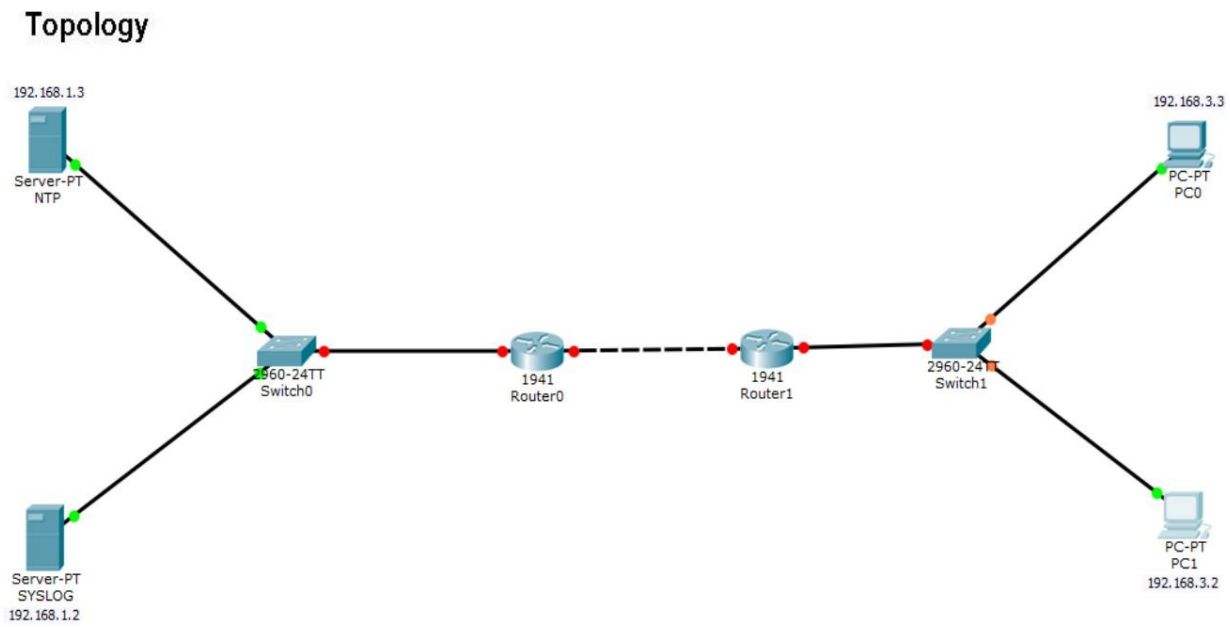
INDEX

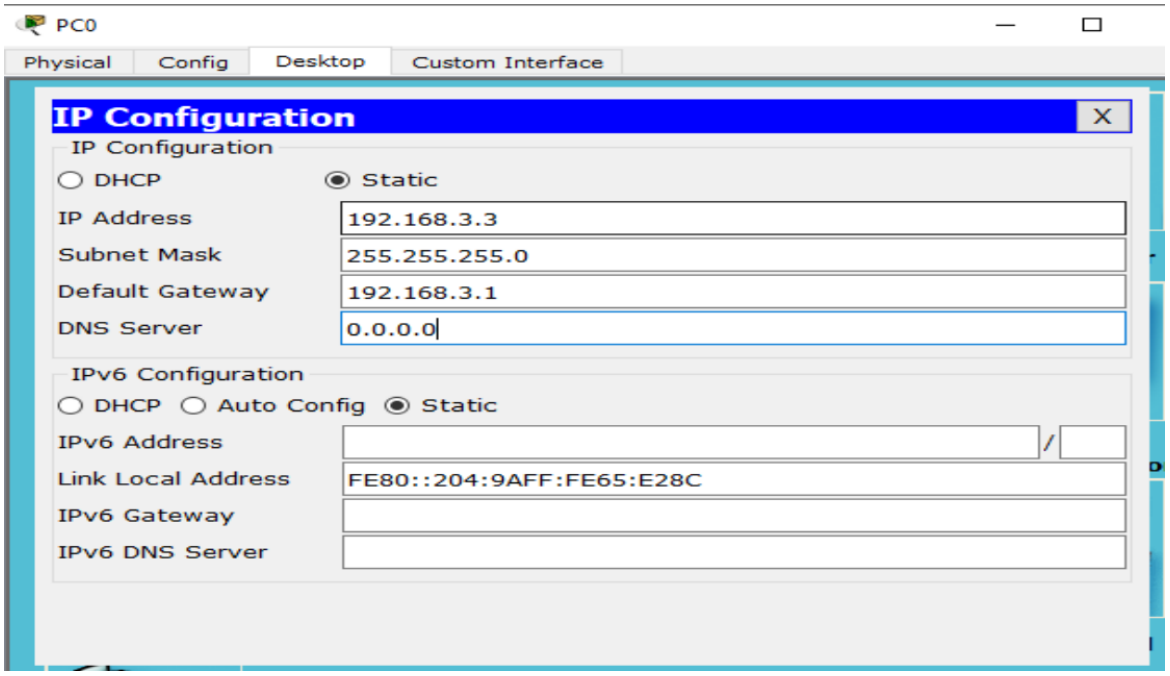
Practical	Title	Sign
1	Configure Cisco Routers for Syslog, NTP, and SSH Operation	
2	Configure AAA Authentication on Cisco Routers	
3	Configuring Extended ACLs : Configure, Apply and Verify an Extended Numbered ACL	
4	Configure IP ACLs to Mitigate Attacks	
5	Configure IPV6 ACLs	
6	Configuring a Zone-Based policy Firewall (ZPF)	
7	Configuring IOS Intrusion Prevention System (IPS) Using the CLI : a) Enable IOS IPS b) Modify an IPS signature	
8	Layer 2 Security a) Assign the Central switch as the root bridge. B) Secure spanning-tree parameters to prevent STP manipulation attacks. C) Enable port security and disable unused ports.	

Practical 1

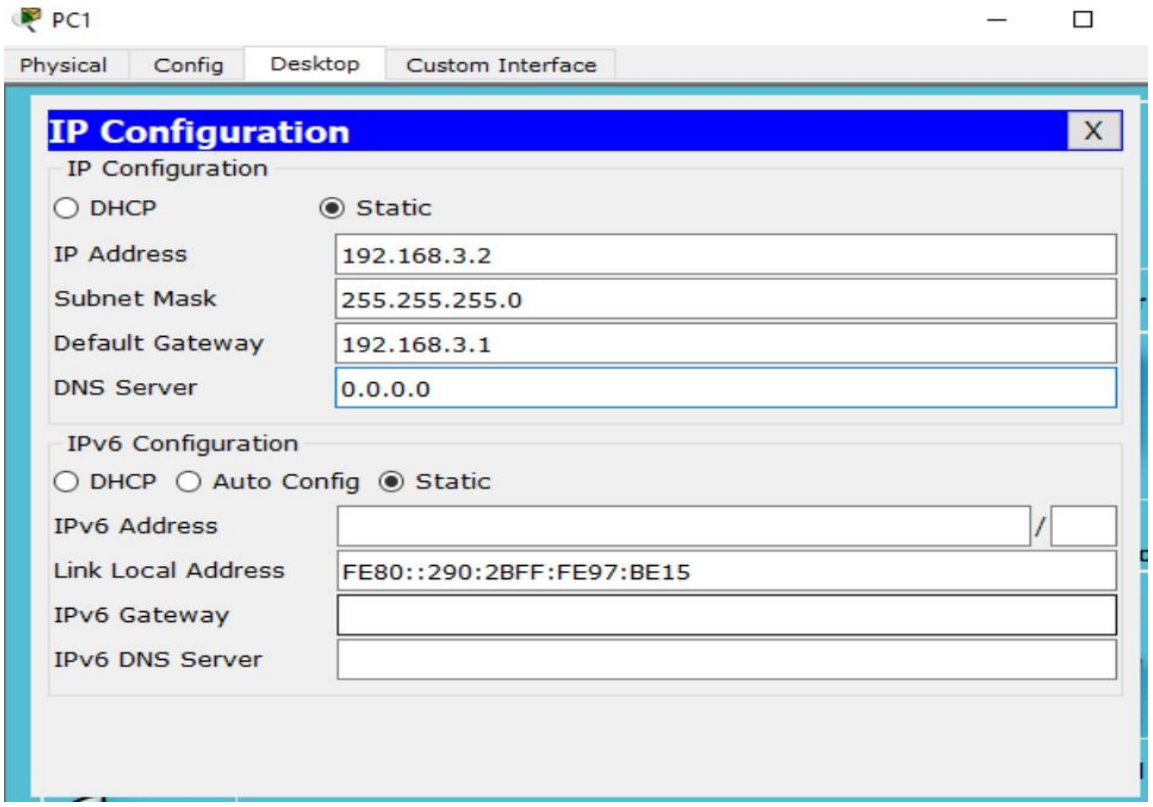
Configure Cisco Routers for Syslog, NTP, and SSH Operation



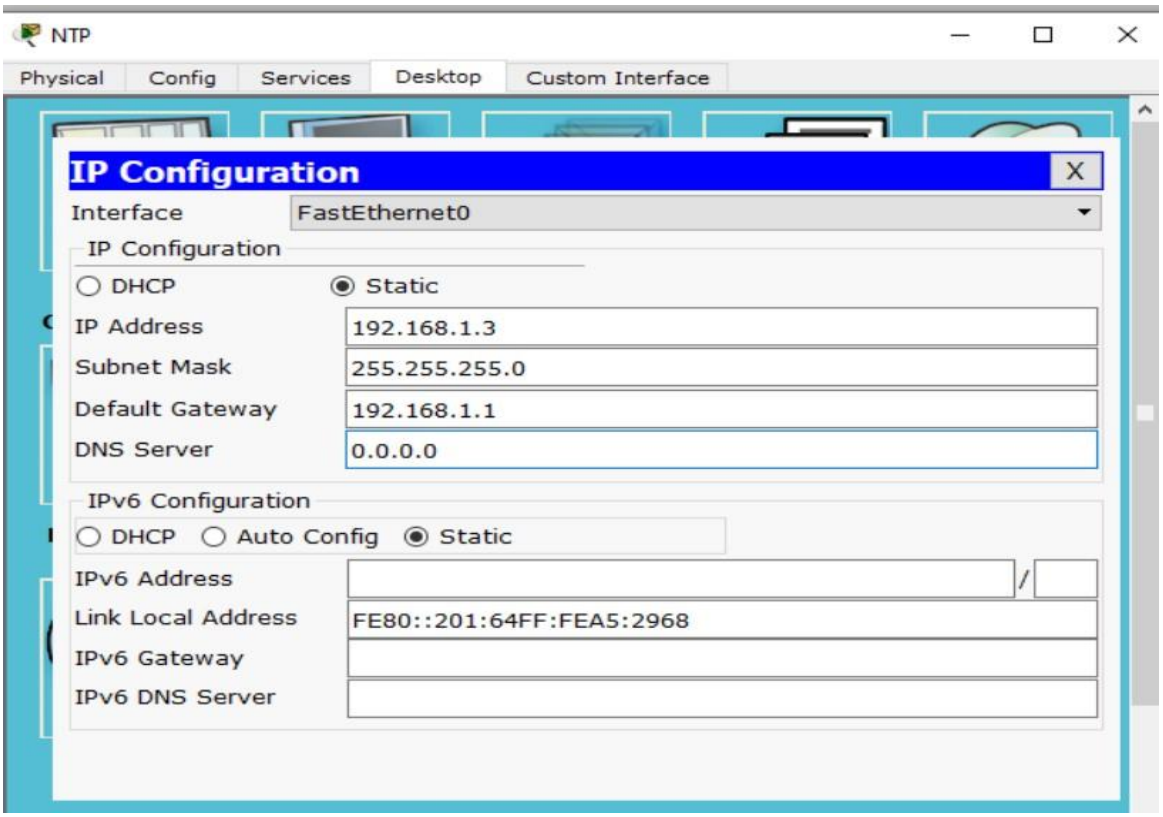
Configure PC0



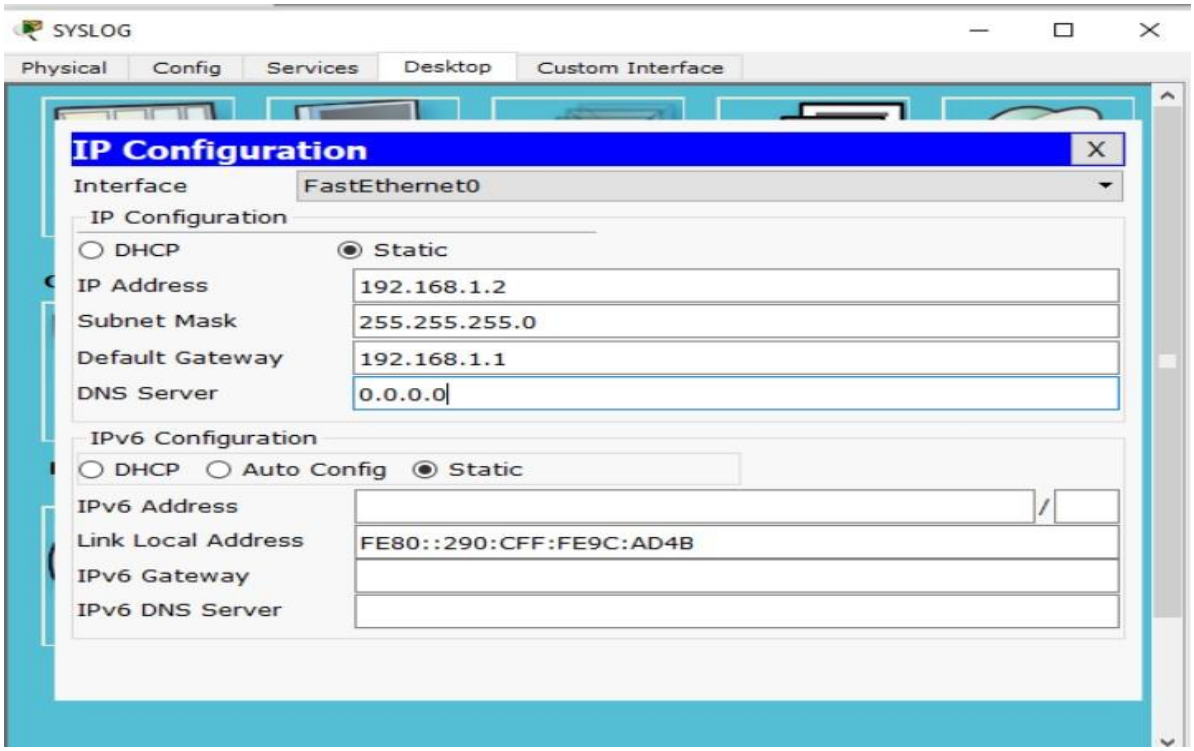
Configure PC1



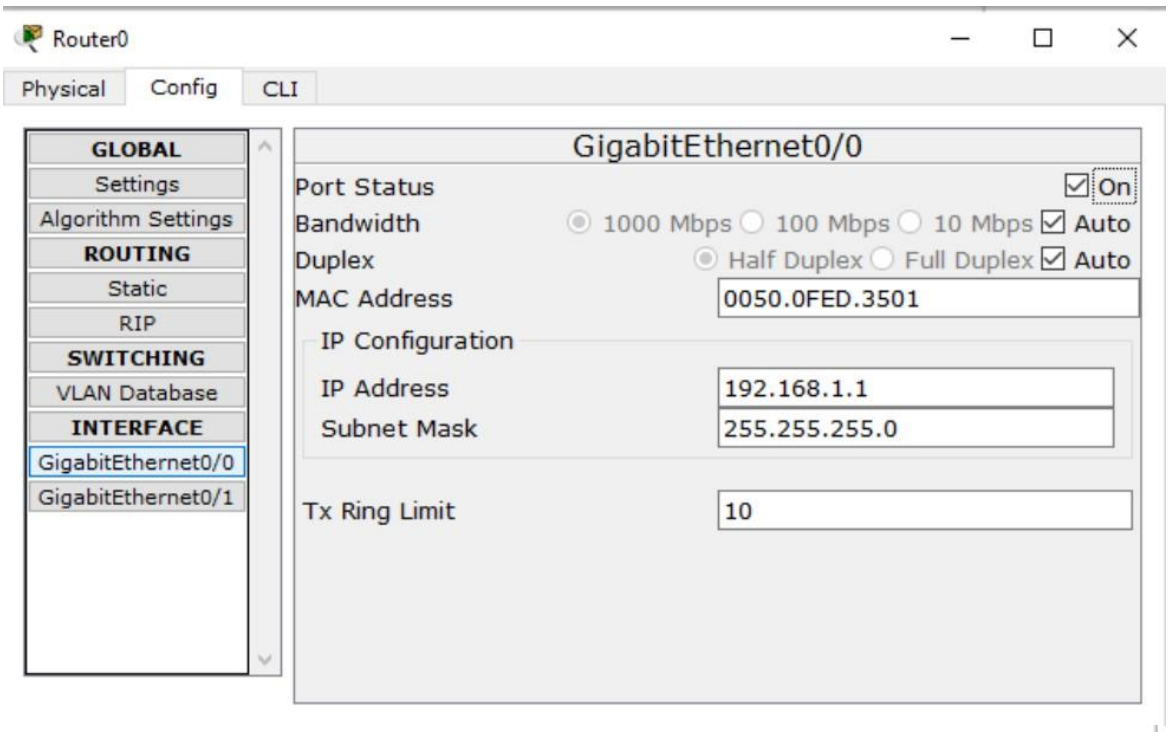
Configure NTP Server

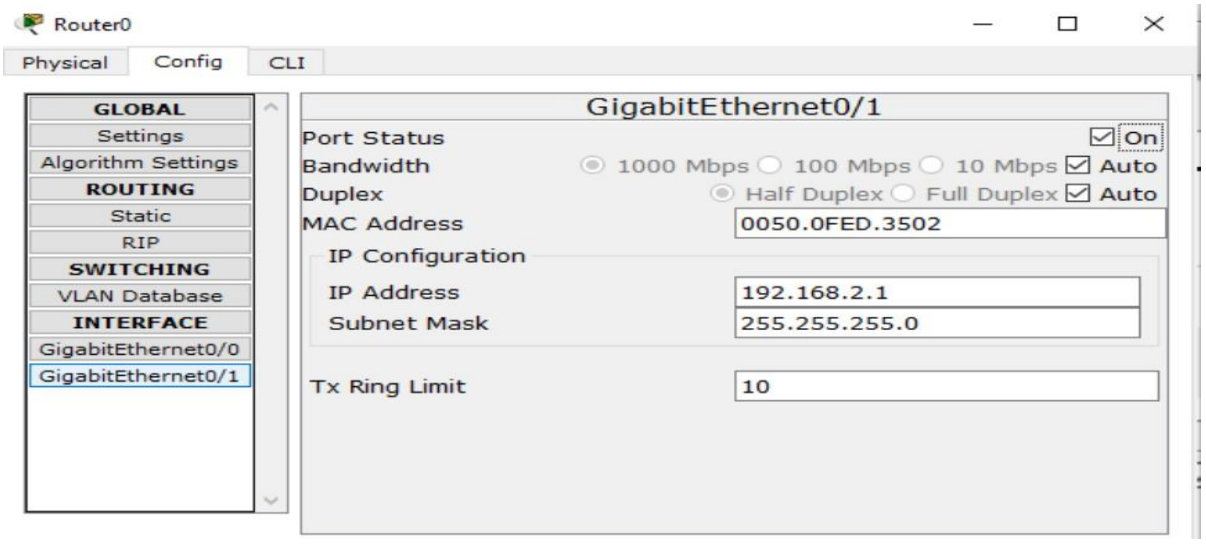


Configure SYSLOG Server

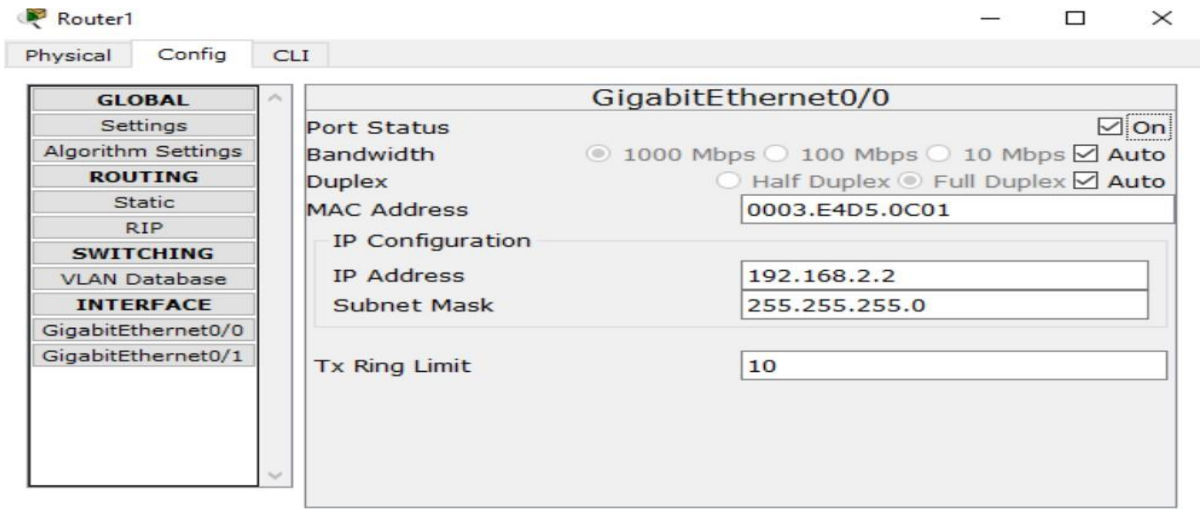


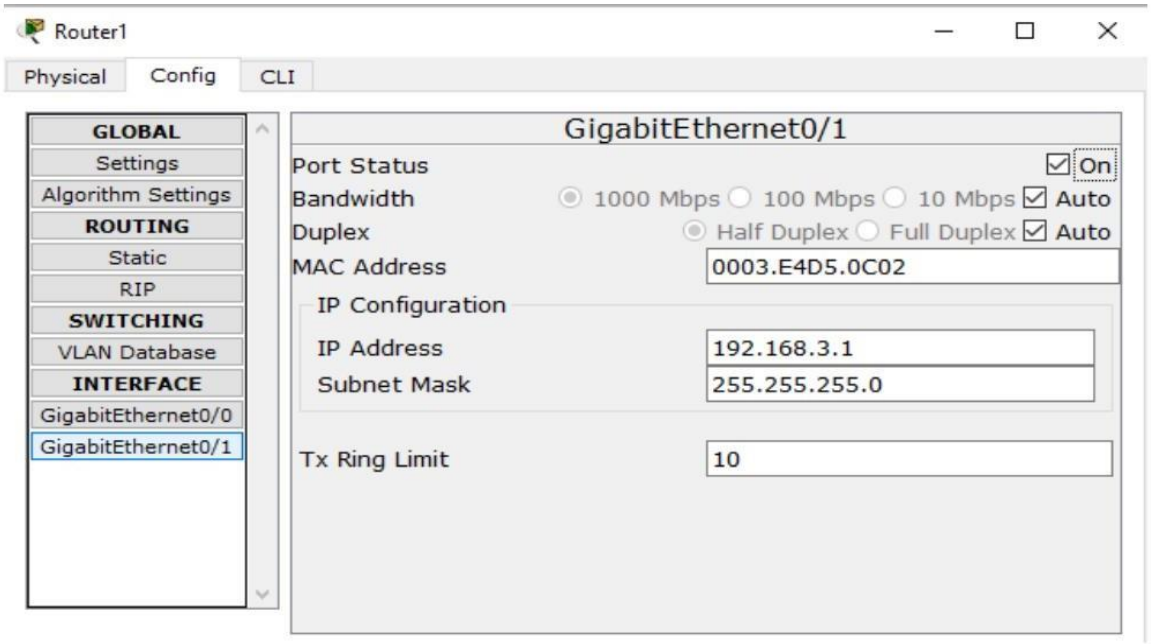
Configure Router 0





Configure Router 1





Part 1: Configure OSPF MD5 Authentication

ROUTER 0: Type the following command in the CLI mode

Router>en

Router#conf t

Router(config)#router ospf 1

Router(config-router)#network 192.168.1.0 0.255.255.255 area 1

Router(config-router)#network 192.168.2.0 0.255.255.255 area 1

Router(config-router)#exit

Router(config)#exit

Router#

Part 1: Configure OSPF MD5 Authentication

ROUTER 0: Type the following command in the CLI mode

Router>en

Router#conf t

Router(config)#router ospf 1

Router(config-router)#network 192.168.1.0 0.255.255.255 area 1

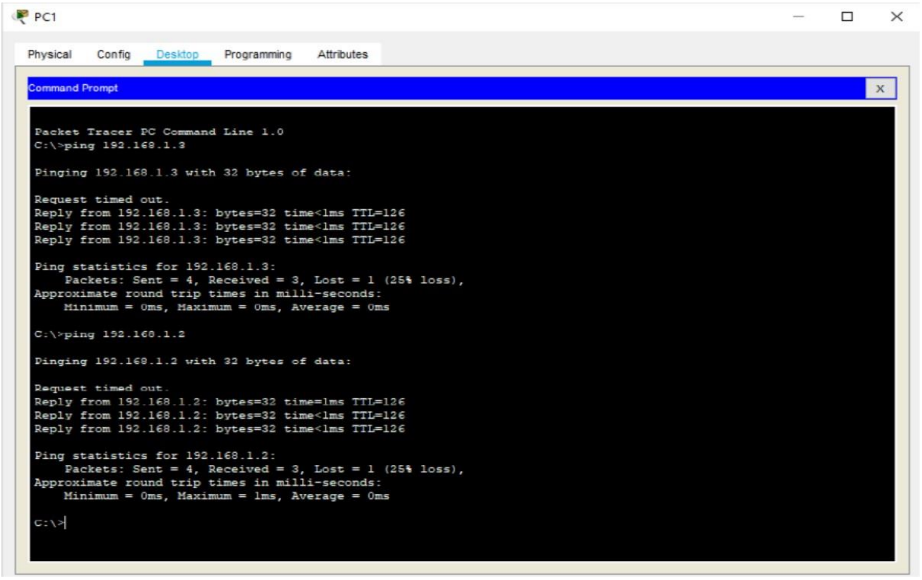
Router(config-router)#network 192.168.2.0 0.255.255.255 area 1

Router(config-router)#exit

Router(config)#exit

Router#

Now we verify the connectivity by using the following



Hence OSPF has been verified

MD5 Authentication

ROUTER0: Type the following command in the CLI mode

Router>enable

Router# conf t

Router(config)#int g0/0

Router(config-if)#ip ospf authentication message-digest

Router(config-if)#ip ospf message-digest-key 1 md5 dalmia

Router(config-if)#exit

Router(config)#int g0/1

Router(config-if)#ip ospf authentication message-digest

Router(config-if)#ip ospf message-digest-key 1 md5 dalmia

Router(config)#exit

ROUTER1: Type the following command in the CLI mode

Router>enable

Router# conf t

Router(config)#int g0/0

Router(config-if)#ip ospf authentication message-digest

Router(config-if)#ip ospf message-digest-key 1 md5 dalmia

Router(config-if)#exit

Router(config)#int g0/1

Router(config-if)#ip ospf authentication message-digest

Router(config-if)#ip ospf message-digest-key 1 md5 dalmia

Router(config)#exit

Verify the MD5 Authentication using the following command in the CLI mode of Router0

We get the following output:

GigabitEthernet0/1 is up, line protocol is up

Internet address is 192.168.2.1/24, Area 1

Process ID 1, Router ID 192.168.2.1, Network Type BROADCAST, Cost: 1

Transmit Delay is 1 sec, State BDR, Priority 1

Designated Router (ID) 192.168.3.1, Interface address 192.168.2.2

Backup Designated Router (ID) 192.168.2.1, Interface address 192.168.2.1

Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5

Hello due in 00:00:06

Index 2/2, flood queue length 0

Next 0x0(0)/0x0(0)

Last flood scan length is 1, maximum is 1

Last flood scan time is 0 msec, maximum is 0 msec

Neighbor Count is 1, Adjacent neighbor count is 1

Adjacent with neighbor 192.168.3.1 (Designated Router)

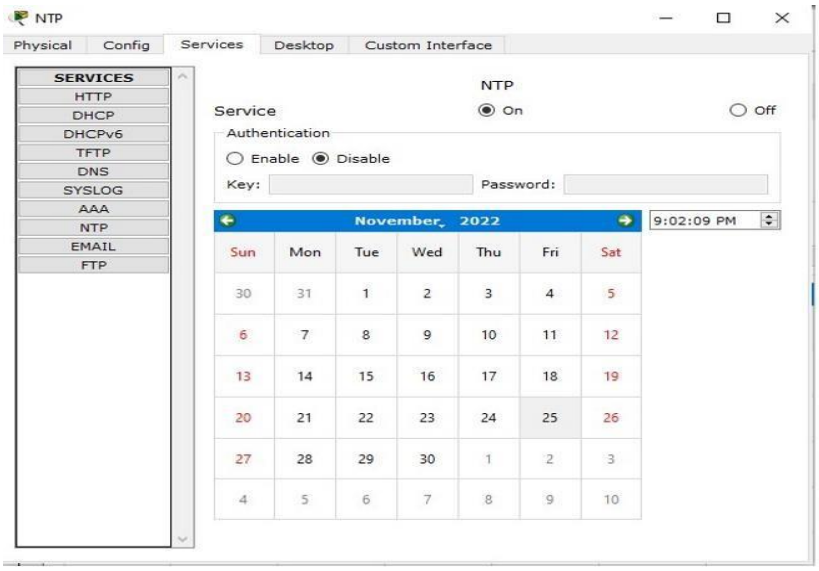
Suppress hello for 0 neighbor(s)

