NATIONAL UNIVERSITY OF COMPUTER & EMERGING SCIENCE

Computer Network Lab (CL307)

Awais Ahmed || Faizan Yousuf || Munim Ali

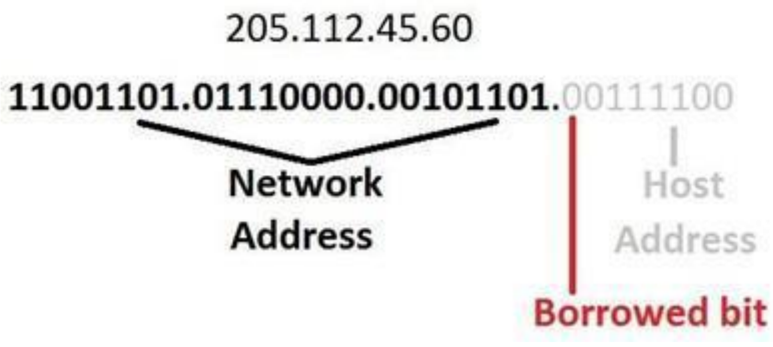
awais.ahmed@nu.edu.pk || faizan.yousuf@nu.edu.pk || munim.ali@nu.edu.pk

Lab Session 11

Subnetting and the Subnet Mask

To subnet a network is to create logical divisions of the network. Subnetting, therefore, involves dividing the network into smaller portions called subnets. Subnetting applies to IP addresses because this is done by borrowing bits from the host portion of the IP address. In a sense, the IP address then has three components - the network part, the subnet part and, finally, the host part.

We create a subnet by logically grabbing the last bit from the network component of the address and using it to determine the number of subnets required. In the following example, a Class C address normally has 24 bits for the network address and eight for the host, but we are going to borrow the left-most bit of the host address and declare it as identifying the subnet.



If the bit is a 0, then that will be one subnet; if the bit is a 1 that would be the second subnet. Of course, with only one borrowed bit we can only have two possible subnets. By the same token, that also reduces the number of hosts we can have on the network to 127 (but actually 125 useable addresses given all zeros and all ones are not recommended addresses), down from 255.

So how can you tell how many bits should be borrowed, or, in other words, how many subnets we want to have on our network?

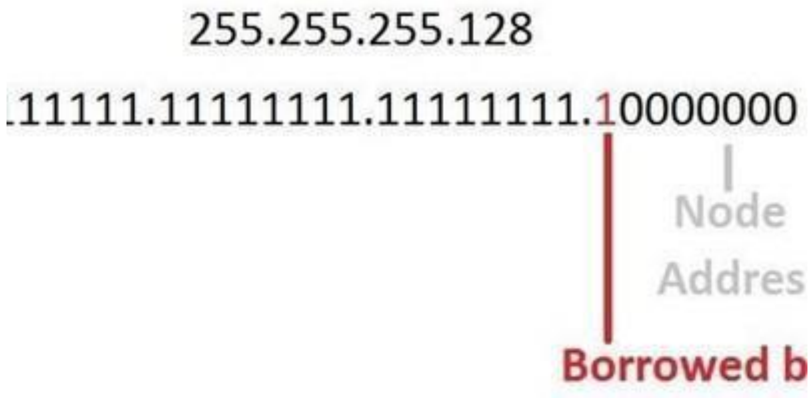
The answer is with a subnet mask.

Subnet masks sound a lot scarier than they really are. All that a subnet mask does is indicate how many bits are being "borrowed" from the host component of an IP address.

If you can't remember anything about subnetting, remember this concept. It is the foundation of all subnetting.

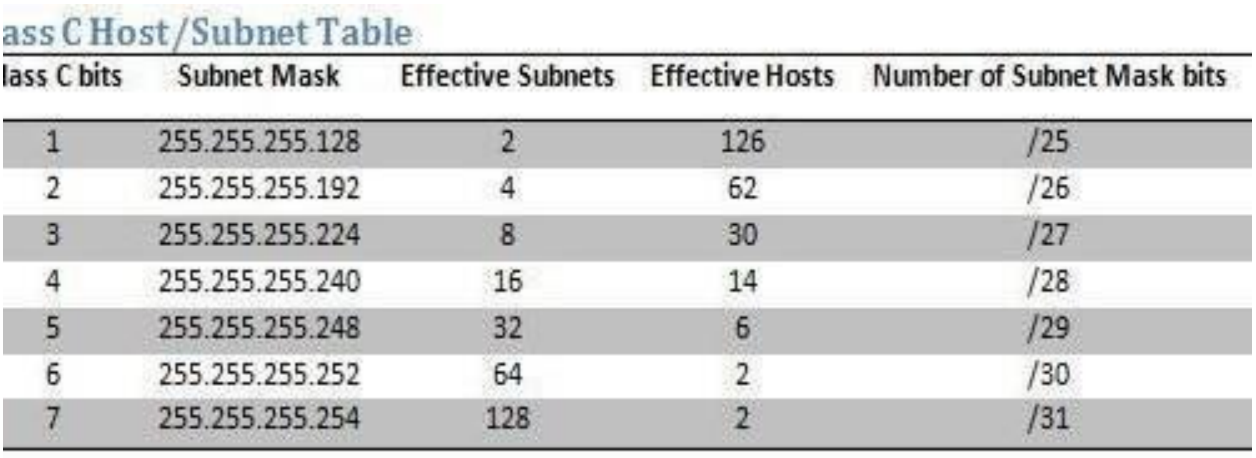
The reason a subnet mask has this name is that it literally masks out the host bits being borrowed from the host address portion of the IP address.

In the following diagram, there is a subnet mask for a Class C address. The subnet mask is 255.255.255.128 which, when translated into bits, indicates which bits of the host part of the address will be used to determine the subnet number.



Of course, more bits borrowed means fewer individually addressable hosts that can be on the network. Sometimes, all the combinations and permutations can be confusing, so here are some tables of subnet possibilities.

