

## Báo cáo thực tập tuần 5 (22/7 – 28/7) : Tìm hiểu về network thực hành cấu hình network cơ bản sử dụng packet tracer

### Mục lục

1. Sử dụng subnet mask định tuyến trong packet tracer .....	1
1.1. Lí thuyết.....	1
1.2. Subnet mask .....	2
1.3. Thực hiện chia subnetmask trên packet tracer .....	4
2. VLAN .....	11
2.1. Lí thuyết.....	11
2.2. VLAN trên packet tracer .....	12

### 1. Sử dụng subnet mask định tuyến trong packet tracer

#### 1.1. Lí thuyết

Trong lớp 3 của mô hình OSI (lớp Network), muốn định tuyến được gói tin đến đúng địa chỉ thì ta phải có được địa chỉ IP của mạng do đó, nó có một vai trò vô cùng quan trọng. Địa chỉ IP là một chuỗi 32 bit được chia làm 4 phần, mỗi phần 8 bit.

**172 . 16 . 254 . 1**  
↓        ↓        ↓        ↓  
**10101100 . 00010000 . 11111110 . 00000001**

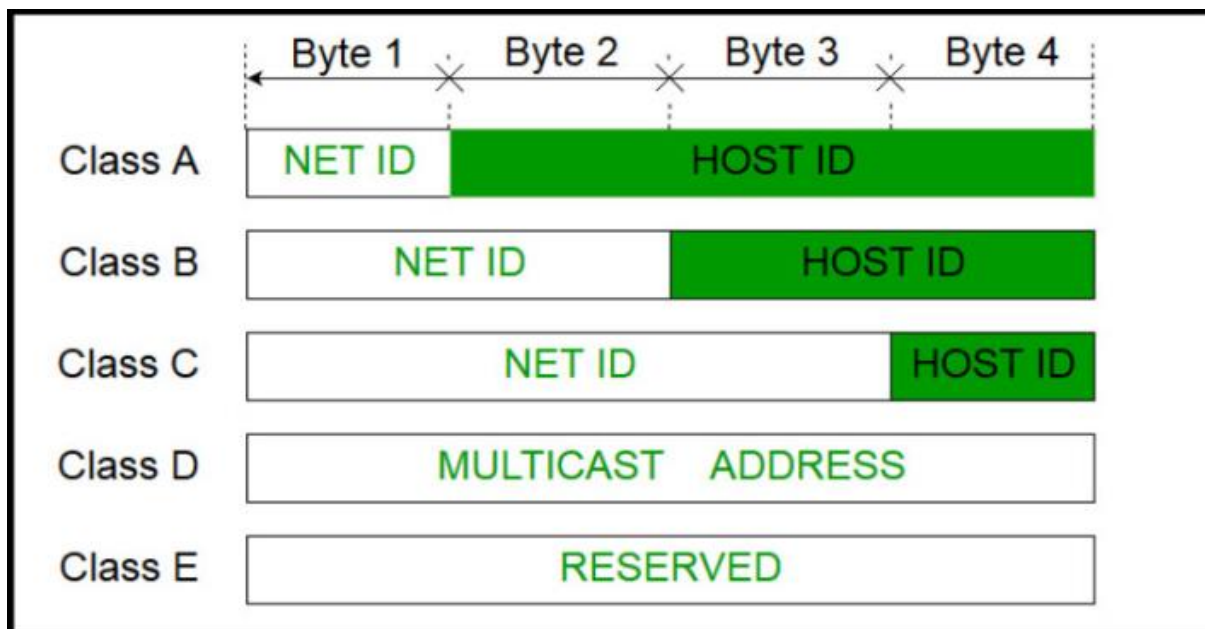
Mỗi phần 8 bit đó được chuyển sang dạng thập phân và được viết gọn như trên hình mà ta thấy (172.16.254.1)

Về cấu tạo thì 32 bit đó được chia làm 2 phần NetID (số duy trì xác định một mạng, các máy tính trong cùng một mạng có phần NetID giống nhau) và HostID (số được gán riêng cho từng máy tính trong mạng). Cụ thể, địa chỉ IP phân làm 5 phân lớp địa chỉ A,B,C,D,E như sau

- Lớp A: có định dạng **NetID**.HostID. HostID. HostID trong đó 8 bit đầu sẽ là địa chỉ mạng và 24 bit sau sẽ là địa chỉ host. Dạng nhị phân của địa chỉ mạng sẽ là 0xxxxxxx do vậy lớp A có 126 địa chỉ mạng (trừ đi 2 trường

hợp đặc biệt 0 và 127) từ 1.0.0.0 đến 126.0.0.0. Phần host có 24 bit vậy nên có 16777214 host (trừ đi 2 trường hợp đặc biệt toàn bit 0 và toàn bit 1)

- Lớp B: có định dạng **NetID.NetID**. HostID. HostID trong đó 16 bit đầu sẽ là địa chỉ mạng và 16 bit sau sẽ là địa chỉ host. Dạng nhị phân của địa chỉ mạng sẽ là 10xxxxxx do vậy lớp B có 16384 địa chỉ mạng từ 128.0.0.0 đến 191.255.0.0. Phần host có 16 bit vậy nên có 65534 host (trừ đi 2 trường hợp đặc biệt toàn bit 0 và toàn bit 1)
- Lớp C: có định dạng **NetID.NetID.NetID**.HostID trong đó 24 bit đầu sẽ là địa chỉ mạng và 8 bit sau sẽ là địa chỉ host. Dạng nhị phân của địa chỉ mạng sẽ là 110xxxxx do vậy lớp C có địa chỉ từ 192.0.0.0 đến 223.255.255.0 Phần host có 8 bit vậy nên có 254 host (trừ đi 2 trường hợp đặc biệt toàn bit 0 và toàn bit 1)



Từ điều trên ta có thể thấy từ một dải địa chỉ, ta có thể xác định được địa chỉ mạng và số host có thể có.

## 1.2. Subnet mask

Trong phần trên, ta đã biết cách xác định địa chỉ mạng và host từ địa chỉ IP. Vậy ngược lại nếu được yêu cầu thiết lập một hay nhiều mạng cho một công ty và đã biết trước số máy thì sao. Ví dụ công ty có 3 phòng ban, phòng IT có 30 máy, phòng kế toán có 10 máy và phòng sales có 10 máy thì ta nên làm thế nào. Ta có thể dùng 3 địa chỉ mạng khác nhau cho mỗi phòng. Địa chỉ 192.168.1.0/24 cho phòng IT, 192.168.2.0/24 cho phòng kế toán, 192.168.3.0/24 cho phòng sales. Đúng vậy, ta có thể làm được thế này. Tuy nhiên nếu xét về kinh phí phải trả thì sẽ tốn kém rất nhiều do phải thuê cả 3 dải địa chỉ mạng và cả 3 đều không

dùng hết số host có thể có của dải mạng (192.168.2.0/24 là địa chỉ lớp C do đó có thể chứa 254 host trong khi phòng IT cần 30 host như vậy sẽ thừa 224 host gây lãng phí. Ngoài ra thì địa chỉ IP không được nhiều lắm do có 32 bit nên nếu làm vậy sẽ không tối ưu). Do vậy subnet mask ra đời để giải quyết vấn đề trên. Mục đích của việc này là chia nhỏ địa chỉ mạng ra làm nhiều địa chỉ hơn với số host phù hợp.

Như ở ví dụ ở trên, có 3 phòng với một số máy tính cụ thể. Giờ đây, với dải địa chỉ 192.168.1.0/24 (đây là địa chỉ lớp C với 24 bit đầu là địa chỉ NetID và 8 bit HostID - 11000000. 10101000.00000001.00000000) ta mượn 1 bit từ địa chỉ host. Khi này, địa chỉ mạng NetID sẽ là 25 bit đầu và hostID sẽ là 7 bit cuối (11000000. 10101000.00000001.00000000). Khi này, ta có 2 dải mạng như sau 192.168.1.0/25 và 192.168.1.128/25 với các thông số như sau:

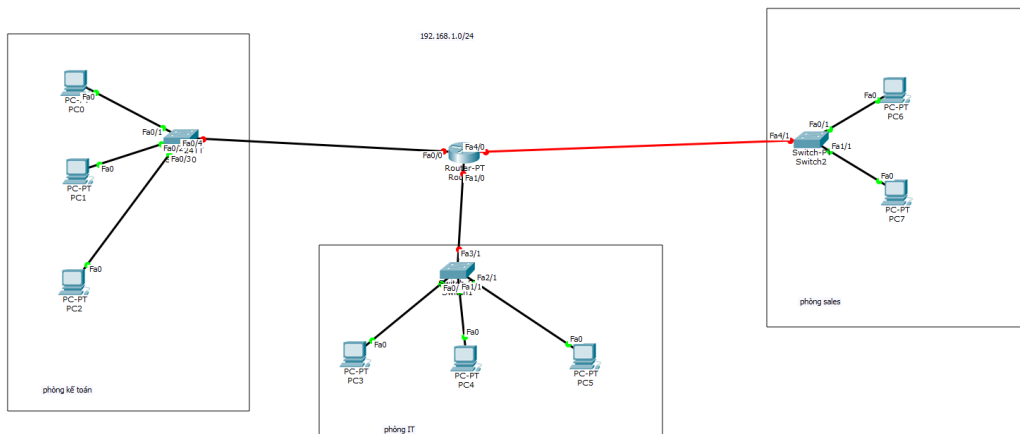
- 192.168.1.0/25 : dải chỉ host từ 192.168.1.1/25 – 192.168.1.126/25 (từ 192.168.1.0 do là địa chỉ mạng và 192.168.1.127 do là địa chỉ broadcast) tổng cộng chứa 126 host
- 192.168.1.128/25 : dải chỉ host từ 192.168.1.129 – 192.168.1.254 (từ 192.168.1.128 do là địa chỉ mạng và 192.168.1.255 do là địa chỉ broadcast) và chứa 126 host

Tuy nhiên đây mới chỉ có 2 dải mạng, ta có 3 phòng ban nên cần thêm 1 dải mạng nữa. Vì thế ta có thể mượn 2 bit thay vì mượn 1 bit. Khi này địa chỉ mạng như sau: 11000000. 10101000.00000001.00000000. Lúc này ta có 4 dải mạng (thỏa mãn điều kiện  $\geq 3$  dải mạng) và 62 host mỗi dải (thỏa mãn điều kiện). Các dải mạng như sau:

- 192.168.1.0/26 : địa chỉ host từ 192.168.1.1 – 192.168.1.62 (192.168.1.0 là địa chỉ mạng và 192.168.1.63 là địa chỉ broadcast) có 62 host
- 192.168.1.64/26 : địa chỉ host từ 192.168.1.65 – 192.168.1.126 (192.168.1.64 là địa chỉ mạng và 192.168.1.127 là địa chỉ broadcast) có 62 host
- 192.168.1.128/26 : địa chỉ host từ 192.168.1.129 – 192.168.1.190 (192.168.1.128 là địa chỉ mạng và 192.168.1.191 là địa chỉ broadcast) có 62 host
- 192.168.1.192/26 : địa chỉ host từ 192.168.1.193 – 192.168.1.254 (192.168.1.192 là địa chỉ mạng và 192.168.1.255 là địa chỉ broadcast) có 62 host

Như vậy thay vì sử dụng 3 dải mạng thì ta chỉ cần sử dụng 1 dải duy nhất tuy nhiên vẫn đảm bảo được yêu cầu đưa ra. Và sẽ tiết kiệm được khá khá kinh phí.

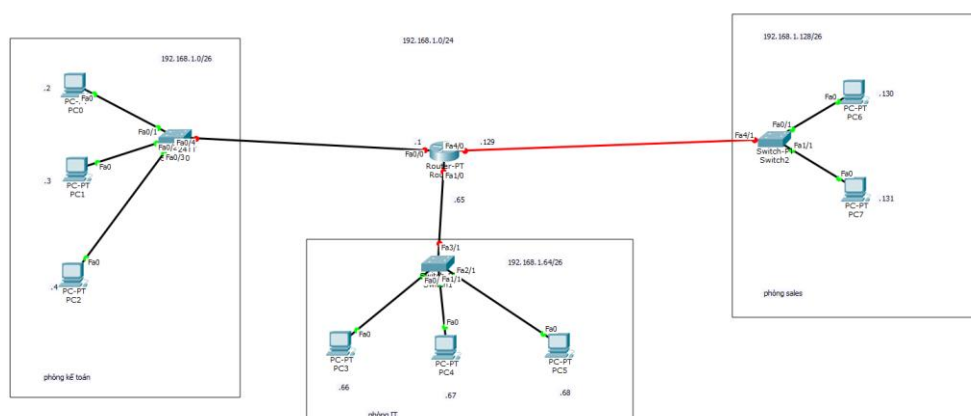
### 1.3. Thực hiện chia subnetmask trên packet tracer



Ta có mô hình như sau, có 3 phòng kế toán, IT và sales. Giờ ta định thực hiện định tuyến sử dụng subnetmask 192.168.1.0/24

Theo lí thuyết thì ta cần mượn 2 bit để tạo ra 4 mạng con và gán 3 trong số 4 mạng con đó cho 3 phòng. 4 mạng gồm 192.168.1.0/26, 192.168.1.64/26, 192.168.1.128/26, 192.168.1.192/26:

- 192.168.1.0/26 cho phòng kế toán
- 192.168.1.64/26 cho phòng IT
- 192.168.1.128/26 cho phòng sales



Ta sẽ bắt đầu gán IP và đầu tiên sẽ là router

```
Router>ena  
Router#conf t
```

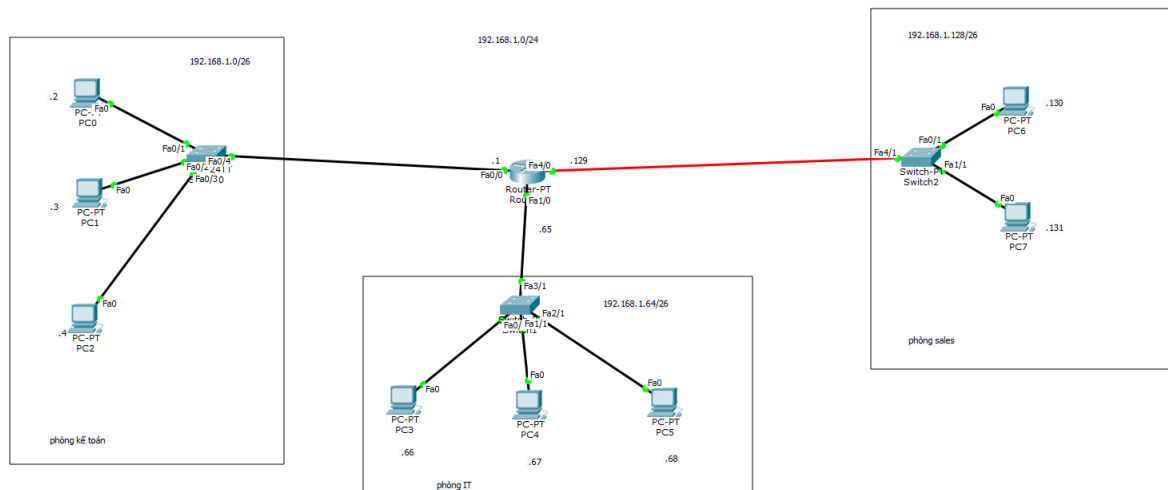
Sau khi thực hiện xong lệnh này, ta sẽ vào mode config của router. Lúc nào ta sẽ vào từng Interface và gán IP cho từng interface

```
Router(config)#int f0/0  
Router(config-if)#ip add 192.168.1.1 255.255.255.192  
Router(config-if)#no shut
```

