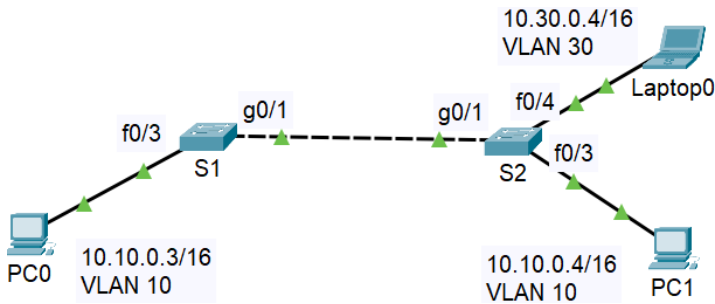# L2 Operation

VLANs

For now, you need two (any) switches (name them S1 and S2, connect them with a crossover cable, g0/1 on both sides). Connect a computer to the port f0/3 on each switch (PC0 and PC1 in the picture below). The address for the computer connected to S1 is 10.10.0.3/16, for the computer connected to S2 is 10.10.0.4/16. The default gateway is 10.10.0.1.

Ports: g0/1 will be the trunk on both switches Set up VLANs on the switches and configure them. First we create them:

```
vlan 10
      name IT
vlan 20
      name IoT
vlan 30
      name others
vlan 40
      name guests
vlan 99
      name management
```

Verify that they are accessible to each other.

Next, on each port, determine which ports will belong to a specific VLAN and which will be trunk ports:

```
int g0/1
      switchport mode trunk
      switchport trunk native vlan 1
      switchport trunk allowed vlan 10,20,30,40,99
                  ; range of VLANs, e.g.: switchport trunk allowed vlan 10-40,99
                  ; adding VLANs: switchport trunk allowed vlan add 50
int f0/3
      switchport mode access
      switchport access vlan 10
...

int vlan99                   ; management vlan
      ip addr 10.99.0.101 255.255.0.0        ; intended for S1
      ip addr 10.99.0.102 255.255.0.0        ; intended for S2
      no shut
      exit
ip default-gateway 10.99.0.1
```

Show commands for VLAN:

```
sh int trunk
sh int status | include trunk
sh vlan brief
```

Possibly other variants.

If the switch shows a message that the port cannot be set to trunk (this is likely to be a multilayer switch), this command should be executed on the port:

```
switchport trunk encap dot1q
do sh int ... switchport  ; verify that encapsulation is set to dot1q (i.e. IEEE 802.1Q)
```

Deleting VLAN on switch:

```
no vlan 30
```

If some ports belong to this VLAN, they become inactivated!

Deleting all VLANs (normal VLAN range):

```
delete vlan.dat
```

This applies to the normal VLAN range. Extended VLANs are stored in running-config or startup-config.

Router-on-a-stick

The purpose is to enable routing between L2_VLAN/L3_subnets, we need L3 device.

On the switch side there will be a trunk with certain VLANs enabled, on the router side one interface divided into subinterfaces – one subinterface for each VLAN.

Connect a router to the switch S1 (g0/2 on the switch, g0/0 on the router, or similar). Connect another computer (port f0/4) to the switch S2. On the computer, set the IP address to 10.30.0.4/16, the gateway to 10.30.0.1, and on switch S2, set VLAN 30 on port f0/4. Verify if 10.10.0.3 and 10.30.0.4 are available to each other.

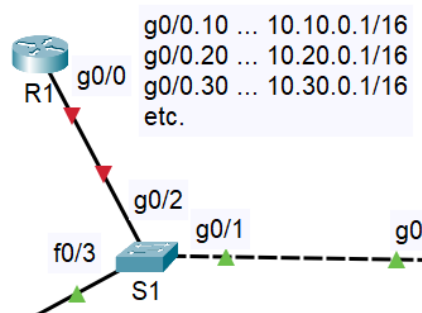On the switch S2, set the trunk leading to the router:

```
int g0/2
     switchport mode trunk
     switchport trunk native vlan 1
     switchport trunk allowed vlan 10,20,30,40,99
```


g0/0.10 ... 10.10.0.1/16
g0/0.20 ... 10.20.0.1/16
g0/0.30 ... 10.30.0.1/16
etc.