Message digest authentication enabled

Youngest key id is 1

MD5 Authentication has been verified

Part 2: Configure NTP Server and enable the NTP service



We must disable the NTP service on other servers' else output won't be obtained

Now Go to CLI Mode of both the routers and type the following commands:-

```
Router#config
Router#configure t
Router#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#ntp server 192.168.1.3
Router(config)#ntp up
Router(config)#ntp update-calendar
Router(config)#exit
Router#
```

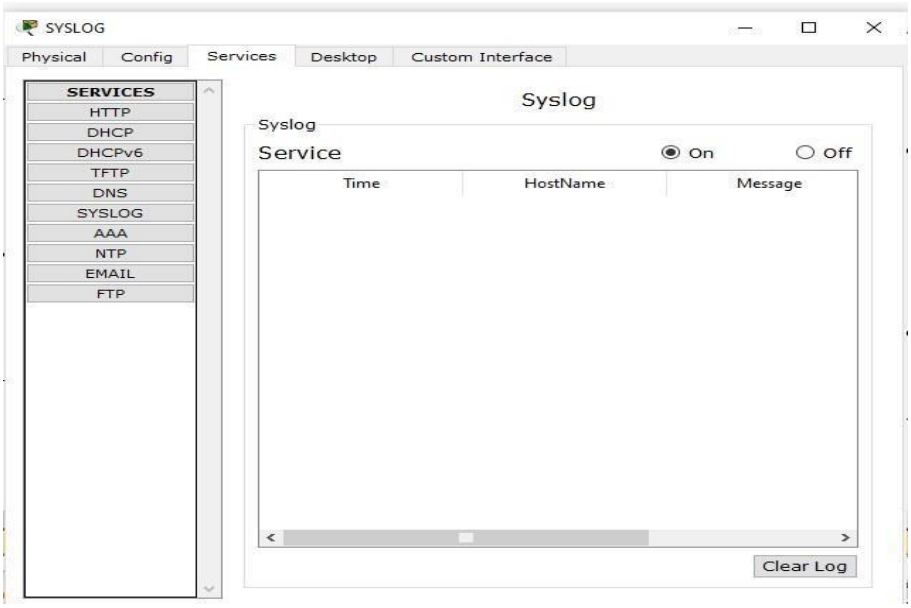
To verify the Output, we use the following command

Router#show clock

*21:7:3.987 UTC Fri Nov 25 2022

Router#

Part 3: Configure SYSLOG Server and enable the service
Turn ON the SYSLOG service on the server

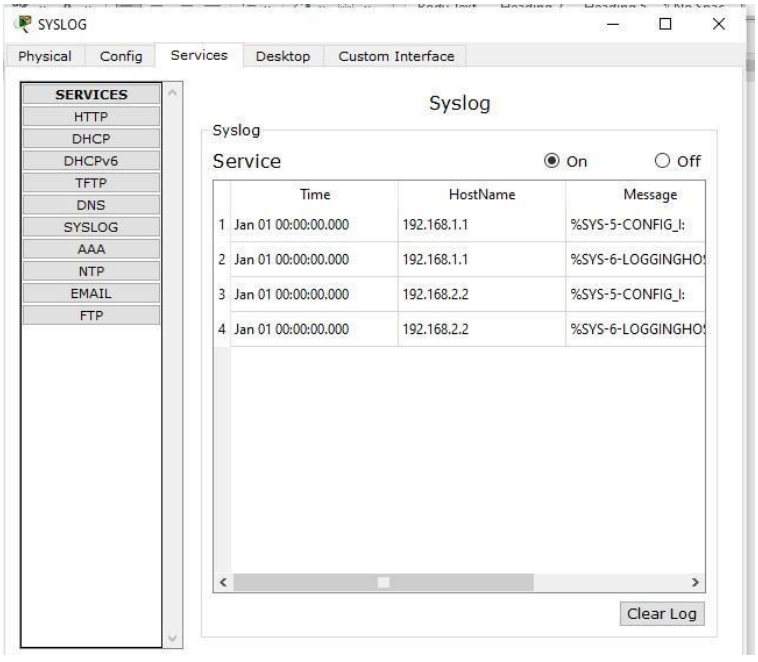


And Turn OFF on all other Servers

Now Go to CLI Mode of both the Routers and type the following commands: -

```
Router#
Router#configure terminal
Router(config)#logging 192.168.1.2
Router(config)#exit
Router#
```

Output:



Part 4: Configure SSH on Router1
Go to CLI Mode of Router1 and type the following commands: -

Router#conf t

Router(config)#ip domain-name dalmia.com

Router(config)#hostname R1

R1(config)#

R1(config)#crypto key generate rsa

The name for the keys will be: R0.dalmia.com

Choose the size of the key modulus in the range of 360 to 2048 for your
General Purpose Keys. Choosing a key modulus greater than 512 may take a
few minutes.
How many bits in the modulus [512]: 1024

% Generating 1024 bit RSA keys, keys will be non-exportable...[OK]

R1(config)#line vty 0 4

*Nov 25 21:19:48.169: %SSH-5-ENABLED: SSH 1.99 has been enabled

R1(config-line)#transport input ssh

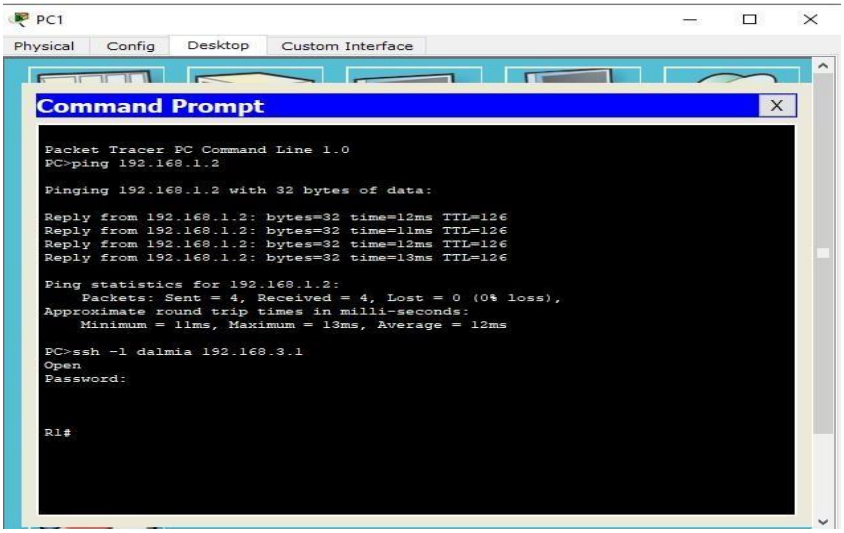
R1(config-line)#login local

R1(config-line)#exit

R1(config)#username dalmia privilege 15 password cisco

R1(config)#

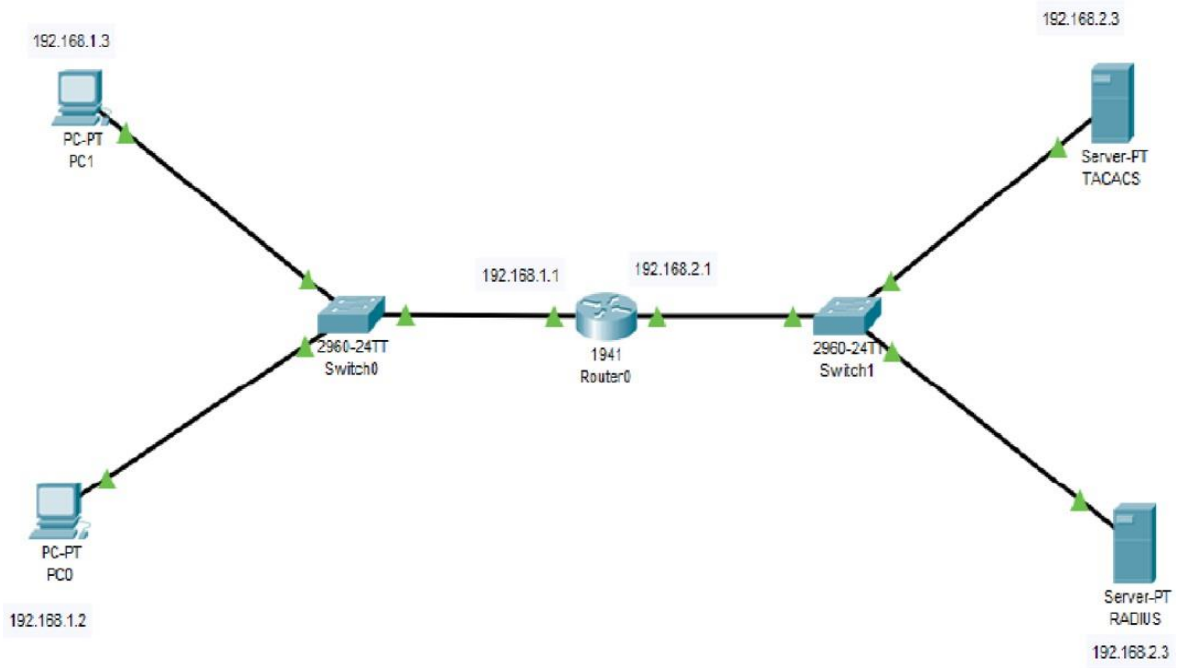
Output: Go to cmd of PC1 and type the command
ssh –l dalmia 192.168.3.1 and type the password cisco



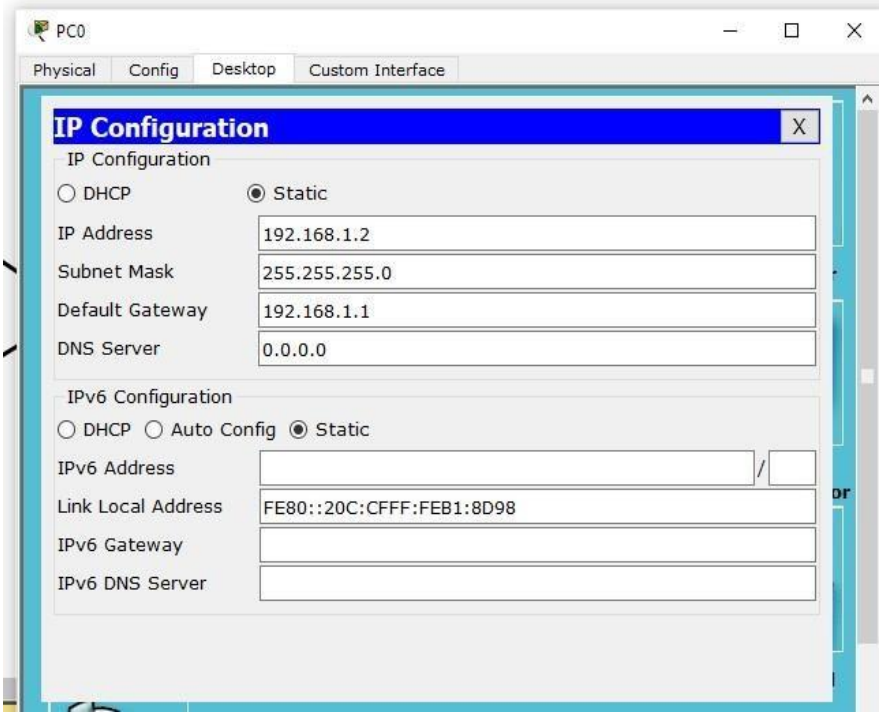
Hence SSH is also verified

Practical 2
Configure AAA Authentication on Cisco Routers

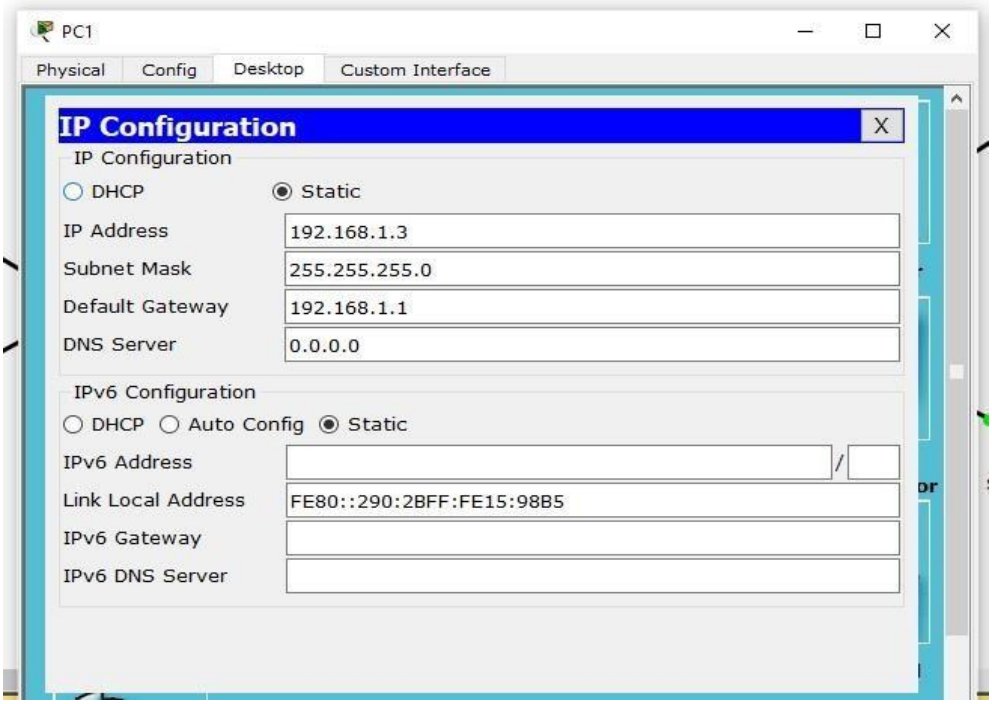
Topology



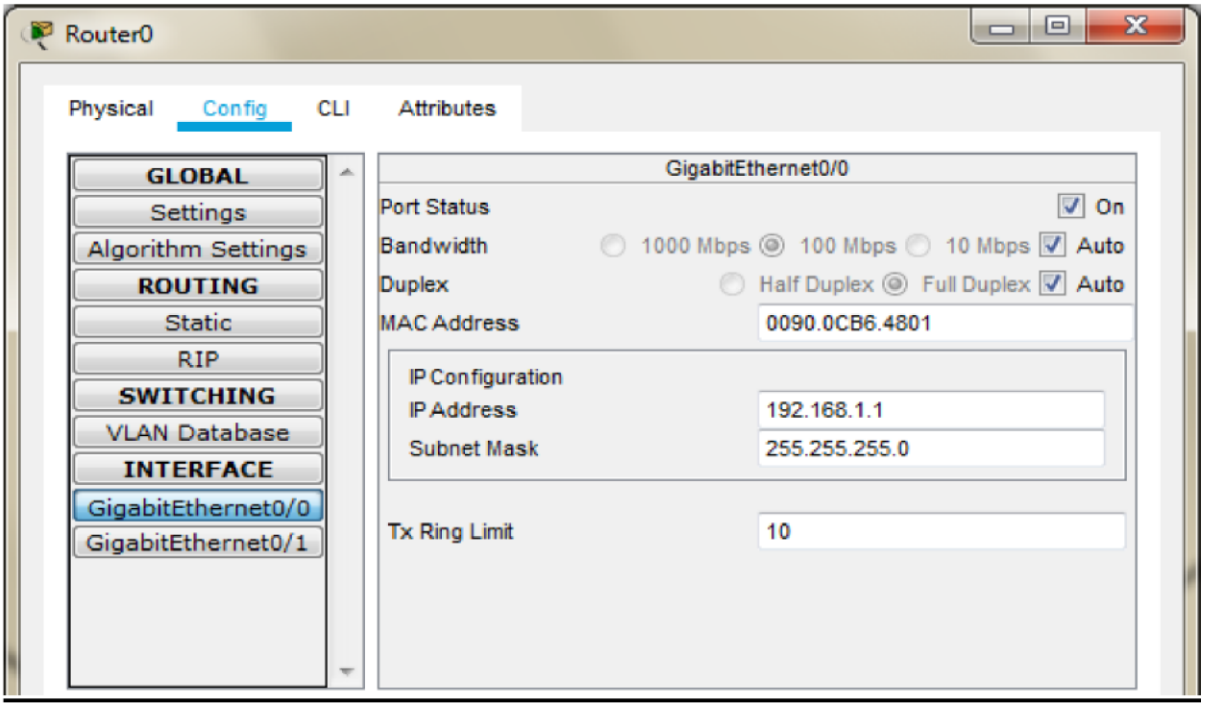
Configuring PC0

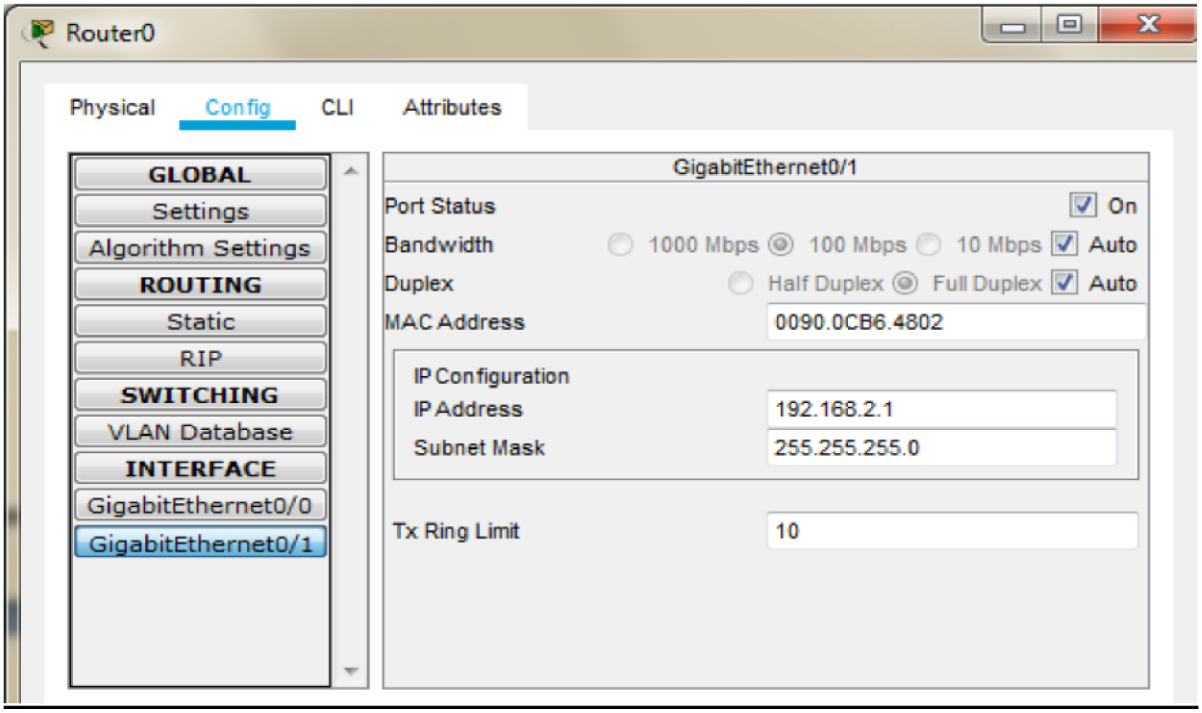


Configuring PC1



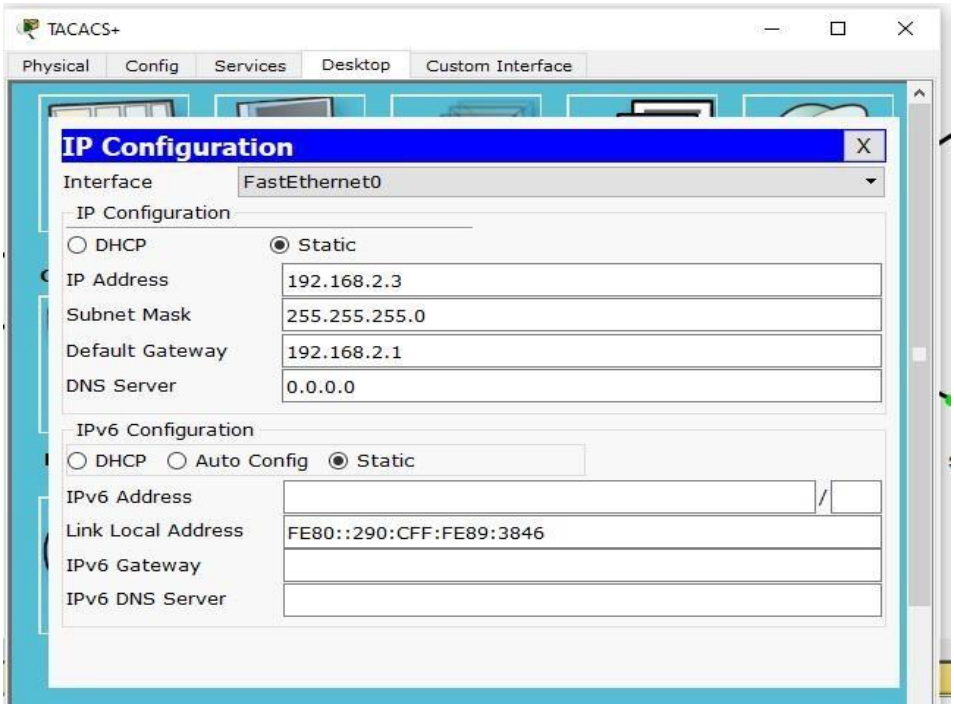
Router0

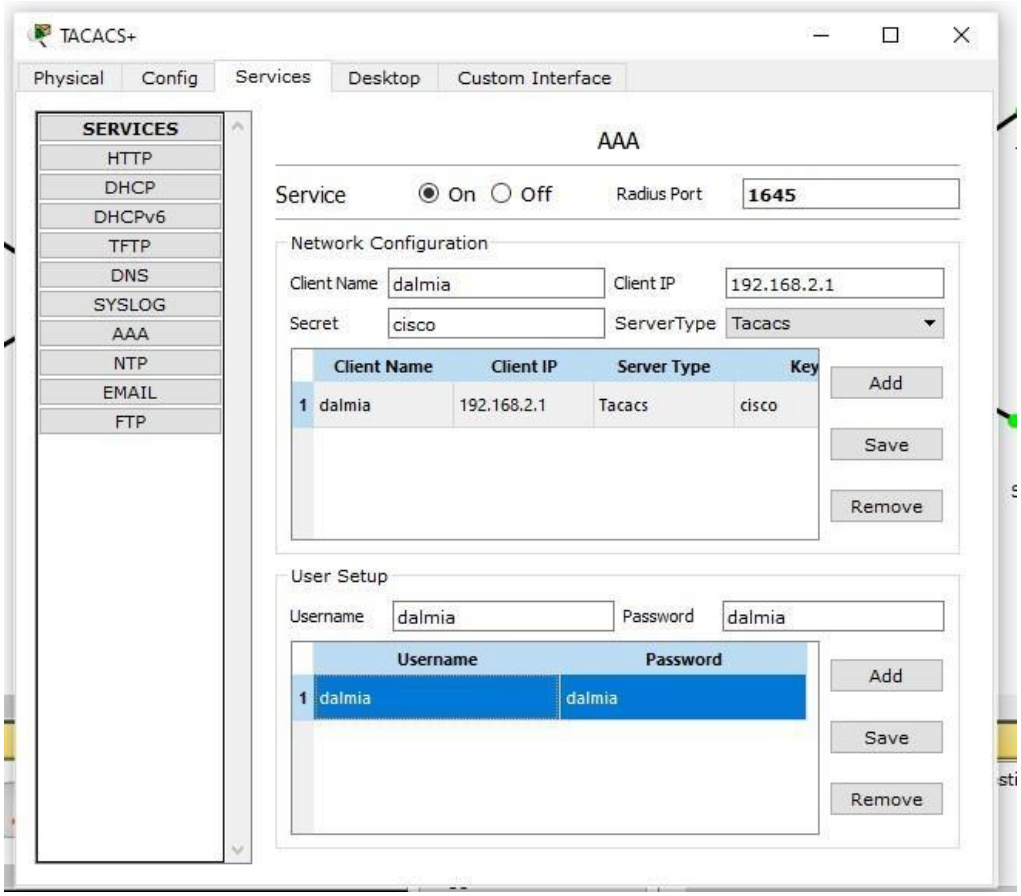




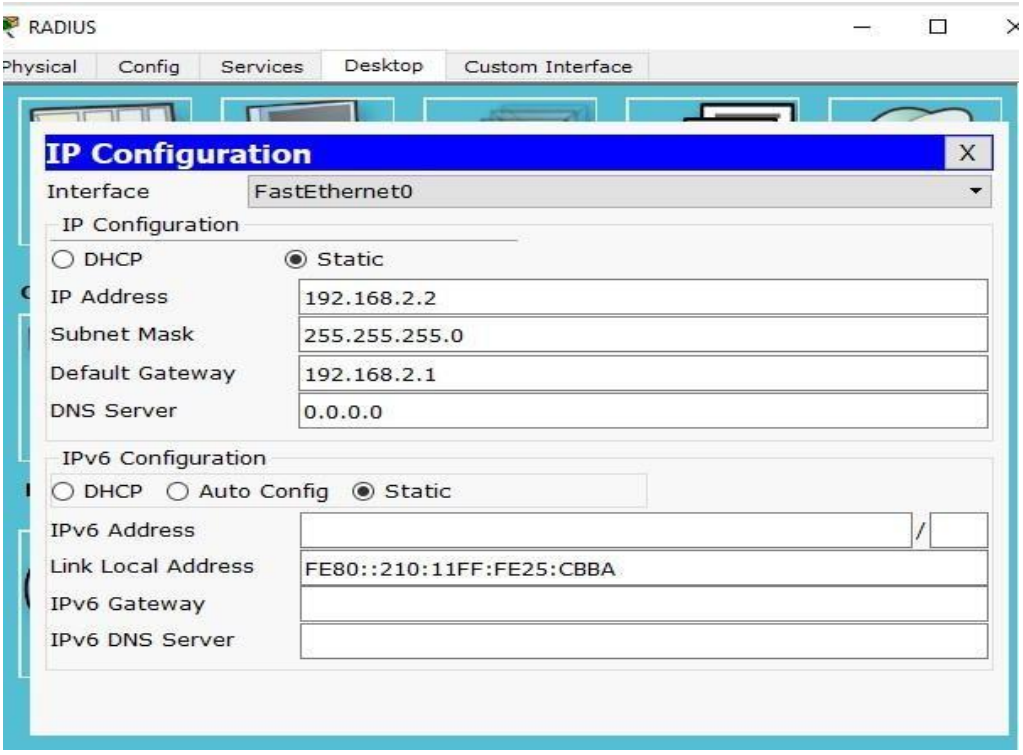
Server0(As TACACS)

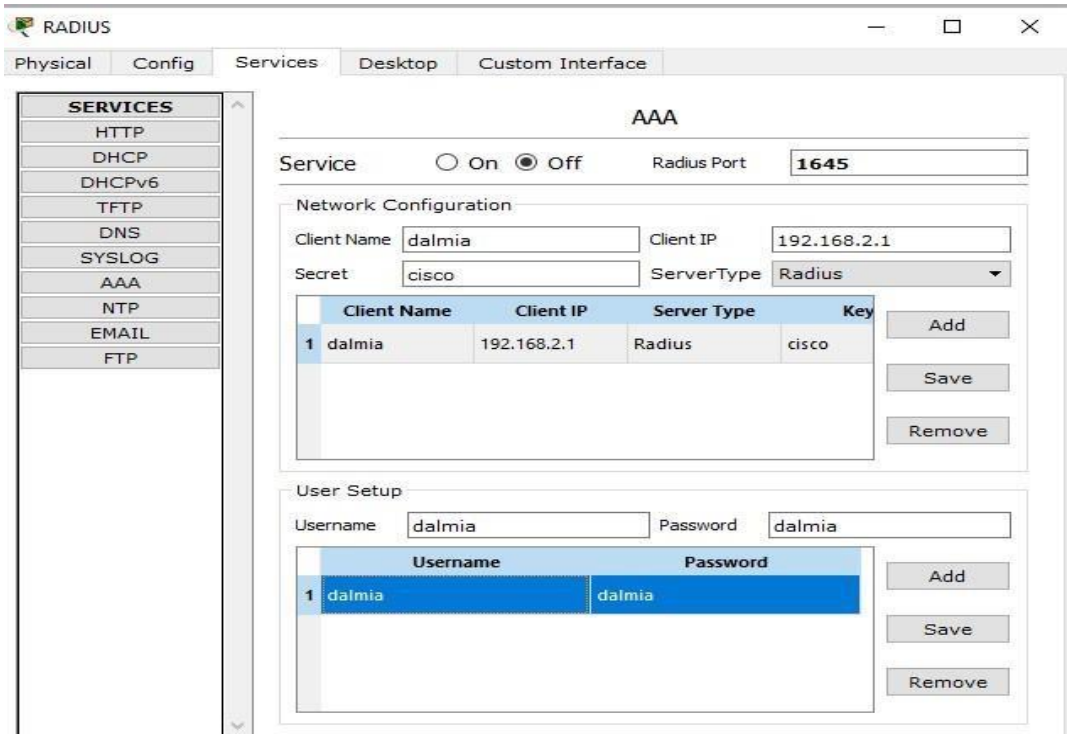
While configuring the TACACS/RADIUS server the Client IP address must be the Router IP.