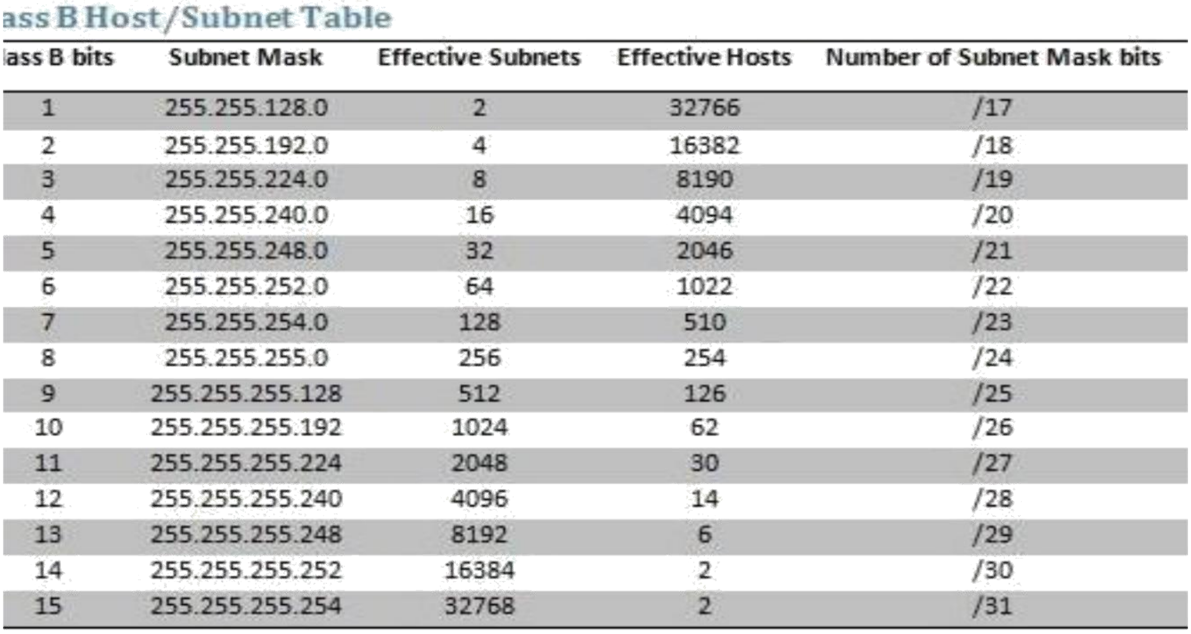
CLASS C HOST/Subnet Table



ass C Host/Subnet Table

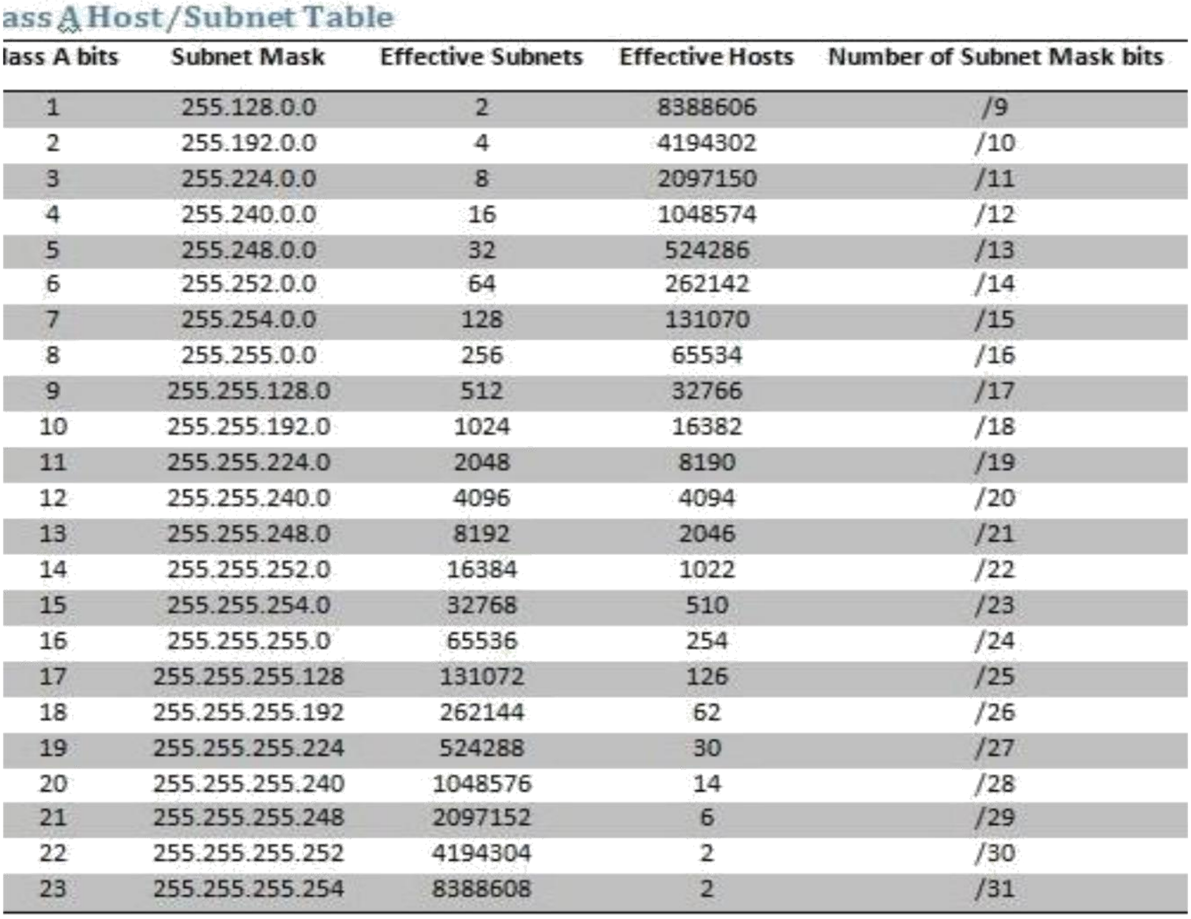
lass C bits	Subnet Mask	Effective Subnets	Effective Hosts	Number of Subnet Mask bits
1	255.255.255.128	2	126	/25
2	255.255.255.192	4	62	/26
3	255.255.255.224	8	30	/27
4	255.255.255.240	16	14	/28
5	255.255.255.248	32	6	/29
6	255.255.255.252	64	2	/30
7	255.255.255.254	128	2	/31

CLASS B and CLASS A HOST/Subnet Table



ass B Host/Subnet Table

lass B bits	Subnet Mask	Effective Subnets	Effective Hosts	Number of Subnet Mask bits
1	255.255.128.0	2	32766	/17
2	255.255.192.0	4	16382	/18
3	255.255.224.0	8	8190	/19
4	255.255.240.0	16	4094	/20
5	255.255.248.0	32	2046	/21
6	255.255.252.0	64	1022	/22
7	255.255.254.0	128	510	/23
8	255.255.255.0	256	254	/24
9	255.255.255.128	512	126	/25
10	255.255.255.192	1024	62	/26
11	255.255.255.224	2048	30	/27
12	255.255.255.240	4096	14	/28
13	255.255.255.248	8192	6	/29
14	255.255.255.252	16384	2	/30
15	255.255.255.254	32768	2	/31



ass A Host/Subnet Table

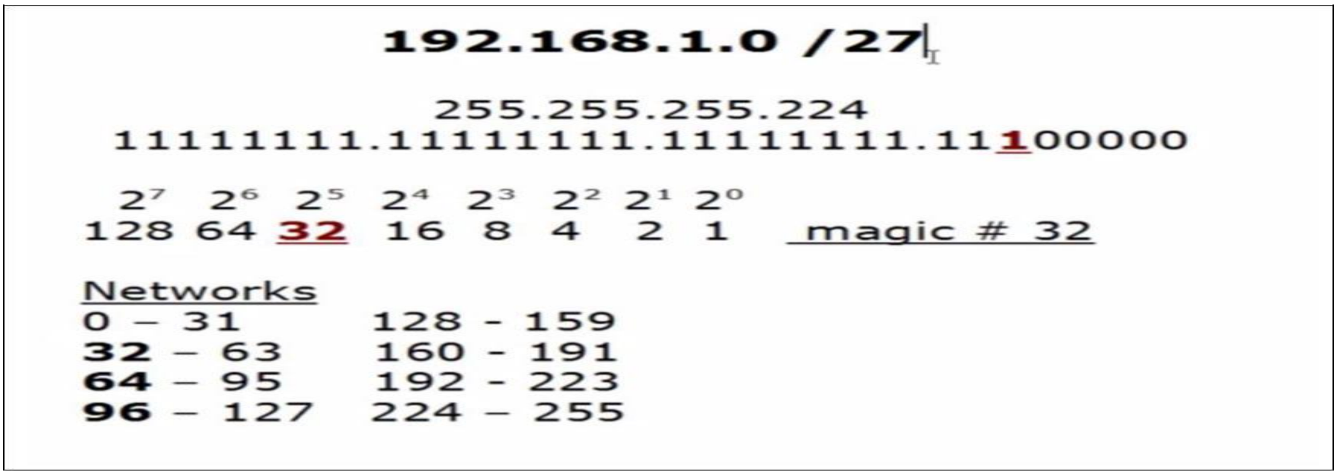
lass A bits	Subnet Mask	Effective Subnets	Effective Hosts	Number of Subnet Mask bits
1	255.128.0.0	2	8388606	/9
2	255.192.0.0	4	4194302	/10
3	255.224.0.0	8	2097150	/11
4	255.240.0.0	16	1048574	/12
5	255.248.0.0	32	524286	/13
6	255.252.0.0	64	262142	/14
7	255.254.0.0	128	131070	/15
8	255.255.0.0	256	65534	/16
9	255.255.128.0	512	32766	/17
10	255.255.192.0	1024	16382	/18
11	255.255.224.0	2048	8190	/19
12	255.255.240.0	4096	4094	/20
13	255.255.248.0	8192	2046	/21
14	255.255.252.0	16384	1022	/22
15	255.255.254.0	32768	510	/23
16	255.255.255.0	65536	254	/24
17	255.255.255.128	131072	126	/25
18	255.255.255.192	262144	62	/26
19	255.255.255.224	524288	30	/27
20	255.255.255.240	1048576	14	/28
21	255.255.255.248	2097152	6	/29
22	255.255.255.252	4194304	2	/30
23	255.255.255.254	8388608	2	/31

Note that this combination of IP addresses and subnet masks in the charts are written as two separate values, such as Network Address = 205.112.45.60, Mask = 255.255.255.128, or as an IP address with the number of bits indicated as being used for the mask, like 205.112.45.60/25.