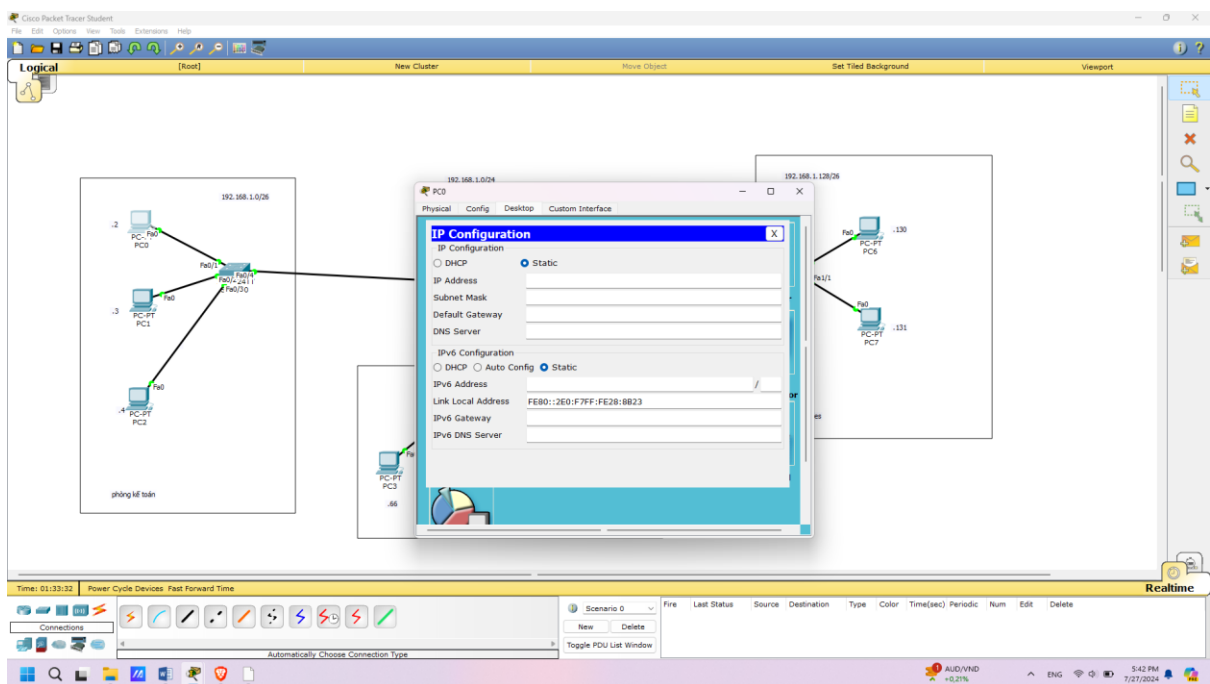
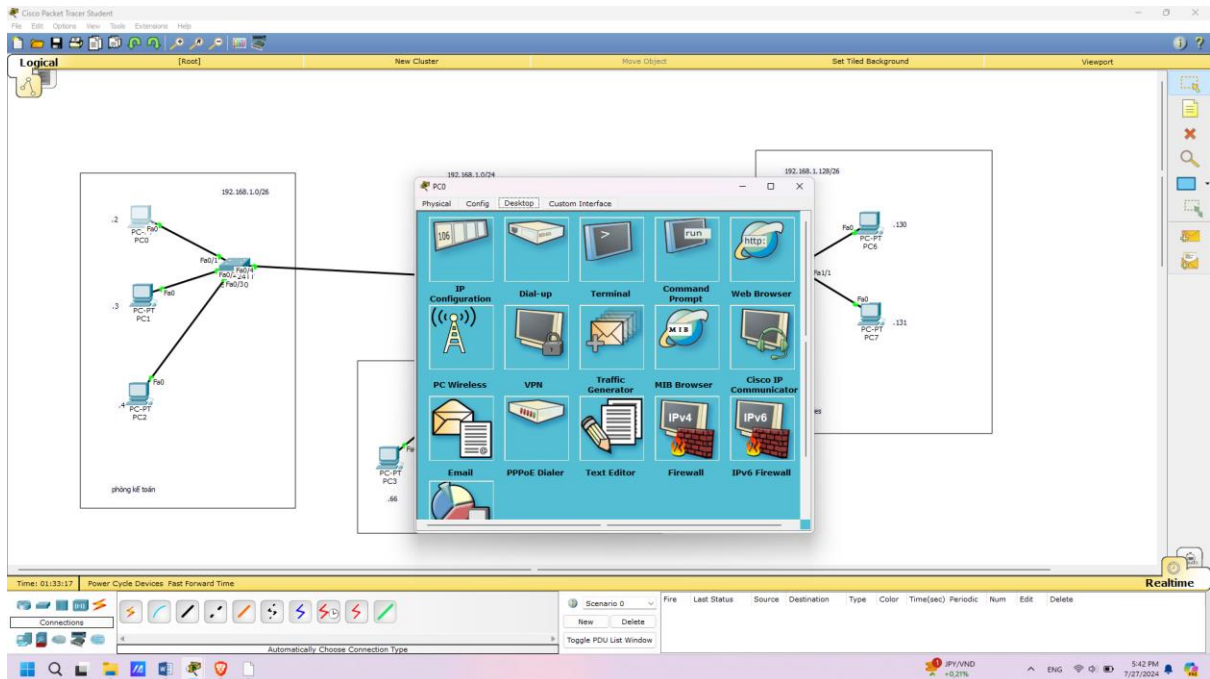
```
Router(config)#int f4/0  
Router(config-if)#ip add 192.168.1.129 255.255.255.192  
Router(config-if)#no shut
```

```
Router(config-if)#int f1/0  
Router(config-if)#ip add 192.168.1.65 255.255.255.192  
Router(config-if)#no shut
```

Sau khi gõ xong lệnh thì tất cả các interface đều xanh



Tiếp theo là gán IP cho từng host. Ta chọn vào PC, chuyển sang phần desktop và chọn IP configuration



Ta chỉnh IP thành IP đã định theo lý thuyết

Phòng kế toán:

- 192.168.1.2/26
- 192.168.1.3/26

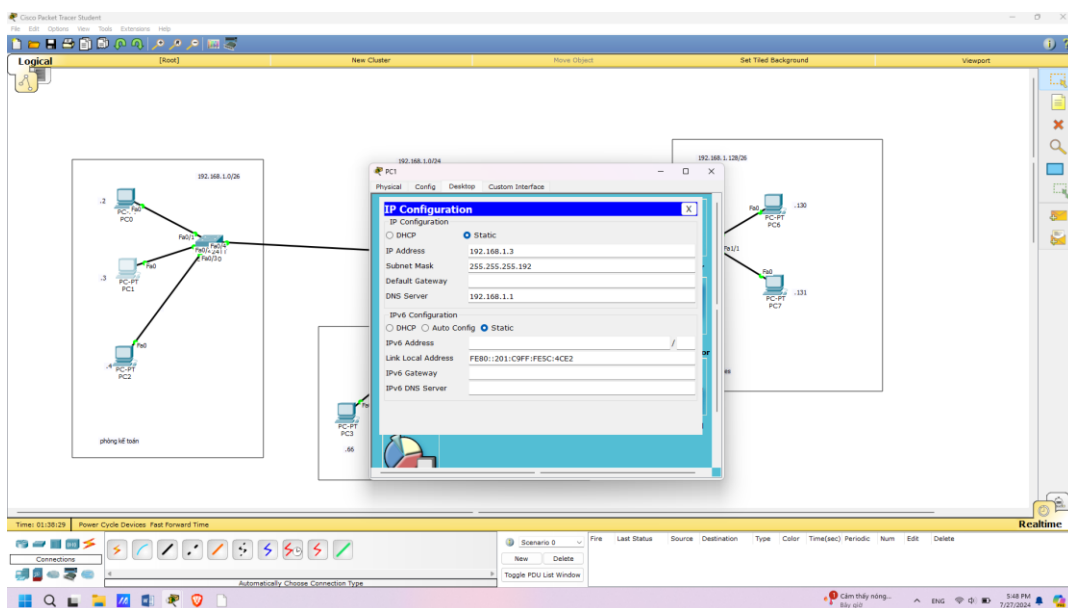
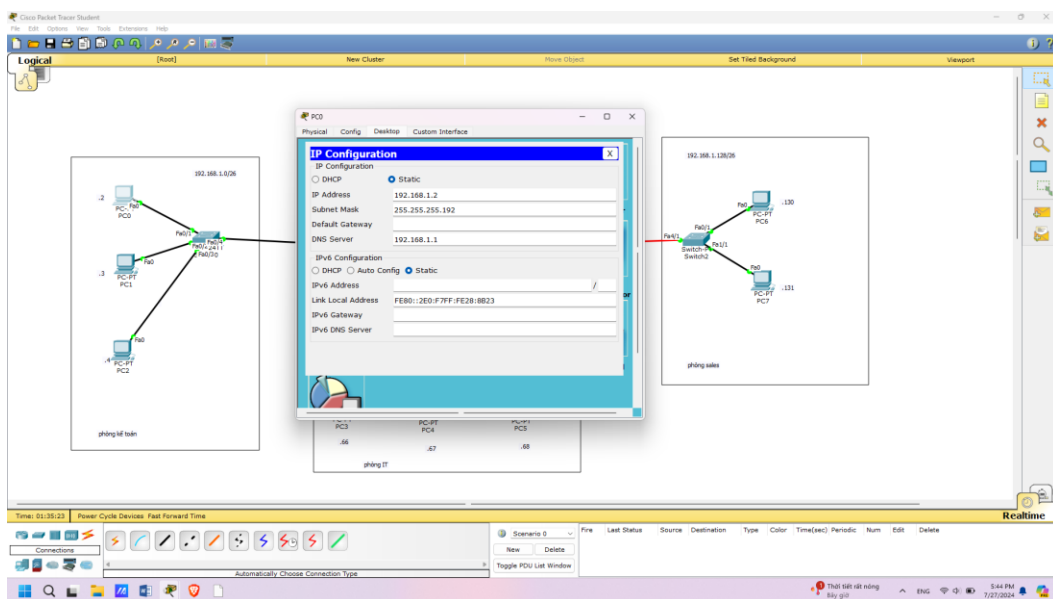
- 192.168.1.4/26

