IP address on G0/0.

- Enter interface configuration mode for G0/0
- Configure the new IP address with a /30 subnet mask
- Use "SHOW IP INTERFACE BRIEF" again
- The new IP address is correctly configured on G0/0
- The interface is up/up

```
R1(config)#interface g0/0
R1(config-if)#ip address 192.168.1.194 255.255.255.252
R1(config-if)#do show ip interface brief
Interface              IP-Address      OK? Method Status                Protocol
GigabitEthernet0/0     192.168.1.194   YES manual up                    up
GigabitEthernet0/0.10  unassigned      YES manual deleted               down
GigabitEthernet0/0.20  unassigned      YES manual deleted               down
GigabitEthernet0/0.30  unassigned      YES manual deleted               down
GigabitEthernet0/1     unassigned      YES NVRAM  administratively down down
GigabitEthernet0/2     unassigned      YES NVRAM  administratively down down
GigabitEthernet0/3     unassigned      YES NVRAM  administratively down down
R1(config-if)#
```

- **Now let's look at the switch's side of the point-to-point connection.**

This configuration is done on SW2, the multilayer switch.

Resetting SW2's G0/1 interface.

First:

- I use DEFAULT INTERFACE G0/1 Reason,
- G0/1 was previously configured as a **trunk**
- That was for router on a stick
- We no longer need that configuration

```
SW2(config)#default interface g0/1
Interface GigabitEthernet0/1 set to default configuration
SW2(config)#ip routing
SW2(config)#interface g0/1
SW2(config-if)#no switchport
SW2(config-if)#ip address 192.168.1.193 255.255.255.252
SW2(config-if)#do show ip interface brief
Interface              IP-Address      OK? Method Status                Protocol
GigabitEthernet0/0     unassigned      YES unset  up                    up
GigabitEthernet0/2     unassigned      YES unset  up                    up
GigabitEthernet0/3     unassigned      YES unset  up                    up
GigabitEthernet0/1     192.168.1.193   YES manual up                    up
GigabitEthernet1/0     unassigned      YES unset  up                    up
GigabitEthernet1/1     unassigned      YES unset  up                    up
GigabitEthernet1/2     unassigned      YES unset  up                    up
GigabitEthernet1/3     unassigned      YES unset  up                    up
GigabitEthernet2/0     unassigned      YES unset  up                    up
GigabitEthernet2/1     unassigned      YES unset  up                    up
GigabitEthernet2/2     unassigned      YES unset  up                    up
GigabitEthernet2/3     unassigned      YES unset  up                    up
```

After that, the ip routing command is enabled on SW2. This step is very important because it allows the switch to perform Layer 3 routing and build its own routing table. Without enabling ip routing, the switch would not be able to route traffic between VLANs even if IP addresses are configured.

Next, interface G0/1 is configured again, but this time as a Layer 3 routed port. The no switchport command is used to convert the interface from a Layer 2 switchport into a Layer 3 interface, similar to a router interface. Only after this conversion is it possible to assign an IP address to the interface.

Then, the IP address 192.168.1.193/30 is assigned to G0/1, which creates a point-to-point Layer 3 connection between SW2 and R1. When show ip interface brief is used, the interface appears with an IP address and shows an up/up status, confirming that it now behaves like a router interface.

Finally, this configuration allows SW2 to route traffic internally between VLANs using SVIs, and forward traffic destined outside the LAN toward R1 through this routed interface.

- **Configuring the default route on SW2.**

Last configuration step:

```
SW2(config-if)#exit
SW2(config)#ip route 0.0.0.0 0.0.0.0 192.168.1.194
SW2(config)#do 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
       a - application route
       + - replicated route, % - next hop override

Gateway of last resort is 192.168.1.194 to network 0.0.0.0

S*    0.0.0.0/0 [1/0] via 192.168.1.194
      192.168.1.0/24 is variably subnetted, 2 subnets, 2 masks
C        192.168.1.192/30 is directly connected, GigabitEthernet0/1
L        192.168.1.193/32 is directly connected, GigabitEthernet0/1
SW2(config)#
```

After finishing the interface configuration, a default route is added on SW2 using ip route 0.0.0.0 0.0.0.0 192.168.1.194, which tells SW2 that any traffic destined outside the local LAN should be forwarded to R1.
When show ip route is used, SW2 now displays its own routing table, including the default route pointing to R1 and the connected/local routes, confirming that Layer 3 routing is working correctly on SW2.