Subnet masks work because of the magic of Boolean logic. To best understand how a subnet mask actually does its thing, you must remember that a subnet mask is only relevant when getting to a subnet. In other words, determining what subnet an IP address lives on is the only reason for a subnet mask. It’s devices like routers and switches that make use of subnet masks.

Implementation of Subnetting in Cisco Packet Tracer

Consider an IP of Class C 192.168.1.0/27, using above IP calculate the subnets and implement the scenario in Cisco Packet Tracer.



192.168.1.0 /27

255.255.255.224
11111111.11111111.11111111.11100000

2^7 2^6 2^5 2^4 2^3 2^2 2^1 2^0
128 64 **32** 16 8 4 2 1 magic # 32

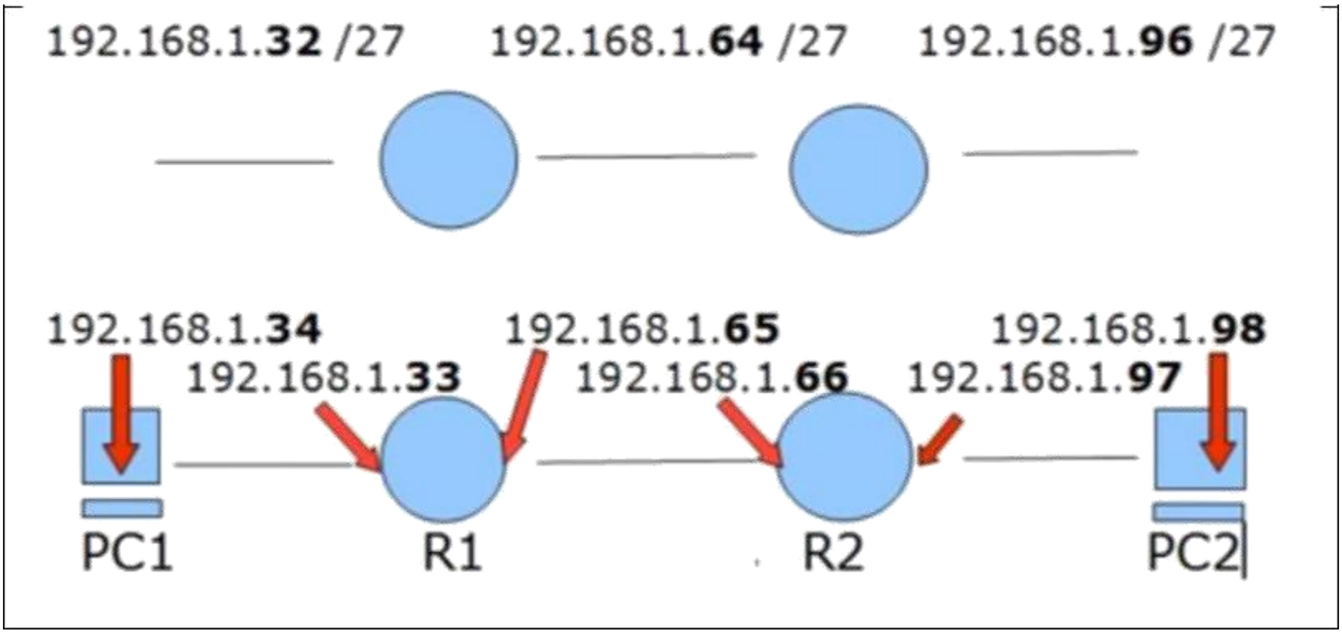
Networks

0 – 31
32 – 63
64 – 95
96 – 127
128 - 159
160 - 191
192 - 223
224 – 255

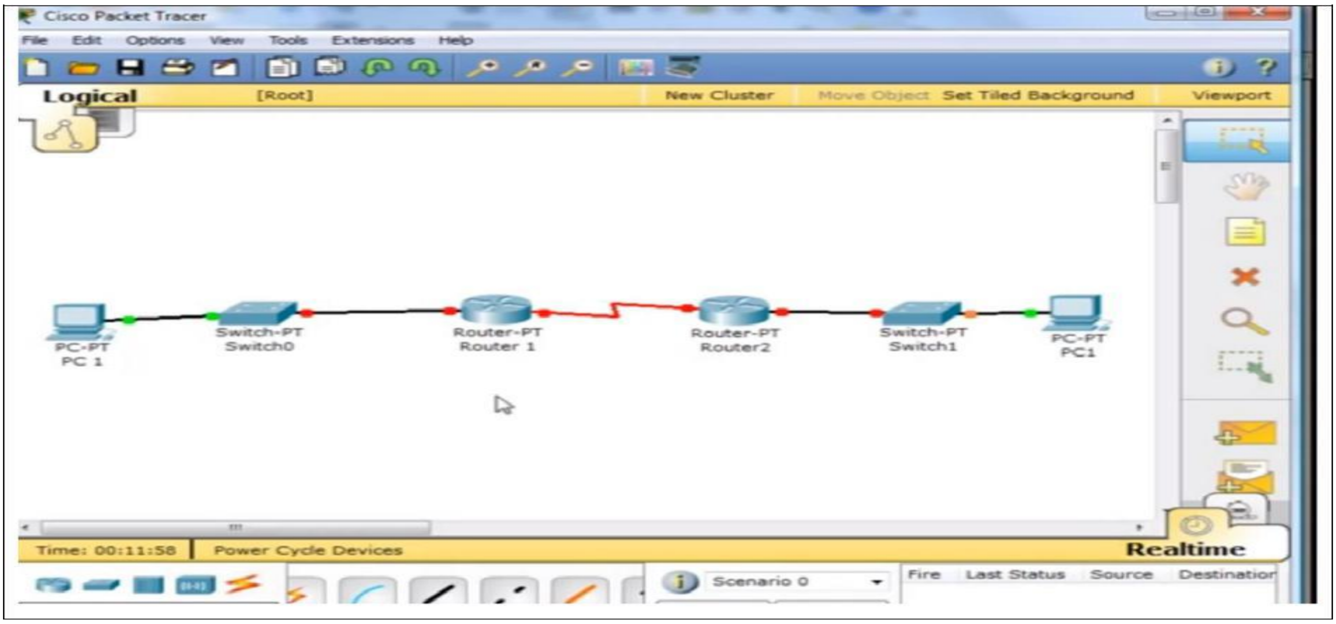
Calculation:

From above, we have:
Possible Subnets:2_n = 2_3 =8
Possible Hosts = 32
Usable Hosts in each Subnet = 32 - 2 = 30
Note: 1st address of every subnet shows network address and last address shows Broadcast address. e.g. 0,32,64 & 96 represent Network address where 31,63,95 &127 represent Broadcast address.
Custom Subnet Mask = 255.255.255.224

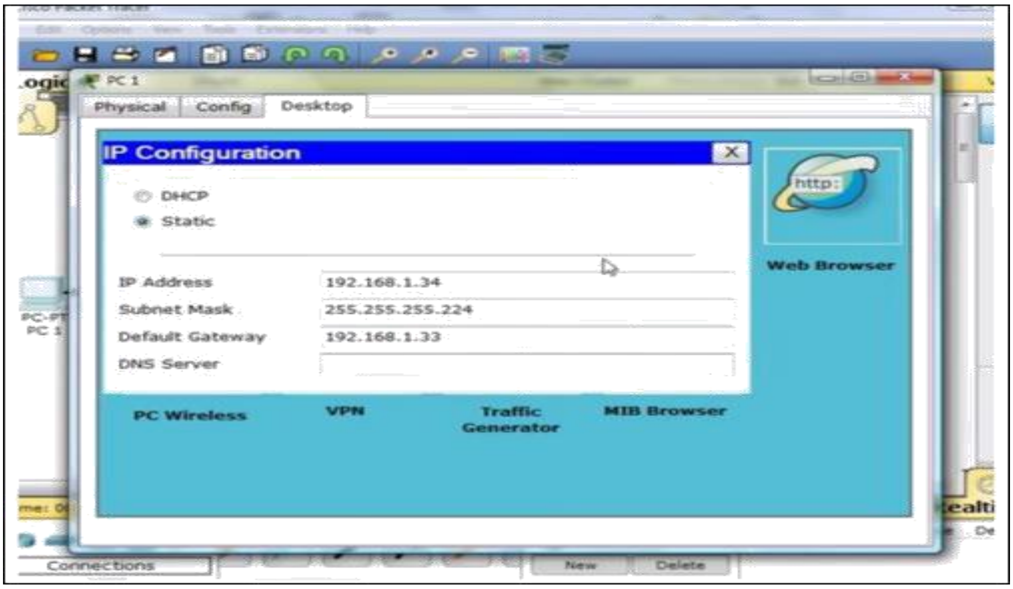
Now implementing below scenario in Cisco packet Tracer.