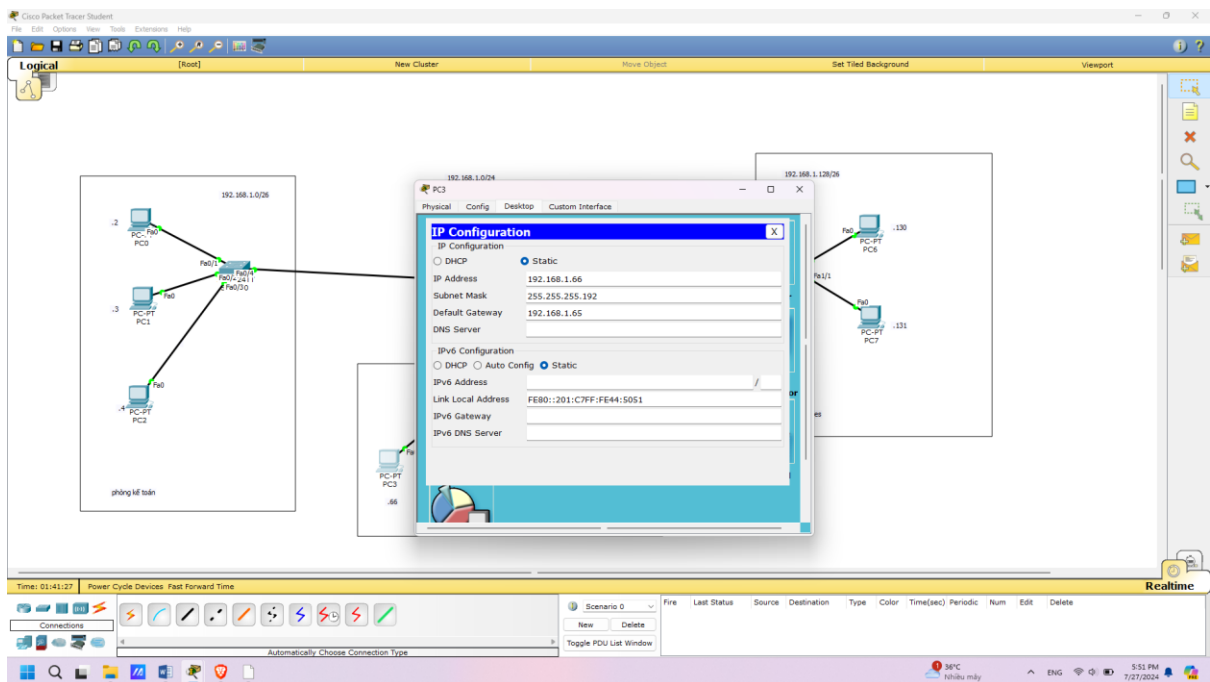
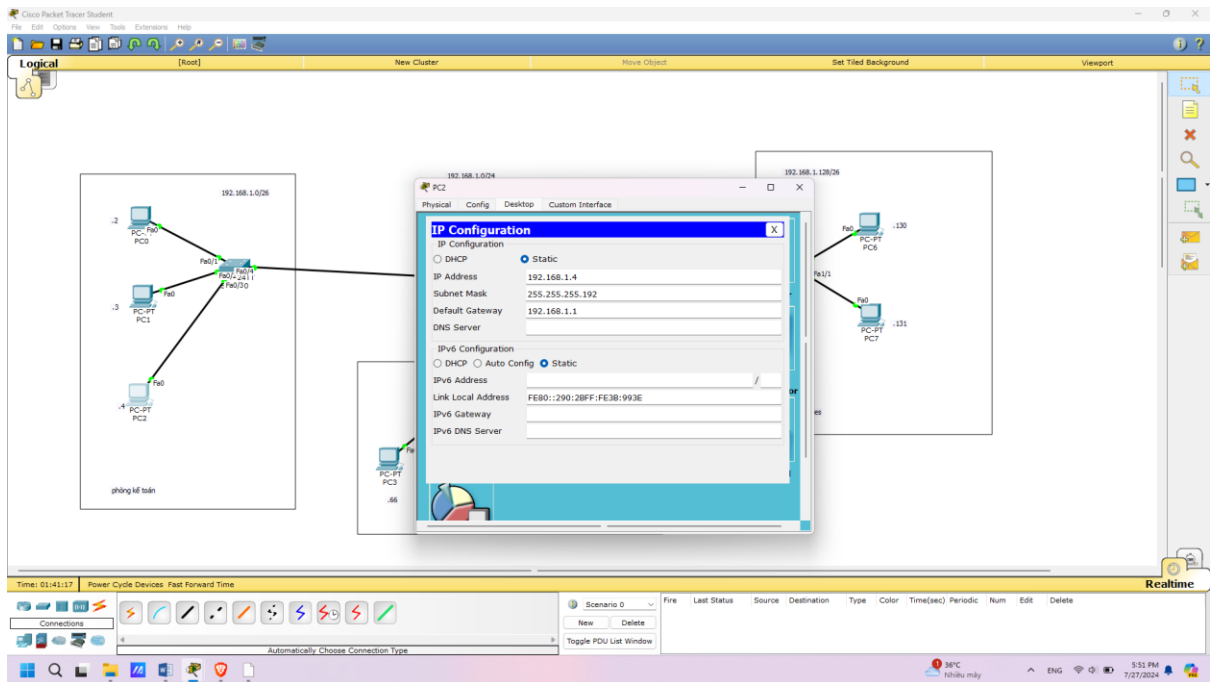
Phòng IT:

- 192.168.1.66/26
- 192.168.1.67/26
- 192.168.1.68/26

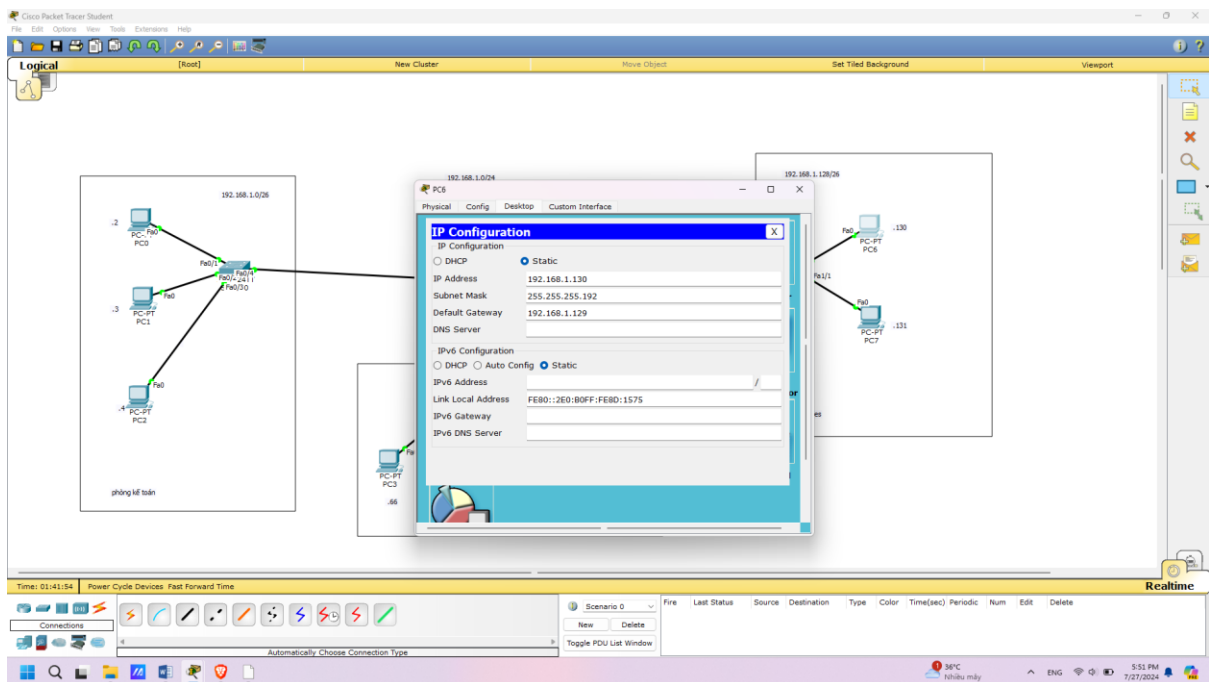
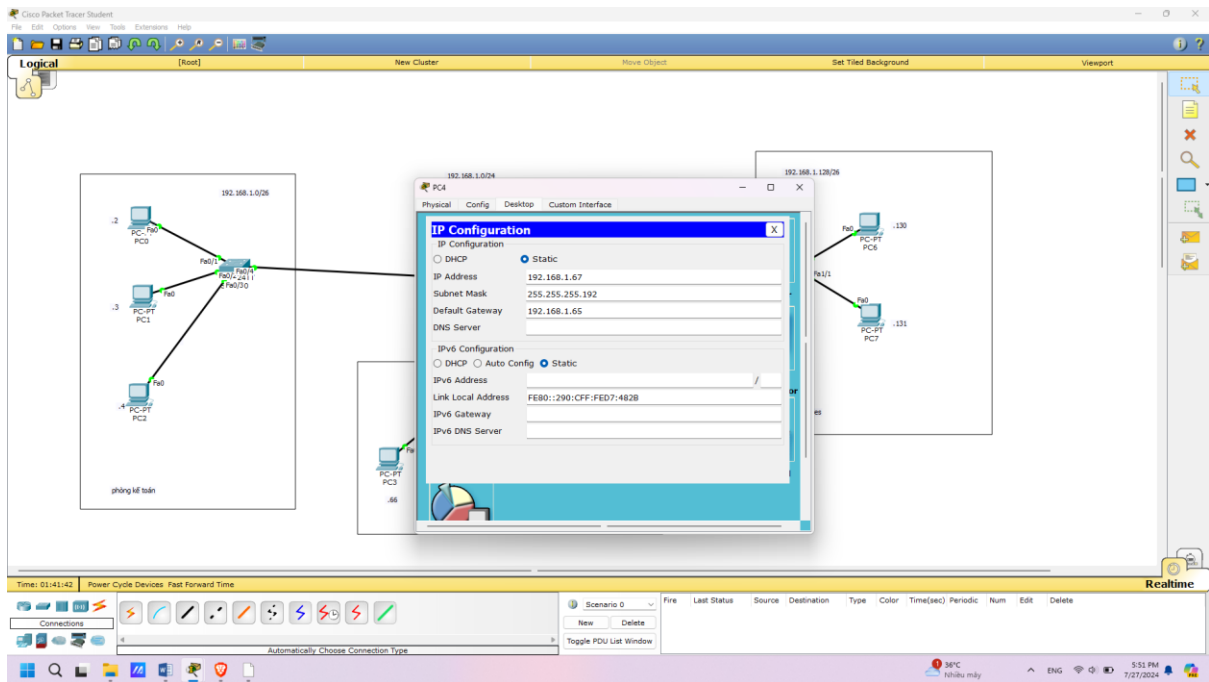
Phòng sales:

- 192.168.1.130/26
- 192.168.1.131/26









Vậy là đã gán xong IP, giờ ta thực hiện định tuyến cho mạng

**Router>ena**

**Router#conf t**

**Router(config)#router rip**

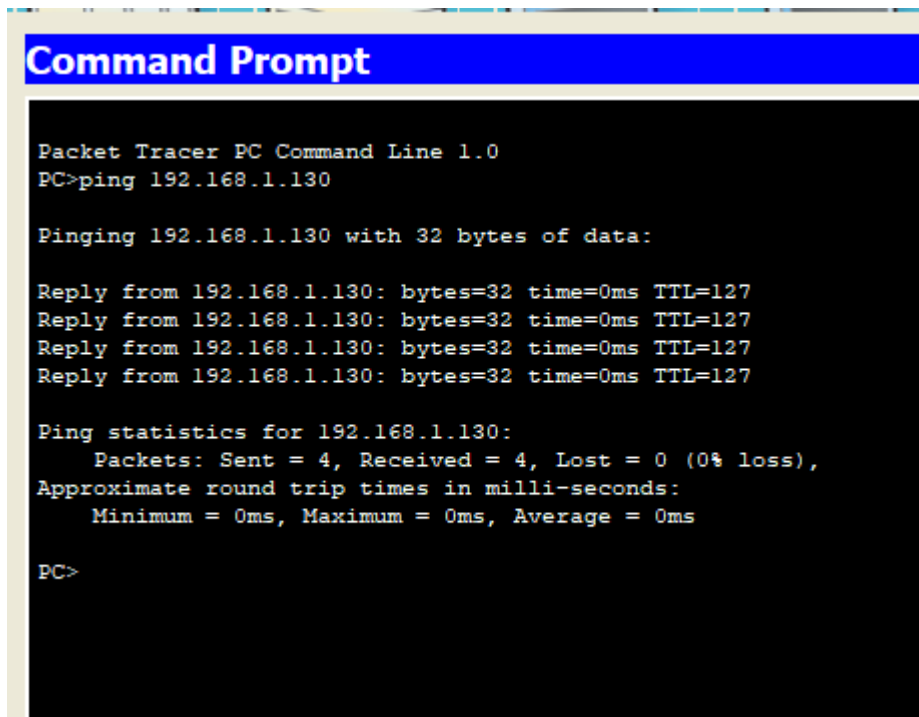
**Router(config-router)#version 2**

**Router(config-router)#network 192.168.1.0**

**Router(config-router)#network 192.168.1.64**

**Router(config-router)#network 192.168.1.128**

Thực hiện ping thử



```
Command Prompt

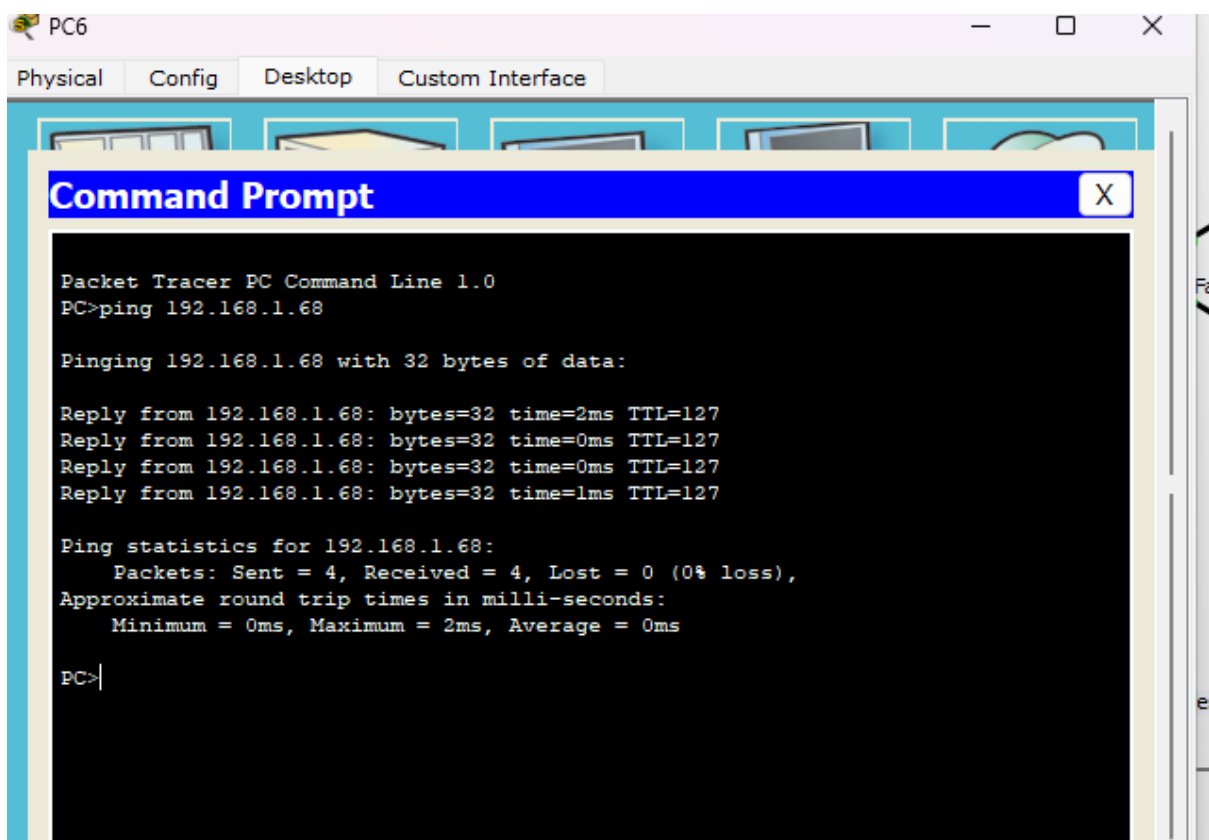
Packet Tracer PC Command Line 1.0
PC>ping 192.168.1.130

Pinging 192.168.1.130 with 32 bytes of data:

Reply from 192.168.1.130: bytes=32 time=0ms TTL=127
Reply from 192.168.1.130: bytes=32 time=0ms TTL=127
Reply from 192.168.1.130: bytes=32 time=0ms TTL=127
Reply from 192.168.1.130: bytes=32 time=0ms TTL=127

Ping statistics for 192.168.1.130:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms

PC>
```



```
PC6

Physical Config Desktop Custom Interface

Command Prompt

Packet Tracer PC Command Line 1.0
PC>ping 192.168.1.68

Pinging 192.168.1.68 with 32 bytes of data:

Reply from 192.168.1.68: bytes=32 time=2ms TTL=127
Reply from 192.168.1.68: bytes=32 time=0ms TTL=127
Reply from 192.168.1.68: bytes=32 time=0ms TTL=127
Reply from 192.168.1.68: bytes=32 time=1ms TTL=127

Ping statistics for 192.168.1.68:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 2ms, Average = 0ms

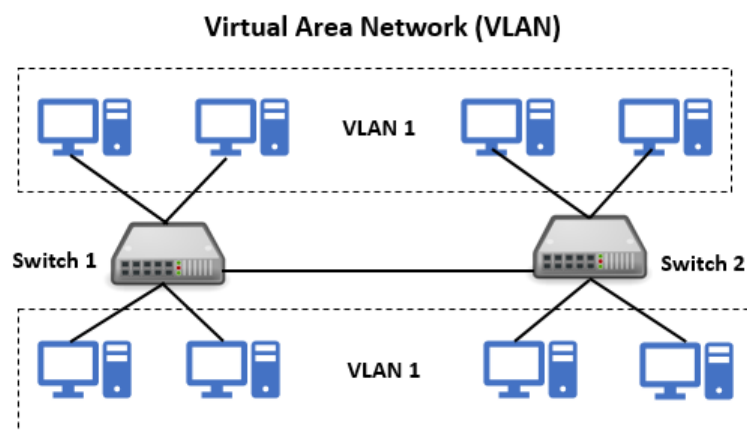
PC>
```

