

Introduction

VLAN playlist:

https://www.youtube.com/watch?v=_PPaArOxHhw&list=PLSNNzog5eydurp2zcB4xs6gdeeVW3cMOW

Packet Tracer playlist:

<https://www.youtube.com/watch?v=7y0DEbAur-M&list=PLQQoSbMrXmrysEaVNia7KVwf85qATli1V&index=4>

Source of below commands:

<https://community.cisco.com/t5/networking-knowledge-base/how-to-configure-vlans-on-the-catalyst-switches/ta-p/3131780>

Create a VLAN

- 1) Enter enable mode using "enable"
- 2) Issue the "**configure terminal**" command in order to enter the global configuration mode.

Create and configure an interface for a VLAN

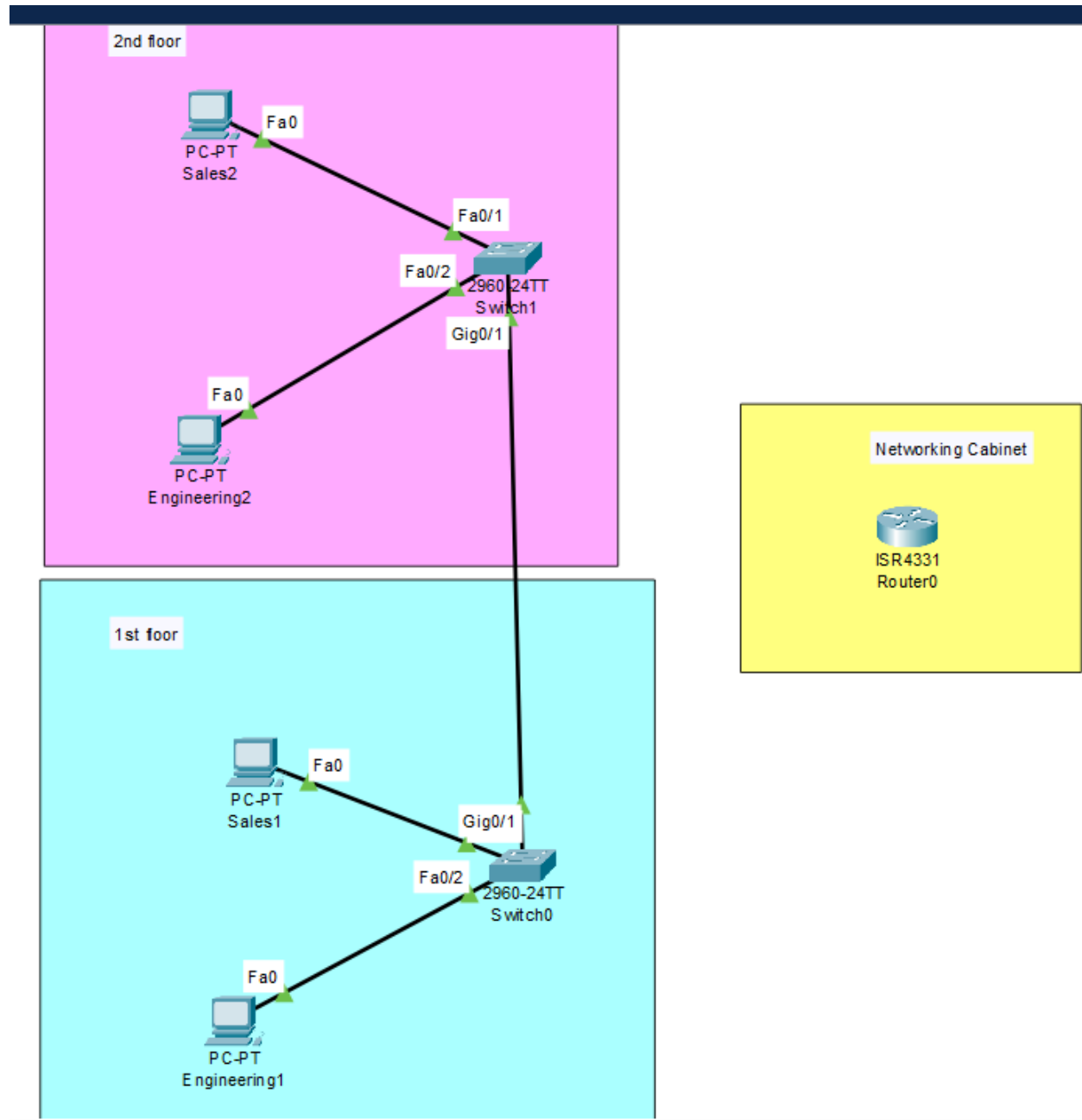
```
SW1(config)#vlan 11
SW1(config-vlan)#name Accounting
SW1(config-vlan)#exit
SW1(config)#int fa1/0
SW1(config-if)#switchport mode access
SW1(config-if)#switchport access vlan 11
SW1(config-if)#end
```