We have taken two routers R1 & R2 and connected their Fast Ethernet interface Fa 0/0 with the switch. While routers connected with their serial interface 2/0.



Now configuring PC1.

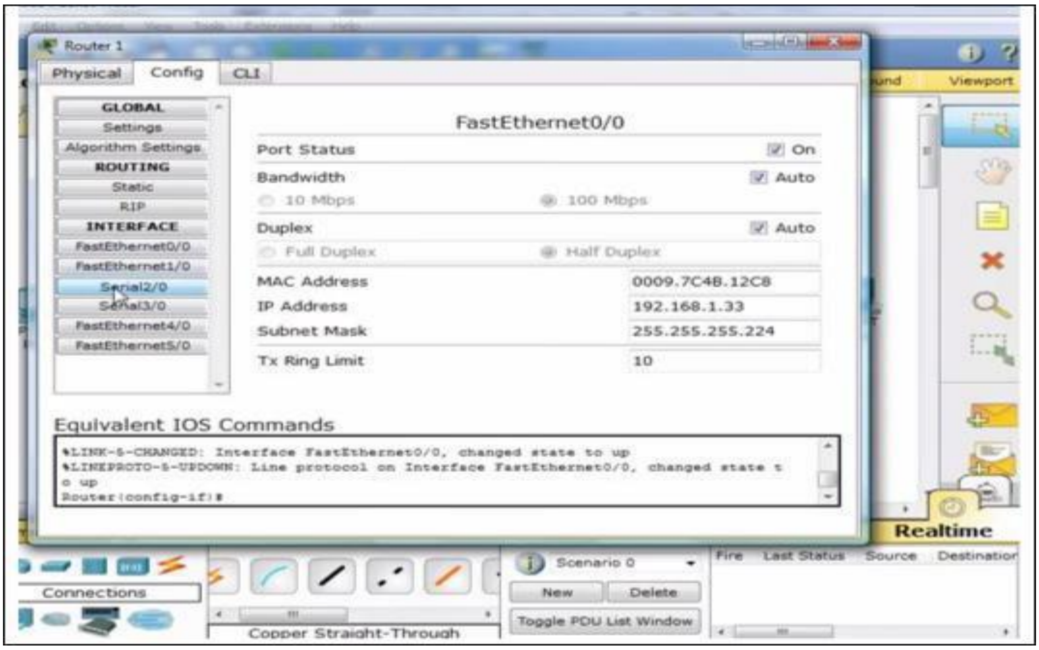


IP Configuration

○ DHCP
● Static

IP Address	192.168.1.34
Subnet Mask	255.255.255.224
Default Gateway	192.168.1.33
DNS Server	

Now configure the Interface FastEthernet0/0 of Router R1.



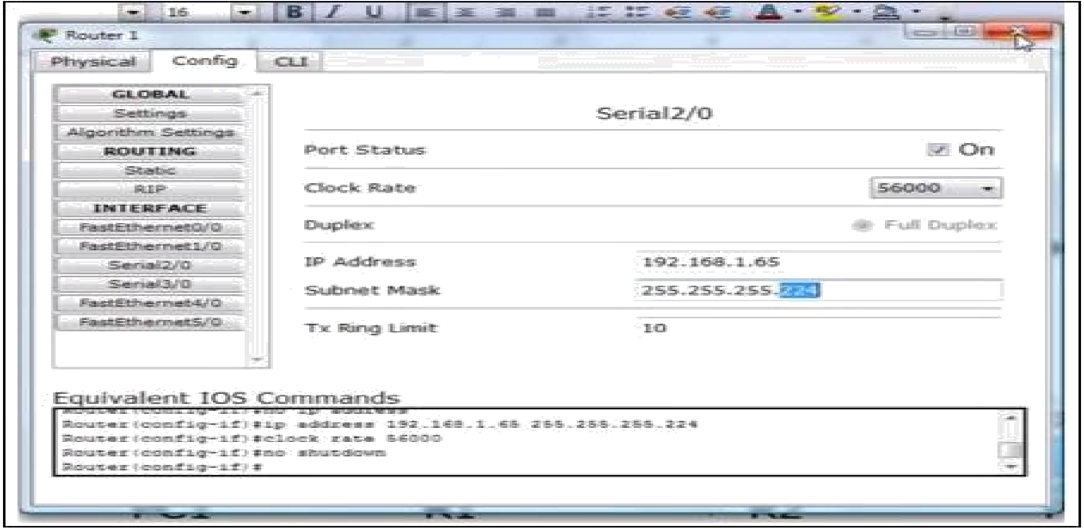
FastEthernet0/0

Port Status	☑ On
Bandwidth	☑ Auto
○ 10 Mbps	● 100 Mbps
Duplex	☑ Auto
○ Full Duplex	● Half Duplex
MAC Address	0009.7C48.12C8
IP Address	192.168.1.33
Subnet Mask	255.255.255.224
Tx Ring Limit	10

Equivalent IOS Commands

```
%LINK-5-CHANGED: Interface FastEthernet0/0, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/0, changed state t
o up
Router(config-if)#
```

Now configure the Interface Serial2/0 of Router R1.