Server1(As RADIUS)





Click on Router 0 go to CLI tab and press enter and enter the following commands:-

Router>en

Router#conf t

Router(config)#aaa new-model

Router(config)#tacacs-server host 192.168.2.3 key cisco Router(config)#radius-server host 192.168.2.2 key cisco

Router(config)#aaa authentication login dalmia group tacacs+ group radius local

Router(config)#line vty 0 4

Router(config-line)#login authentication dalmia

Router(config-line)#exit

Router(config)#

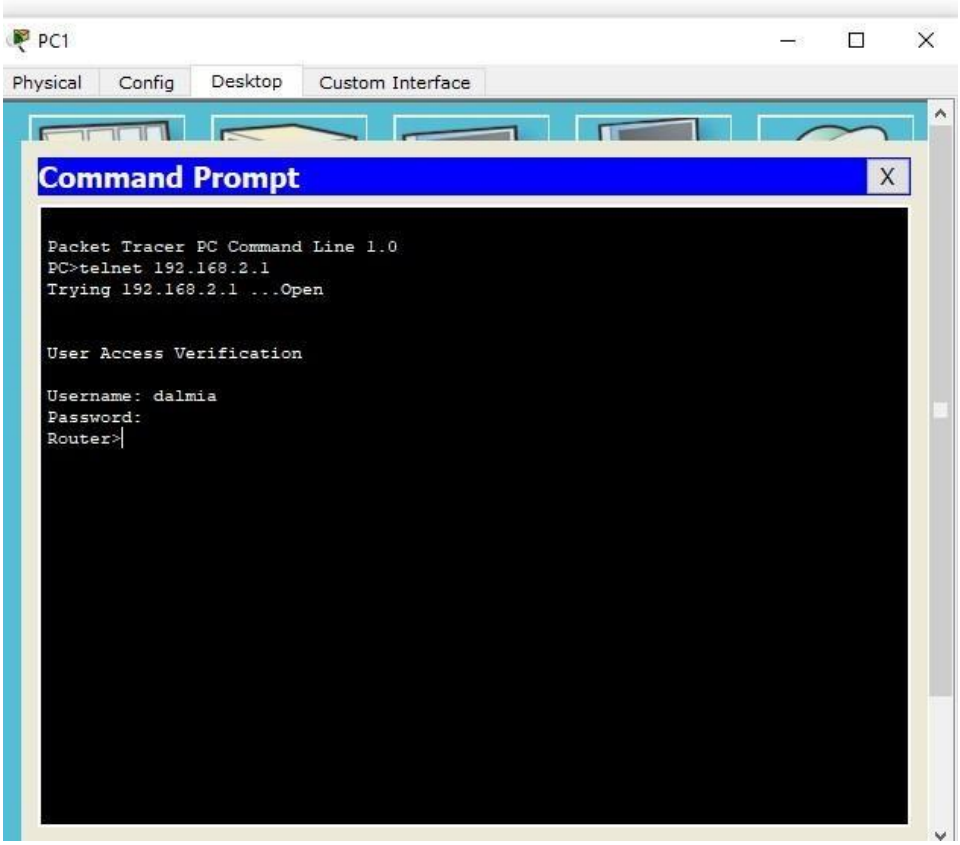
The Authentication can be done by typing the command **telnet 192.168.2.1** (the Router IP) in any of the PCs

We get a prompt to type the username and password, the username and password set in TACACS are entered

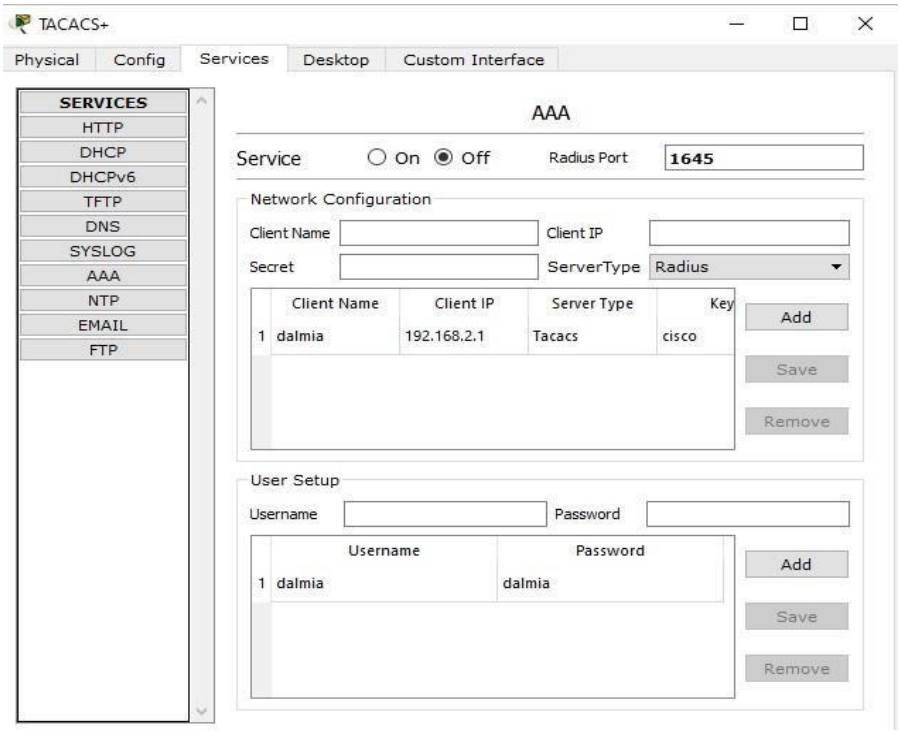
Username: dalmia

Password: dalmia

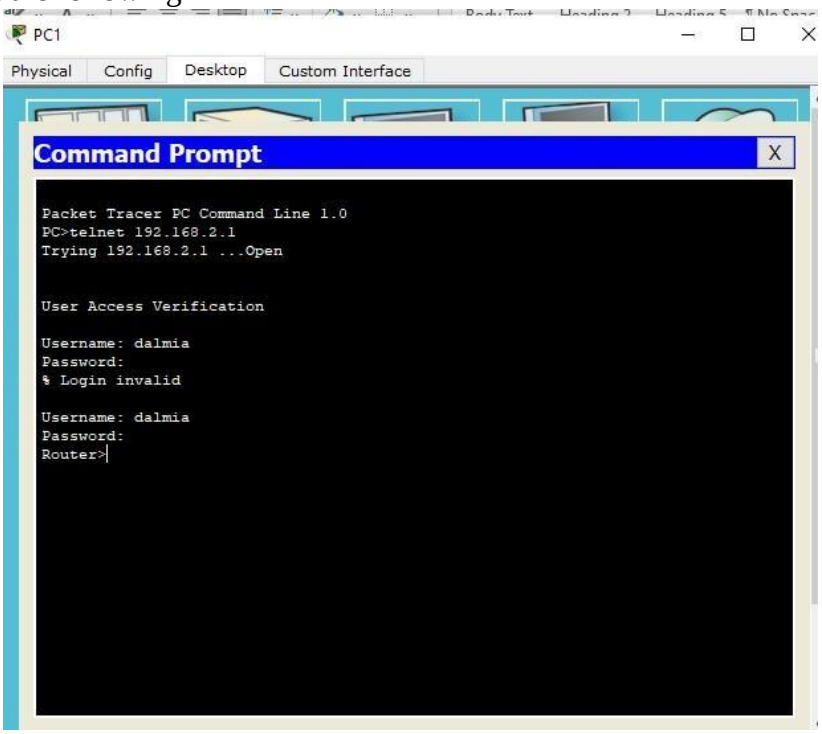
We get the following



In order to authenticate the RADIUS server, we need to turn OFF the TACACS service



We again enter the command **telnet 192.168.2.1** (the Router IP) and enter the username and password of the RADIUS server (Username: dalmia , Password: dalmia) We get the following

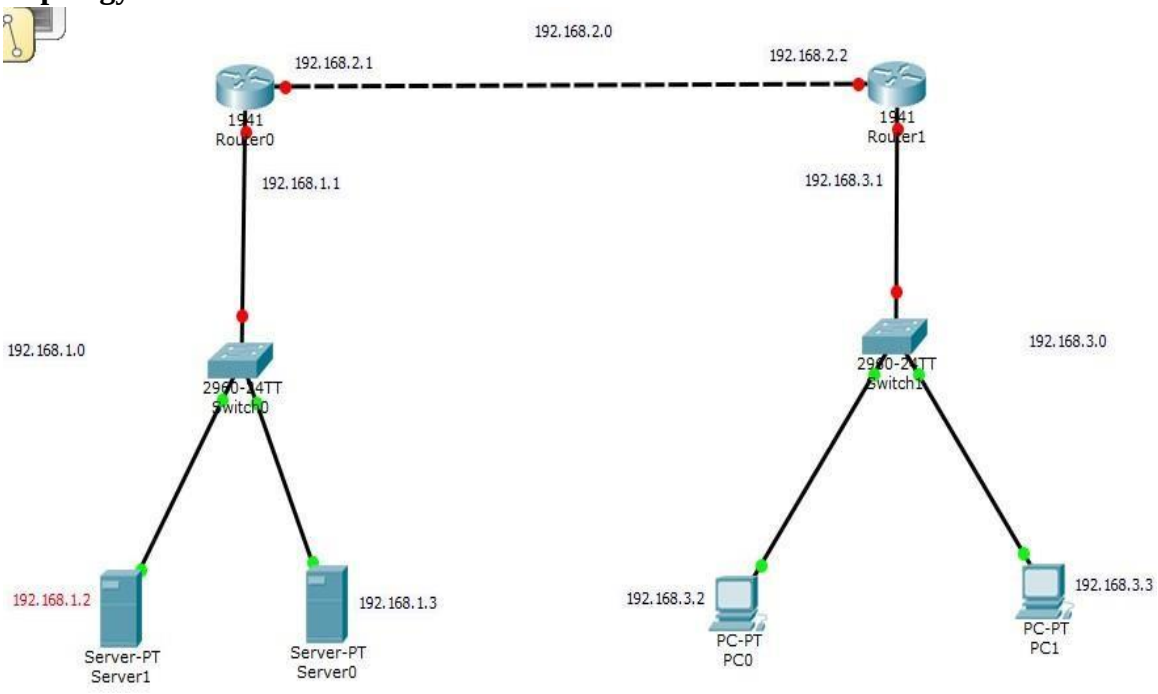


The local login can also be verified by turning OFF both TACACS and RADIUS service. Hence the authentication through both TACACS and RADIUS.

Practical 3

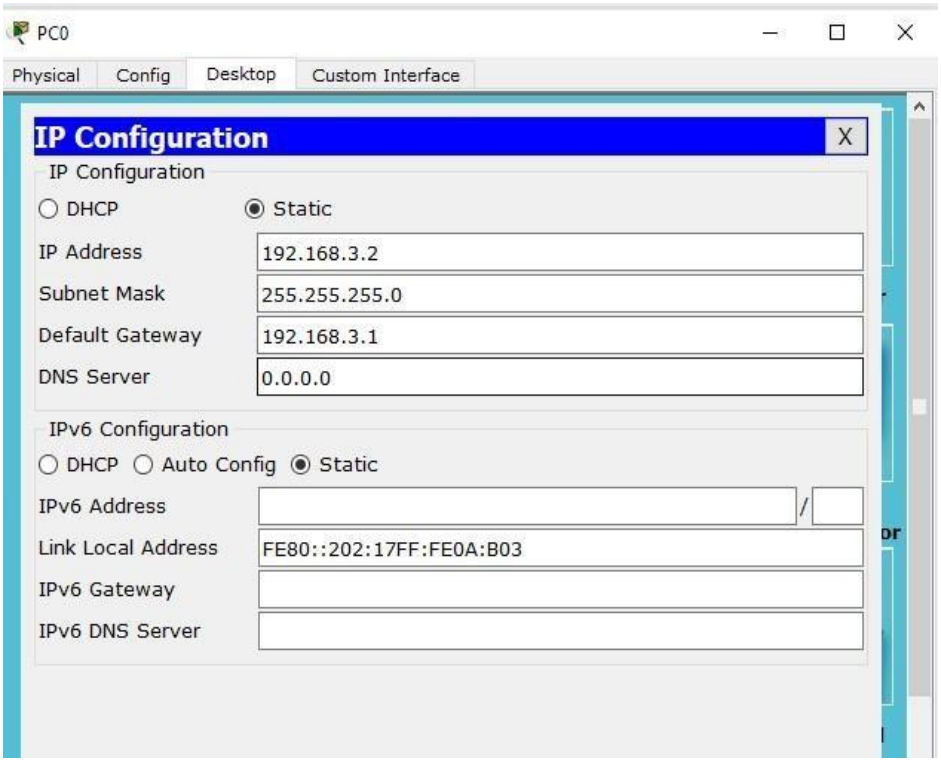
Configuring Extended ACLs : Configure, Apply and Verify an Extended Numbered ACL