"show vlan" for the list of existing vlans on the switch.

Save the configuration on the switch using "**copy running-config startup-config**"

Configure Trunk on a Switch

```
Switch(config)# interface Gig0/1
Switch(config-if)# switchport mode trunk
```



Connecting the Router to two Switches with Static IP addresses and with Router-on-a-Stick model

InterVLAN Routing

Step1: Create sub-interfaces on the router and assign IP addresses for these sub-interfaces.

Router:

Router>enable

Router#configure terminal

Enter configuration commands, one per line. End with CNTL/Z.

Router(config)#interface gig0/0/0.1

Router(config-subif)#encapsulation dot1q 10

Router(config-subif)#ip address 10.0.1.1 255.255.255.0

Router(config-subif)#interface gig0/0/0.2

Router(config-subif)#encapsulation dot1q 20

Router(config-subif)#ip address 10.0.2.1 255.255.255.0

Router(config-subif)#int gig0/0/0

Router(config-if)#no shutdown

<not sure if these commands are necessary>

Router(config-if)#no ip address

Router(config-if)#duplex auto

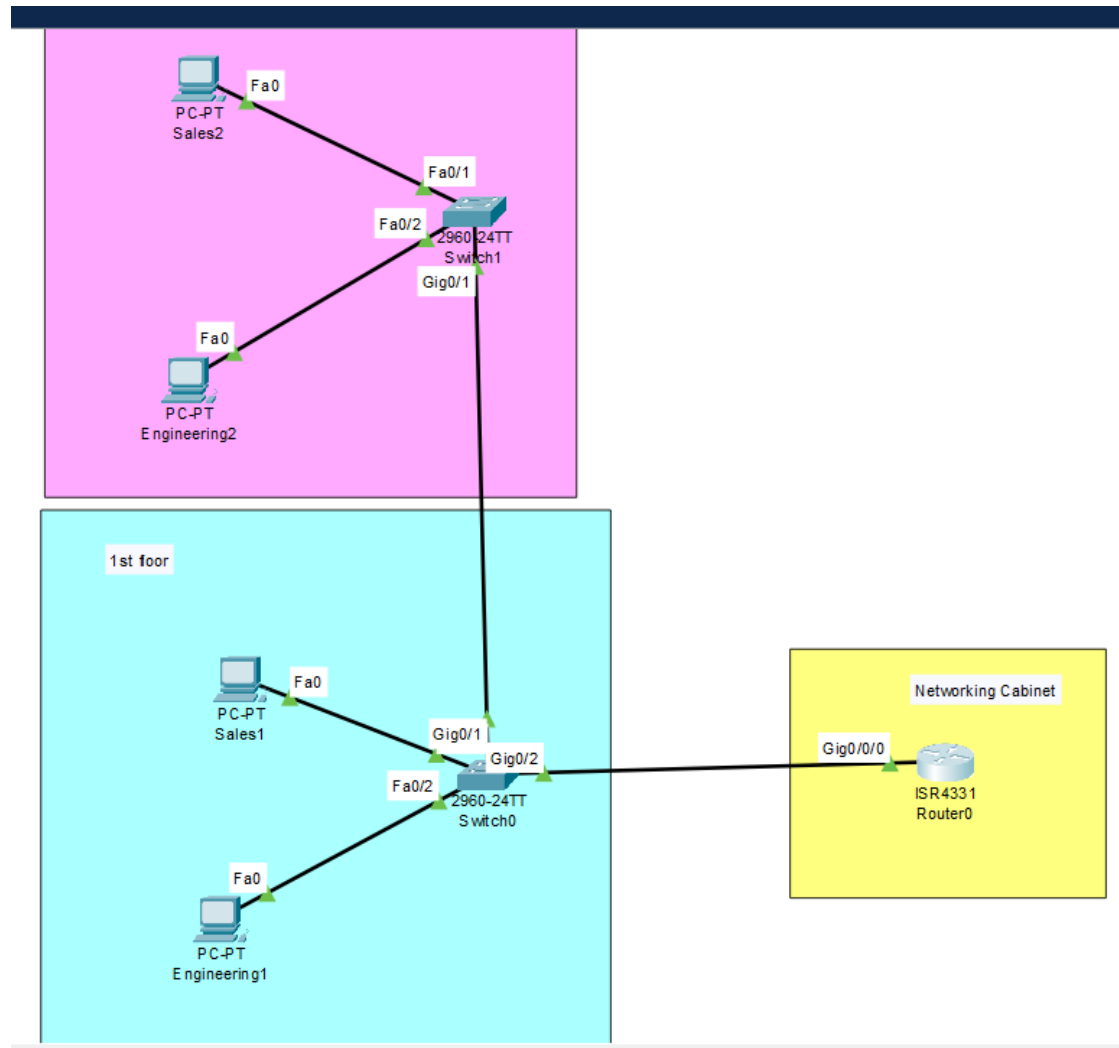
Router(config-if)#speed auto

Switch0:

Switch(config)#interface gig0/2

Switch(config-if)#switchport mode trunk

Don't forget to configure default gateway (10.0.1.1 and 10.0.2.1 on each of the respective hosts for Routing between the subnets)



Configuring DHCP on Router for each Subnet

```
Router>enable
Router#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#ip dhcp pool Sales-vlan10
Router(dhcp-config)#network 10.0.1.0 255.255.255.0
Router(dhcp-config)#default-router 10.0.1.1
Router(dhcp-config)#dns-server 1.1.1.1
Router(dhcp-config)#exit
Router(config)#ip dhcp excluded-address 10.0.1.1
Router(config)#ip dhcp excluded-address 10.0.1.255
Router(config)#
```

