

This is a **Router-on-a-Stick** lab that demonstrates VLANs, 802.1Q trunking, router subinterfaces and inter-VLAN routing.

Objective

Students will configure:

- two VLANs on a single switch (VLAN 10 and VLAN 20),
- a trunk link between the switch and router,
- router subinterfaces with encapsulation dot1Q so the router provides gateway IPs for each VLAN,
- PC IPs and default gateways,
- verify inter-VLAN communication (PC1 ↔ PC2).

Equipment (in Packet Tracer)

- 1 × Router (Cisco 1941 or similar)
- 1 × Switch (2960)
- 2 × PCs (PC1, PC2)
- Copper straight-through cables (PCs → switch, switch → router)

IP Plan

- VLAN 10 (Admin): 192.168.10.0 /24
 - Router SVI (subinterface) / Gateway: 192.168.10.1
 - PC1: 192.168.10.10 mask 255.255.255.0 gateway 192.168.10.1
- VLAN 20 (Sales): 192.168.20.0 /24
 - Router SVI (subinterface) / Gateway: 192.168.20.1
 - PC2: 192.168.20.10 mask 255.255.255.0 gateway 192.168.20.1

Step-by-Step Implementation

1) Build the physical topology

- Drag a **2960 switch** to the workspace.
- Drag a **Router 1941** (or similar).

- Place **PC1** and **PC2**.
- Connect:
 - PC1 → Switch Fa0/2 (straight-through)
 - PC2 → Switch Fa0/3 (straight-through)
 - Router G0/0 → Switch Fa0/1 (straight-through)

Tip: use Fa0/x on the switch and Gig0/0 on router. The important thing is the logical placement: router ↔ switch trunk on Fa0/1.

2) Configure VLANs on the switch

Open the switch → CLI:

Switch> enable

Switch# configure terminal

! create VLANs

Switch(config)# vlan 10

Switch(config-vlan)# name Admin

Switch(config-vlan)# exit

Switch(config)# vlan 20

Switch(config-vlan)# name Sales

Switch(config-vlan)# exit

! assign PCs to access ports

Switch(config)# interface fa0/2

Switch(config-if)# switchport mode access

Switch(config-if)# switchport access vlan 10

Switch(config-if)# exit

Switch(config)# interface fa0/3

```
Switch(config-if)# switchport mode access
Switch(config-if)# switchport access vlan 20
Switch(config-if)# exit
! configure the router link as trunk
Switch(config)# interface fa0/1
Switch(config-if)# switchport trunk encapsulation dot1q ! (if supported; many 2960
default to dot1Q)
Switch(config-if)# switchport mode trunk
! optionally limit allowed VLANs:
Switch(config-if)# switchport trunk allowed vlan 10,20
Switch(config-if)# no shutdown
Switch(config-if)# exit
```

```
Switch(config)# end
Switch# write memory
```

Note: If switchport trunk encapsulation dot1q is not accepted (some 2960s in PT don't need it), just use switchport mode trunk.

3) Configure router subinterfaces (Router-on-a-Stick)

Open Router → CLI:

bash

CopyEdit

Router> enable

Router# configure terminal

! ensure physical interface is up

Router(config)# interface gigabitEthernet 0/0

```
Router(config-if)# no shutdown
```

```
Router(config-if)# exit
```

```
! subinterface for VLAN 10
```

```
Router(config)# interface gigabitEthernet 0/0.10
```

```
Router(config-subif)# encapsulation dot1Q 10
```

```
Router(config-subif)# ip address 192.168.10.1 255.255.255.0
```

```
Router(config-subif)# exit
```

```
! subinterface for VLAN 20
```

```
Router(config)# interface gigabitEthernet 0/0.20
```

```
Router(config-subif)# encapsulation dot1Q 20
```

```
Router(config-subif)# ip address 192.168.20.1 255.255.255.0
```

```
Router(config-subif)# exit
```

```
Router(config)# end
```

```
Router# write memory
```

Important: encapsulation dot1Q <vlan-id> tells the router which tagged VLAN traffic is for that subinterface.