Topology

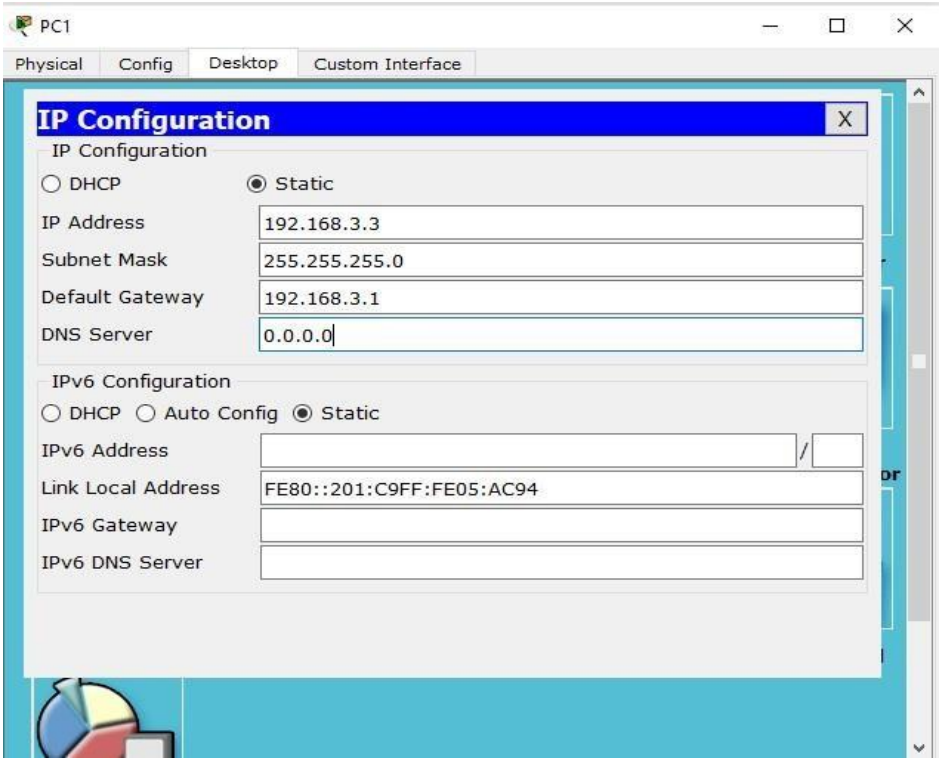


Topology Configuration

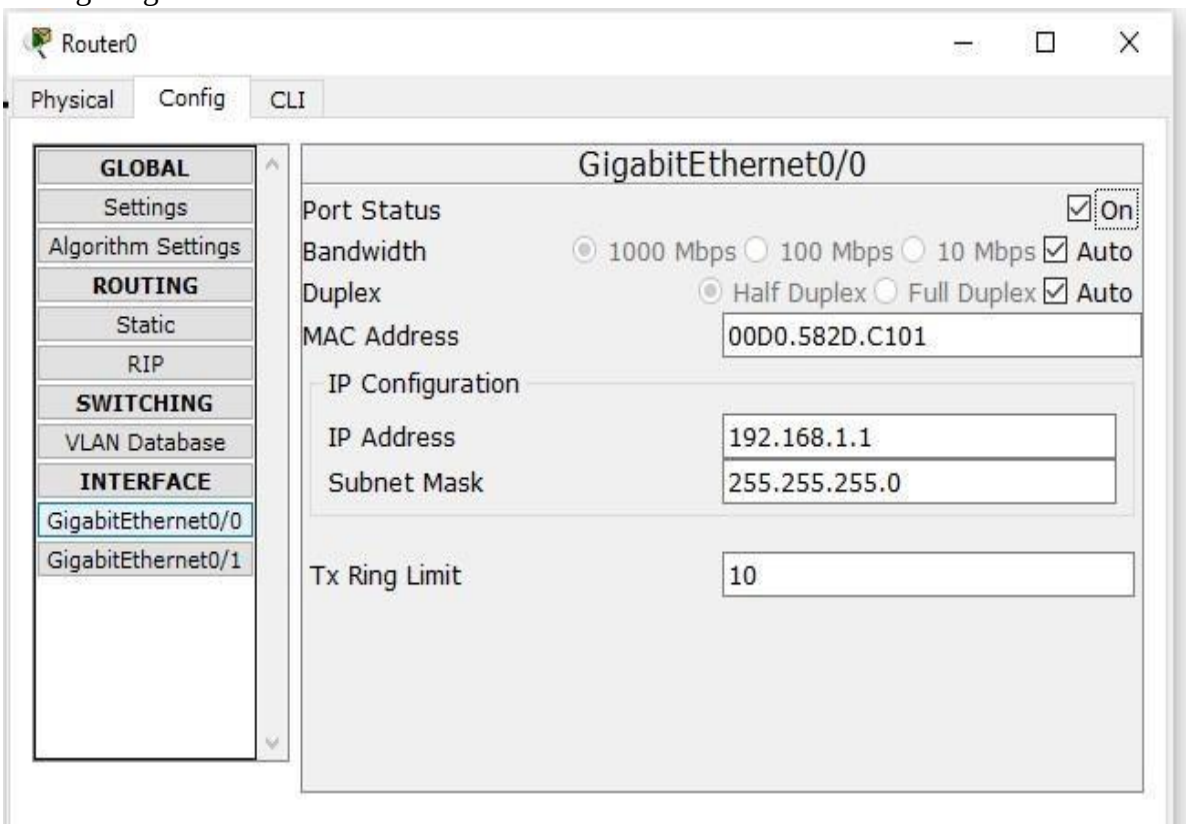
Configuring PC0

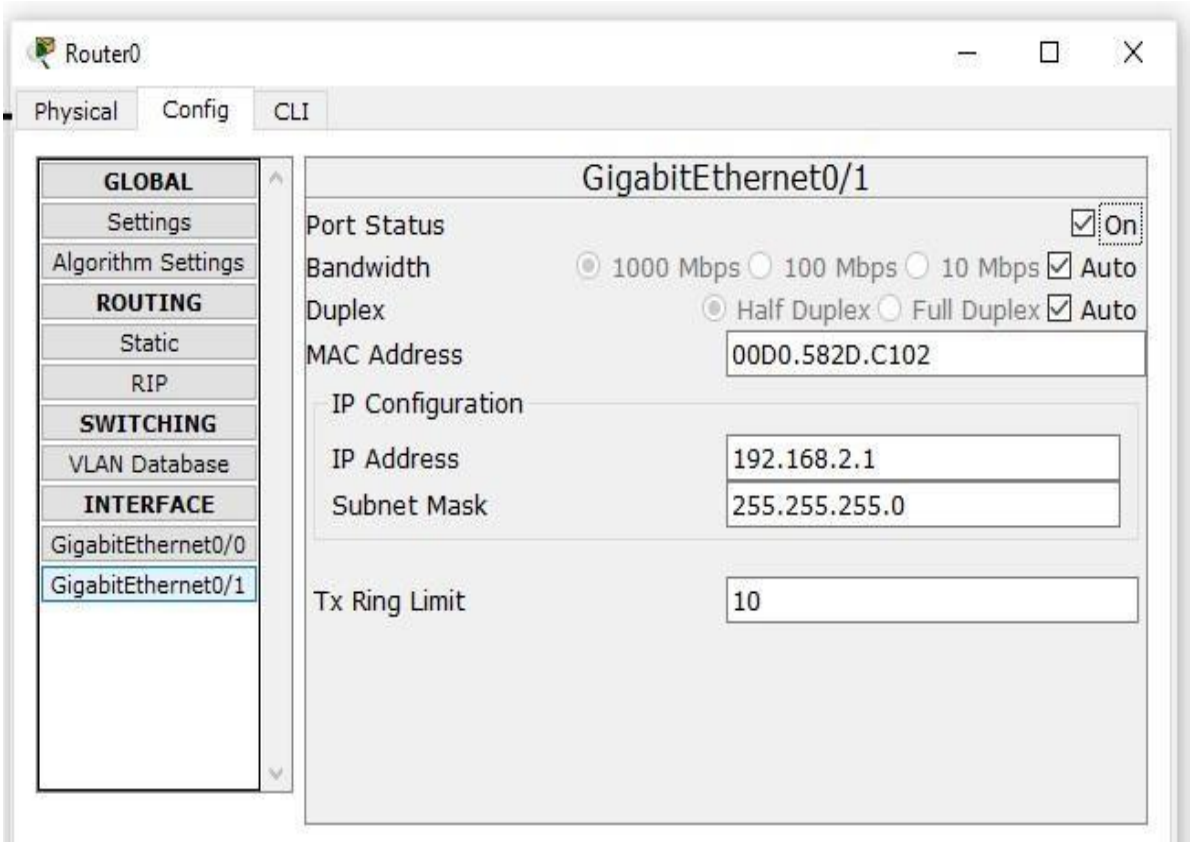


Configuring PC1

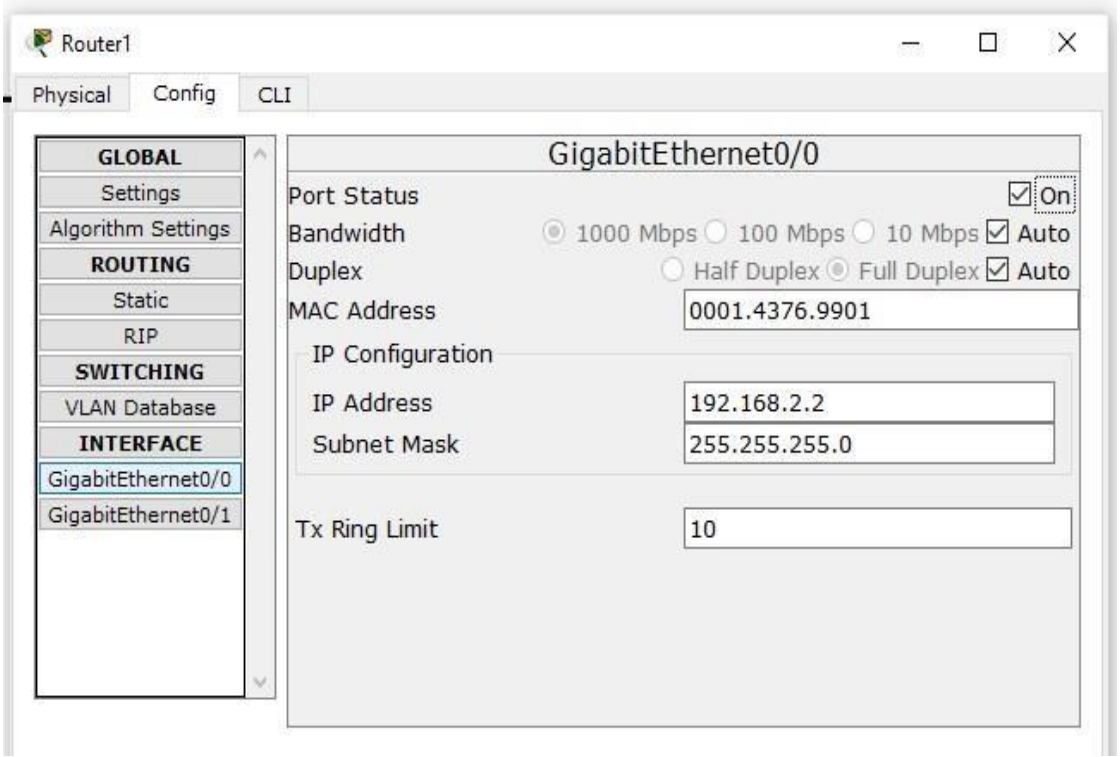


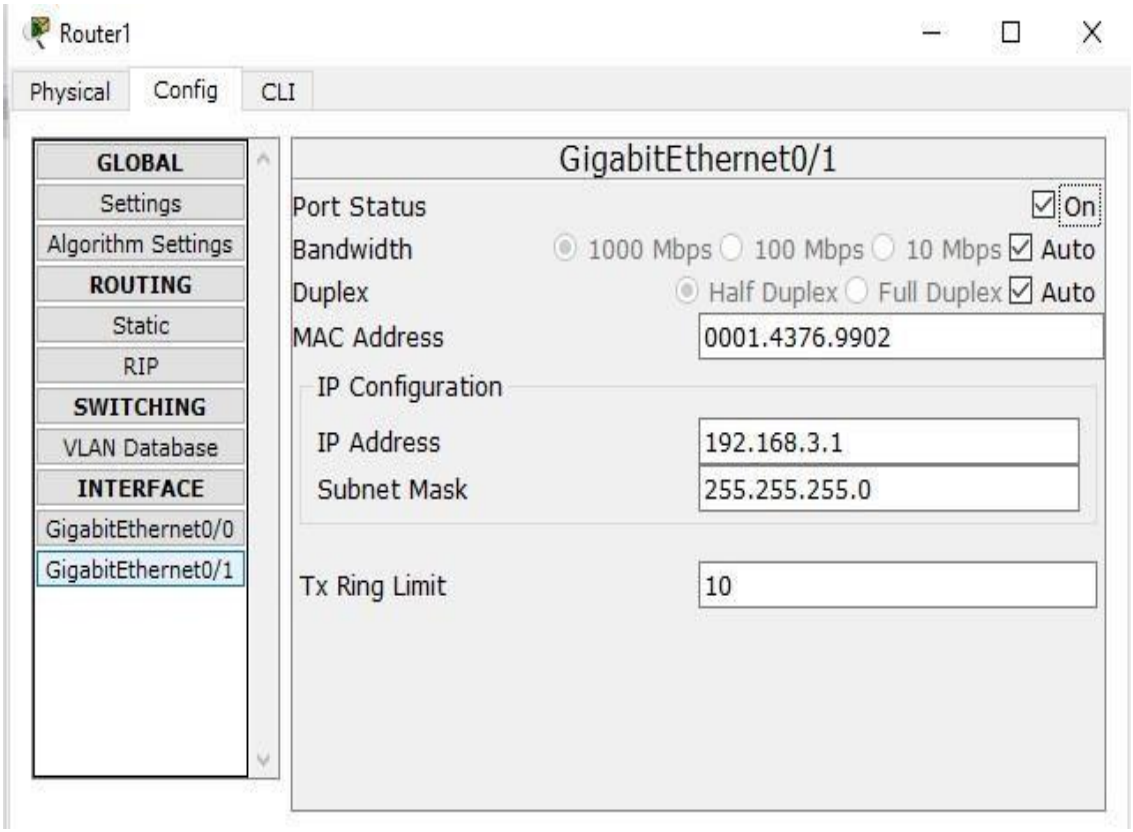
Configuring Router0



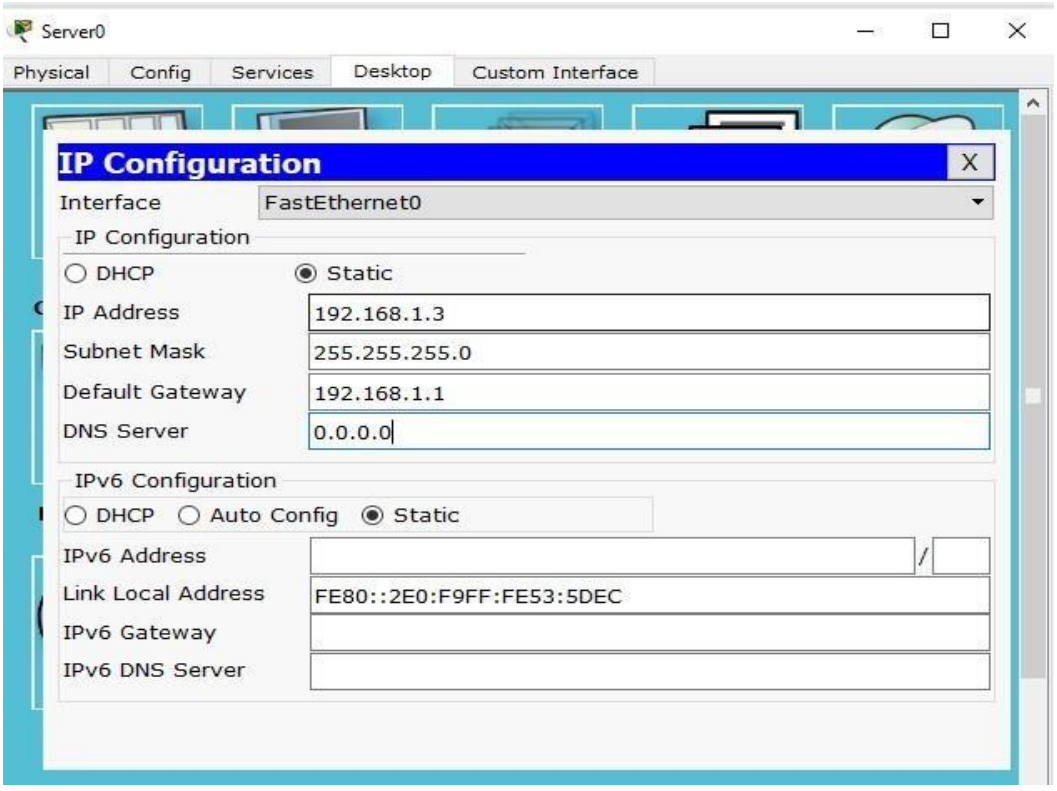


Configuring Router1

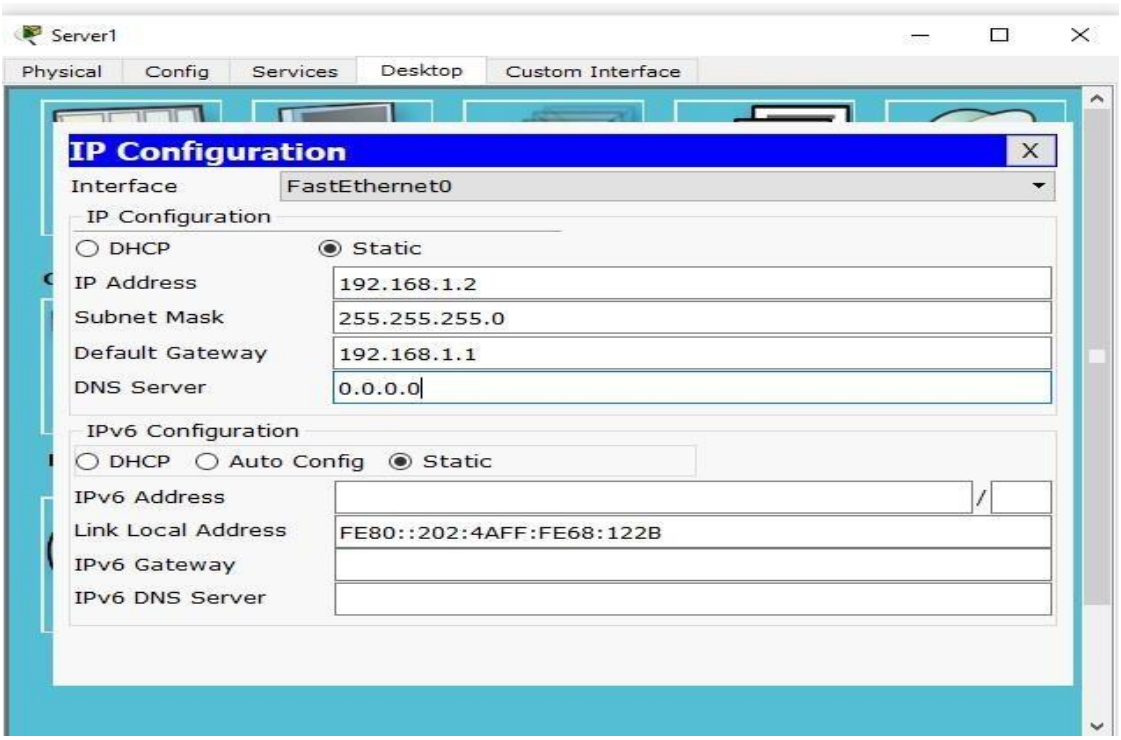




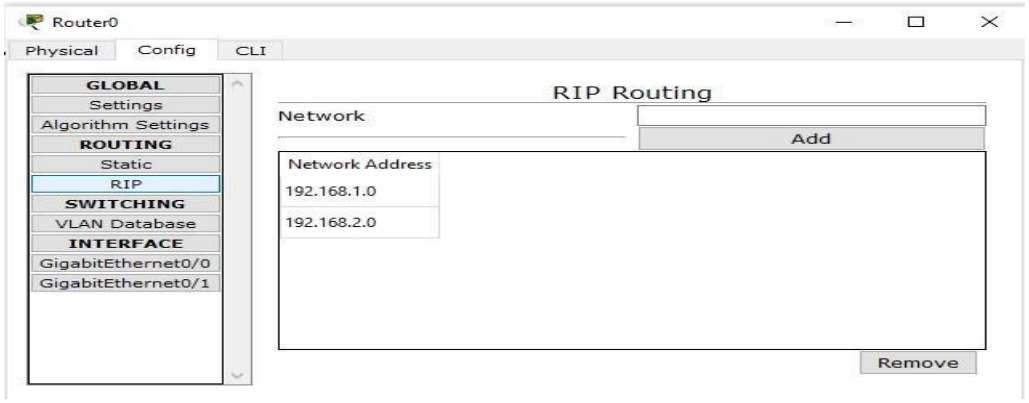
Configuring Server0

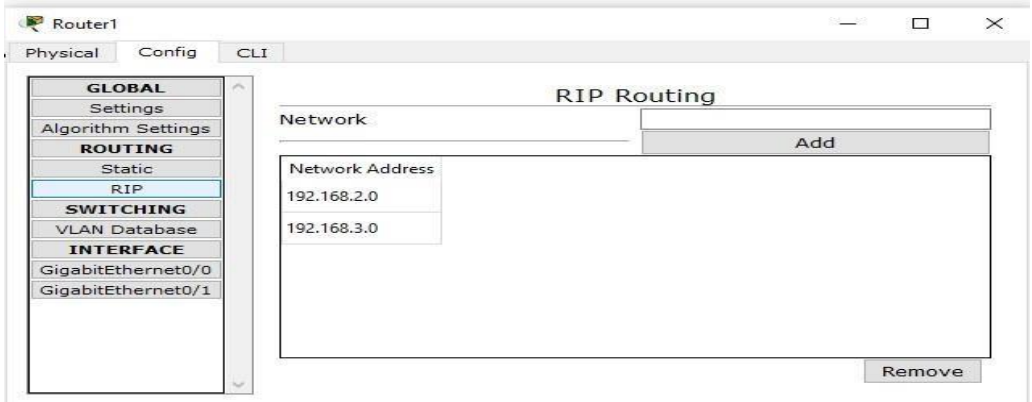


Configuring Server1



Set the RIP protocol on both the Routers as follows





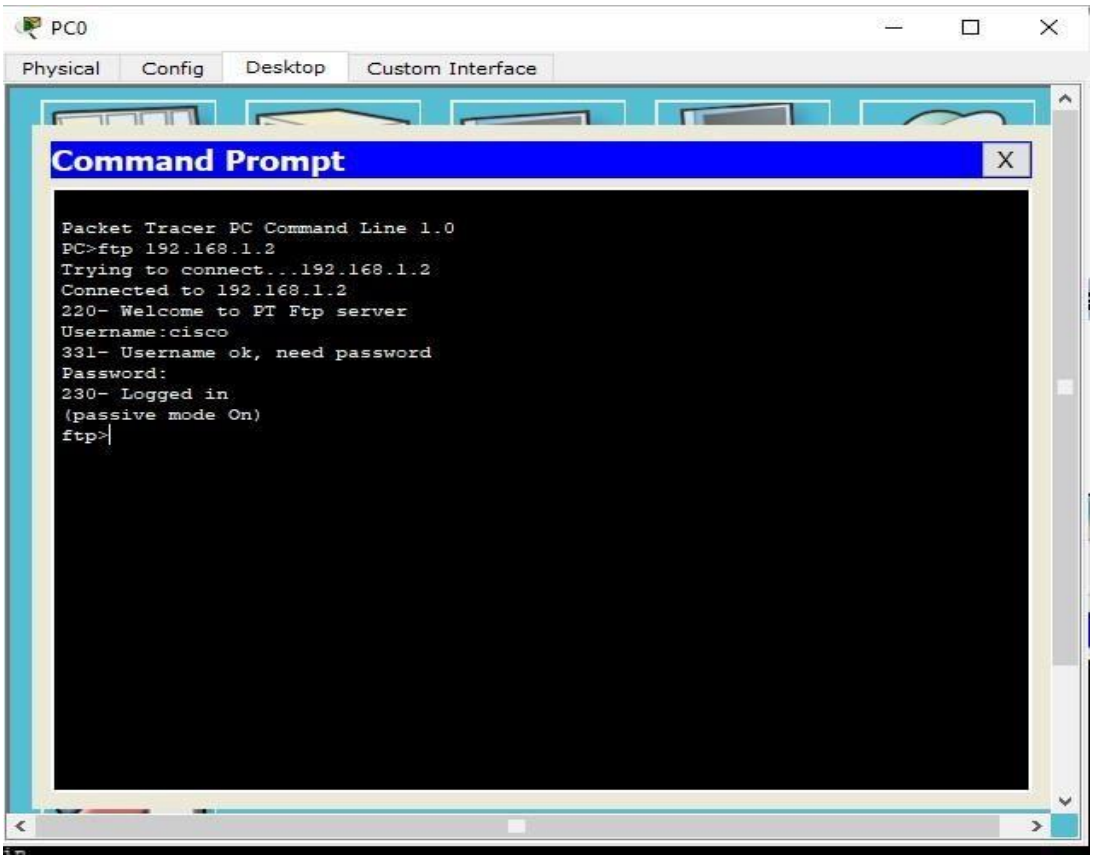
Check the connectivity by using the ping command

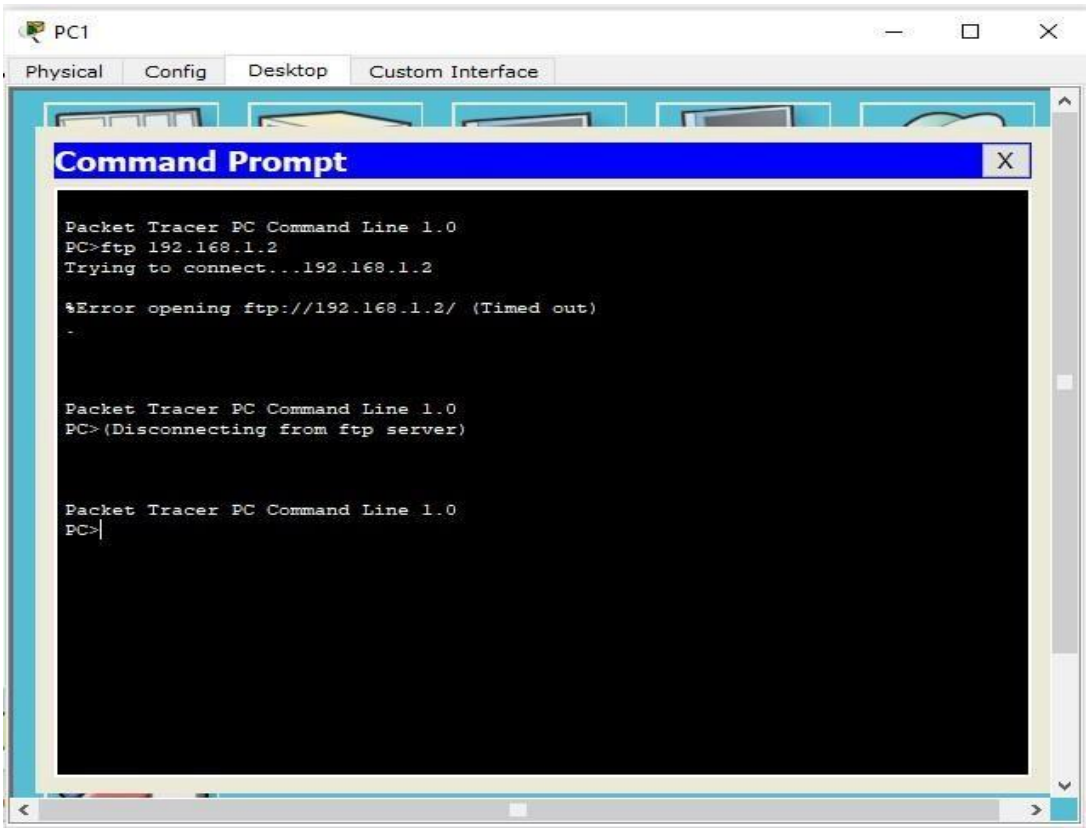
Part 1: Configure, Apply and Verify an Extended Numbered ACL

Click on Router1 go to CLI tab and press enter and enter the following commands: -

```
Router>en
Router# conf t
Ro uter(config)# access -list 100 permit tcp host 192.168.3.2 host 192.168.1.2 eq ftp
Router(config)#    interface GigabitEthernet0/ 0
Router(config)#    ip access -group 100 out
Router(config -line)#exit
Router(config)#
```

Now verify the ftp (ftp 192.168.1.2) command from both the PCs, one would be successful (PC0) and other (PC1) would fail.





Part 2: Configure, Apply and Verify an Extended Named ACL

We use the same topology for this case

Click on Router1 go to CLI tab and press enter and enter the following commands: -

Router>en

Router#conf t

Router(config)# ip access-list extended DALMIA

Router(config-ext-nacl)# permit tcp host 192.168.3.3 host 192.168.1.3 eq www

Router(config-ext-nacl)#exit

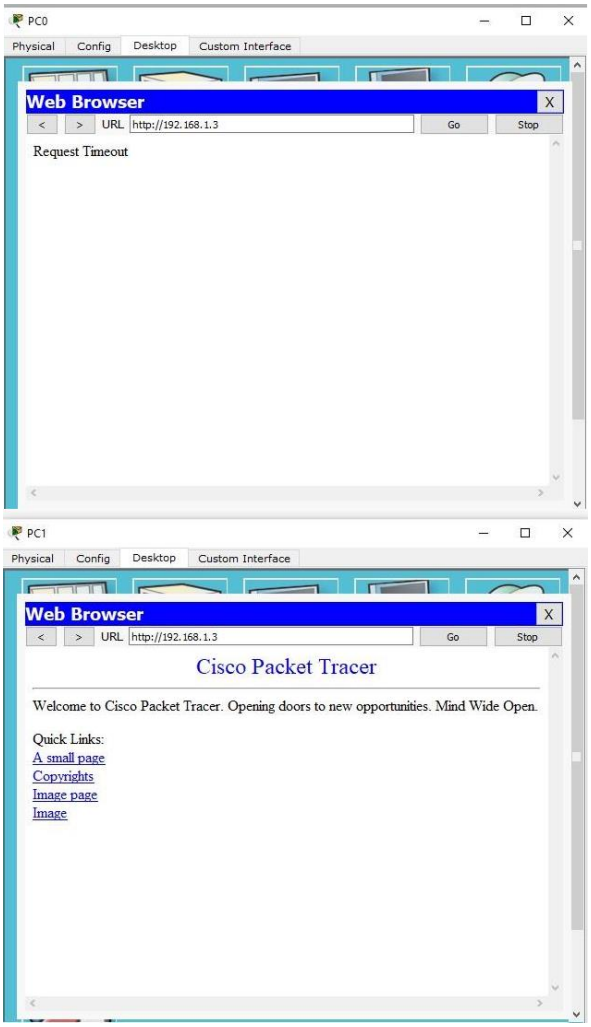
Router(config)#

Router(config)#interface GigabitEthernet0/0

Router(config-if)#ip access-group DALMIA out

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

Now verify the www (192.168.1.3) command from both the PCs browser, one would be successful (PC1) and other (PC0) would fail.

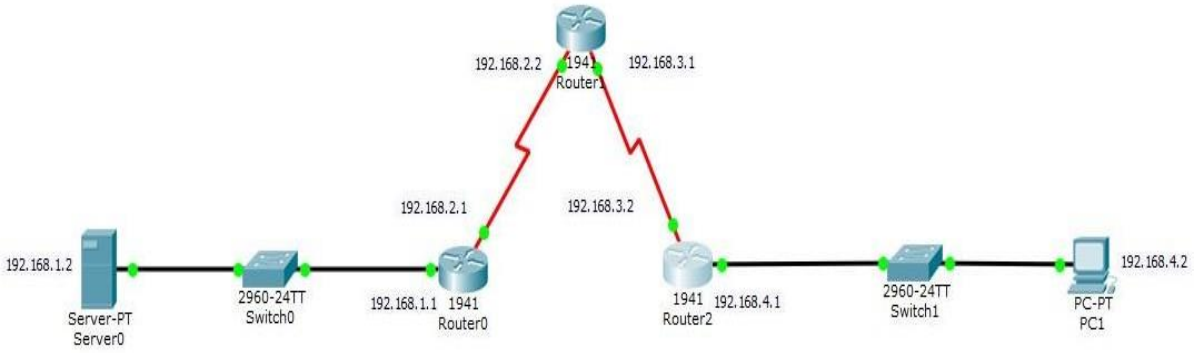


Hence Extended Numbered ACLs as well as Extended Named ACLs have been verified

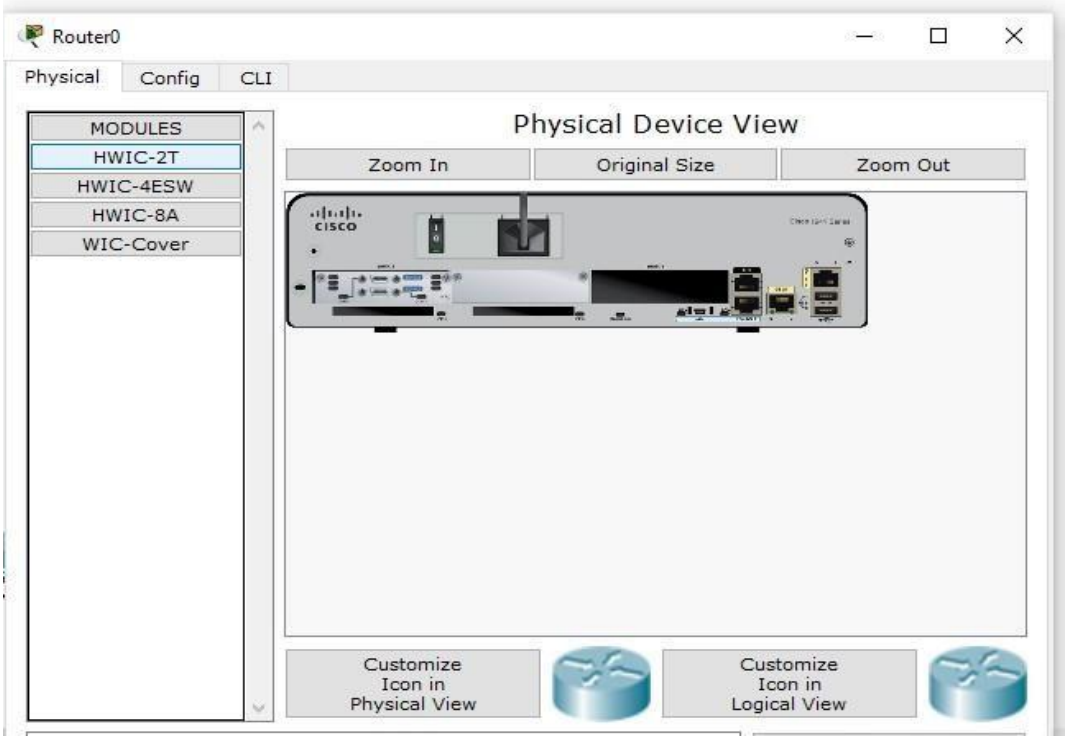
Practical 4

Configure IP ACLs to Mitigate Attacks

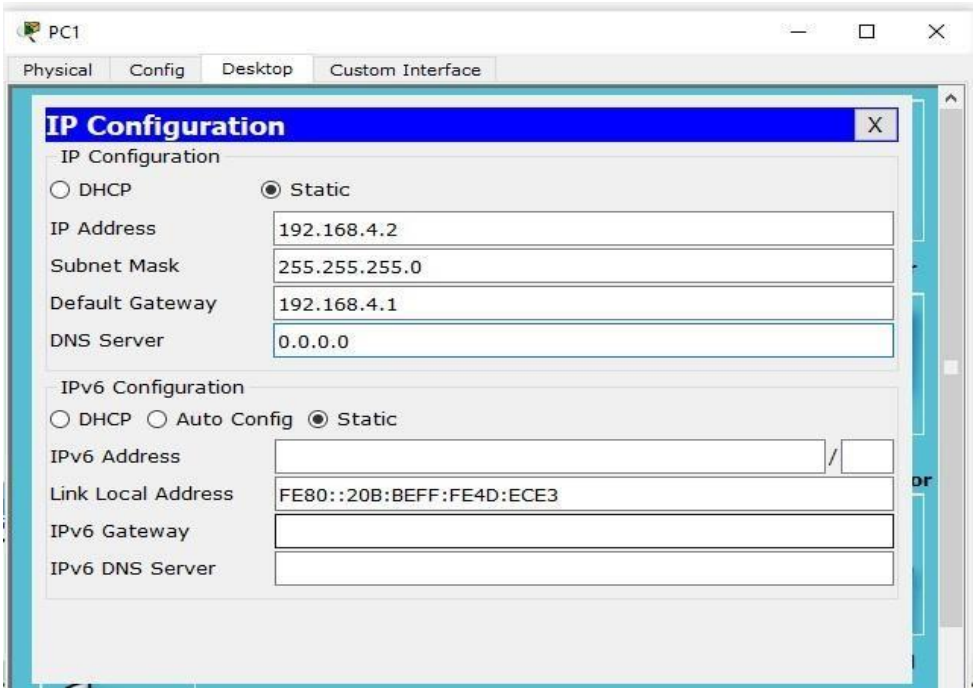
Topology



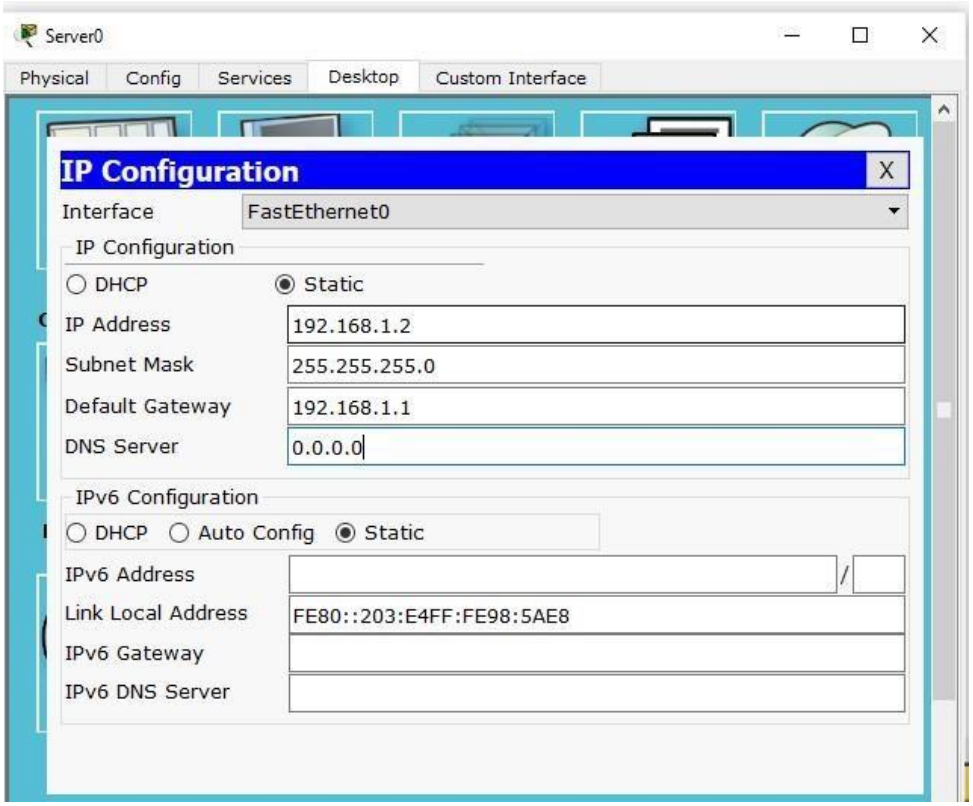
The serial interface in each Router is added as follows