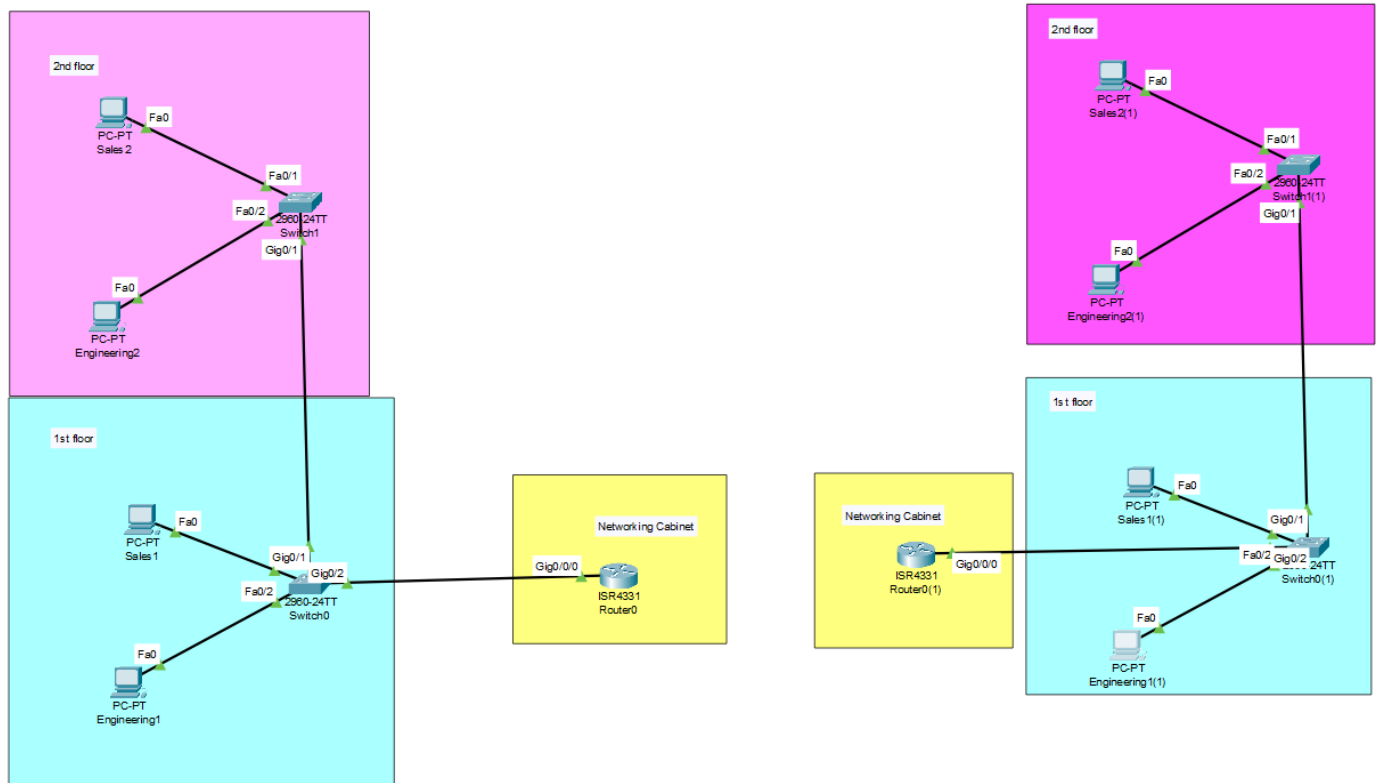
```

Router(config)#ip dhcp pool Engineering-vlan20
Router(dhcp-config)#network 10.0.2.0 255.255.255.0
Router(dhcp-config)#default-router 10.0.2.1
Router(dhcp-config)#dns-server 1.1.1.1
Router(dhcp-config)#exit
Router(config)#ip dhcp excluded-address 10.0.2.1
Router(config)#ip dhcp excluded-address 10.0.2.255
Router(config)#
Router(config)#
Router(config)#end

```

That's it! Change the config on each host to use DHCP and the IP will be added automatically!

Connecting two Routers



I've copied and pasted the whole network on the left to the right and configured the sub-interface on the right Router to use 20.0.1.0 and 20.0.2.0 networks for the right network.

Step 1: Assign IP address to Router0 interface

Router>enable

Router#configure terminal

Enter configuration commands, one per line. End with CNTL/Z.

```
Router(config)#interface gig0/0/1
```

```
Router(config-if)#ip address 15.0.0.1 255.255.255.0
```

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

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

```
%LINK-5-CHANGED: Interface GigabitEthernet0/0/1, changed state to up
```

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

Do the same for Router0(1) interface (assigning ip address 15.0.0.2)

Step 2: Configure RIP protocol on both routers:

Router0:

```
Router#configure terminal
```

Enter configuration commands, one per line. End with CNTL/Z.

```
Router(config)#router rip
```

```
Router(config-router)#network 10.0.1.0
```

```
Router(config-router)#network 10.0.2.0
```

```
Router(config-router)#network 15.0.0.0
```

```
Router(config-router)#end
```

Router0(1):

```
Router#configure terminal
```

Enter configuration commands, one per line. End with CNTL/Z.

```
Router(config)#router rip
```

```
Router(config-router)#network 20.0.1.0
```

```
Router(config-router)#network 20.0.2.0
```

```
Router(config-router)#network 15.0.0.0
```

```
Router(config-router)#end
```