Serial2/0

Port Status ☑ On

Clock Rate 56000

Duplex ● Full Duplex

IP Address 192.168.1.65

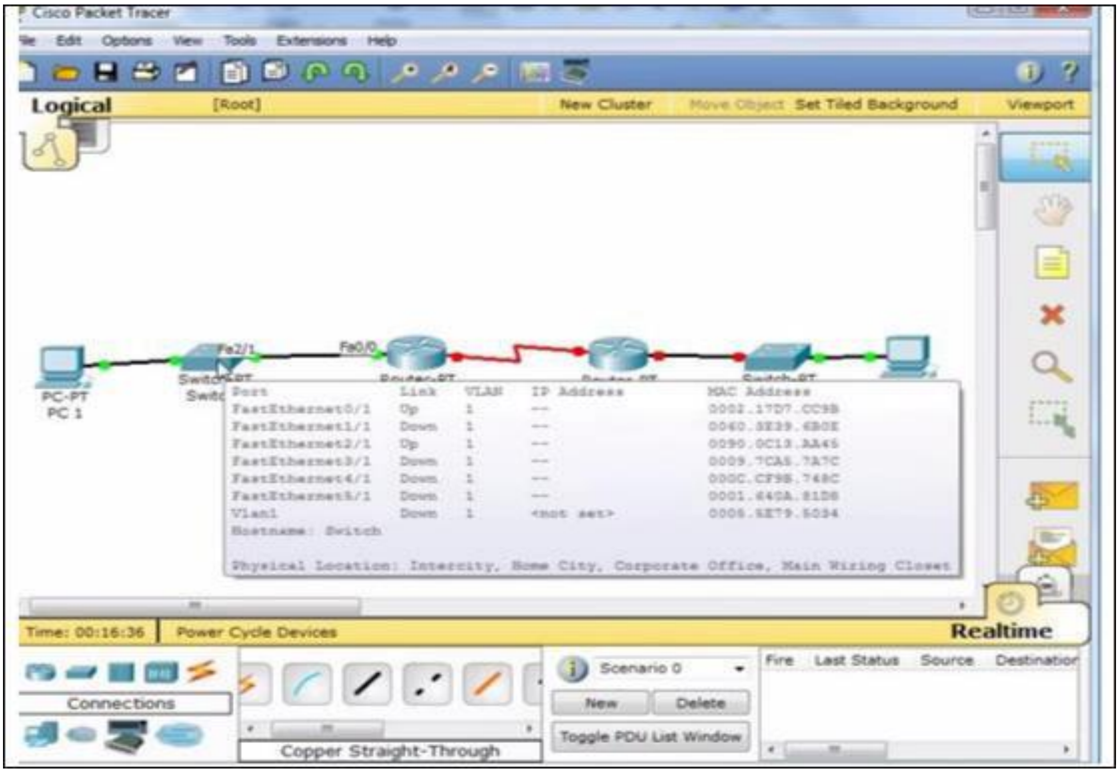
Subnet Mask 255.255.255.224

Tx Ring Limit 10

Equivalent IOS Commands

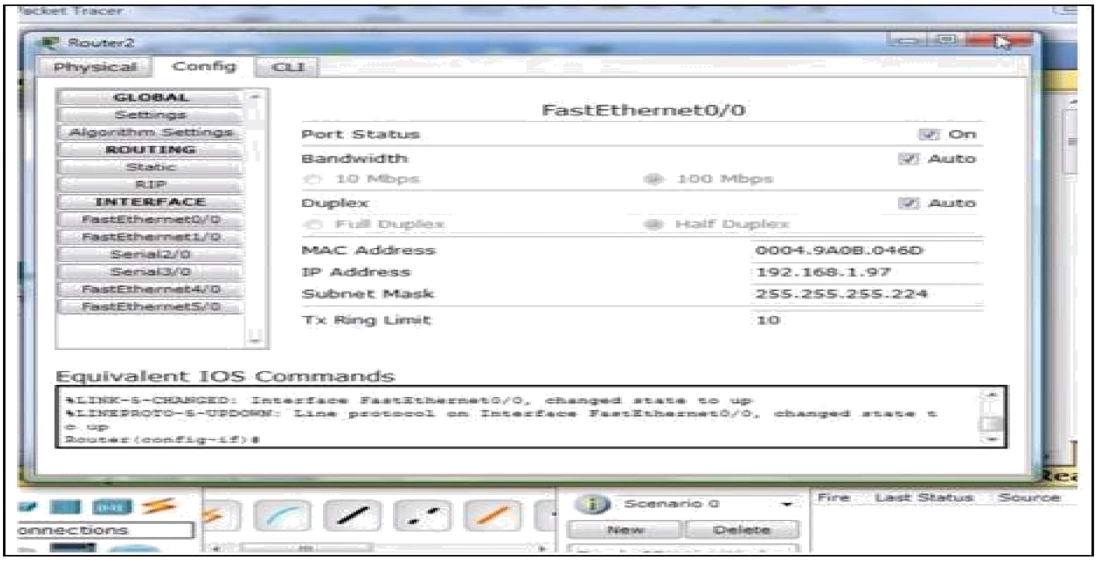
```
Router(config-if)#no ip address
Router(config-if)#ip address 192.168.1.65 255.255.255.224
Router(config-if)#clock rate 56000
Router(config-if)#no shutdown
Router(config-if)#
```

After the configuring the Interface FastEthernet0/0 and Serial2/0 of Router R1.



```
Port                Link   VLAN   IP Address          MAC Address
FastEthernet0/1     Up     1      --                  0002.17D7.CC9B
FastEthernet1/1     Down   1      --                  0060.3E39.6B0E
FastEthernet2/1     Up     1      --                  0090.0C13.AA45
FastEthernet3/1     Down   1      --                  0009.7CA5.7A7C
FastEthernet4/1     Down   1      --                  000C.CF9B.748C
FastEthernet5/1     Down   1      --                  0001.640A.81D8
Vlan1               Down   1      <not set>           0006.5E79.5034
Hostname: Switch

Physical Location: Intercity, Home City, Corporate Office, Main Wiring Closet
```

Now configure the Interface FastEthernet0/0 of Router R



FastEthernet0/0

Port Status ☑ On

Bandwidth ☑ Auto

○ 10 Mbps ● 100 Mbps

Duplex ☑ Auto

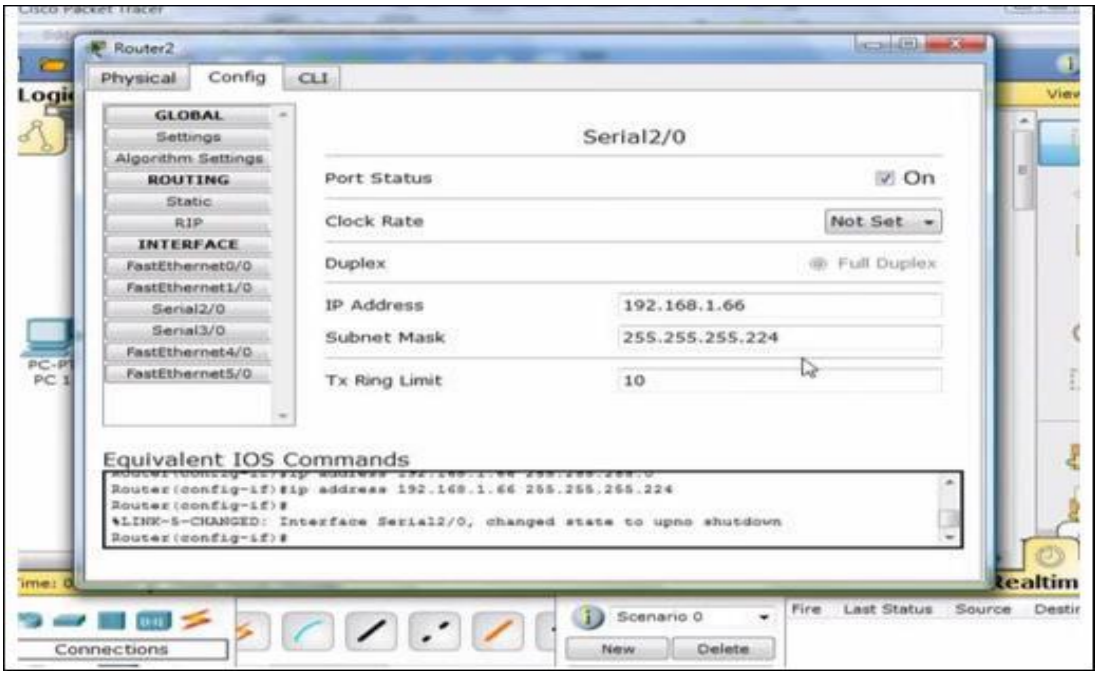
○ Full Duplex ● Half Duplex

MAC Address	0004.9A0B.046D
IP Address	192.168.1.97
Subnet Mask	255.255.255.224
Tx Ring Limit	10

Equivalent IOS Commands

```
%LINK-5-CHANGED: Interface FastEthernet0/0, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/0, changed state t
o up
Router(config-if)#
```

Configure the Interface Serial2/0 of Router R2.



Serial2/0

Port Status ☑ On

Clock Rate Not Set

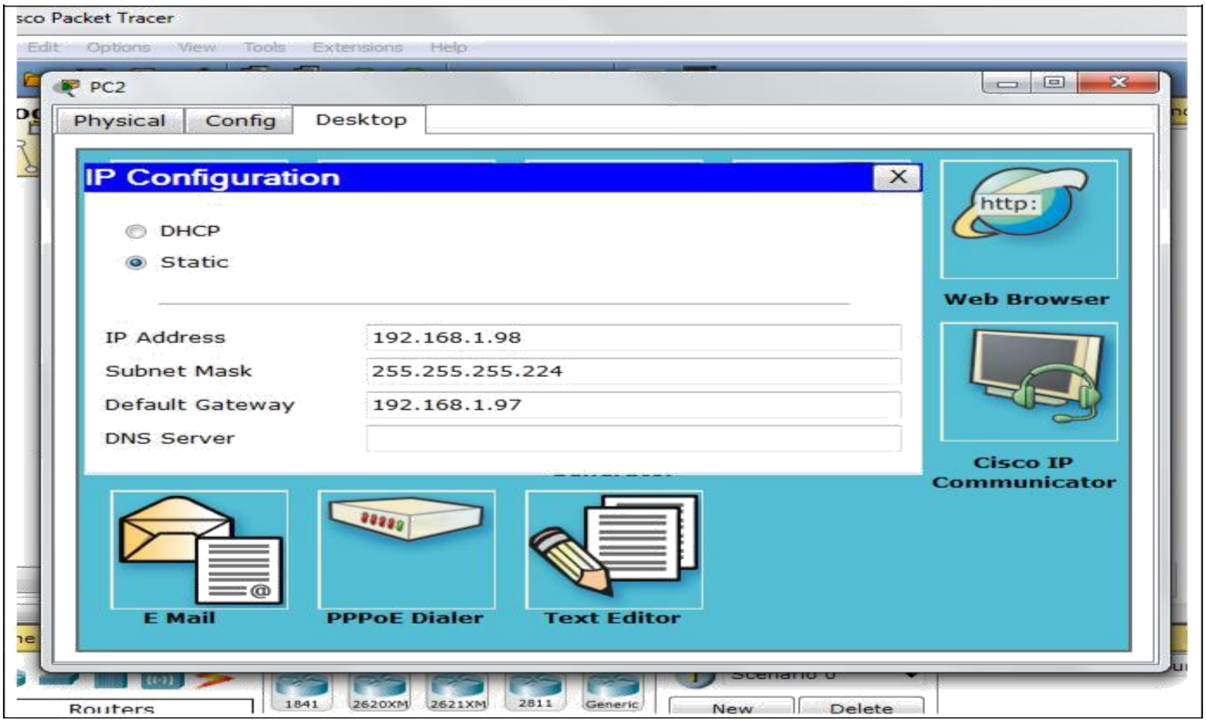
Duplex ● Full Duplex

IP Address	192.168.1.66
Subnet Mask	255.255.255.224
Tx Ring Limit	10

Equivalent IOS Commands

```
Router(config-if)#ip address 192.168.1.66 255.255.255.0
Router(config-if)#ip address 192.168.1.66 255.255.255.224
Router(config-if)#
%LINK-5-CHANGED: Interface Serial2/0, changed state to upno shutdown
Router(config-if)#
```

Now configuring PC2.