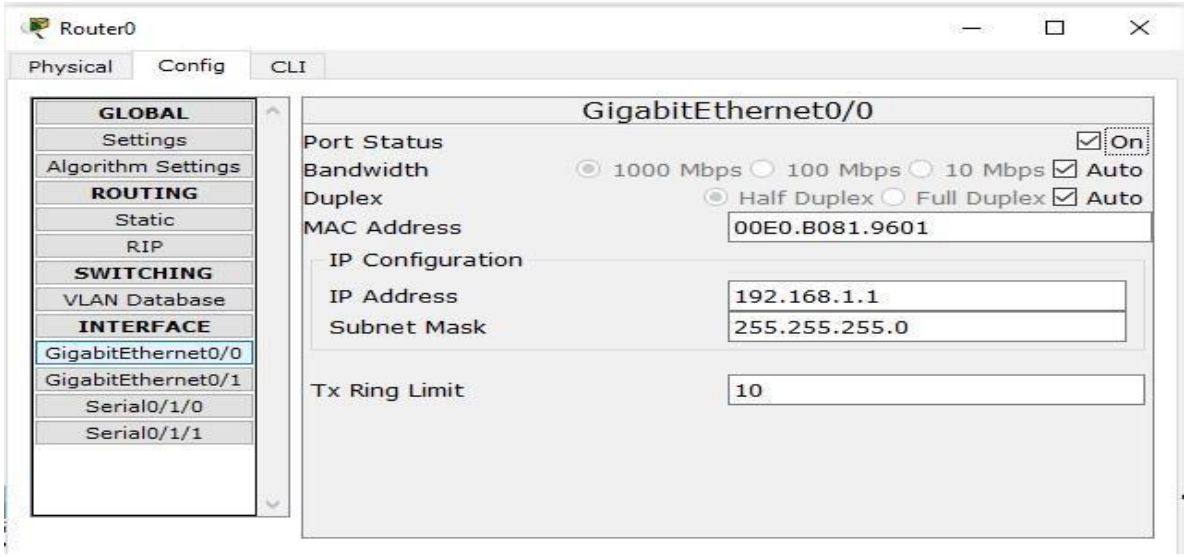
Configuring PC1

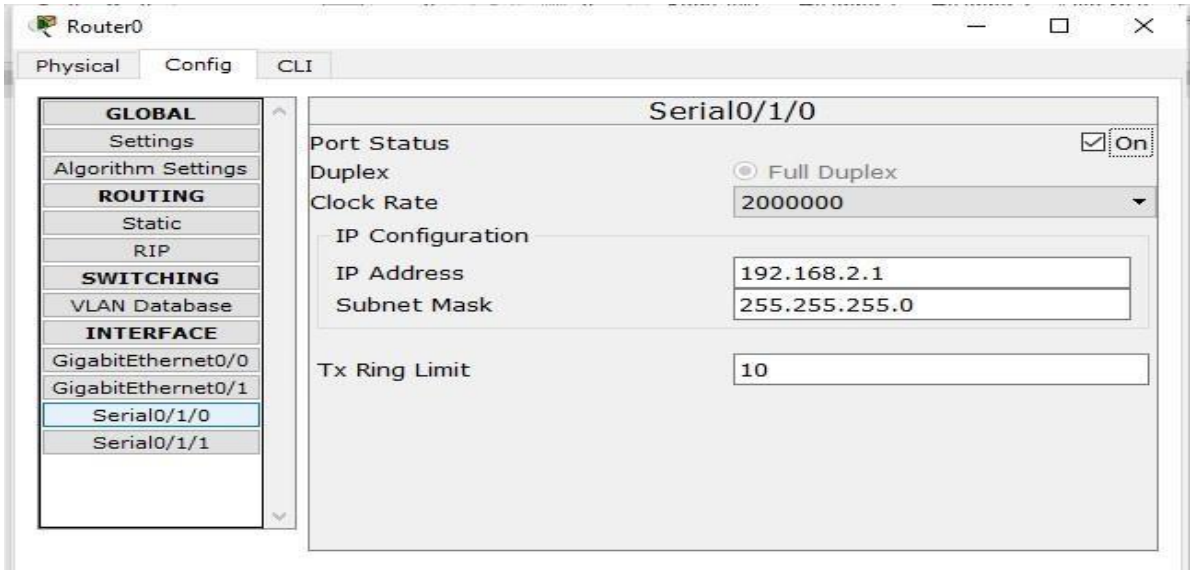


Configuring Server0

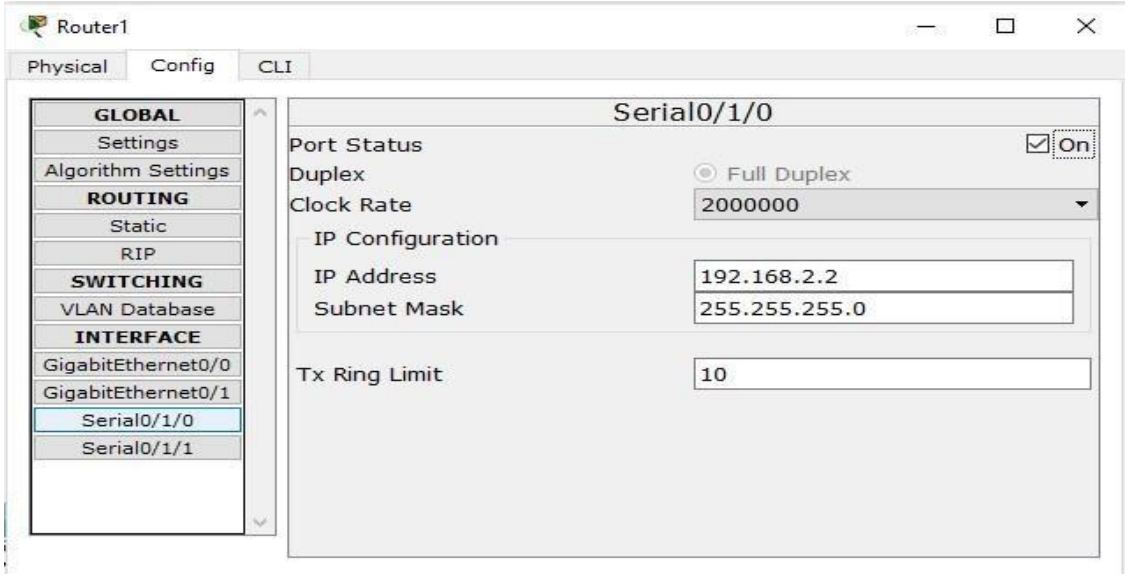


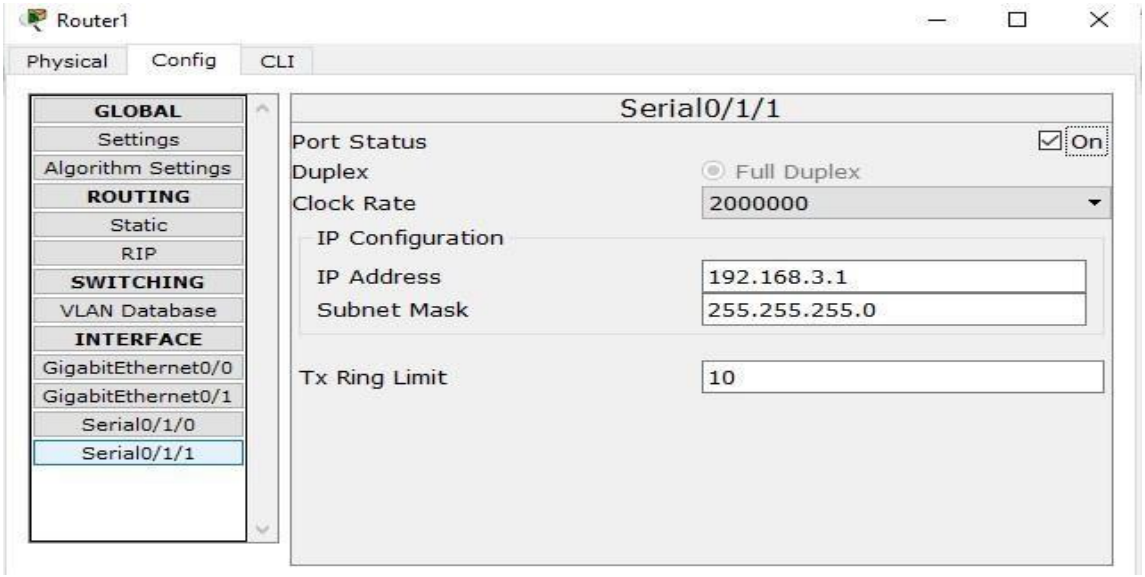
Router0



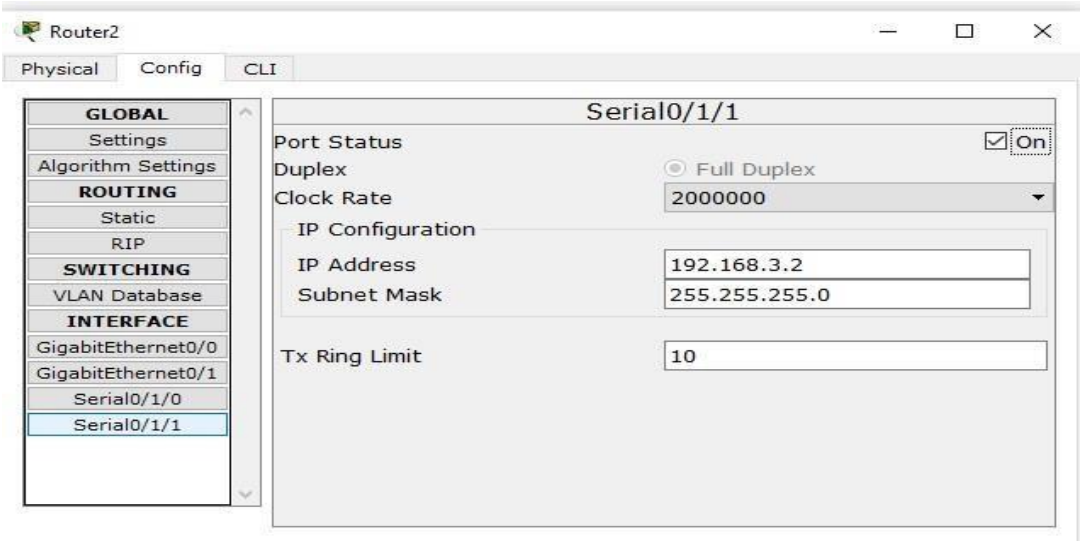


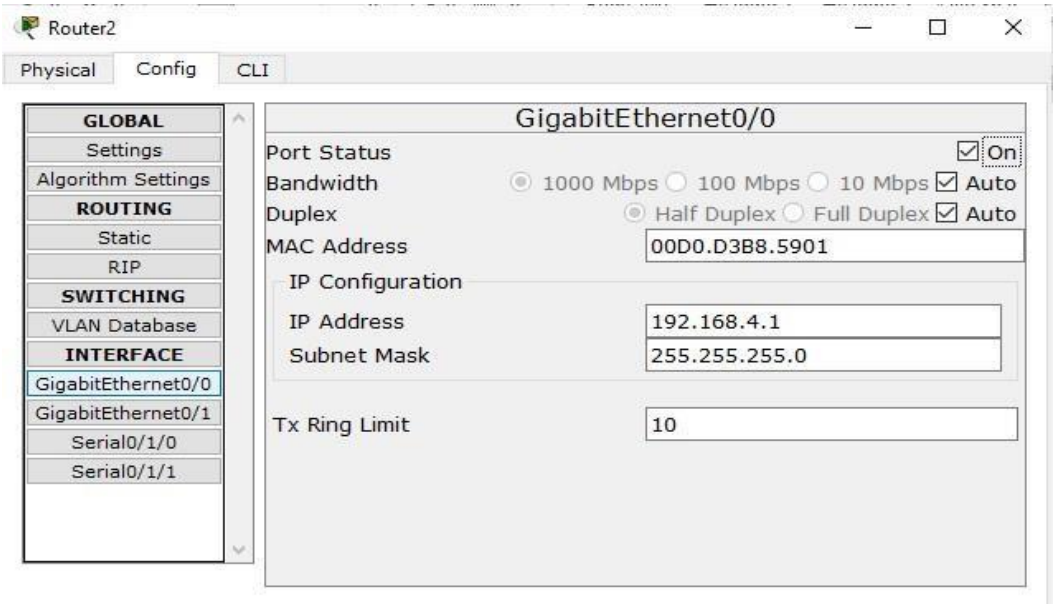
Router1



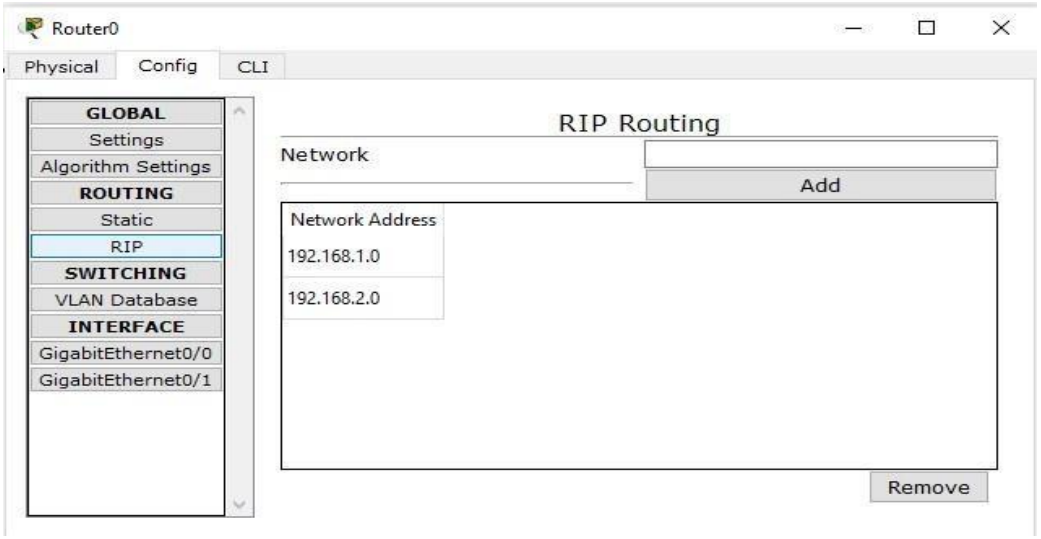

Router1
Physical
Config
CLI
GLOBAL
Settings
Algorithm Settings
ROUTING
Static
RIP
SWITCHING
VLAN Database
INTERFACE
GigabitEthernet0/0
GigabitEthernet0/1
Serial0/1/0
Serial0/1/1
Serial0/1/1
Port Status
On
Duplex
Full Duplex
Clock Rate
2000000
IP Configuration
IP Address
192.168.3.1
Subnet Mask
255.255.255.0
Tx Ring Limit
10

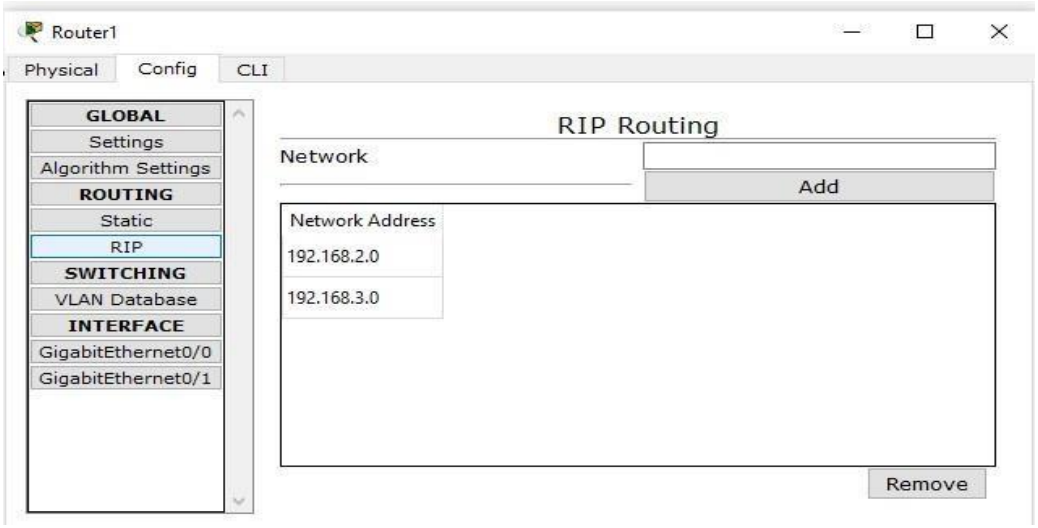
Router2

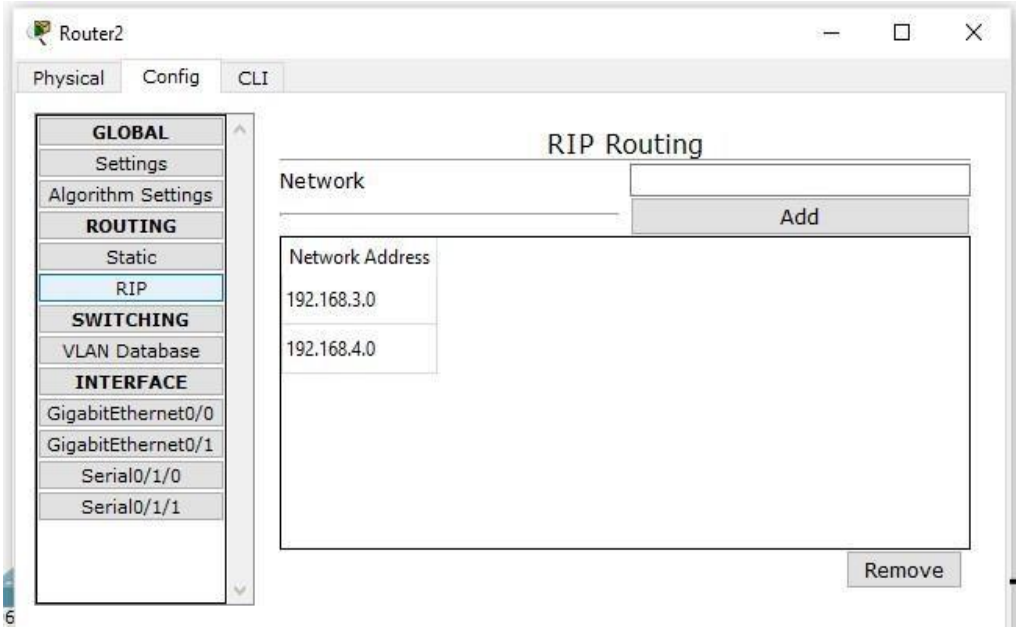

Router2
Physical
Config
CLI
GLOBAL
Settings
Algorithm Settings
ROUTING
Static
RIP
SWITCHING
VLAN Database
INTERFACE
GigabitEthernet0/0
GigabitEthernet0/1
Serial0/1/0
Serial0/1/1
Serial0/1/1
Port Status
On
Duplex
Full Duplex
Clock Rate
2000000
IP Configuration
IP Address
192.168.3.2
Subnet Mask
255.255.255.0
Tx Ring Limit
10


Router2
Physical
Config
CLI
GLOBAL
Settings
Algorithm Settings
ROUTING
Static
RIP
SWITCHING
VLAN Database
INTERFACE
GigabitEthernet0/0
GigabitEthernet0/1
Serial0/1/0
Serial0/1/1
GigabitEthernet0/0
Port Status
On
Bandwidth
1000 Mbps
100 Mbps
10 Mbps
Auto
Duplex
Half Duplex
Full Duplex
Auto
MAC Address
00D0.D3B8.5901
IP Configuration
IP Address
192.168.4.1
Subnet Mask
255.255.255.0
Tx Ring Limit
10

Set the RIP protocol on all the Routers as follows

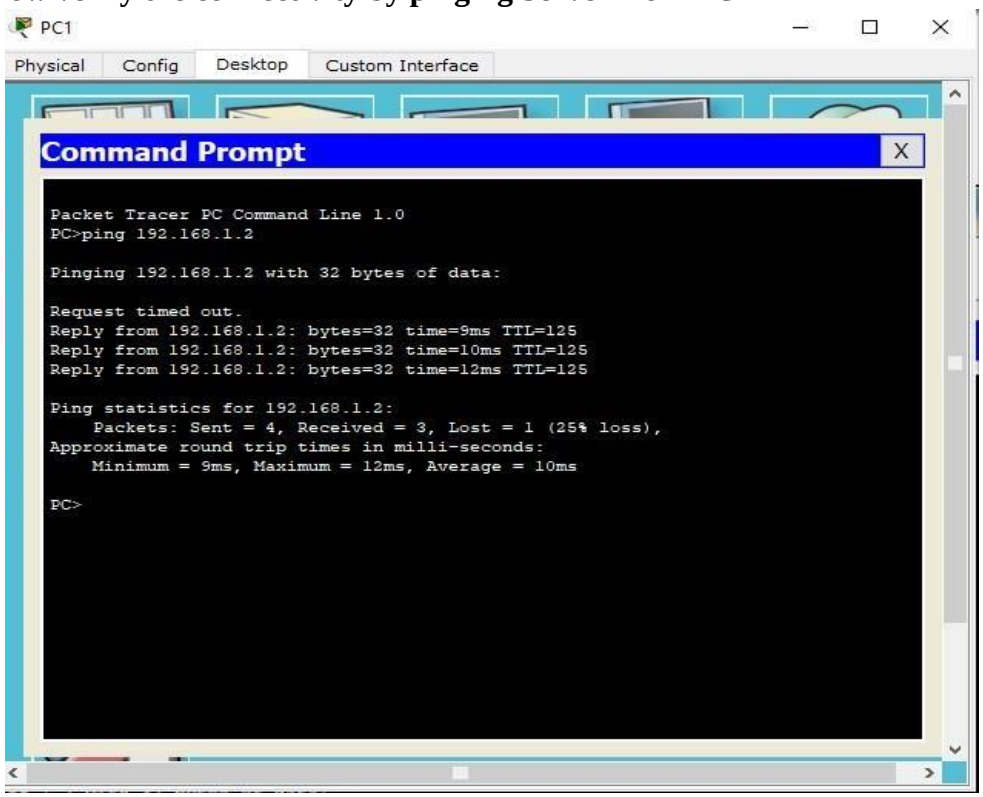




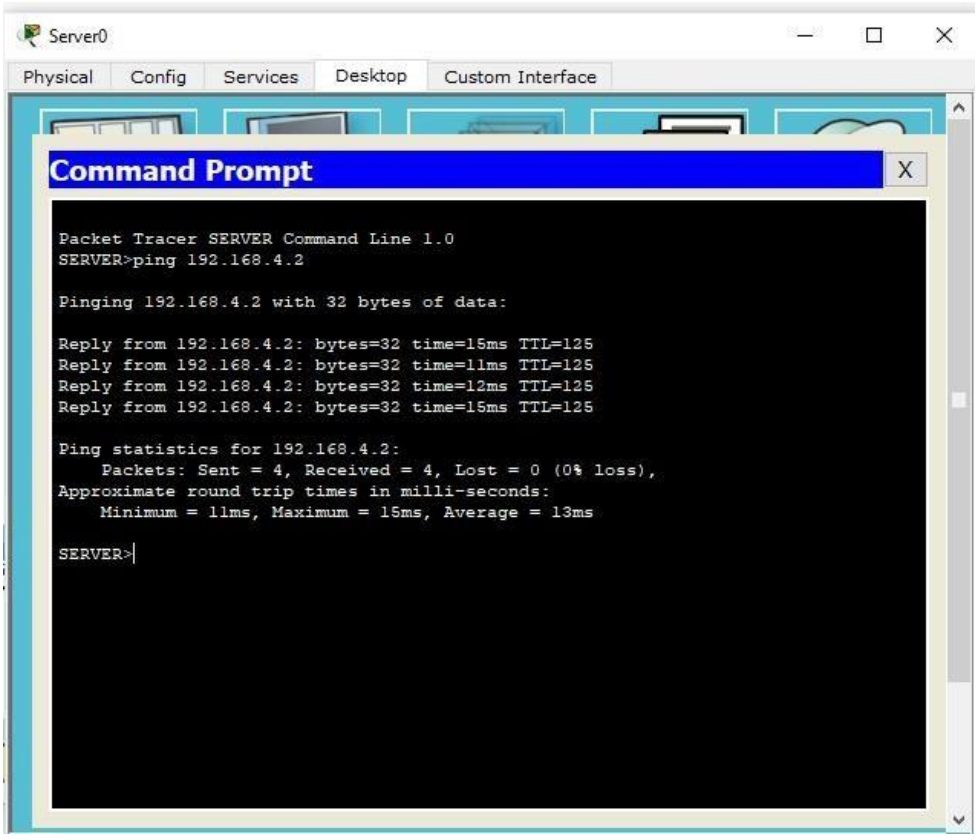


Part 1: Verify Basic Connectivity

We can now verify the connectivity by pinging Server from PC



We can now verify the connectivity by pinging PC from Server



Part 2: Secure Access to Routers

We configure ACL 10 to block all remote access to the Routers and allow remote access only from PC. We type the following commands in all the Routers (Router0, Router1, and Router2). This part is divided in 2 subparts.

Part a) Set up the SSH protocol

Enter the following commands in CLI mode of all Routers.

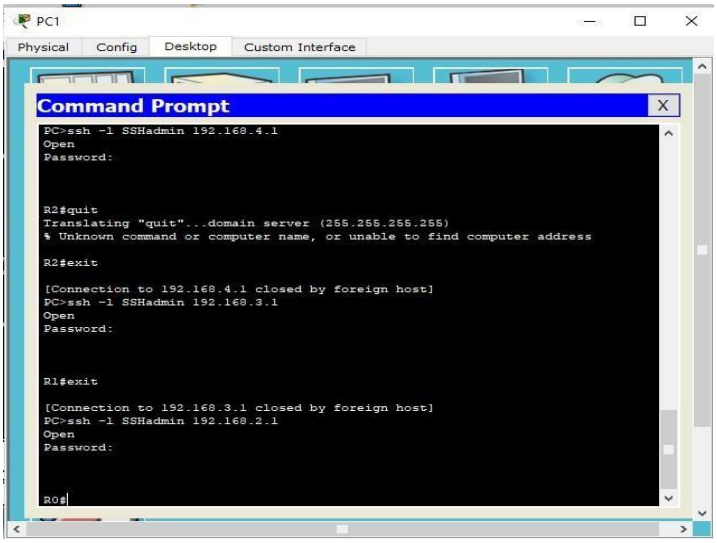
```
Router>en
Router# conf t
Ro uter(config)# ip domain -name dalmia .com
Router(config)#    hostname R0
R 0(config)#  crypto key generate rsa
R 0(config)#  line vty 0 4
R 0(config -line)# transport input ssh
R 0(config -line)# login local
R 0(config -line)# exit
R 0(config)#   username SSHadmin privilege 15 password        dalmia
R0(config)#exit
```

Part b) Create an ACL 10 to permit remote access to PC only

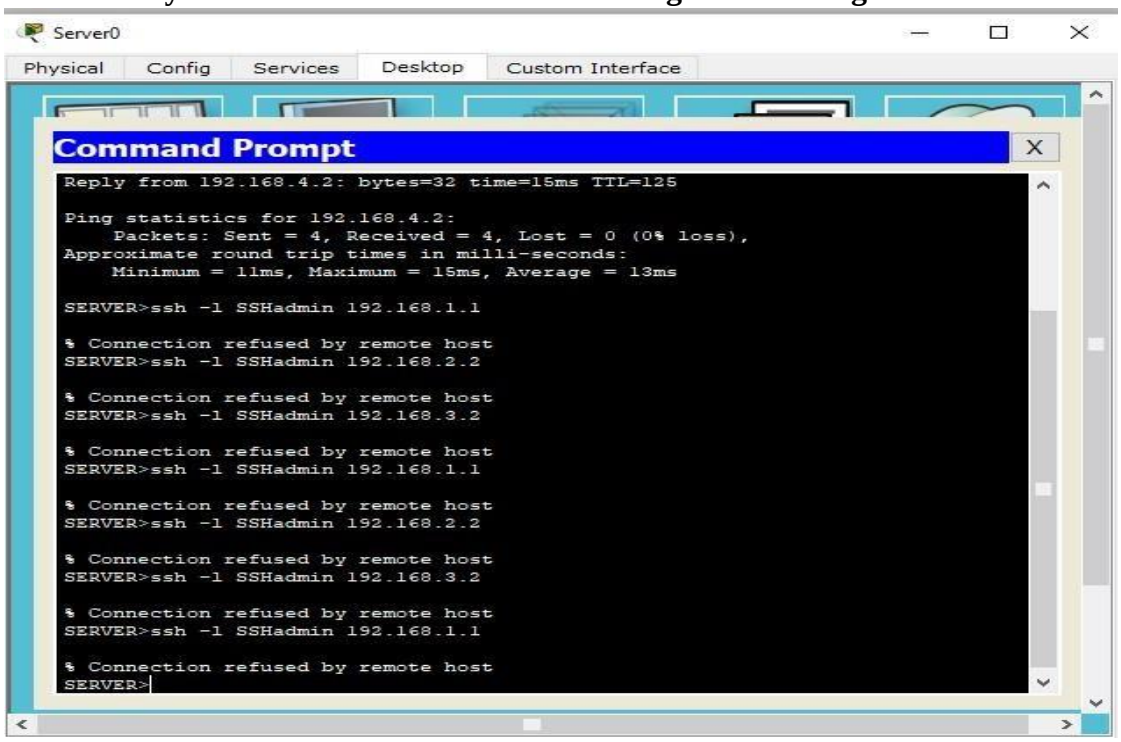
Enter the following commands in CLI mode of all Routers

```
Router>en
Router# conf t
Ro uter(config)#  access -list 10 permit host 192.168.4.2
R outer(config)# line vty 0 4
R outer(config -line)# access -class 10 in
```

Now we verify the remote access from PC using the following and find it to be successful.



Now we verify the remote access from Server using the following and find it to be failure



Part 3: Create a Numbered IP ACL 120 on R1
We need to perform the following in this part

1) Create an IP ACL numbered 120 on R1 using the following rules
2) Permit any outside host to access DNS, SMTP, and FTP services on server
3) Deny any outside host access to HTTPS services on **server**
4) Permit **PC1 to** access **R1** via SSH. (done in previous part)

Enter the following commands in the CLI mode of Router1

R1>enable

R1#

R1#configure terminal

R1(config)#access-list 120 permit udp any host 192.168.1.2 eq domain

R1(config)#access-list 120 permit tcp any host 192.168.1.2 eq smtp

R1(config)#access-list 120 permit tcp any host 192.168.1.2 eq ftp

R1(config)#access-list 120 deny tcp any host 192.168.1.2 eq 443

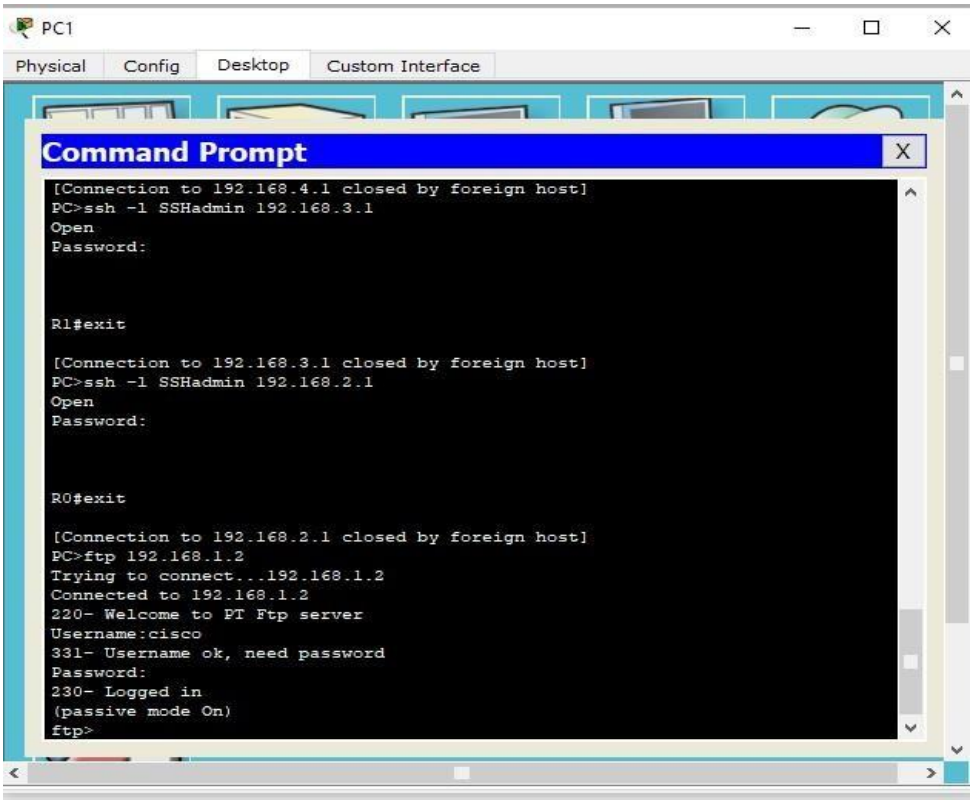
R1(config)#exit

R1#configure terminal

R1(config)#interface Serial0/1/1

R1(config-if)#ip access-group 120 in

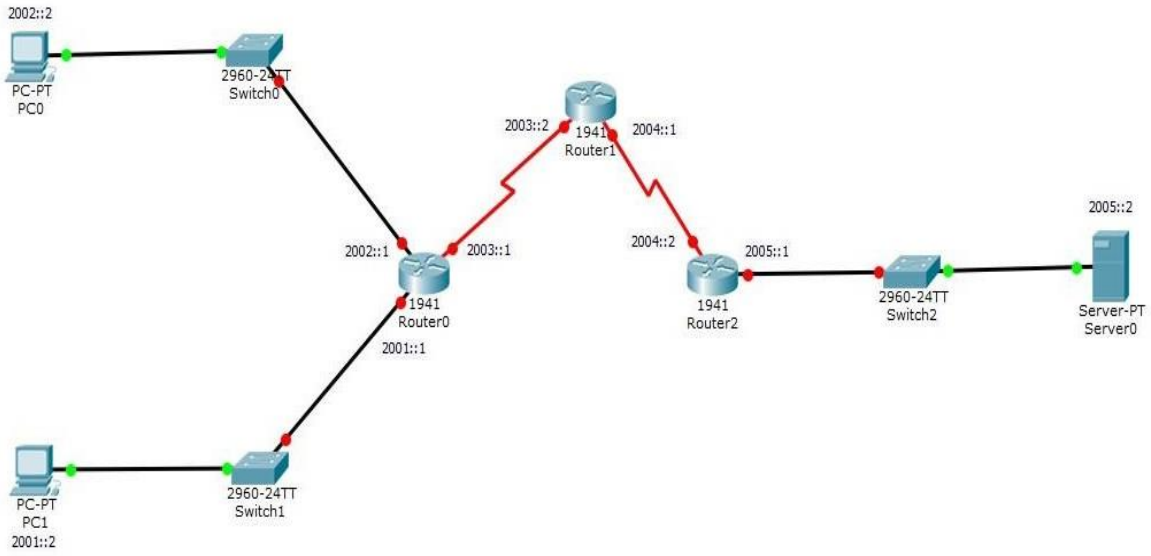
Verify the above entering the following commands in the PC



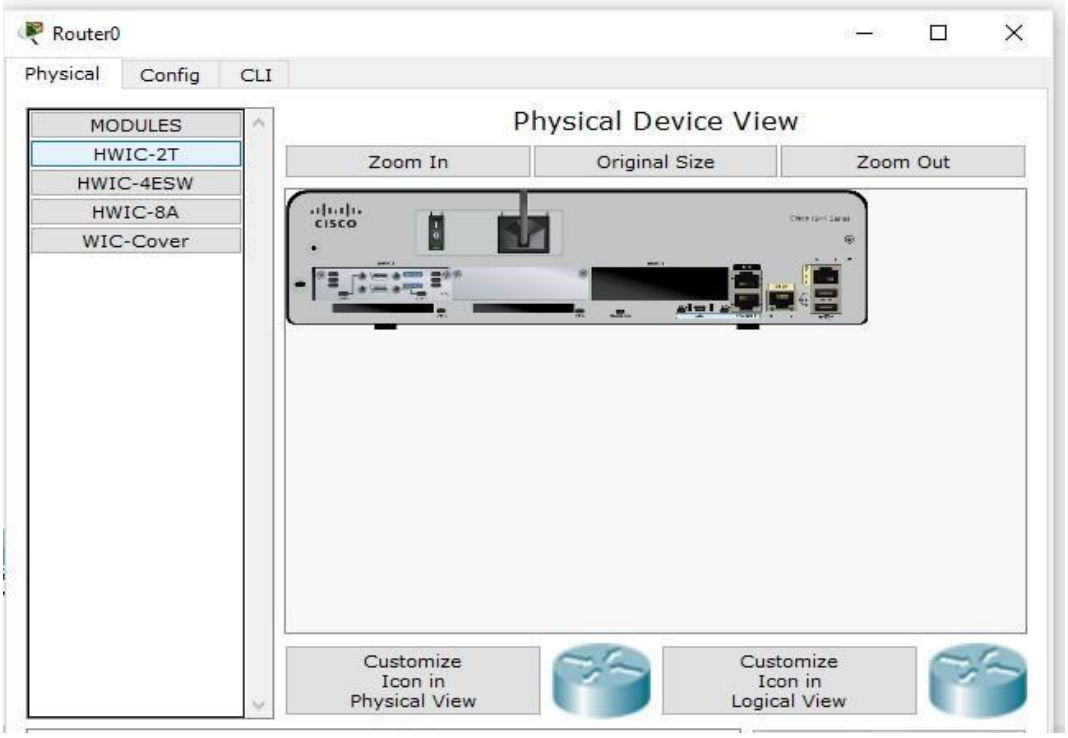
Hence, we have applied and verified all the required ACLs.