Can confirm by looking at the routing table:

```
Router#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, E - EGP

i - IS-IS, 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

Gateway of last resort is not set

```
R 10.0.0.0/8 [120/1] via 15.0.0.1, 00:00:14, GigabitEthernet0/0/1
```

```
15.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
```

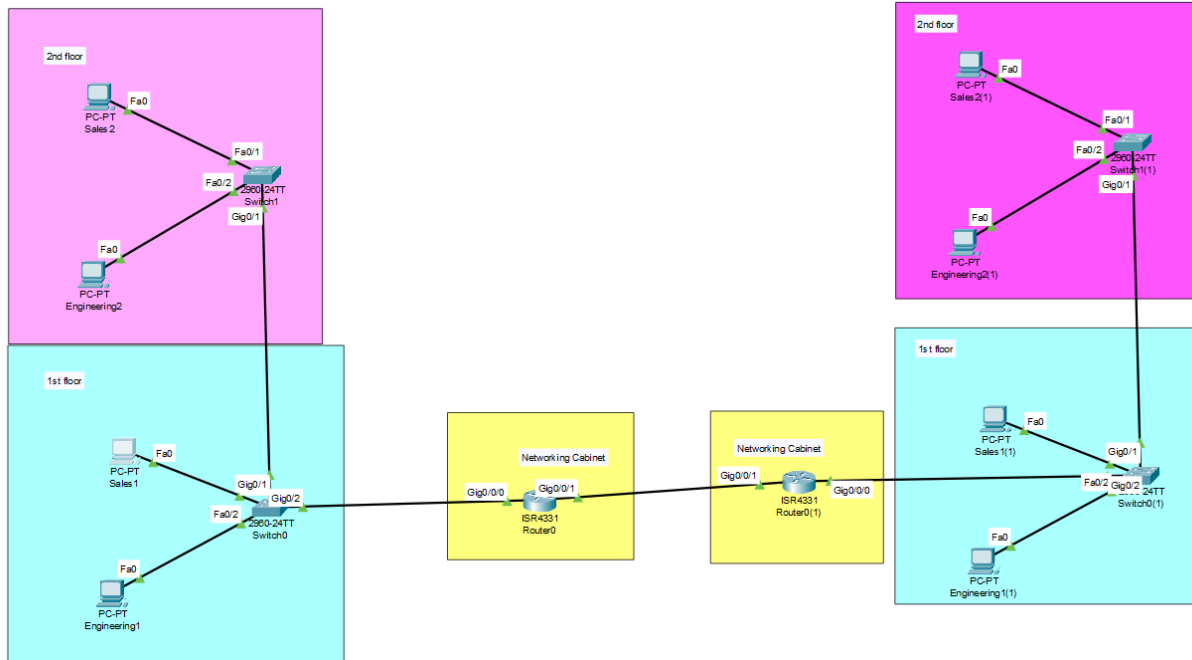
```
C 15.0.0.0/24 is directly connected, GigabitEthernet0/0/1
```