PC2

Physical | Config | Desktop

IP Configuration

○ DHCP

● Static

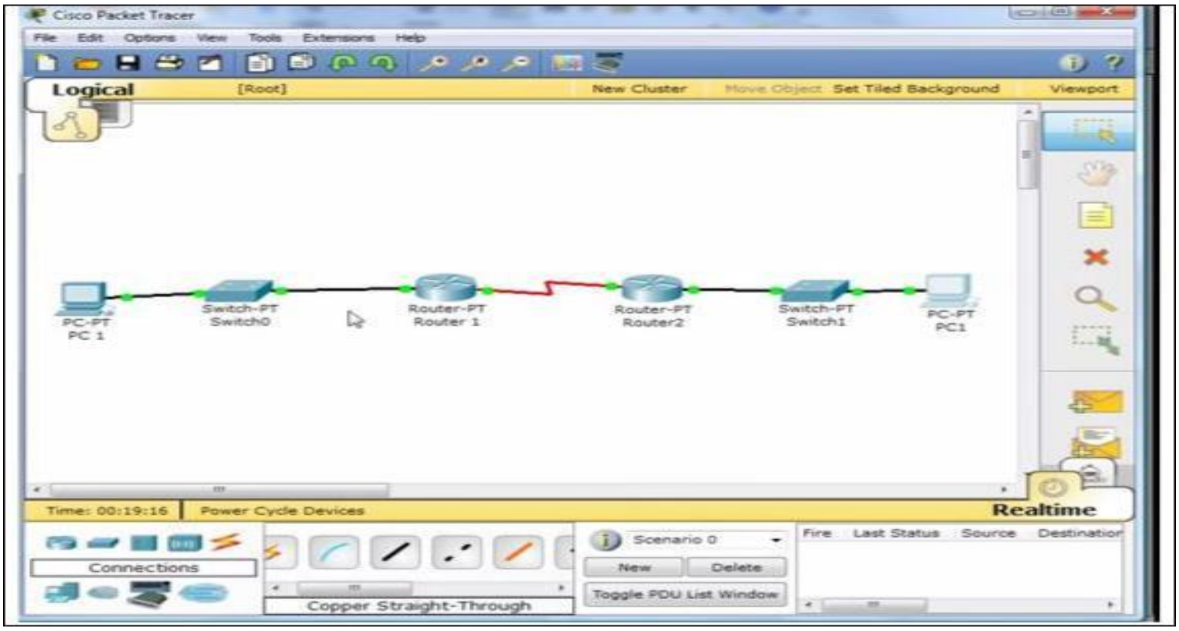
IP Address 192.168.1.98

Subnet Mask 255.255.255.224

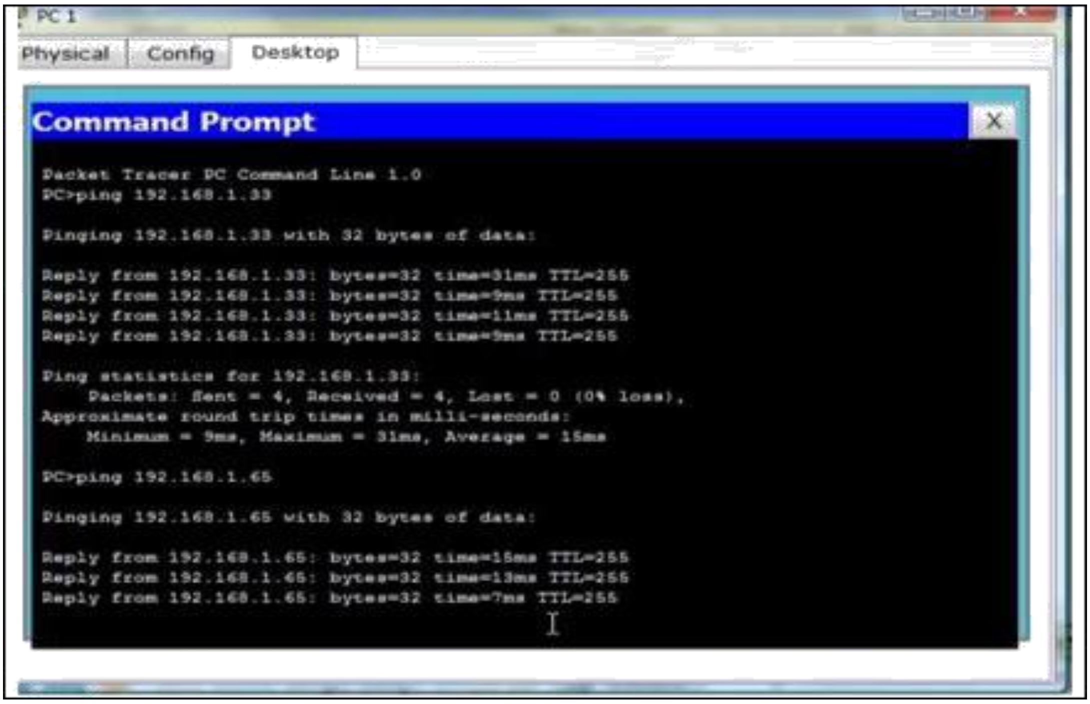
Default Gateway 192.168.1.97

DNS Server

Now we have gone through the entire configuration, all the interfaces are up.

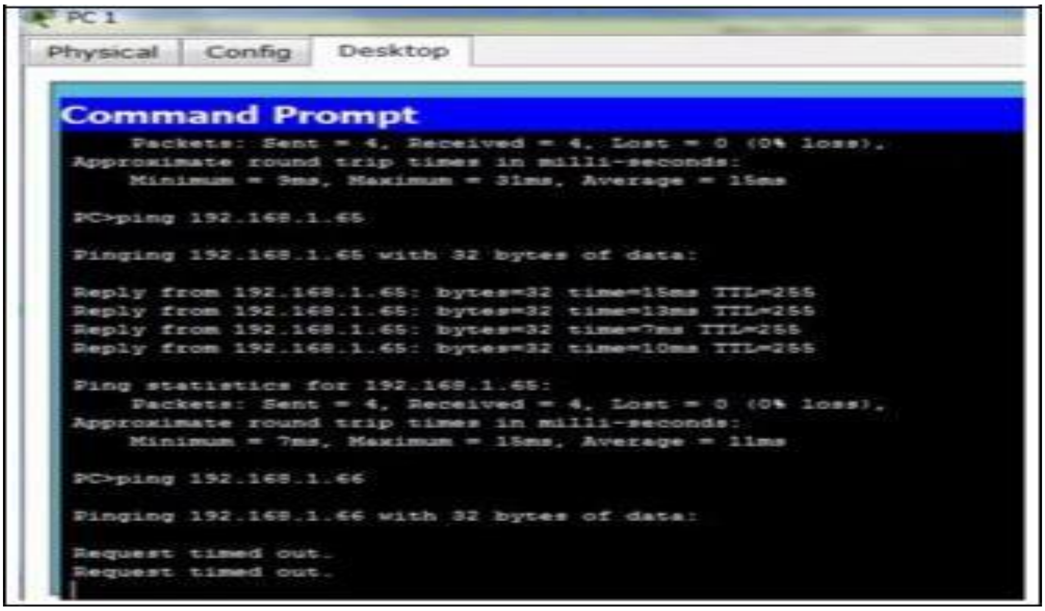


Now let start the pinging the interfaces from PC1.As we ping 192.168.1.33 and 192.168.1.65 we got the reply because these interface are directly connected to Router R1.