Vậy là ping đã thành công.

## 2. VLAN

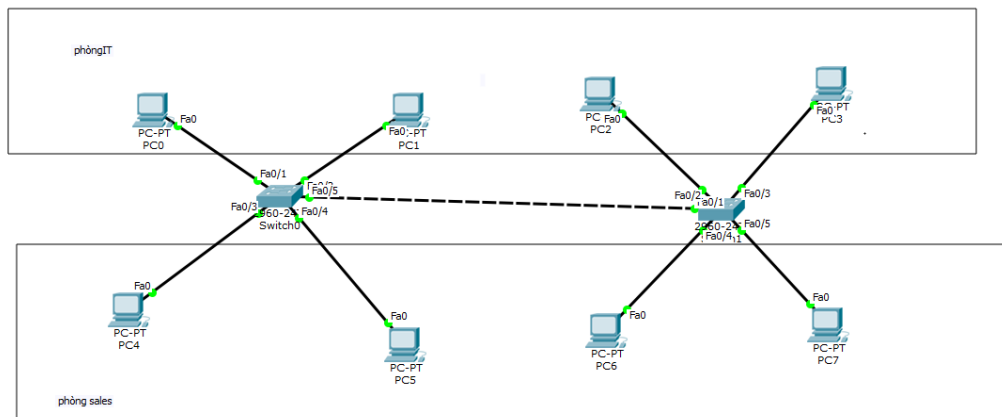
### 2.1. Lí thuyết

**VLAN** là viết tắt của **Virtual Local Area Network** - nghĩa là **mạng LAN ảo**, là một mạng tùy chỉnh, được hình thành từ một hoặc nhiều mạng LAN, cho phép các nhóm thiết bị khả dụng kết nối cùng với một mạng dù không đặt cạnh nhau. Từ đó tạo nên mạng VLAN được quản lý tương tự với mạng LAN vật lý. VLAN hay Virtual LAN giúp sử dụng tài nguyên mạng hiệu quả hơn và hữu ích khi kết nối quá nhiều thiết bị cùng một mạng.



Thông thường Router sẽ có vai trò là miền quảng bá. Tuy nhiên, trong mạng VLAN, thiết bị chuyển mạch Switch sẽ có chức năng tương tự tạo nên miền quảng bá. Vậy nên xét về mặt kỹ thuật, mạng VLAN được xem là một miền quảng bá được tạo ra bởi Switch. Hiểu đơn giản là việc ta nhóm một số port của switch lại thành một mạng nhỏ và chỉ cho những máy tính trong cùng một nhóm đó giao tiếp với nhau. Nhờ VLAN ta có thể thực hiện điều này và giúp giảm chi phí hơn so với việc sử dụng LAN thường.

## 2.2. VLAN trên packet tracer



Ta có một sơ đồ mạng như sau. Ta có 2 switch được đặt ở 2 tầng khác nhau. Mỗi switch được nối với 4 host. Ta muốn PC0, PC1 ở SW0 và PC2, PC3 ở SW1 chỉ giao tiếp với nhau. Những PC còn lại cũng chỉ giao tiếp với nhau và không giao tiếp với các PC khác. Tượng trưng cho các máy phòng IT thì nhận được thông tin của nhau và không nhận thông tin phòng sales và ngược lại, PC phòng sales nhận thông tin của nhau và không nhận thông tin phòng IT.

Ta có 4 bước phải làm như sau:

- Bước 1: tạo các VLAN tại switch
- Bước 2 : gán vlan cho các interface
- Bước 3: cài IP cho các PC
- Bước 4 : Trunking

Bây giờ ta sẽ bắt đầu từng bước

### a. Bước 1: Tạo các VLAN

Ta chọn vào switch0 và ấn các lệnh sau:

```
Switch>ena
```

```
Switch#conf t
```

```
Switch(config)#vlan 10
```

```
Switch(config-vlan)#name IT
```

```
Switch(config-vlan)#vlan 20
```

```
Switch(config-vlan)#name SALES
```

Ta làm tương tự với switch1 các câu lệnh trên

Sau đó ta kiểm tra việc khởi tạo thành công chưa bằng lệnh

**Switch#show vlan brief**

```
Switch#show vlan brief
```

VLAN	Name	Status	Ports
1	default	active	Fa0/1, Fa0/2, Fa0/3, Fa0/4 Fa0/5, Fa0/6, Fa0/7, Fa0/8 Fa0/9, Fa0/10, Fa0/11, Fa0/12 Fa0/13, Fa0/14, Fa0/15, Fa0/16 Fa0/17, Fa0/18, Fa0/19, Fa0/20 Fa0/21, Fa0/22, Fa0/23, Fa0/24 Gig0/1, Gig0/2
10	IT	active	
20	SALES	active	
1002	fddi-default	active	
1003	token-ring-default	active	
1004	fddinet-default	active	
1005	trnet-default	active	

```
Switch#|
```

Bảng VLAN hiện lên 2 VLAN ta mới tạo kia là thành công

## **b. Bước 2 : gán vlan cho các interface**

Trong sơ đồ mạng, ta thấy PC0 và PC1 sẽ là vlan 10, PC4 và PC5 sẽ là vlan 20. Do vậy interface kết nối switch0 với PC0 và PC1 sẽ được gán vào vlan 10. Interface kết nối switch0 với PC4 và PC5 sẽ được gán vào vlan 20. Tương tự so với switch1

Ta chọn vào switch0 và viết lệnh

```
Switch(config)#int f0/1
```

```
Switch(config-if)#switchport mode access
```

```
Switch(config-if)#switchport access vlan 10
```

```
Switch(config)#int f0/2
```

```
Switch(config-if)#switchport mode access
```

```
Switch(config-if)#switchport access vlan 10
```

```
Switch(config)#int f0/3
```

```
Switch(config-if)#switchport mode access
```

```
Switch(config-if)#switchport access vlan 20
```

```
Switch(config)#int f0/4
```

```
Switch(config-if)#switchport mode access
```

```
Switch(config-if)#switchport access vlan 20
```

Ta chuyển interface sang mode access và gán vào vlan 10 và 20. Ta làm tương tự với các switch1. Để kiểm tra ta gõ lệnh

### Switch#show vlan brief

```
Switch#show vlan brief
```

VLAN	Name	Status	Ports
1	default	active	Fa0/2, Fa0/5, Fa0/6, Fa0/7 Fa0/8, Fa0/9, Fa0/10, Fa0/11 Fa0/12, Fa0/13, Fa0/14, Fa0/15 Fa0/16, Fa0/17, Fa0/18, Fa0/19 Fa0/20, Fa0/21, Fa0/22, Fa0/23 Fa0/24, Gig0/1, Gig0/2
10	IT	active	Fa0/1
20	SALES	active	Fa0/3, Fa0/4
1002	fddi-default	active	
1003	token-ring-default	active	
1004	fddinet-default	active	
1005	trnet-default	active	