```
L 15.0.0.2/32 is directly connected, GigabitEthernet0/0/1
```

```
20.0.0.0/8 is variably subnetted, 4 subnets, 2 masks
```

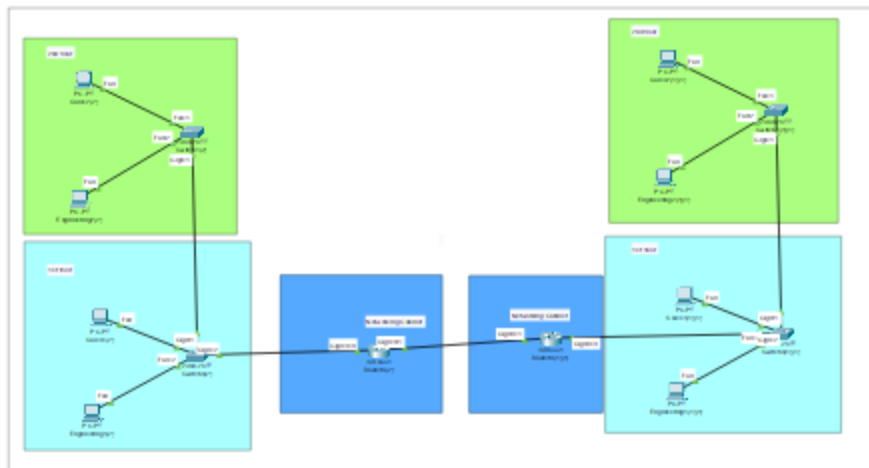
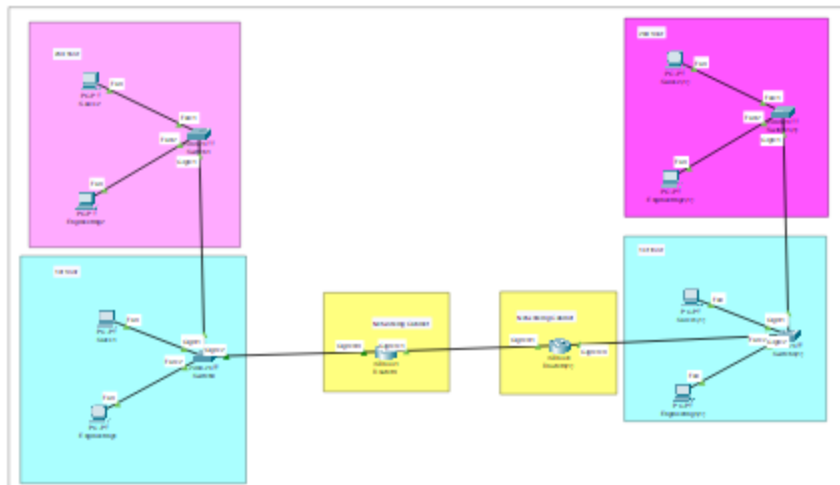
C 20.0.1.0/24 is directly connected, GigabitEthernet0/0/0.1
L 20.0.1.1/32 is directly connected, GigabitEthernet0/0/0.1
C 20.0.2.0/24 is directly connected, GigabitEthernet0/0/0.2
L 20.0.2.1/32 is directly connected, GigabitEthernet0/0/0.2

And now, we can ping from host 10.0.1.3 to 20.0.1.3!!!

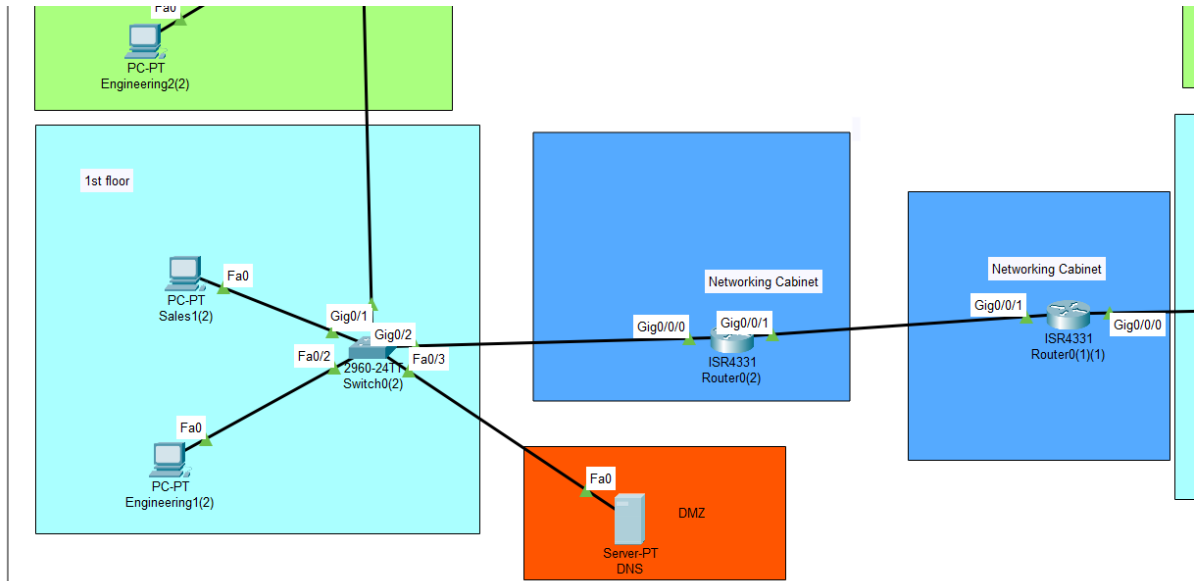


Access Control Lists (ACLs)

Create another similar network as the current one (and using 30.0.1.0,30.0.2.0 and 40.0.1.0,40.0.2.0 networks)