Practical 5
Configure IPV6 ACLs

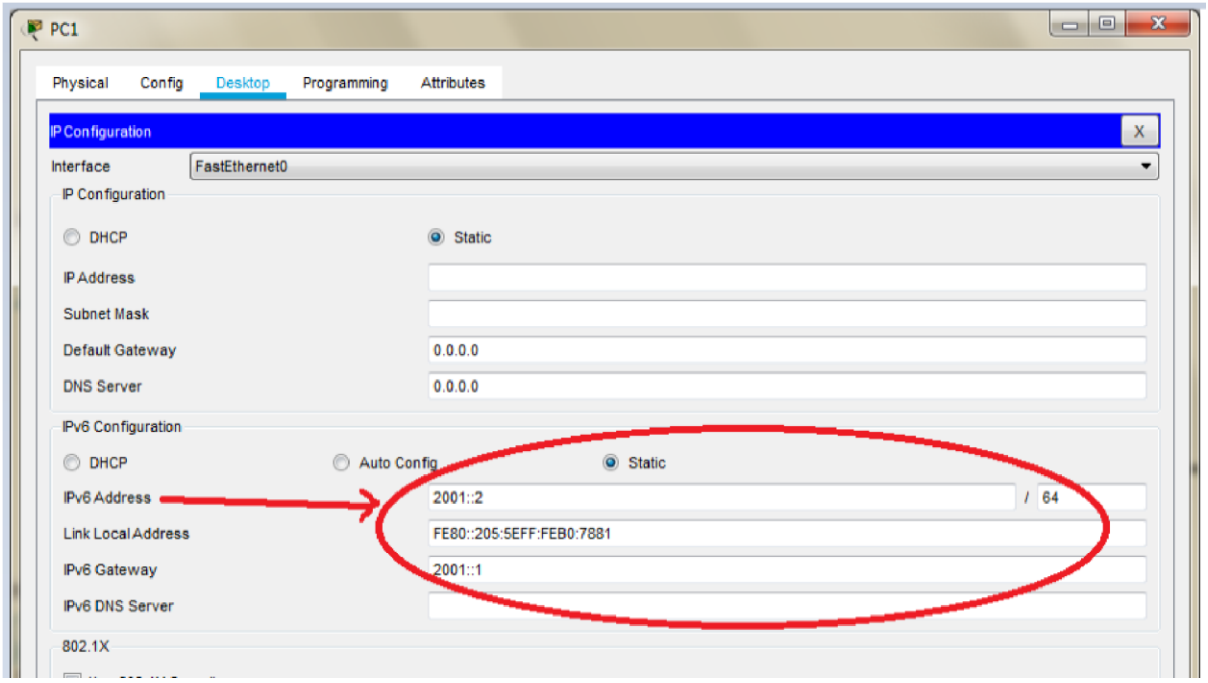
Topology



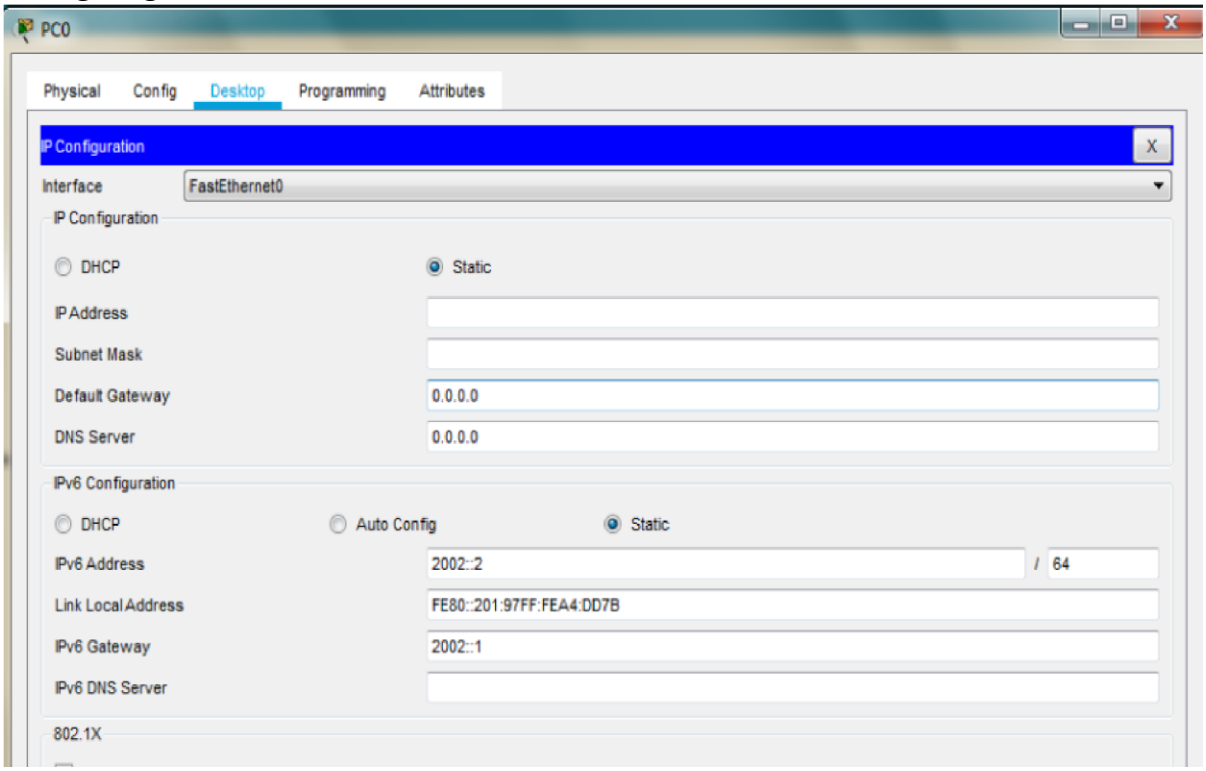
The serial interface in each Router is added as follows



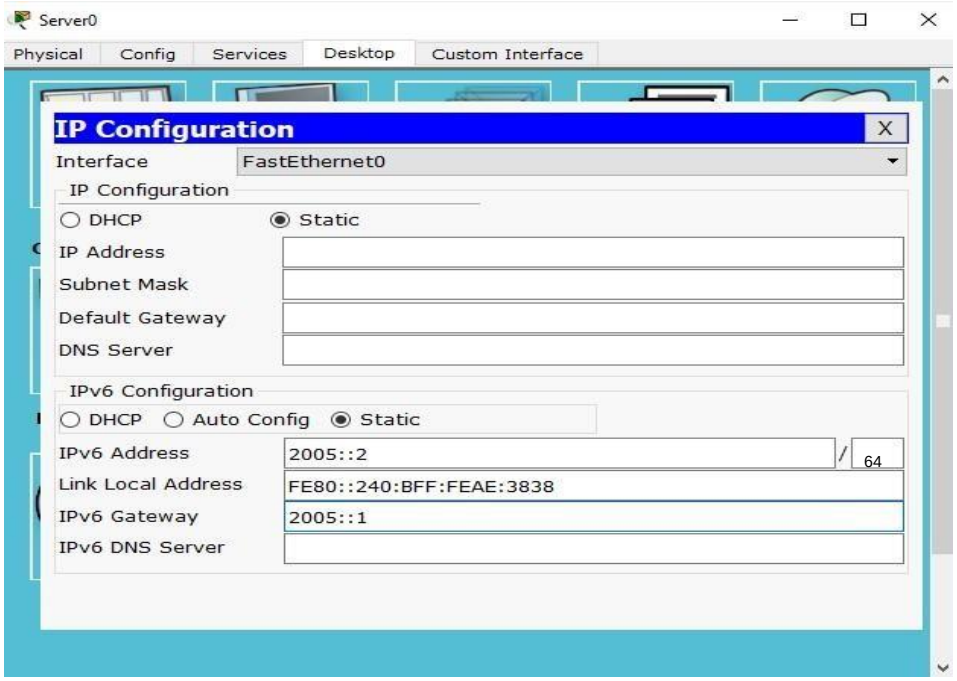
Configuring PC1



Configuring PC0



Configuring Server0



For setting the ipv6 addresses we need to use the CLI mode for each Router as follows

Configuring Router0

```
Router>
Router>en
Router#
Router#conf  t
Router(config)#ipv6 unicast     -routing
Router(config)#int G0/0
Router(config  -if)#ipv6 address 2002::1/64
Router(config  -if)#ipv6 rip a enable
Router(config  -if)#no shut
Router(config  -if)#exit
Router(config)#
Router(config)#int G0/1
Router(config -if)#ipv6 addr ess 2001::1/64
Router(config  -if)#ipv6 rip a enable
Router(config  -if)#no shut
Router(config -if)#exit
Router(config)#
Router(config)#int Se0/1/0
Router(config  -if)#ipv6 address 2003::1/64
Router(config  -if)#ipv6 rip a enable
Router(config  -if)#no shut
Router(co  nfig -if)#exit
Router(config)#
```

Configuring Router1

```
Router>en
Router#conf t
Router(config)#ipv6 unicast-routing
Router(config)#
Router(config)#int Se0/1/0
Router(config-if)#ipv6 address 2003::2/64
Router(config-if)#ipv6 rip a enable
Router(config-if)#no shut
Router(config-if)#
Router(config-if)#exit
Router(config)#
Router(config)#int Se0/1/1
Router(config-if)#ipv6 address 2004::1/64
Router(config-if)#ipv6 rip a enable
Router(config-if)#no shut
Router(config-if)#exit
Router(config)#
```

Configuring Router2

```
Router>en
Router#conf t
Router(config)#ipv6 unicast-routing
Router(config)#
Router(config)#int Se0/1/1
Router(config-if)#ipv6 address 2004::2/64
Router(config-if)#ipv6 rip a enable
Router(config-if)#no shut
Router(config-if)#exit
```

Router(config)#int G0/0

Router(config-if)#ipv6 address 2005::1/64

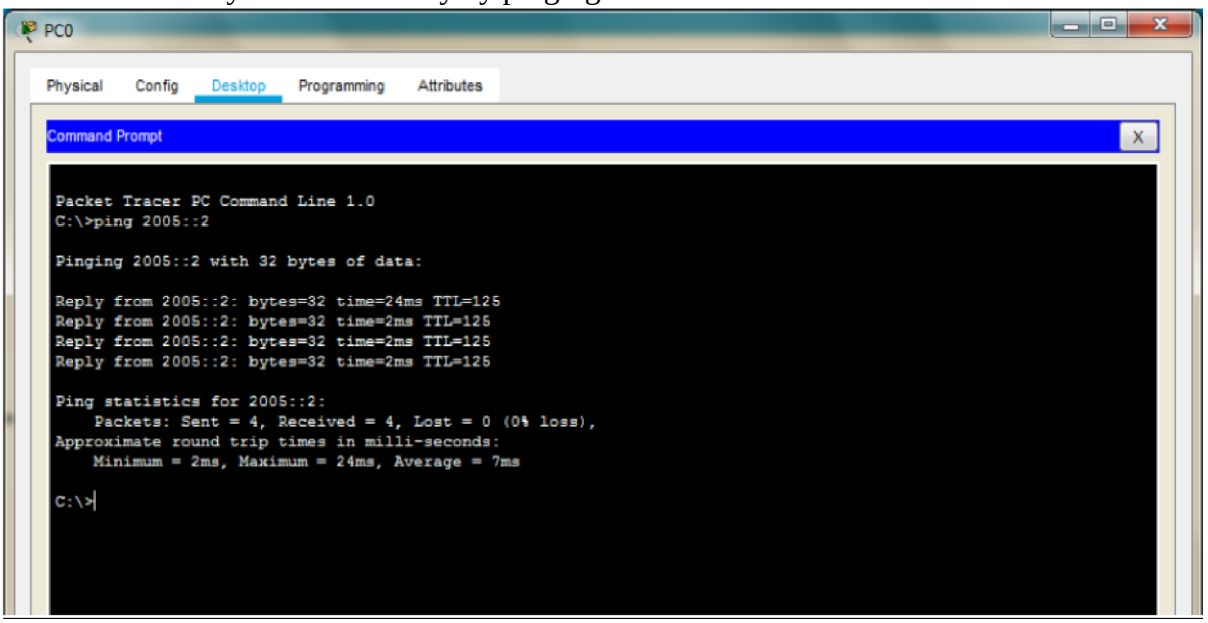
Router(config-if)#ipv6 rip a enable

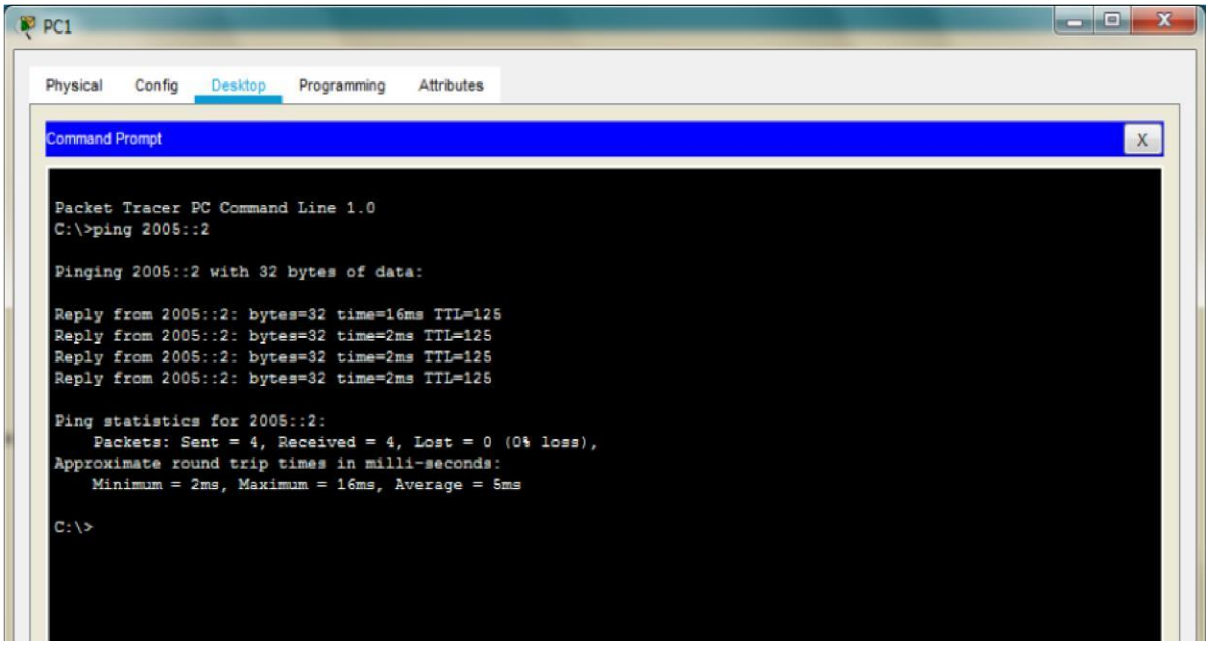
Router(config-if)#no shut

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

Verify Connectivity

We can now verify the connectivity by pinging Server from PCs





And we see that the connectivity is established

We configure the ACL and apply it to the Router1 with the following conditions

1) No HTTP or HTTPS allowed on server by any host
2) No www service accessible on the server by any host
3) Only ipv6 packets allowed towards the server

We enter the following commands in the CLI mode of the Router1 and apply it at the proper interface

Router>

Router>enable

Router#conf t

Router(config)#ipv6 access-list dalmia

Router(config-ipv6-acl)#deny tcp any host 2005::2 eq www

Router(config-ipv6-acl)#deny tcp any host 2005::2 eq 443

Router(config-ipv6-acl)#permit ipv6 any any

Router(config-ipv6-acl)#

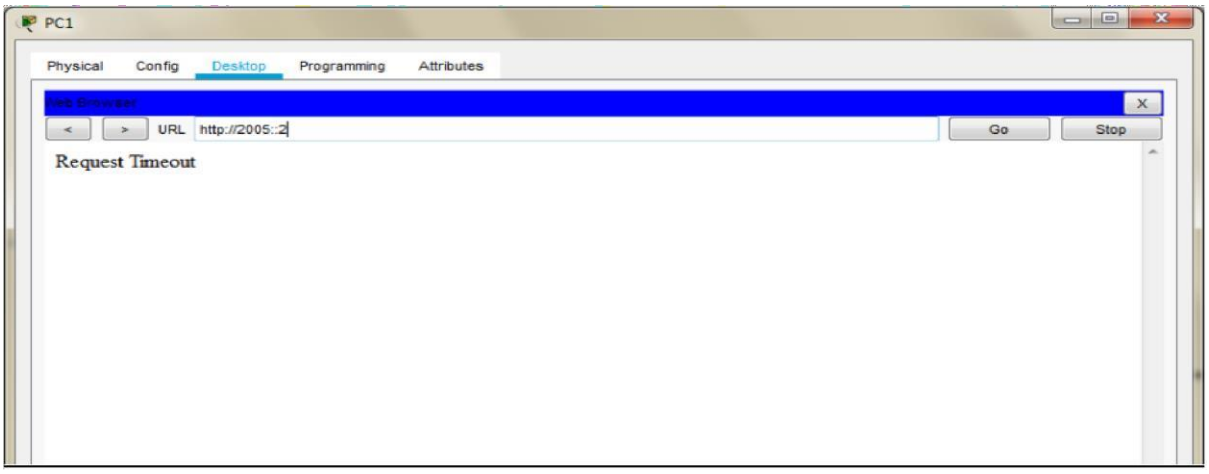
Router(config-ipv6-acl)#exit
Router(config)#

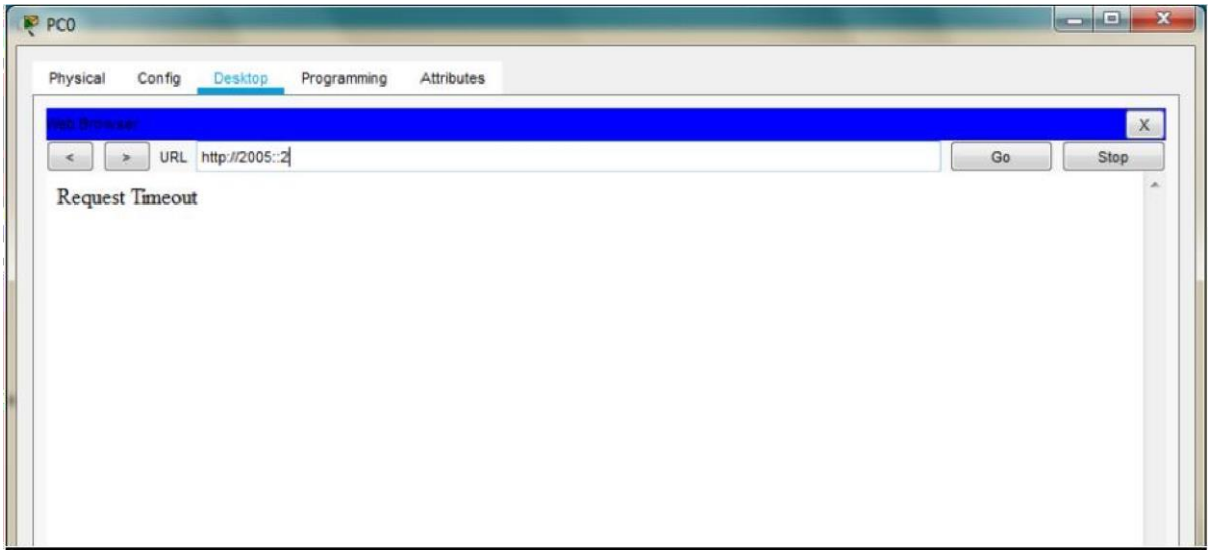
Router(config)#int Se0/1/0
Router(config-if)#ipv6 traffic-filter dalmia in

Router(config-if)#exit

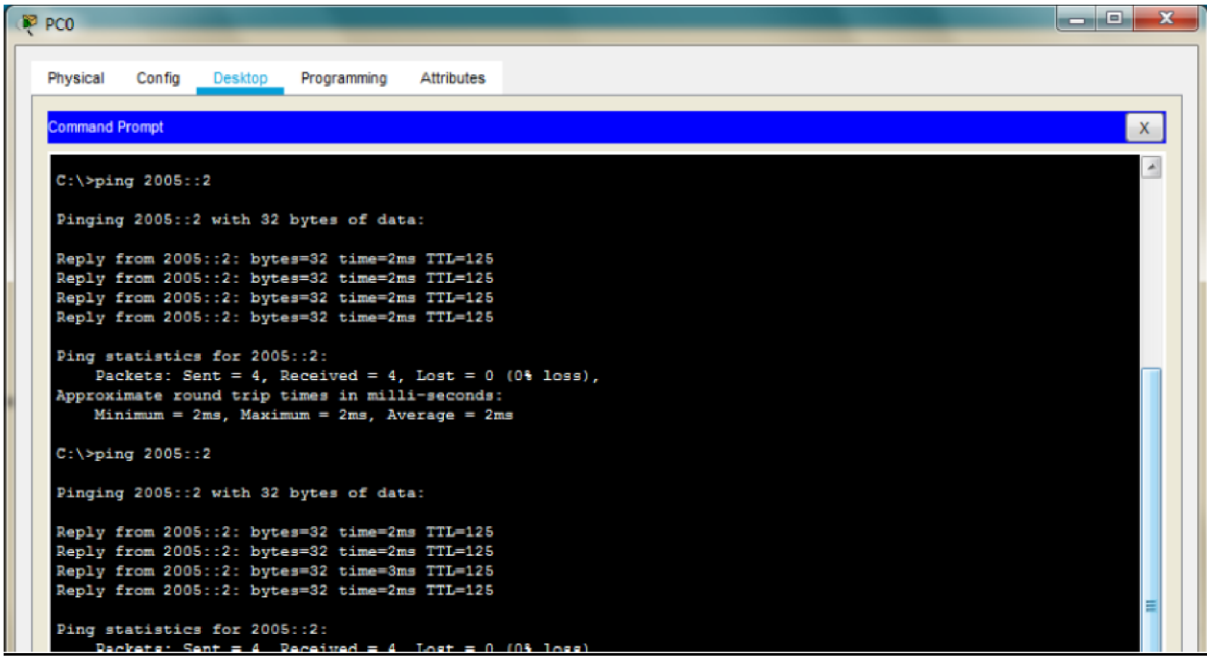
Router(config)#

We verify the configuration by first accessing the www service from the browser of both PCs and get failure.





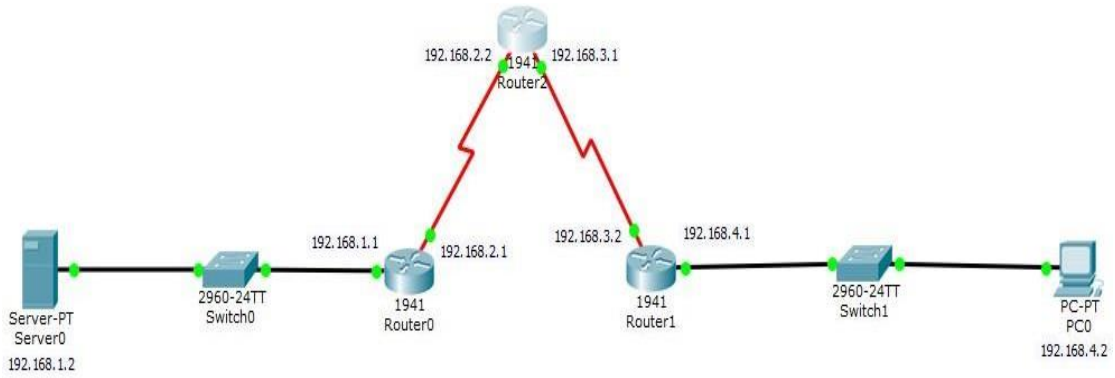
Next, we verify whether the ipv6 protocol works by pinging server from any of the PC (it must be successful)



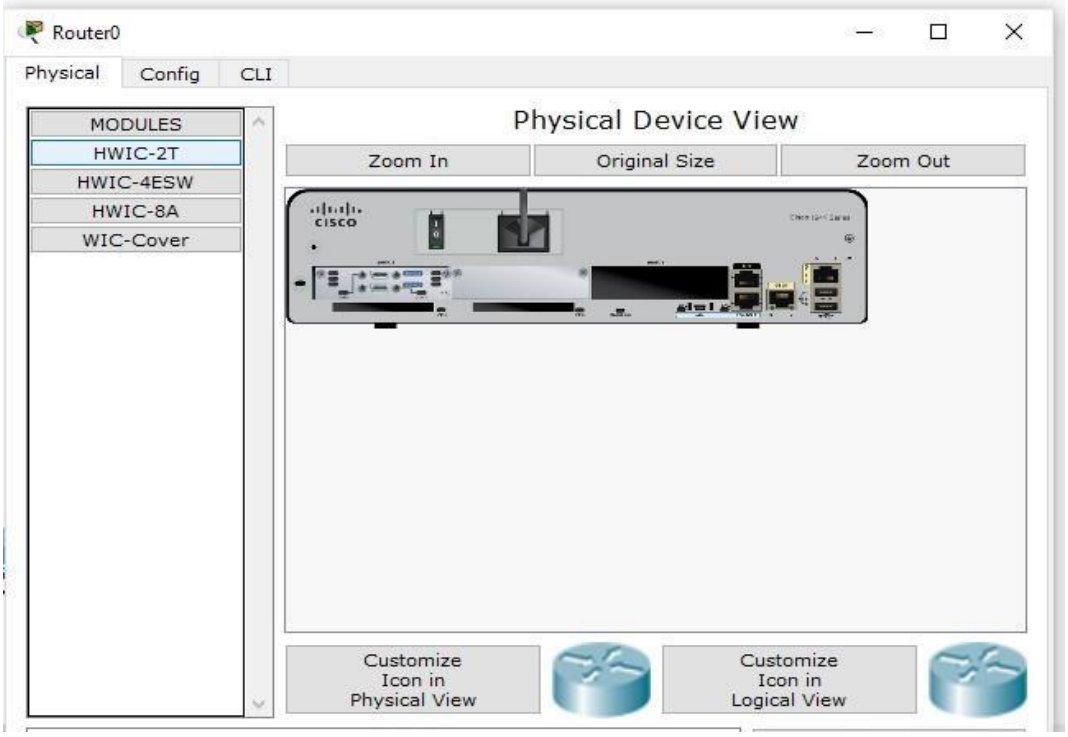
Hence the given ACLs have been applied and verified on host running on ipv6 protocol.