Switch#

Hiện thêm phần port ở Vlan10 và Vlan20 như thế này là thành công

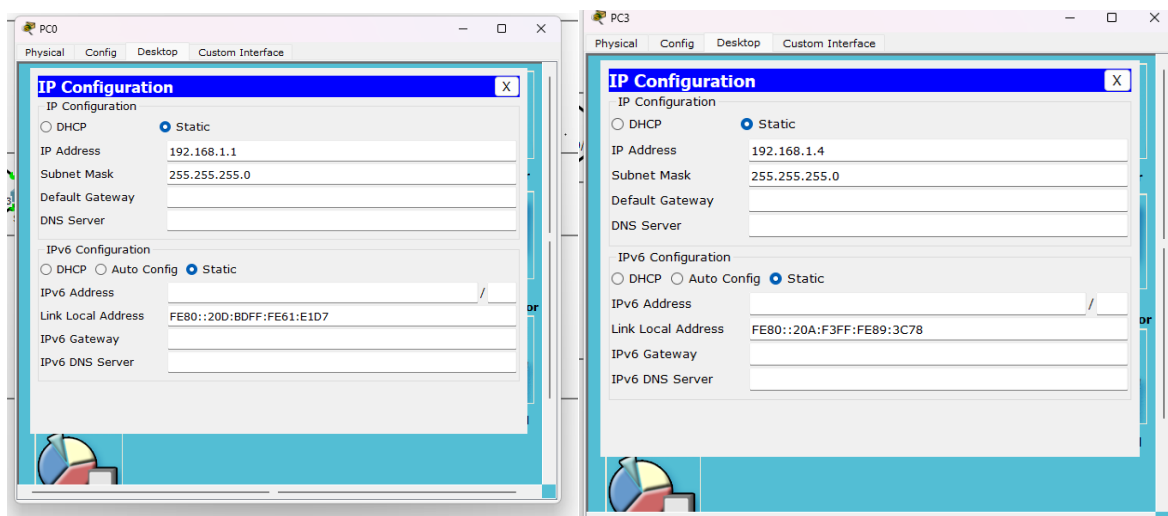
### c. Bước 3: Gán IP cho PC

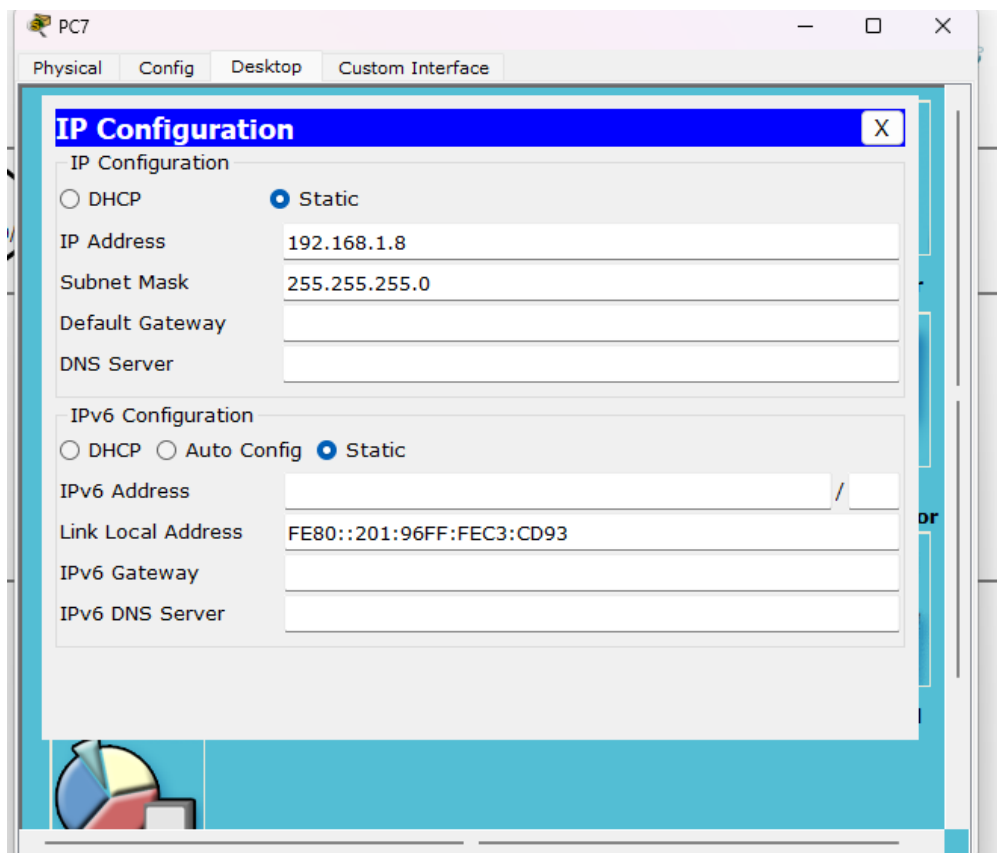
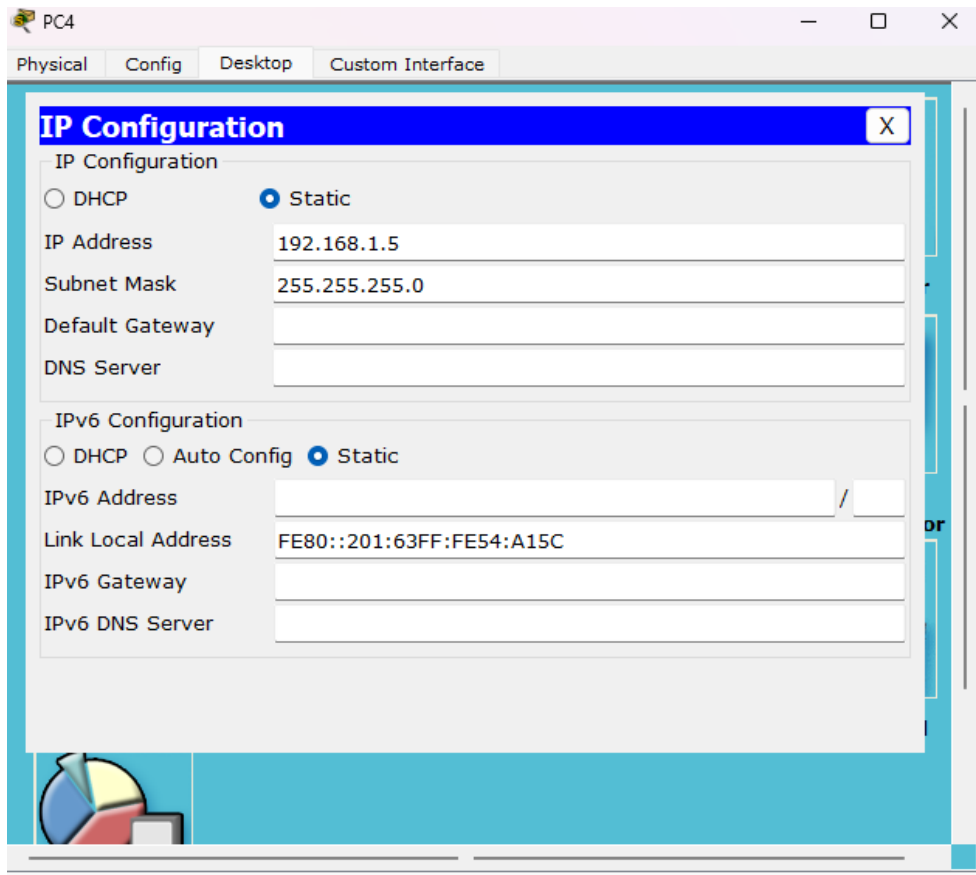
Với 4 PC thuộc Vlan IT ta gán IP lần lượt là:

- 192.168.1.1
- 192.168.1.2
- 192.168.1.3
- 192.168.1.4

Với 4PC thuộc VLAN SALES ta gán IP lần lượt là:

- 192.168.1.5
- 192.168.1.6
- 192.168.1.7
- 192.168.1.8





#### d. Bước 4: Trunking

Tại bước này ta chuyển mode của interface kết nối 2 switch với nhau thành mode trunk để giúp switch có thể giao tiếp với nhau và chuyển tiếp gói tin theo vlan chính xác