- **Final verification command.**

One more useful command:

- SHOW INTERFACES STATUS

```
SW2#show interfaces status

Port      Name               Status       Vlan       Duplex  Speed Type
Gi0/0                        connected    trunk        auto   auto unknown
Gi0/2                        connected    10           auto   auto unknown
Gi0/3                        connected    10           auto   auto unknown
Gi0/1                        connected    routed       auto   auto unknown
Gi1/0                        connected    20           auto   auto unknown
Gi1/1                        connected    1            auto   auto unknown
Gi1/2                        connected    1            auto   auto unknown
Gi1/3                        connected    1            auto   auto unknown
Gi2/0                        connected    1            auto   auto unknown
Gi2/1                        connected    1            auto   auto unknown
Gi2/2                        connected    1            auto   auto unknown
Gi2/3                        connected    1            auto   auto unknown
Gi3/0                        connected    1            auto   auto unknown
Gi3/1                        connected    1            auto   auto unknown
Gi3/2                        connected    1            auto   auto unknown
Gi3/3                        connected    1            auto   auto unknown
SW2#
```

- In the VLAN column
- Interface G0/1 shows ROUTED
- It is no longer a switchport
- It is a Layer 3 routed interface

At this point:

- Router-on-a-stick is fully removed
- The point-to-point Layer 3 link is configured
- SW2 is routing traffic
- The default route to R1 is working

Configure SVIs

- **Now let's move on to configure those SVIs on SW2.**

SVI configuration is very simple.

- **Creating an SVI for VLAN10/20/30**

Use the command INTERFACE VLAN__

```
SW2(config)#interface vlan10
SW2(config-if)#ip address 192.168.1.62 255.255.255.192
SW2(config-if)#no shutdown
SW2(config-if)#interface vlan20
SW2(config-if)#ip address 192.168.1.126 255.255.255.192
SW2(config-if)#no shutdown
SW2(config-if)#interface vlan30
SW2(config-if)#ip address 192.168.1.190 255.255.255.192
SW2(config-if)#no shutdown
```

- This creates an **SVI for VLAN10/20/30**
- Assign an IP address to all vlan
- Use #NO SHUTDOWN to enable it

Important:

- SVIs are shutdown by default
- Always remember to use NO SHUTDOWN

- **Demonstrating a possible problem with SVIs.**

Now, just to demonstrate a problem you might encounter:

```
SW2(config-if)#interface vlan40
SW2(config-if)#ip address 40.40.40.40 255.255.255.0
SW2(config-if)#no shutdown
SW2(config-if)#do show ip interface brief
Interface              IP-Address      OK? Method Status                Protocol
GigabitEthernet0/0     unassigned      YES unset  up                    up
GigabitEthernet0/2     unassigned      YES unset  up                    up
GigabitEthernet0/3     unassigned      YES unset  up                    up
GigabitEthernet0/1     192.168.1.193   YES manual up                    up
```

- I create another SVI for VLAN40 that doesn't exist
- I assign IP address 40.40.40.40/24
- I also enable it with NO SHUTDOWN

Checking the SVI status.

```
Vlan10                    192.168.1.62     YES manual up                   up
Vlan20                    192.168.1.126    YES manual up                   up
Vlan30                    192.168.1.190    YES manual up                   up
Vlan40                    40.40.40.40      YES manual down                 down
```

When we look at the SVI: VLAN40 is **DOWN/DOWN**

Why is that?

Conditions required for an SVI to be UP/UP.

First condition: The VLAN must exist on the switch.

In this case:

- VLAN40 does not exist on the switch
- So the SVI cannot become UP/UP

Important difference:

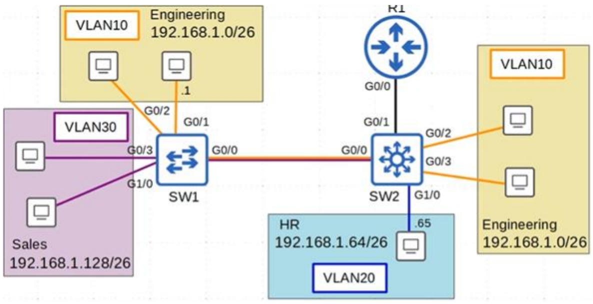
- If you assign an access port to a VLAN that doesn't exist, the switch automatically creates the VLAN
- But if you create an SVI for a VLAN that doesn't exist, the switch will NOT create the VLAN automatically

Second condition: There must be active ports in the VLAN.

The switch must have:

- At least one access port in that VLAN in an up/up state
  OR
- At least one trunk port that allows that VLAN and is up/up

In this topology:



- VLAN10 has connected hosts → SVI can go up
- VLAN20 has connected hosts → SVI can go up
- VLAN30 has no hosts, but:
  - Trunk port G0/0 allows VLAN30
  - So VLAN30's SVI is up

Third condition: The VLAN itself must not be shutdown.