Practical 6
Configuring a Zone-Based policy Firewall (ZPF)

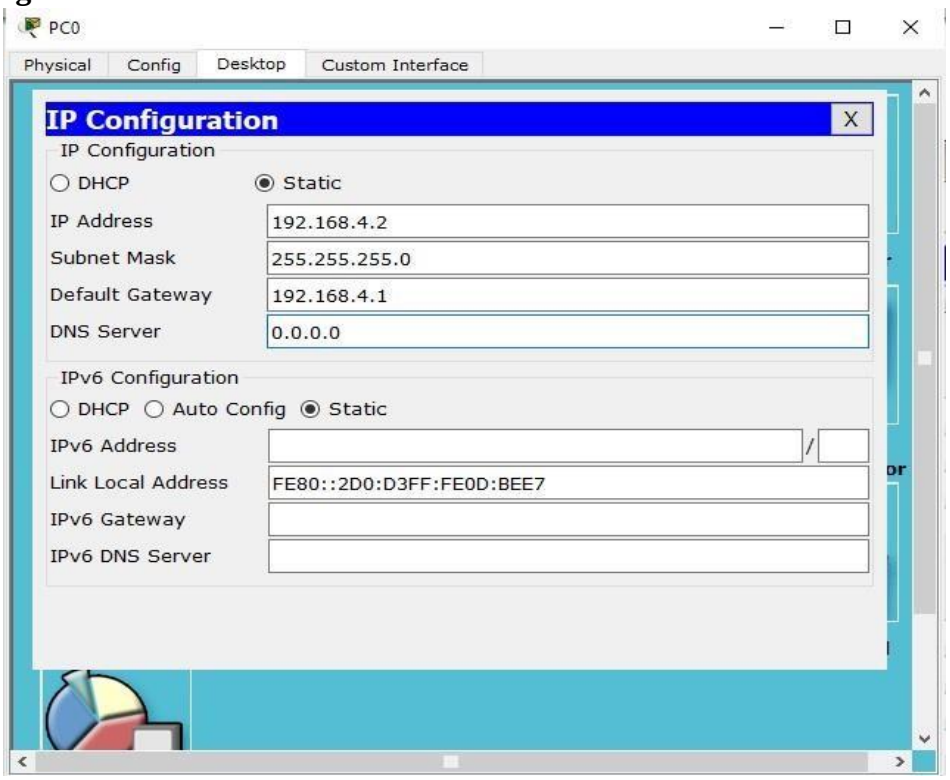
Topology



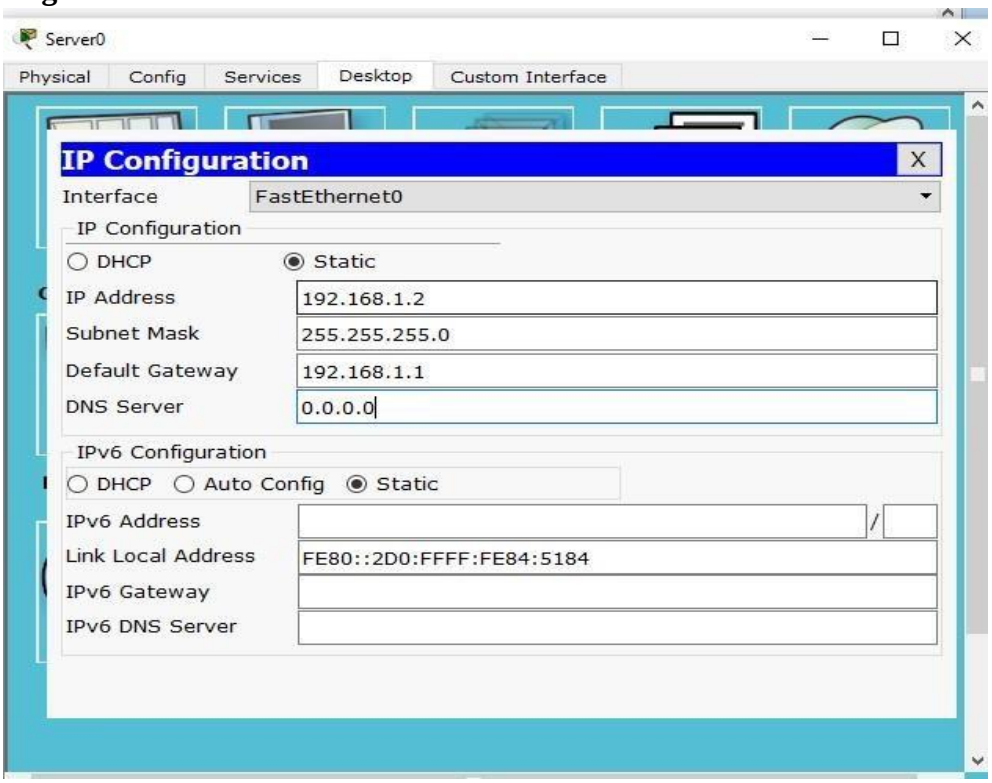
Serial Interface must be added in each Router before configuring it The serial interface in each Router is added as follows



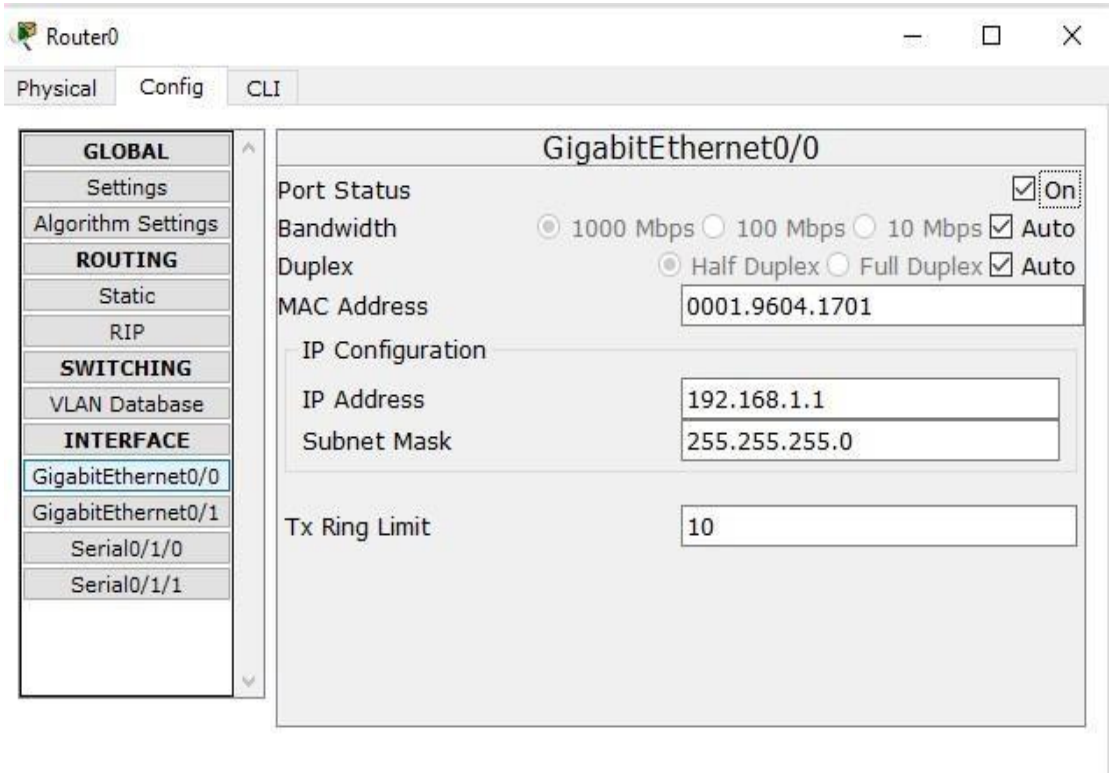
Configuring PC0

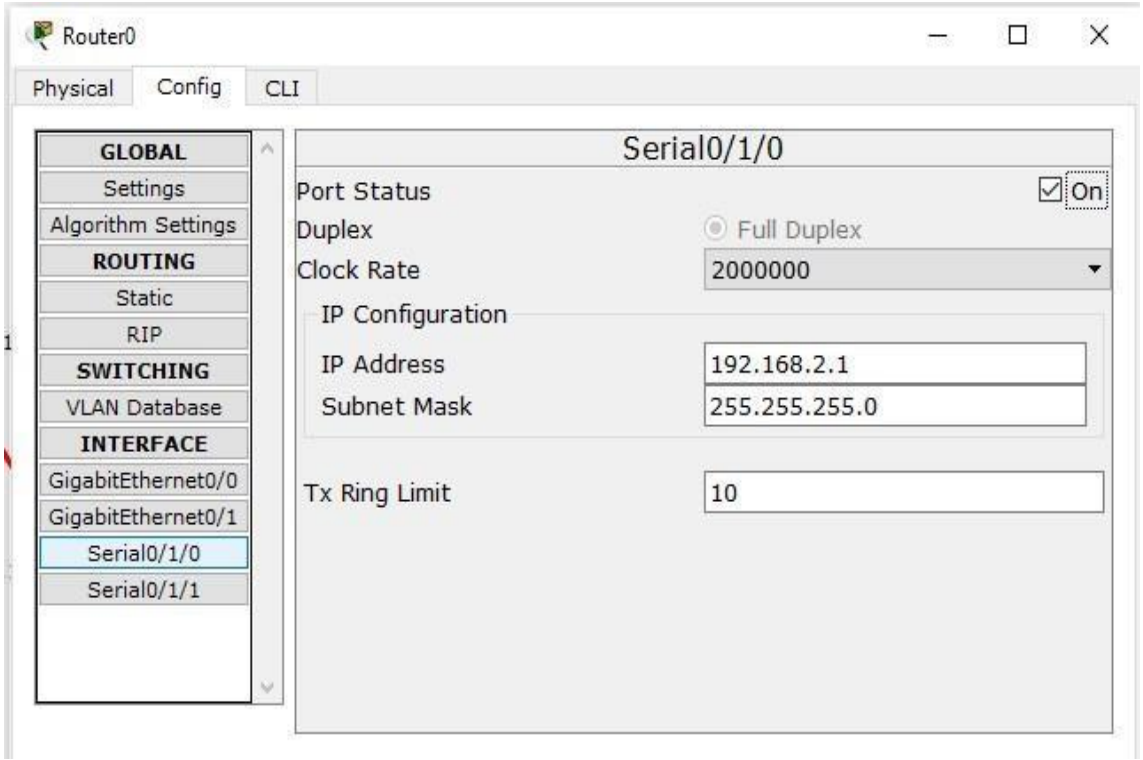


Configuring Server0

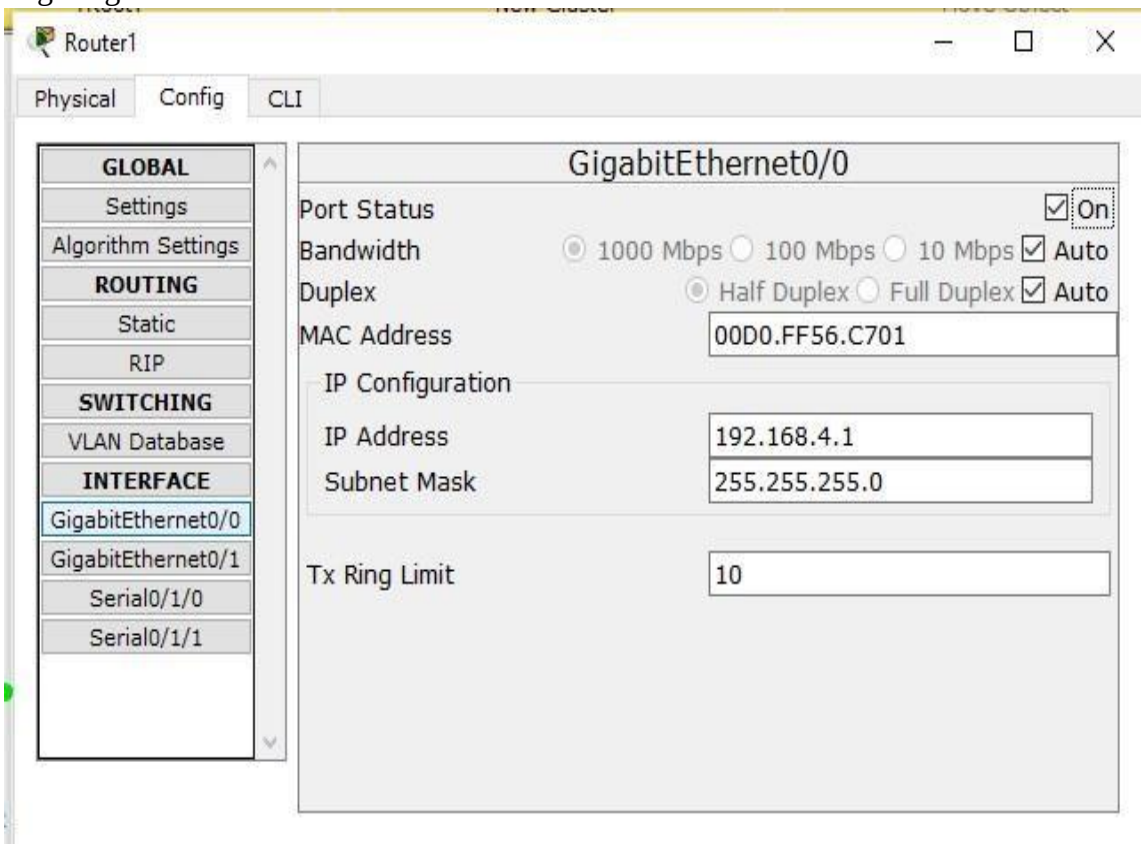


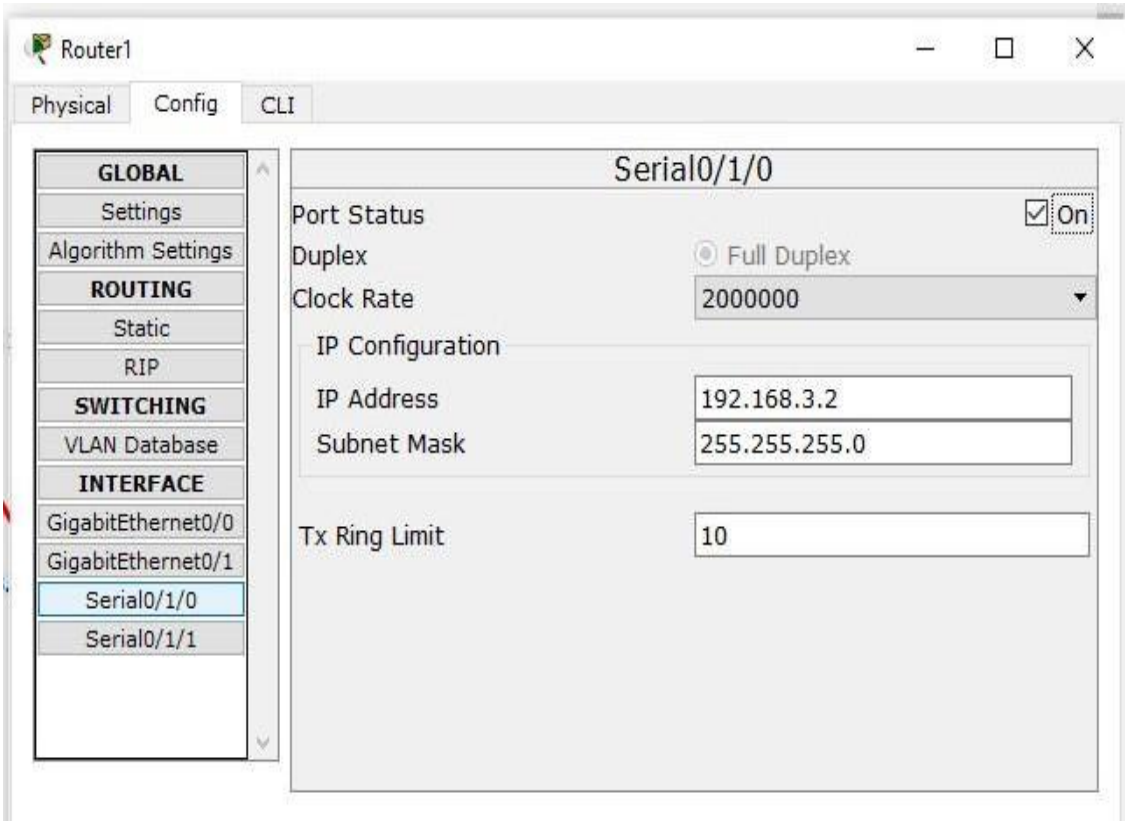
Configuring Router0



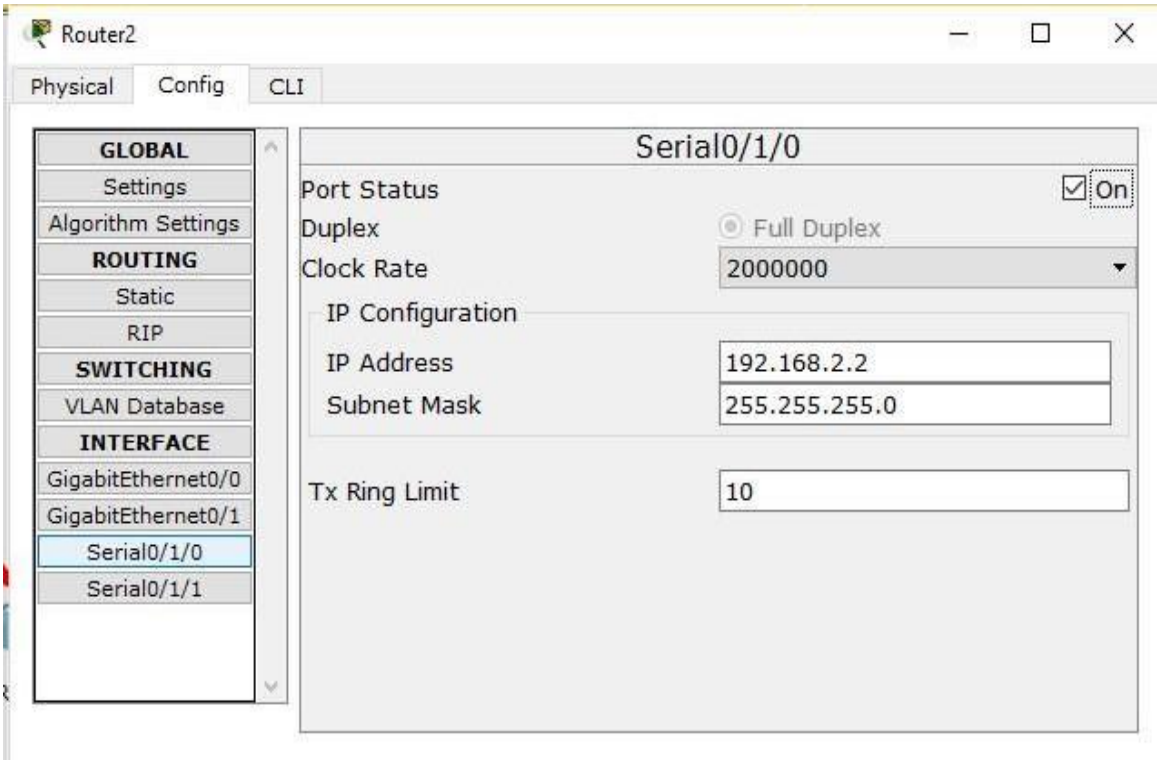


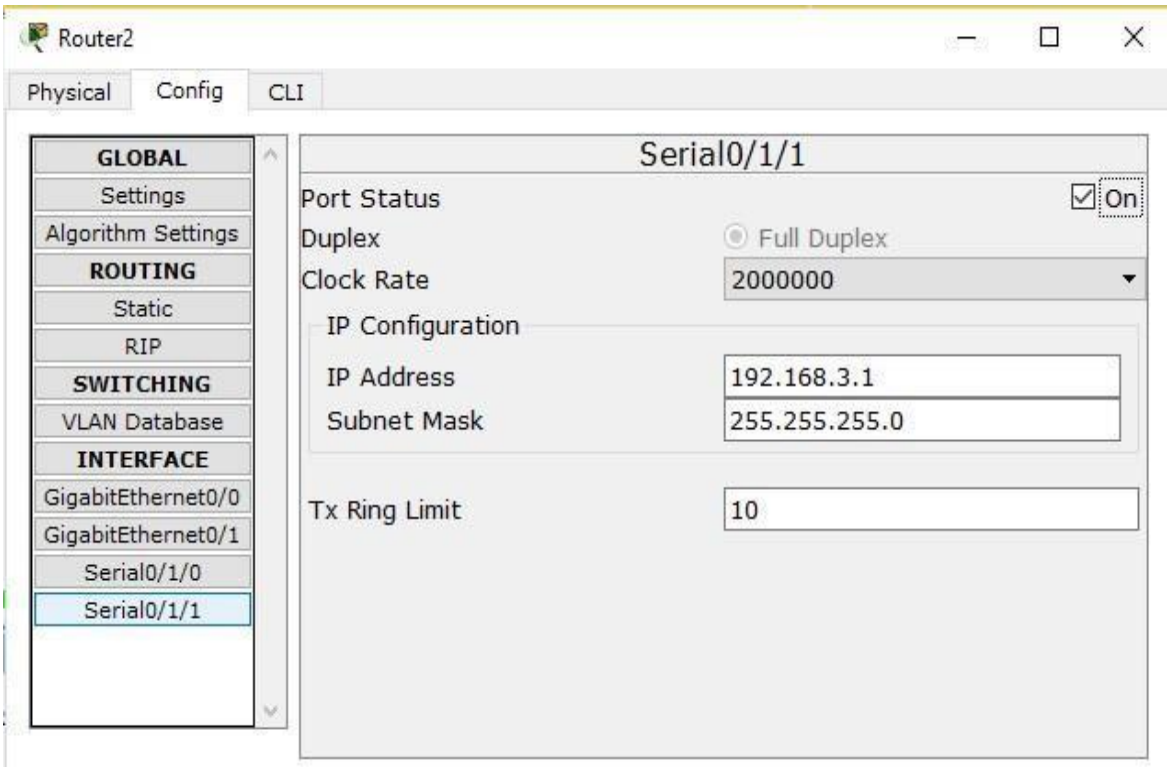
Configuring Router1





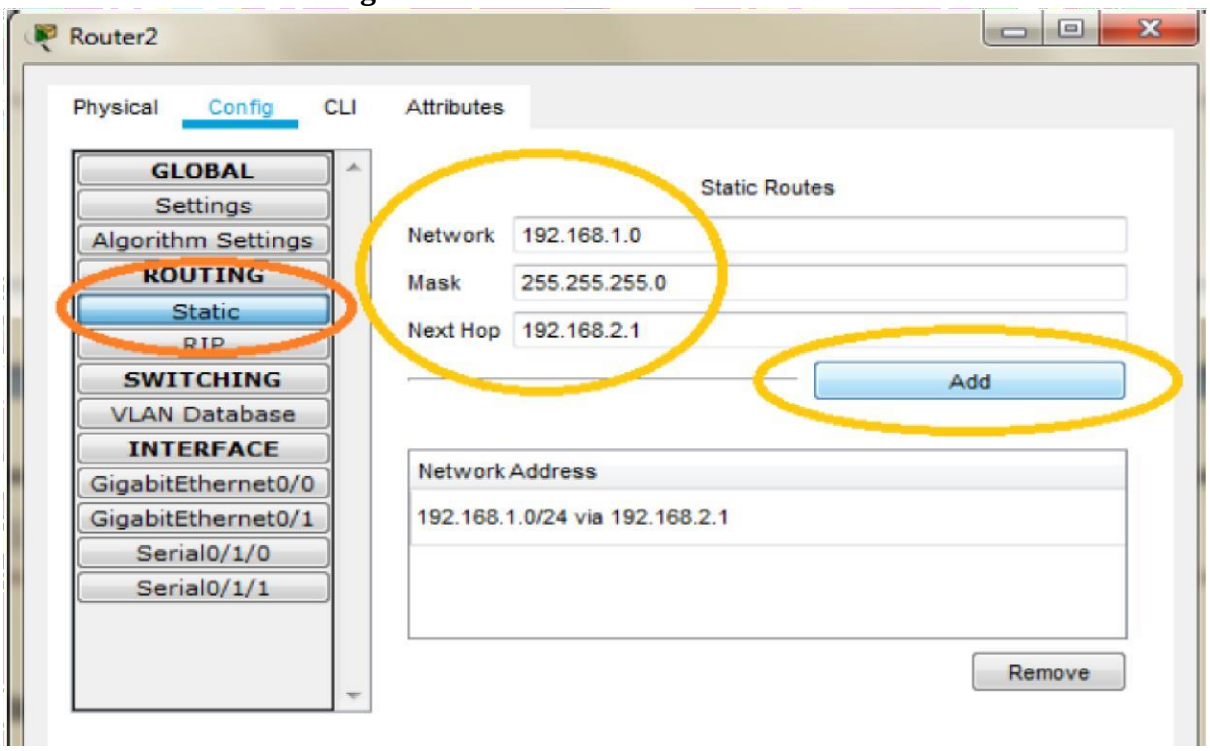
Configuring Router2

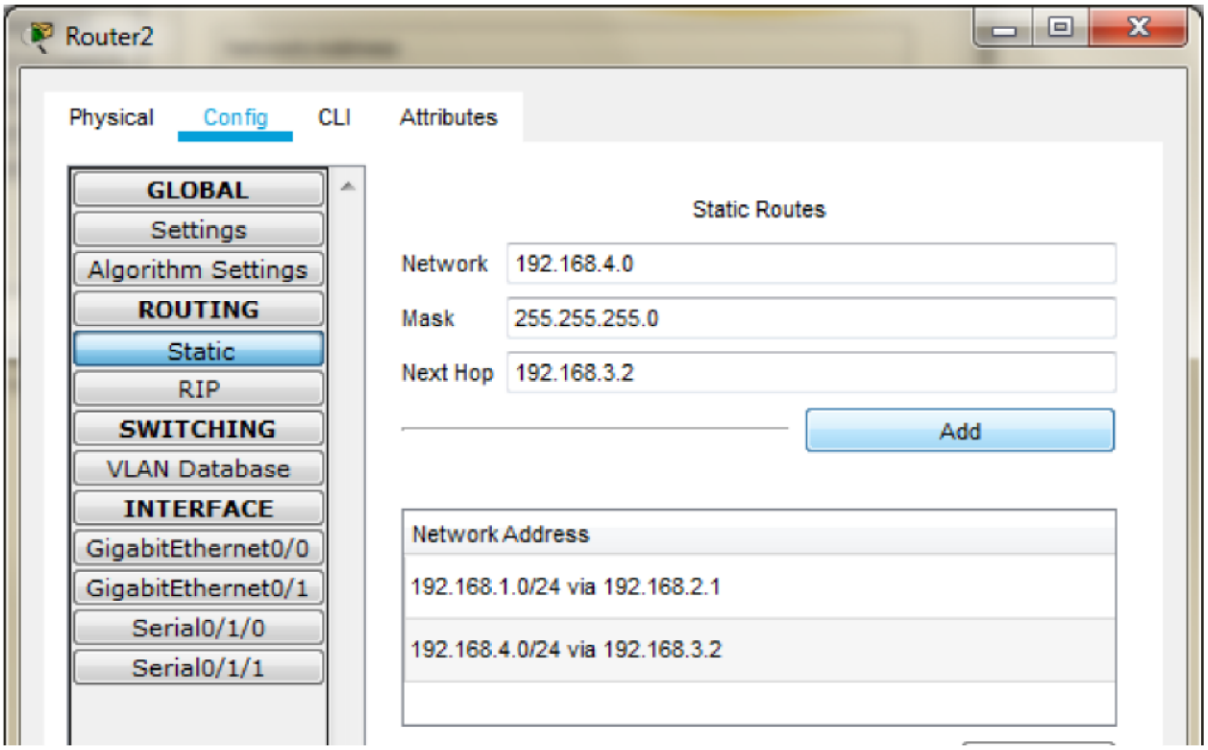




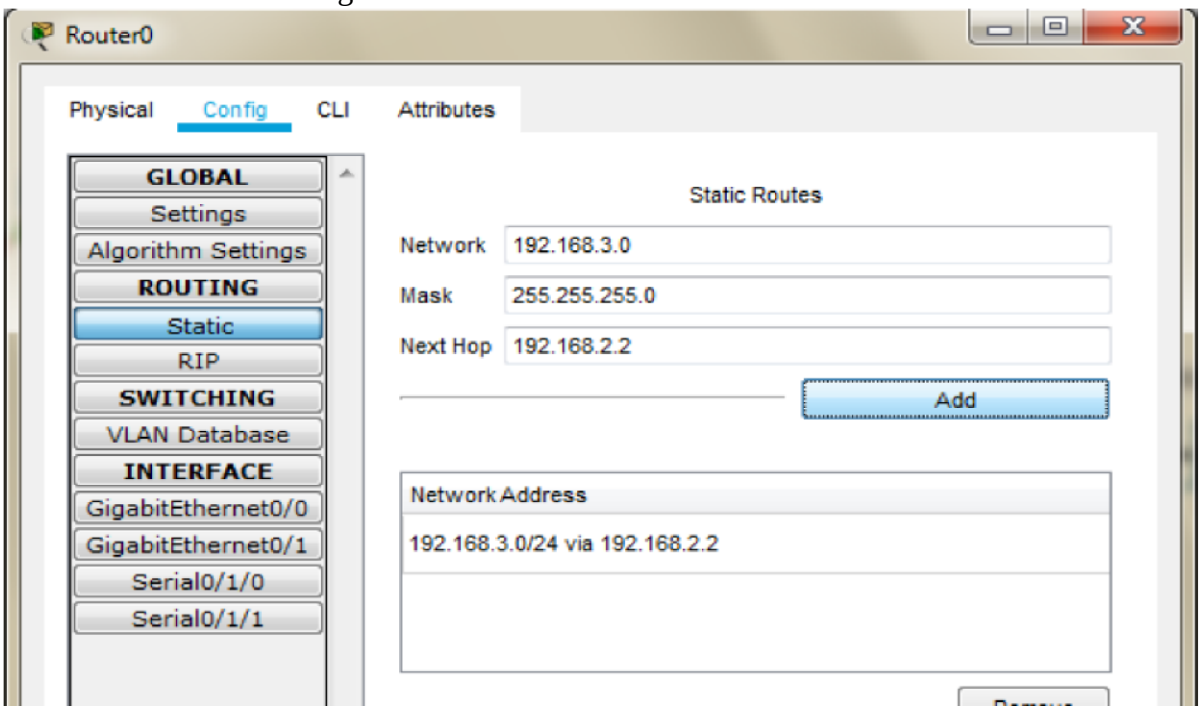
Part 1: Static Routing
Static Routing is done using the following procedure for each Router