4) Configure PC IPs / Gateways

On **PC1** (Desktop → IP Configuration):

- IP: 192.168.10.10
- Mask: 255.255.255.0
- Gateway: 192.168.10.1

On **PC2**:

- IP: 192.168.20.10
- Mask: 255.255.255.0
- Gateway: 192.168.20.1

(Or set to DHCP later—static is fine for this demo.)

5) Verify basic connectivity & status (commands to run)

On the switch:

- show vlan brief → confirms VLAN IDs and ports.
- show interfaces trunk → confirms Fa0/1 is trunking and allowed VLANs.
- show mac address-table → after traffic, shows MAC → port mapping.

On the router:

- show ip interface brief → should show g0/0 and subinterfaces g0/0.10 & g0/0.20 with IPs. Typical status: up up for g0/0 and up up for subinterfaces once trunk active.
- show running-config → review subinterfaces and encapsulation.
- show ip route → networks for 192.168.10.0 and 192.168.20.0 should be in routing table (connected).

From PCs:

- From PC1: ping 192.168.10.1 (gateway) — should reply.
- From PC1: ping 192.168.20.10 (PC2) — should reply (inter-VLAN routing).
- Reverse ping from PC2 to PC1 as well.

Note: the **very first ping** across VLANs may show one lost packet (ARP learning) — subsequent pings succeed. This is normal.

✓ Expected outputs (examples)

show vlan brief (switch):

```
sql
```

CopyEdit

VLAN Name	Status	Ports
-----------	--------	-------

1 default	active	Fa0/1, ...
10 Admin	active	Fa0/2
20 Sales	active	Fa0/3

show interfaces trunk (switch):

nginx

CopyEdit

Port	Mode	Encapsulation	Status	Native vlan
Fa0/1	on	802.1q	trunking	1

Port Vlans allowed on trunk

Fa0/1 1,10,20

Port Vlans allowed and active in management domain

Fa0/1 1,10,20

show ip interface brief (router):

objectivec

CopyEdit

Interface	IP-Address	OK?	Method	Status	Protocol
GigabitEthernet0/0	unassigned	YES	manual	up	
GigabitEthernet0/0.10	192.168.10.1	YES	manual	up	
GigabitEthernet0/0.20	192.168.20.1	YES	manual	up	



Troubleshooting checklist (common student issues)

- **Subinterface not up:**
 - Ensure the **physical router interface** is no shutdown.
 - Ensure the **switch port** connecting router is in **trunk mode** (switchport mode trunk).
 - Ensure encapsulation dot1Q <vlan> on the subinterface is present and matches the switch VLAN.
- **Ping fails between VLANs:**
 - Verify PC gateway is correct.
 - From router, ping <PC IP> to check reachability.
 - Check show mac address-table on switch to see if MACs are learned on correct ports.
- **Native VLAN mismatch:**
 - If you changed native VLAN on one side (switch) and not on the other (router assumes VLAN 1 native), untagged frames may be misinterpreted — keep native VLAN consistent or explicitly set it.
- **Switch rejects encapsulation command:**
 - That command is for the router, not the switch. On some older switches you configure trunk encapsulation; on 2960 in Packet Tracer you may not need to set encapsulation explicitly.

Teaching notes / talking points

- Emphasize that **VLANs are logical broadcast domains**; hosts on different VLANs need a **Layer 3 device** to talk.
- Explain **802.1Q tagging** — trunk carries traffic for multiple VLANs; tags identify VLAN membership.
- Contrast **Router-on-a-Stick** (single physical link with subinterfaces) vs **Layer-3 switch** (SVIs on the switch) — performance and scaling differences.
- Show show mac address-table to illustrate how switch learns MACs per VLAN/port.
- Demonstrate how first ping might drop (ARP) and why.