Add a DNS server and configure it in a separate VLAN (vlan 30) and with a new subinterface.



ACL reference:

https://www.cisco.com/c/en/us/td/docs/app_ntwk_services/waas/waas/v401_v403/command/reference/cmdref/ext_acl.pdf

ACL tutorial video: <https://www.youtube.com/watch?v=vMshgkItW5g>

Next, apply ACL to allow only DNS traffic into the DMZ and no traffic is allowed to go out of DMZ
Router>enable

Router#configure terminal

Enter configuration commands, one per line. End with CNTL/Z.

Router(config)#access-list 102 permit udp any host 1.1.1.1 eq domain //to allow any host to contact DNS server

Router(config)#access-list 102 permit udp host 1.1.1.1 eq domain any //this is required for DNS server to reply back

Router(config)#interface gig0/0/0.3

Router(config-subif)#ip access-group 102 in

Here, we can't even ping the DNS server but only use the service.

Use "show access-lists" to view existing ACLs.

Also make 30.0.2.0/24 network (Engineering network) private by denying all incoming traffic to that network:

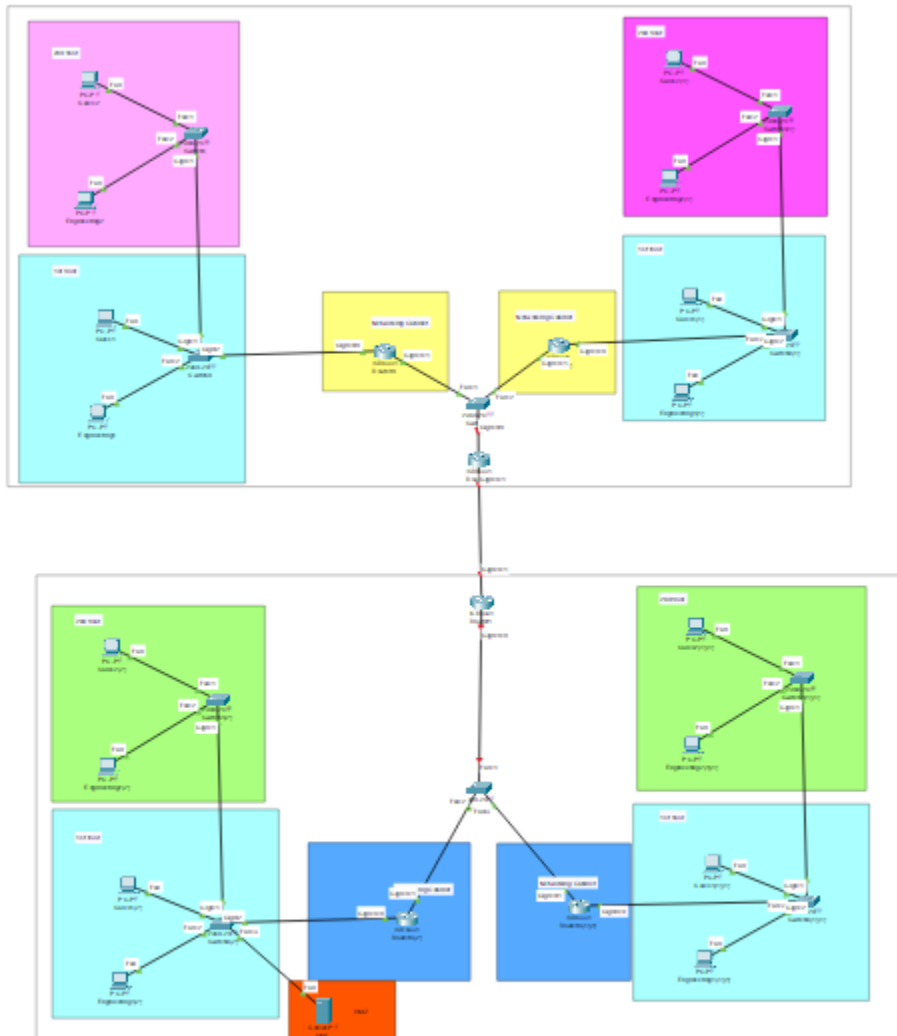
Router(config)#access-list 104 deny ip any 30.0.2.0 0.0.0.255

Router(config)#interface gig0/0/0.2

Router(config-subif)#ip access-group 104 in

Creating Autonomous Systems (AS) and BGP

Create and connect border gateway routers (along with Switches because the selected routers only have two interfaces) as shown below:



Configure border router Router1:

```
Router(config)#interface gig0/0/0
```

```
Router(config-if)#ip address 45.0.0.3 255.255.255.0
```

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

```
Router(config-if)#exit
Router(config)#router rip
Router(config-router)#network 45.0.0.0
Router(config-router)#end
```

Do the same for Router2 border router.

Setup BGP on Router2:

```
Router(config)#router bgp 10
Router(config-router)#network 10.0.1.0
Router(config-router)#network 10.0.2.0
Router(config-router)#network 20.0.1.0
Router(config-router)#network 20.0.2.0
Router(config-router)#network 15.0.0.0
Router(config-router)#neighbor 90.0.0.2 remote-as 30
Router(config-router)#exit
```

Setup BGP on Router1:

```
Router(config)#router bgp 30
Router(config-router)#network 30.0.1.0
Router(config-router)#network 30.0.2.0
Router(config-router)#network 40.0.1.0
Router(config-router)#network 40.0.2.0
Router(config-router)#network 45.0.0.0
Router(config-router)#network 1.1.1.0
Router(config-router)#neighbor 90.0.0.1 remote-as 10
```

Problem: Now, although Router1 and Router2 know what internal networks lie on each other's side, they are not advertising these routes to their respective internal networks.

Maybe I should use Route Redistribution between RIP and BGP? Is that how it works?

Route Redistribution video: https://www.youtube.com/watch?v=dUz-CWp_NaA

“redistribute bgp” wasn't working under “router rip” command on Packet Tracer. My goal was to pass the routes learned through BGP into internal RIP routers.

Setting up Static Routes

However, since this wasn't working, I setup static routes on internal routers as follows:
Following commands on Router0(1) and Router0.

```
Router(config)#ip route 1.1.1.0 255.255.255.0 15.0.0.3
Router(config)#ip route 30.0.1.0 255.255.255.0 15.0.0.3
Router(config)#ip route 30.0.2.0 255.255.255.0 15.0.0.3
```

And following commands on Router0(2)

```
Router(config)#ip route 10.0.1.0 255.255.255.0 45.0.0.3
Router(config)#ip route 10.0.2.0 255.255.255.0 45.0.0.3
Router(config)#ip route 20.0.2.0 255.255.255.0 45.0.0.3
Router(config)#ip route 20.0.1.0 255.255.255.0 45.0.0.3
```

This will make the 1.1.1.1 DNS server and the 30.0.0.0 network reachable to 10.0.0.0 and 20.0.0.0 networks.

Conclusion

In this entire Network, we created two Autonomous Systems and they communicate with each other via BGP and internal routers communicate using RIP. We have a global DNS server 1.1.1.1.