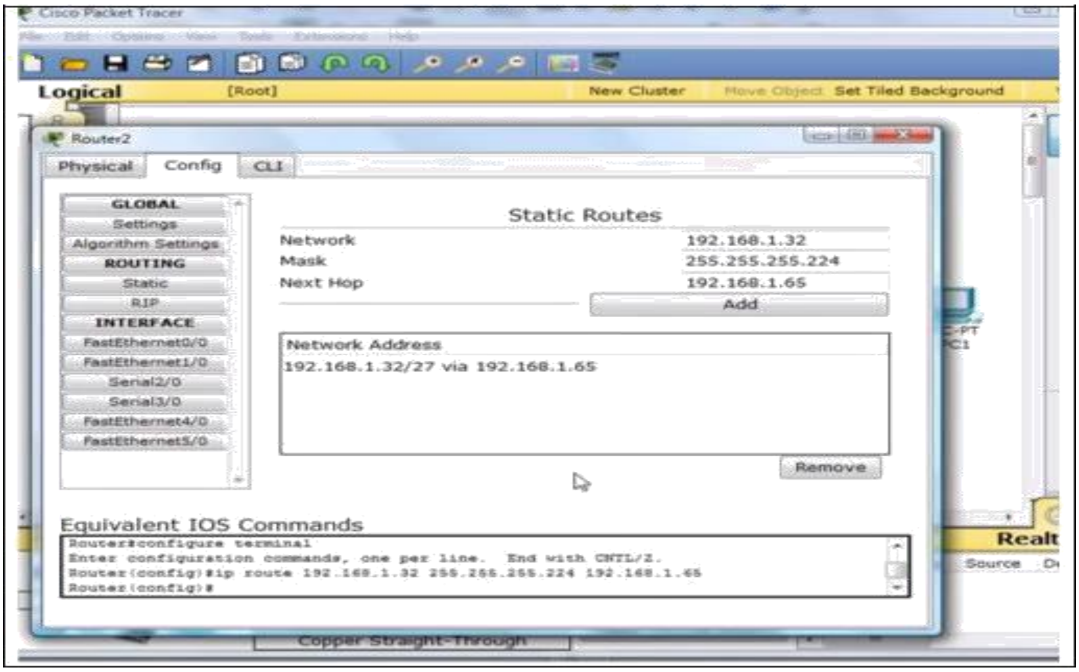
```
Packet Tracer PC Command Line 1.0
PC>ping 192.168.1.33

Pinging 192.168.1.33 with 32 bytes of data:

Reply from 192.168.1.33: bytes=32 time=31ms TTL=255
Reply from 192.168.1.33: bytes=32 time=9ms TTL=255
Reply from 192.168.1.33: bytes=32 time=11ms TTL=255
Reply from 192.168.1.33: bytes=32 time=9ms TTL=255

Ping statistics for 192.168.1.33:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 9ms, Maximum = 31ms, Average = 15ms

PC>ping 192.168.1.65

Pinging 192.168.1.65 with 32 bytes of data:

Reply from 192.168.1.65: bytes=32 time=15ms TTL=255
Reply from 192.168.1.65: bytes=32 time=13ms TTL=255
Reply from 192.168.1.65: bytes=32 time=7ms TTL=255
```

Now we ping 192.168.1.66 we got the Timed out because these interface are not directly connected to Router R1.



```
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 9ms, Maximum = 31ms, Average = 15ms

PC>ping 192.168.1.65

Pinging 192.168.1.65 with 32 bytes of data:

Reply from 192.168.1.65: bytes=32 time=15ms TTL=255
Reply from 192.168.1.65: bytes=32 time=13ms TTL=255
Reply from 192.168.1.65: bytes=32 time=7ms TTL=255
Reply from 192.168.1.65: bytes=32 time=10ms TTL=255

Ping statistics for 192.168.1.65:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 7ms, Maximum = 15ms, Average = 11ms

PC>ping 192.168.1.66

Pinging 192.168.1.66 with 32 bytes of data:

Request timed out.
Request timed out.
```

Therefore, we have to add static route in Router R2.



Static Routes

Network 192.168.1.32

Mask 255.255.255.224

Next Hop 192.168.1.65

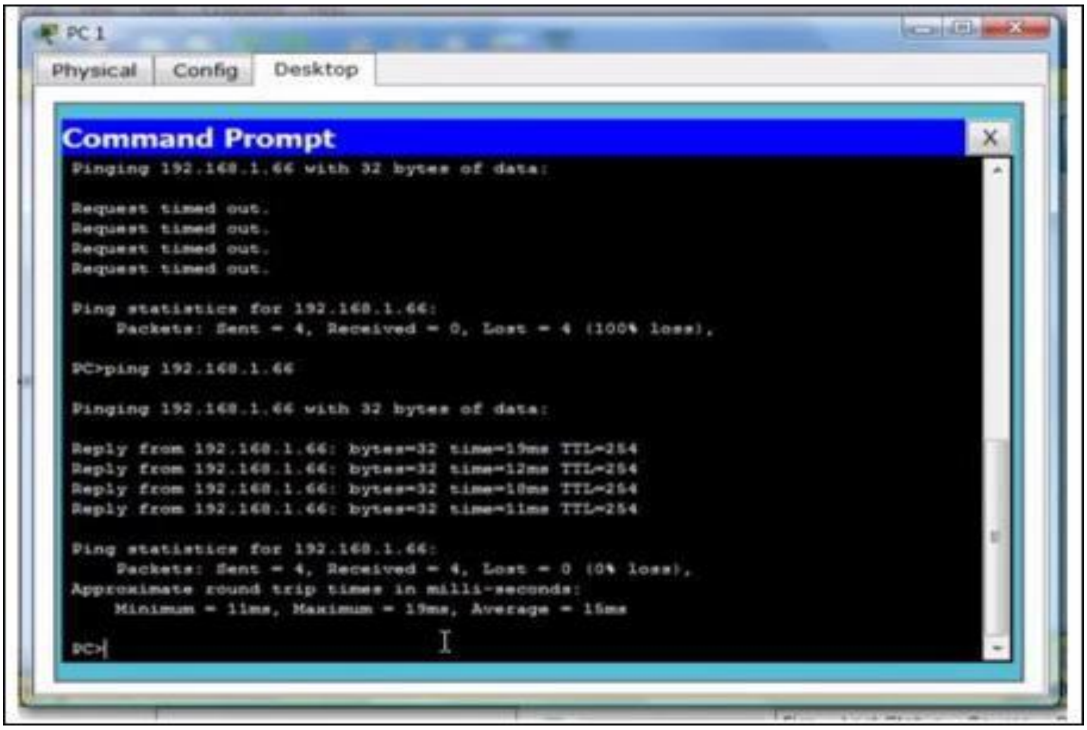
Network Address

192.168.1.32/27 via 192.168.1.65

Equivalent IOS Commands

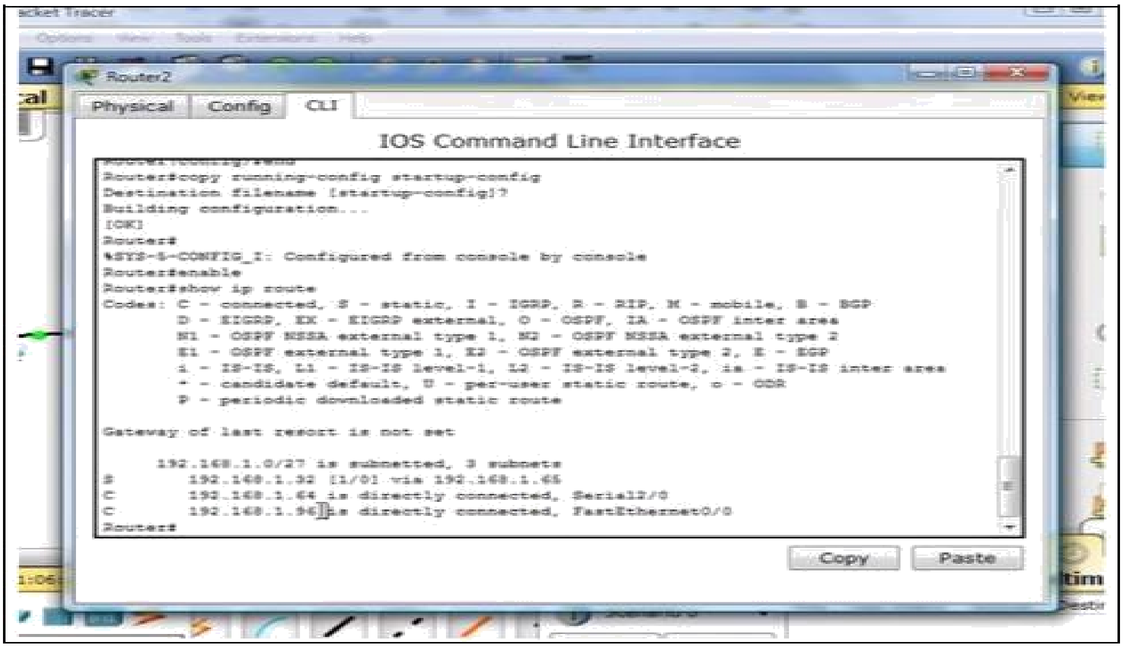
```
Router#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#ip route 192.168.1.32 255.255.255.224 192.168.1.65
Router(config)#
```

As you can see that we got the reply after adding the static route in Router R2.