For the router:

```
int g0/0.10
     encapsulation dot1q 10
     ip addr 10.10.0.1 255.255.0.0
int g0/0.20
     encap dot1q 20
     ip addr 10.20.0.1 255.255.0.0
int g0/0.30
     encap dot1q 30
int g0/0.99
     encap dot1q 99
     ip addr 10.99.0.1 255.255.0.0
int g0/0
     no shut
```

Verify that computers 10.10.0.3 and 10.30.0.4 are available to each other.

We have shown how to create VLANs and configure the trunk manually. In very large switch networks, this can all be automated using VTP and and DTP protocols. VTP (VLAN Trunking Protocol) is used to automatically transfer enabled VLANs among switches, so we don't create VLANs manually. DTP (Dynamic Trunking Protocol) allows to negotiate a trunk with its neighboring switch automatically, so it is not necessary to create a trunk manually.

Multilayer Switch and Inter-VLAN Routing

Routers are a bit more cheaper than multilayer switches, but multilayer switches provide faster operation (links are not shared) and lower latency. There are two possibilities for inter-VLAN routing using L3 switch:

- using SVI interfaces (multilayer switches can have multiple SVIs),
- using routed ports.

In any case, routing must be enabled:

```
ip routing
```

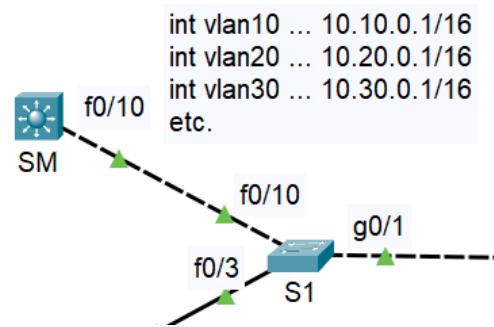## L3 switch for inter-VLAN routing using SVI interface

On S1:

```
int f0/10    ; trunk port
     switchport mode trunk
     switchport trunk native vlan 1
     switchport trunk allowed vlan 10,20,30,40………
```

On SM (L3 switch):

```
vlan 10
     name it
...               ; other VLANs
int vlan 10
     ip addr 10.10.0.1 255.255.0.0
     no shut

int vlan 20
     ip addr 10.20.0.1 255.255.0.0
...               ; other VLAN interfaces

int f0/10    ; trunk port
     switchport trunk encap dot1q    ; trunk on L3 switch needs this settings
     switchport mode trunk
     switchport trunk native vlan 1
     switchport trunk allowed vlan 10,20,30,40………
```



```
int vlan10 ... 10.10.0.1/16
int vlan20 ... 10.20.0.1/16
int vlan30 ... 10.30.0.1/16
etc.
```

L3 switch for inter-VLAN routing using routed ports:

On SM: reserve one port on the L3 switch for each VLAN. Switch these ports to L3 mode and set the IP address.

```
int f0/11
     no switchport       ; L3 operation
           ; for the reverse: switchport ...
     ip addr 10.10.0.1 255.255.0.0
int f0/12
     no switchport
     ip addr 10.20.0.1 255.255.0.0
...
```



```
f0/11 ... 10.10.0.1/16
f0/12 ... 10.20.0.1/16
f0/11 - f0/14  f0/13 ... 10.30.0.1/16
etc.
```
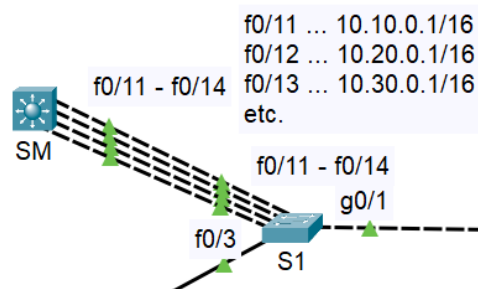
On S1:

```
int f0/11
     switchport mode access
     switchport access vlan 10
int f0/12
     switchport mode access
     switchport access vlan 20
...
```

SM can operate as the both L2 and L3 device, including various show commands, e.g.:

```
sh ip route
```