This refers to:

- The VLAN, not the SVI

If you:

- Enter VLAN configuration mode
- Use the SHUTDOWN command on the VLAN

Then:

- The SVI for that VLAN cannot become UP/UP

Note:

- This command usually cannot be tested in Packet Tracer
- A real Cisco switch is required

Final condition: The SVI itself must not be shutdown.

If:

- The SVI is shutdown

Then:

- It will not be UP/UP

Reminder:

- SVIs are shutdown by default

- Always use NO SHUTDOWN after creating them

🔹 Verifying the routing table.

I use the command SHOW IP ROUTE

```
SW2(config-if)#do 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
       a - application route
       + - replicated route, % - next hop override

Gateway of last resort is 192.168.1.194 to network 0.0.0.0

S*    0.0.0.0/0 [1/0] via 192.168.1.194
      192.168.1.0/24 is variably subnetted, 8 subnets, 3 masks
C        192.168.1.0/26 is directly connected, Vlan10
L        192.168.1.62/32 is directly connected, Vlan10
C        192.168.1.64/26 is directly connected, Vlan20
L        192.168.1.126/32 is directly connected, Vlan20
C        192.168.1.128/26 is directly connected, Vlan30
L        192.168.1.190/32 is directly connected, Vlan30
C        192.168.1.192/30 is directly connected, GigabitEthernet0/1
L        192.168.1.193/32 is directly connected, GigabitEthernet0/1
SW2(config-if)#
```

What you can see:

- Connected routes
- Local routes

These routes are:

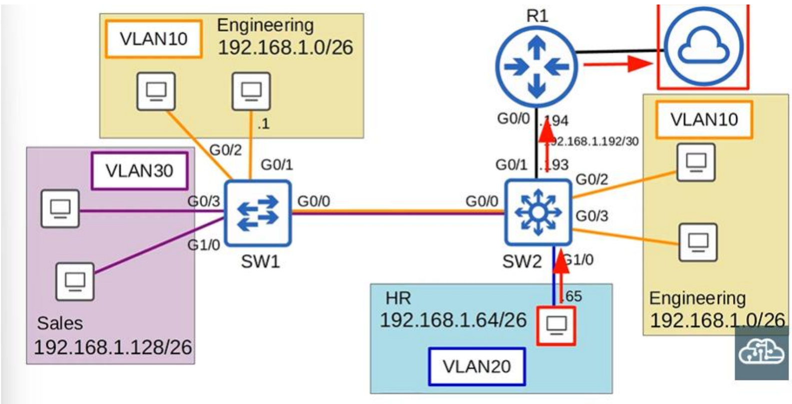
- Automatically added
- Shown as directly connected to the SVI of each VLAN

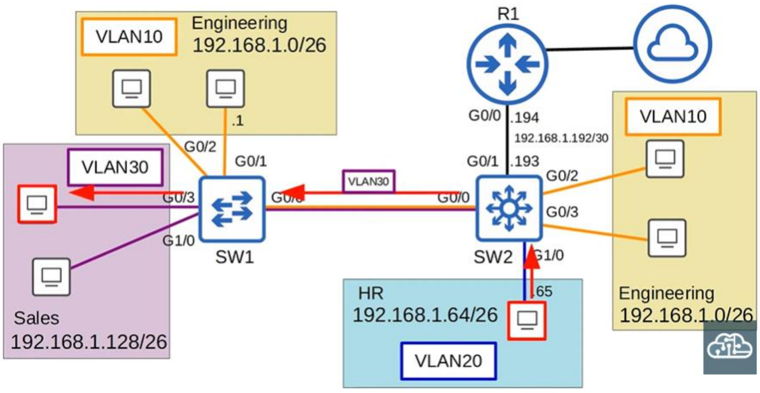
🔹 Configuration status.

At this point:

- All SVI configurations are complete
- Inter-VLAN routing is handled by **SW2**

- **Logical traffic flow (ASCII Diagram).**





- **Traffic behaviour summary.**

If a PC wants to reach:

- A destination outside the LAN
    - Traffic goes to SW2
    - SW2 sends it to R1
    - R1 handles it from there

Note:

- No routes were configured on R1 in this lab
- The focus here is inter-VLAN routing

If a PC wants to reach:

- A destination inside the LAN but in another VLAN
    - SW2 performs the inter-VLAN routing
    - Traffic does not need to go to R1