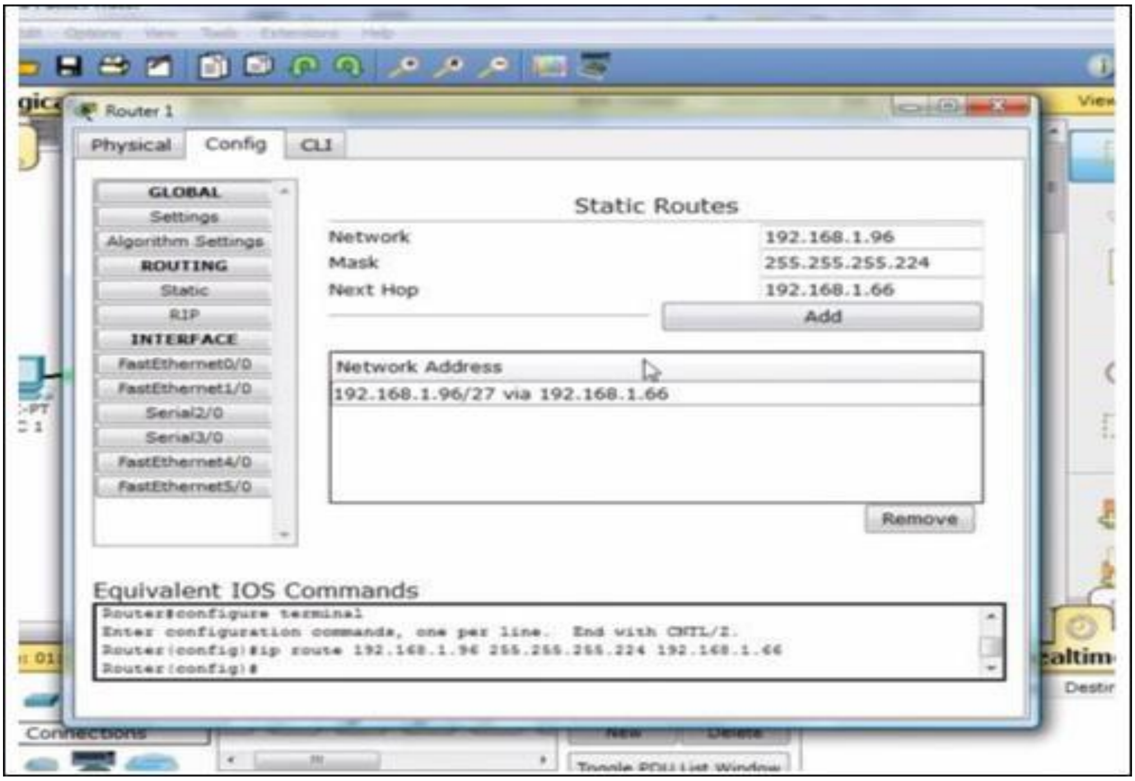
```
Pinging 192.168.1.66 with 32 bytes of data:

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

Ping statistics for 192.168.1.66:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),

PC>ping 192.168.1.66

Pinging 192.168.1.66 with 32 bytes of data:

Reply from 192.168.1.66: bytes=32 time=19ms TTL=254
Reply from 192.168.1.66: bytes=32 time=12ms TTL=254
Reply from 192.168.1.66: bytes=32 time=18ms TTL=254
Reply from 192.168.1.66: bytes=32 time=11ms TTL=254

Ping statistics for 192.168.1.66:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 11ms, Maximum = 19ms, Average = 15ms

PC>
```

Now using show ip route command we can see all the details of routing table saved in R2.



```
Router#copy running-config startup-config
Destination filename [startup-config]?
Building configuration...
[OK]
Router#
%SYS-5-CONFIG_I: Configured from console by console
Router#enable
Router#show ip route
Codes: C - connected, S - static, I - IGRP, 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

     192.168.1.0/27 is subnetted, 3 subnets
S       192.168.1.32 [1/0] via 192.168.1.65
C       192.168.1.64 is directly connected, Serial2/0
C       192.168.1.96 is directly connected, FastEthernet0/0
Router#
```

Now similarly add route in Router R1.



Static Routes

Network 192.168.1.96

Mask 255.255.255.224

Next Hop 192.168.1.66

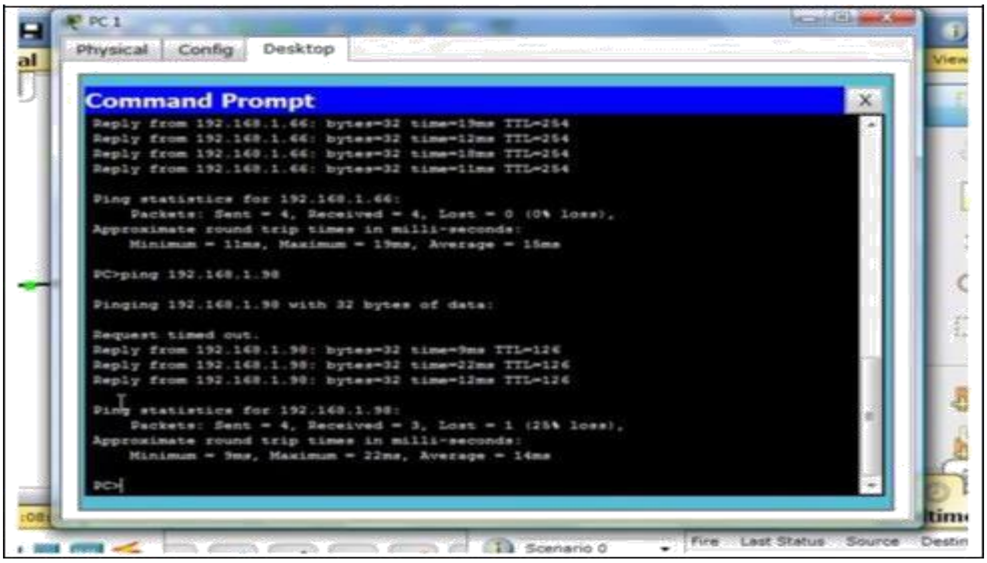
Network Address

192.168.1.96/27 via 192.168.1.66

Equivalent IOS Commands

```
Router#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#ip route 192.168.1.96 255.255.255.224 192.168.1.66
Router(config)#
```

Now we can ping Router R1 from PC



```
Reply from 192.168.1.66: bytes=32 time=19ms TTL=254
Reply from 192.168.1.66: bytes=32 time=12ms TTL=254
Reply from 192.168.1.66: bytes=32 time=10ms TTL=254
Reply from 192.168.1.66: bytes=32 time=11ms TTL=254

Ping statistics for 192.168.1.66:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 11ms, Maximum = 19ms, Average = 15ms

PC>ping 192.168.1.98

Pinging 192.168.1.98 with 32 bytes of data:

Request timed out.
Reply from 192.168.1.98: bytes=32 time=9ms TTL=126
Reply from 192.168.1.98: bytes=32 time=22ms TTL=126
Reply from 192.168.1.98: bytes=32 time=12ms TTL=126

Ping statistics for 192.168.1.98:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 9ms, Maximum = 22ms, Average = 14ms

PC>
```