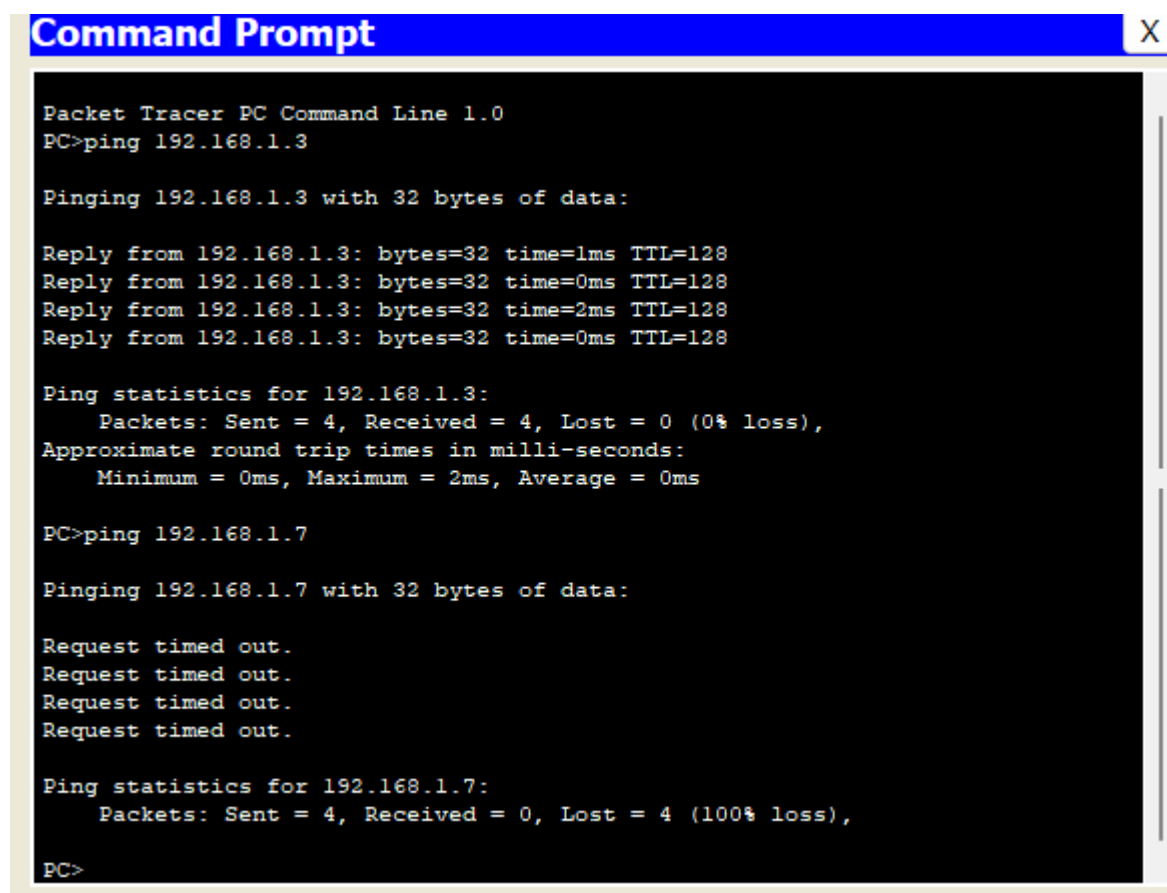
Ta vào switch0 và gõ lệnh:

```
Switch(config)#interface FastEthernet0/5
```

```
Switch(config-if)#switchport mode trunk
```

sau khi hoàn thành bước này thì ta đã thiết lập vlan thành công. Giờ hãy thử ping trong cùng vlan và khác vlan để xem kết quả

ta vào PC0 (192.168.1.1) và ping sang các PC khác

A screenshot of a Packet Tracer Command Prompt window. The title bar is blue with the text "Command Prompt" and a close button "X". The main area is black with white text. The text shows a PC command line interface. The user enters "ping 192.168.1.3". The output shows four successful replies from 192.168.1.3 with varying times (1ms, 0ms, 2ms, 0ms) and TTL=128. Then the user enters "ping 192.168.1.7". The output shows four "Request timed out." messages. Finally, the user enters "ping 192.168.1.7" again, and the output shows "Packets: Sent = 4, Received = 0, Lost = 4 (100% loss)".

```
Packet Tracer PC Command Line 1.0
PC>ping 192.168.1.3

Pinging 192.168.1.3 with 32 bytes of data:

Reply from 192.168.1.3: bytes=32 time=1ms TTL=128
Reply from 192.168.1.3: bytes=32 time=0ms TTL=128
Reply from 192.168.1.3: bytes=32 time=2ms TTL=128
Reply from 192.168.1.3: bytes=32 time=0ms TTL=128

Ping statistics for 192.168.1.3:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 2ms, Average = 0ms

PC>ping 192.168.1.7

Pinging 192.168.1.7 with 32 bytes of data:

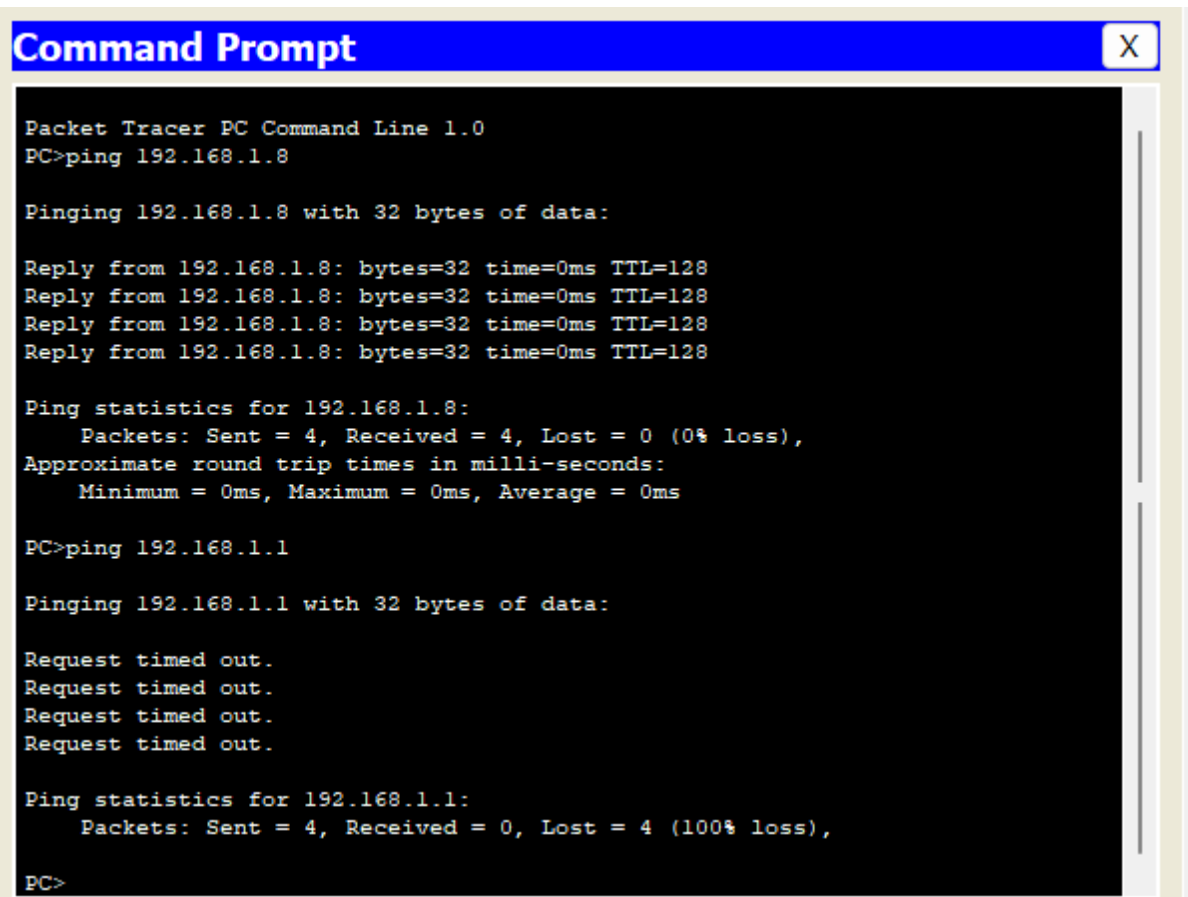
Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 192.168.1.7:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),

PC>
```

Ping sang 192.168.1.3 cùng vlan thì sẽ ping được thành công. Tuy nhiên ping sang 192.168.1.7 thuộc vlan khác sẽ không thành công





The image shows a Packet Tracer PC Command Line window. The title bar is blue with the text 'Command Prompt' and a close button 'X'. The main area is black with white text. The text shows the user entering 'ping 192.168.1.8' and receiving four successful replies with 0ms response times. Then, the user enters 'ping 192.168.1.1' and receives four 'Request timed out' messages, indicating a 100% loss.

```
Packet Tracer PC Command Line 1.0
PC>ping 192.168.1.8

Pinging 192.168.1.8 with 32 bytes of data:

Reply from 192.168.1.8: bytes=32 time=0ms TTL=128
Reply from 192.168.1.8: bytes=32 time=0ms TTL=128
Reply from 192.168.1.8: bytes=32 time=0ms TTL=128
Reply from 192.168.1.8: bytes=32 time=0ms TTL=128

Ping statistics for 192.168.1.8:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms

PC>ping 192.168.1.1

Pinging 192.168.1.1 with 32 bytes of data:

Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 192.168.1.1:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),

PC>
```

Ta vào PC4 (192.168.1.5) thì sẽ có kết quả như vậy