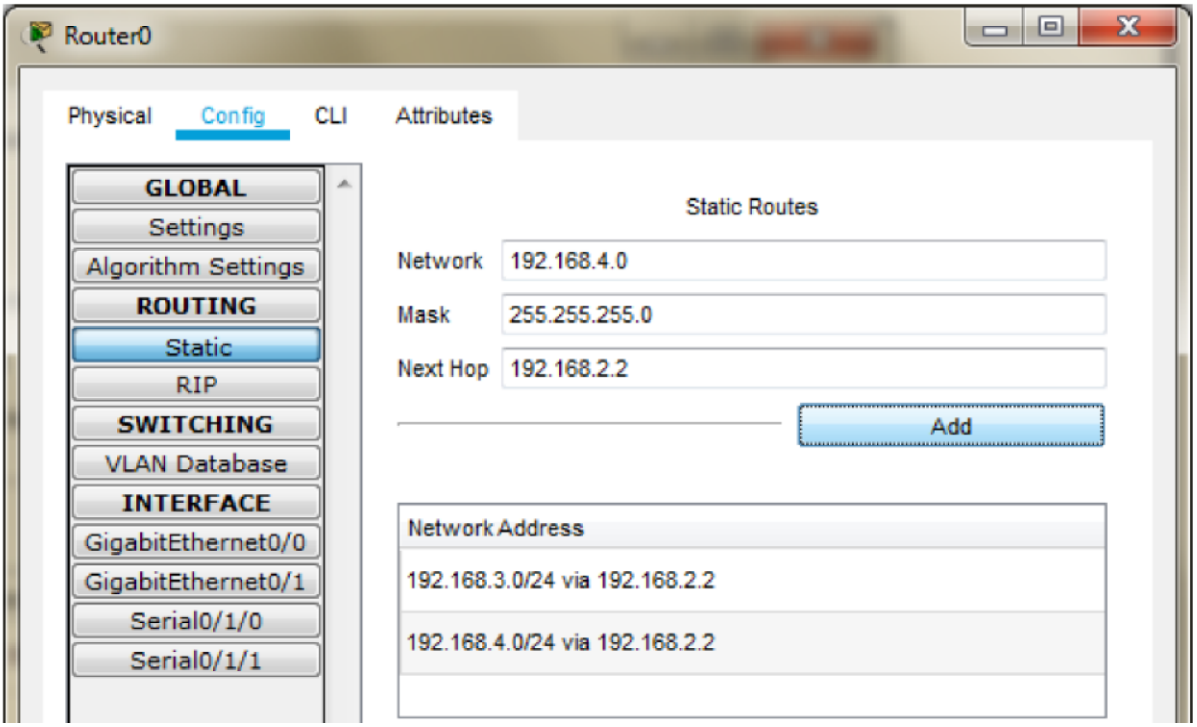
Router 2: Add the following Routes in the Static mode



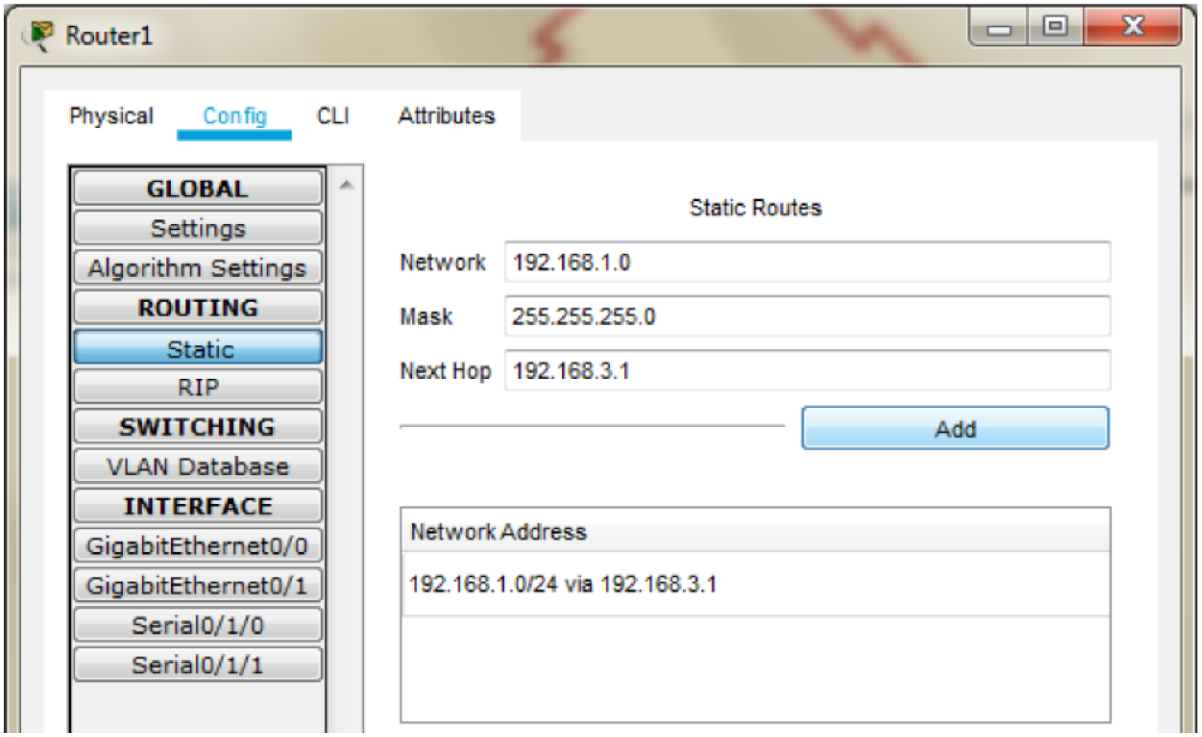


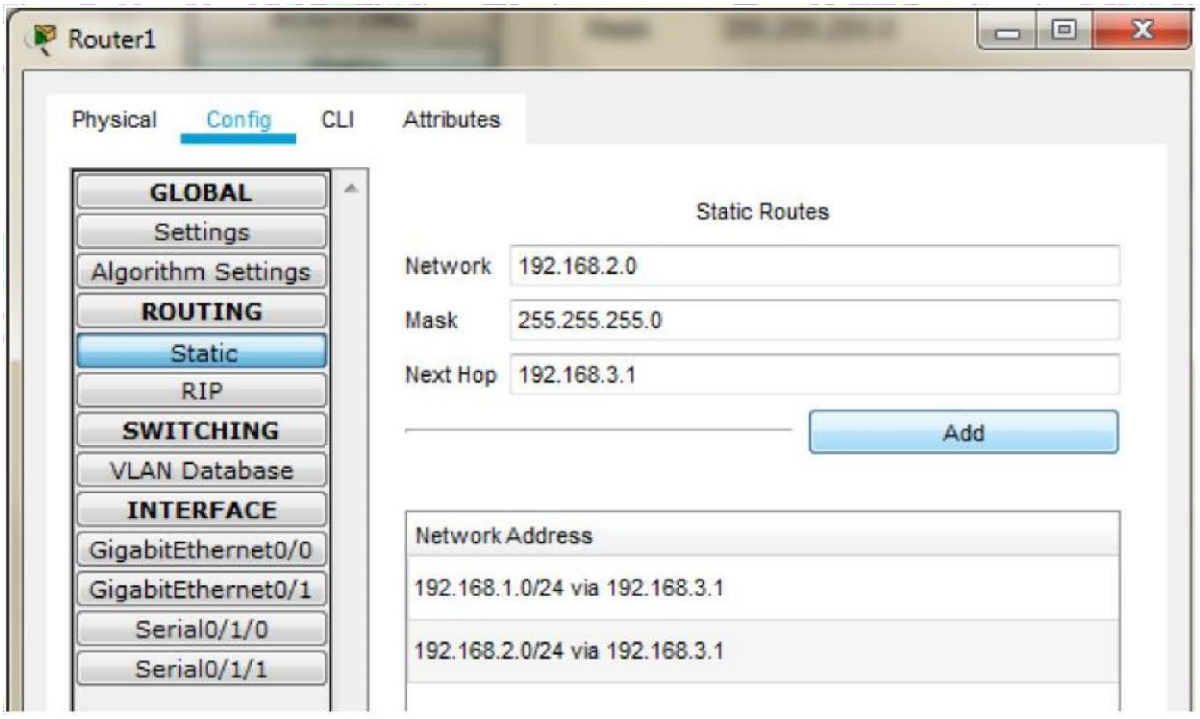
Router 0: Add the following Routes in the Static mode



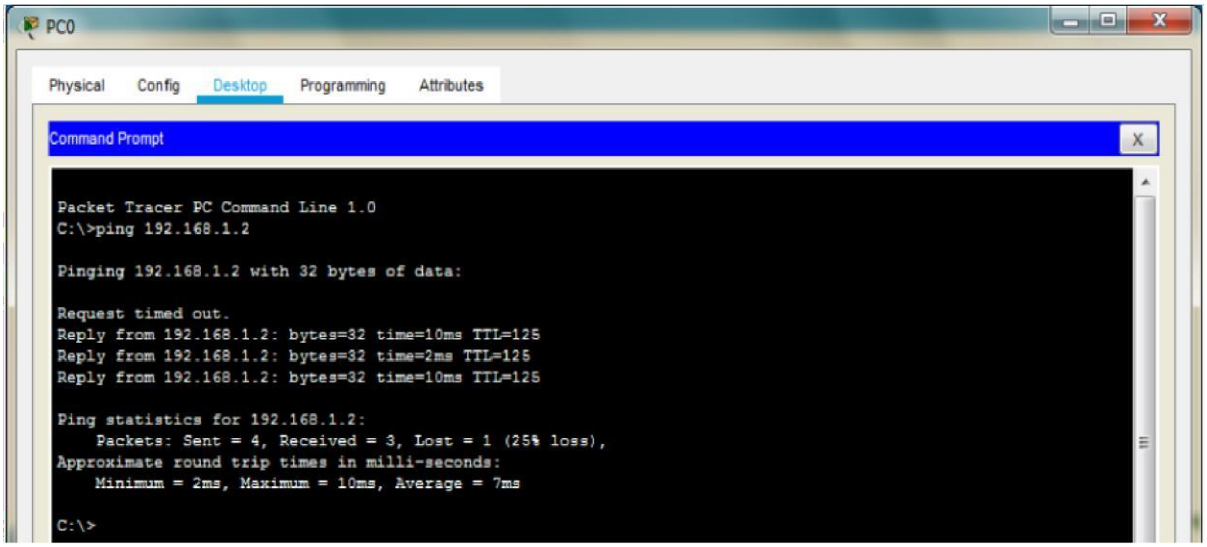


Router 1: Add the following Routes in the Static mode





Now we check the connectivity by pinging the Server from the PC



Part 2: Configuring SSH on Router 2
Type the following commands in the CLI mode of Router2

Router>en

Router#conf t

Router(config)#ip domain-name dalmia.com

Router(config)#hostname R2

R2(config)#crypto key generate rsa

The name for the keys will be: R2.dalmia.com

Choose the size of the key modulus in the range of 360 to 2048 for your General Purpose Keys. Choosing a key modulus greater than 512 may take a few minutes.

How many bits in the modulus [512]: 1024

% Generating 1024 bit RSA keys, keys will be non-exportable...[OK]

R2(config)#line vty 0 4

*Mar 2 0:52:50.777: %SSH-5-ENABLED: SSH 1.99 has been enabled

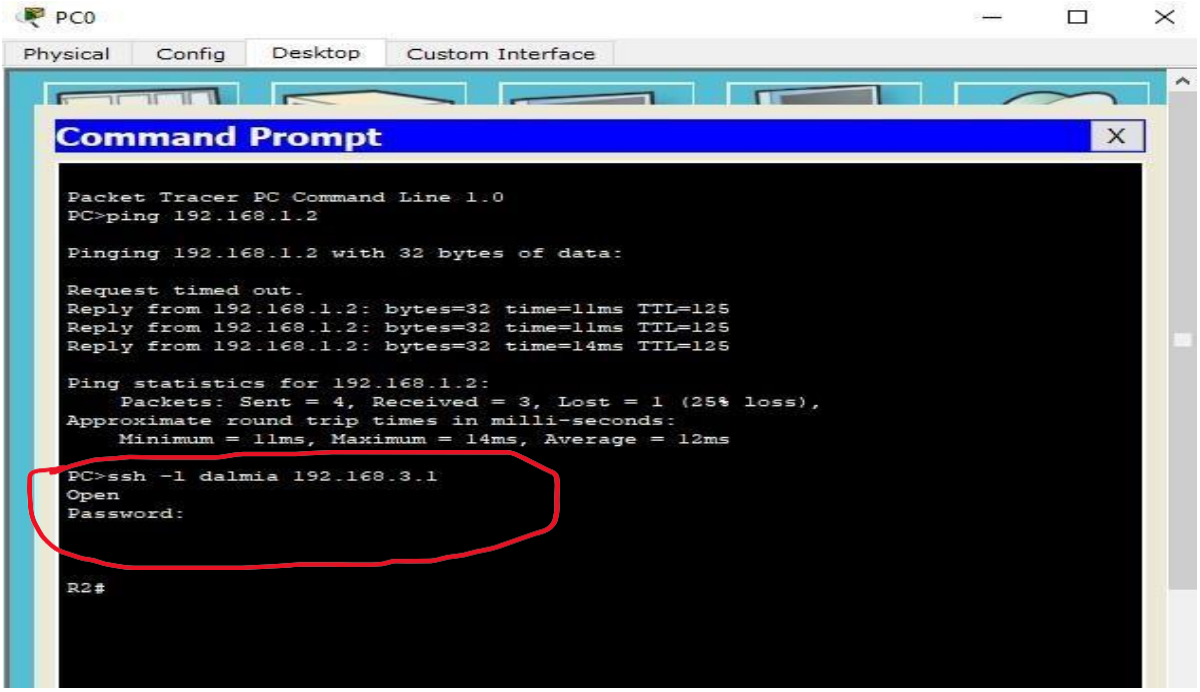
R2(config-line)#transport input ssh

R2(config-line)#login local

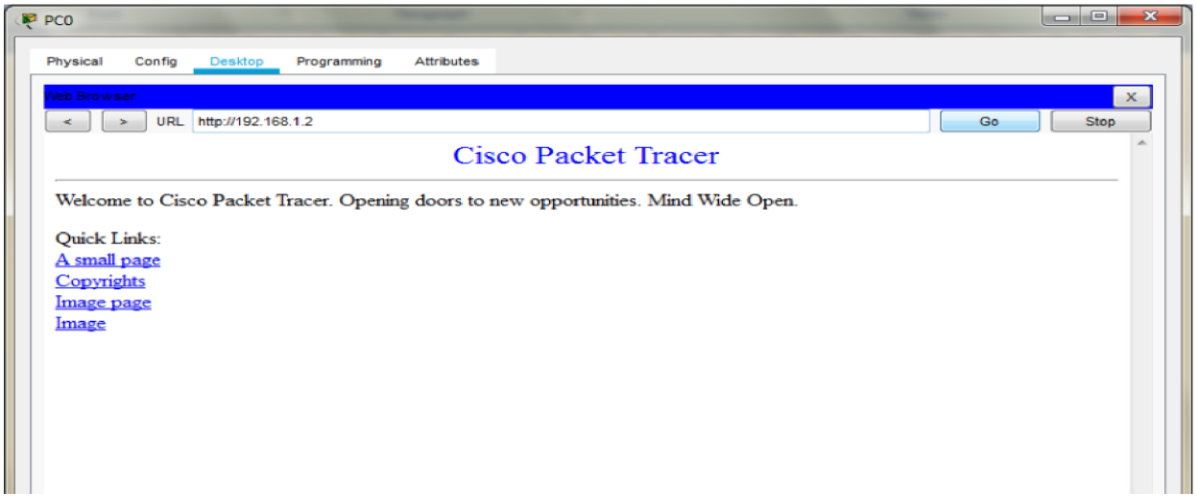
R2(config-line)#exit

R2(config)#username dalmia privilege 15 password cisco

Now we verify the SSH using PC as follows



Next, we access the web services of the Server using the web browser of PC using the following: -



Part 3: Create the Firewall Zones on Router1

Type the following commands in the CLI mode of Router1

Router>en

Router#conf t

Router(config)#license boot module c1900 technology-package securityk9

ACCEPT? [yes/no]: y

Router(config)#exit

Router#copy run start

Press enter when prompted

Router#reload

Continue with configuration dialog? [yes/no]: n

Router>en

Router#conf t

Router(config)#zone security in-zone

Router(config-sec-zone)#exit

Router(config)#zone security out-zone

Router(config-sec-zone)#exit

Router(config)#access-list 101 permit ip 192.168.4.0 0.0.0.255 any

Router(config)#class-map type inspect match-all in-map

Router(config-cmap)#match access-group 101

Router(config-cmap)#exit

Router(config)#policy-map type inspect in-out

Router(config-pmap)#class type inspect in-map

Router(config-pmap-c)#inspect

Router(config-pmap-c)#exit

Router(config-pmap)#exit

Router(config)#

Router(config)#zone-pair security in-out-zone source in-zone destination out-zone

Router(config-sec-zone-pair)#service-policy type inspect in-out

Router(config-sec-zone-pair)#exit

Router(config)#

Router(config)#int G0/0

Router(config-if)#zone-member security in-zone

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

Router(config)#int Se0/1/1
Router(config-if)#zone-member security out-zone

Router(config-if)#exit

Router(config)#exit

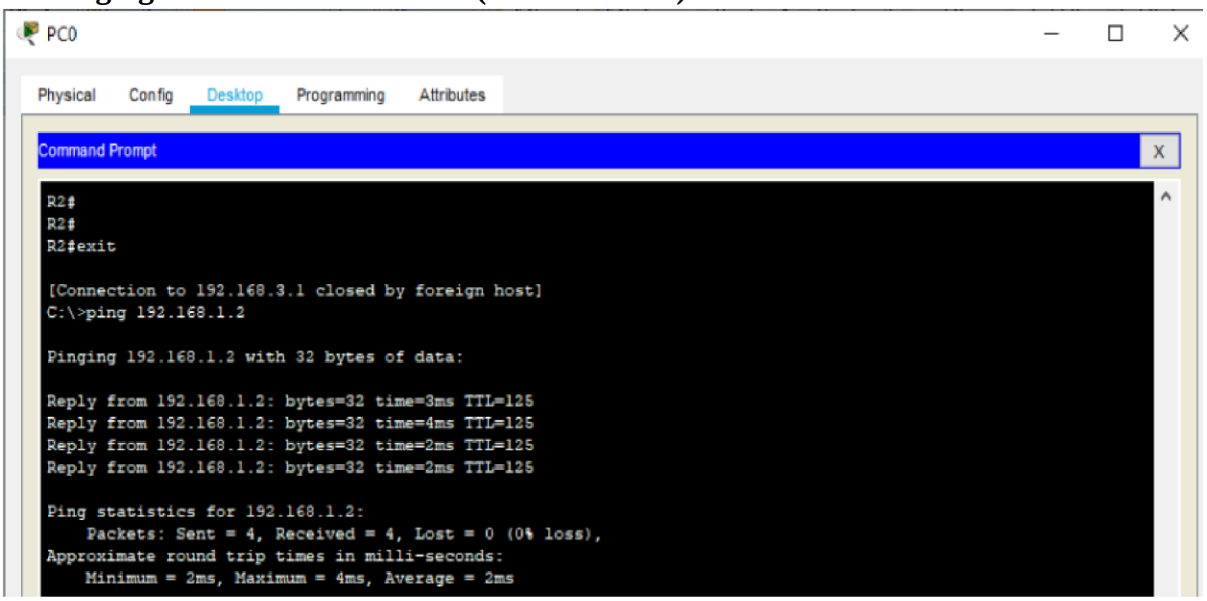
Router#copy running-config startup-config Destination
filename [startup-config]?

Building configuration...

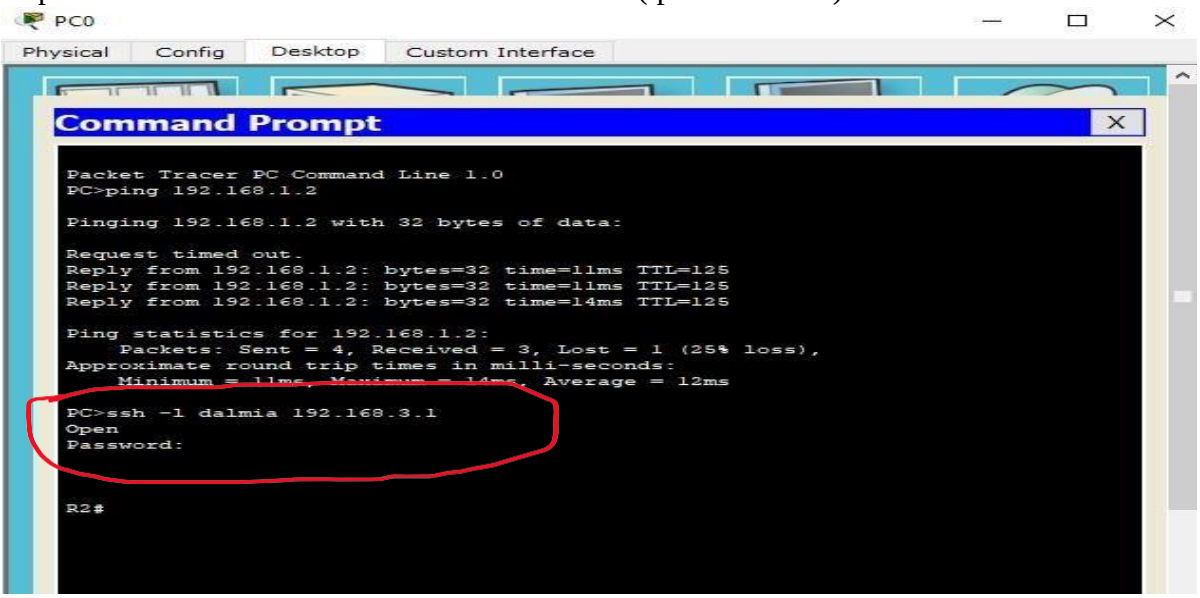
[OK]

Part 4: Testing the Firewall Functionality (from in-zone to out-zone) by the following steps

Step 1: Pinging SERVER from the PC (it will succeed)



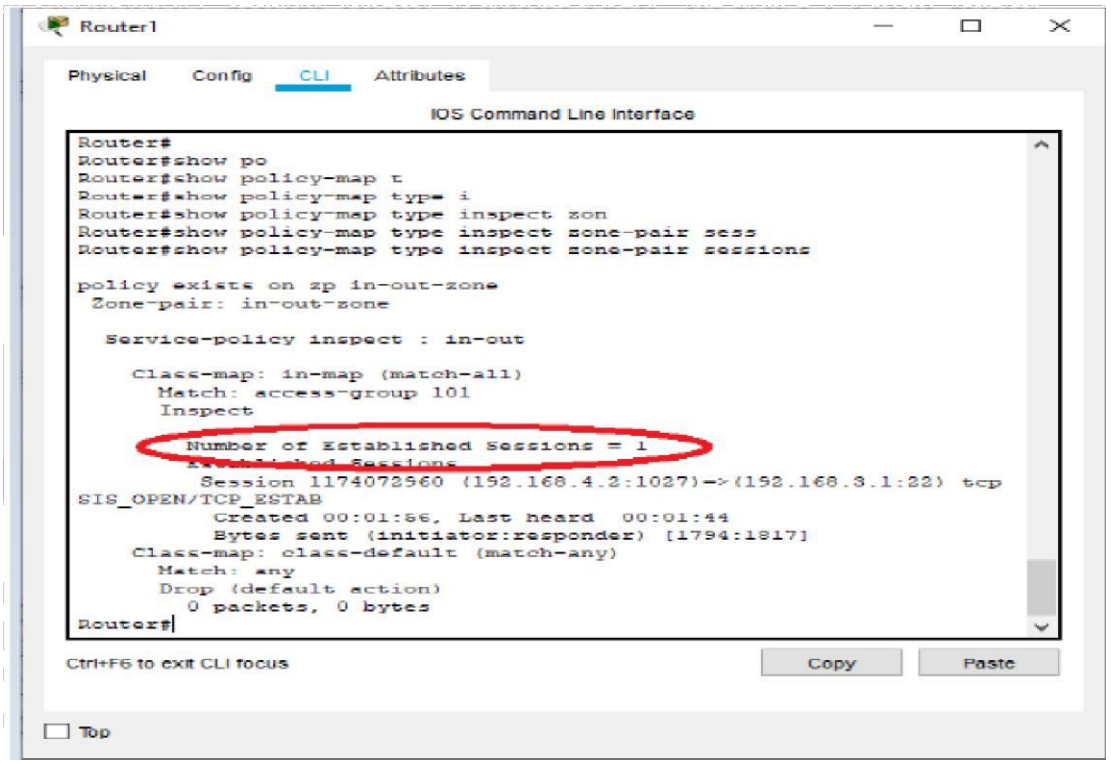
Step 2: Start an SSH session from PC to Router 2 (ip 192.168.1.2)



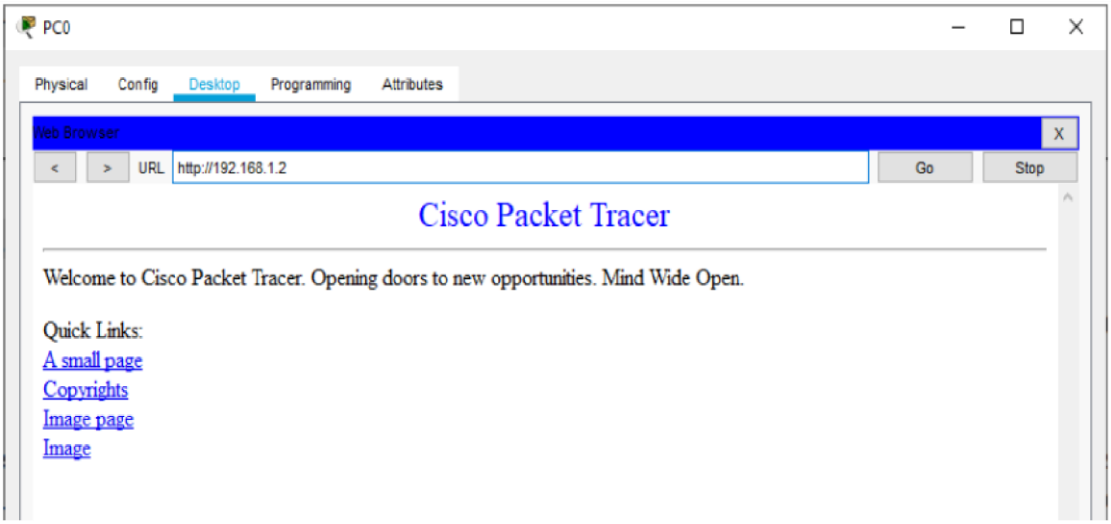
As seen above the session becomes active and we get access to Router2 (Do not exit and the session and continue to Step 3)

Step 3: Type the following command in the CLI mode of Router1

Router#show policy -map type inspect zone -pair sessions

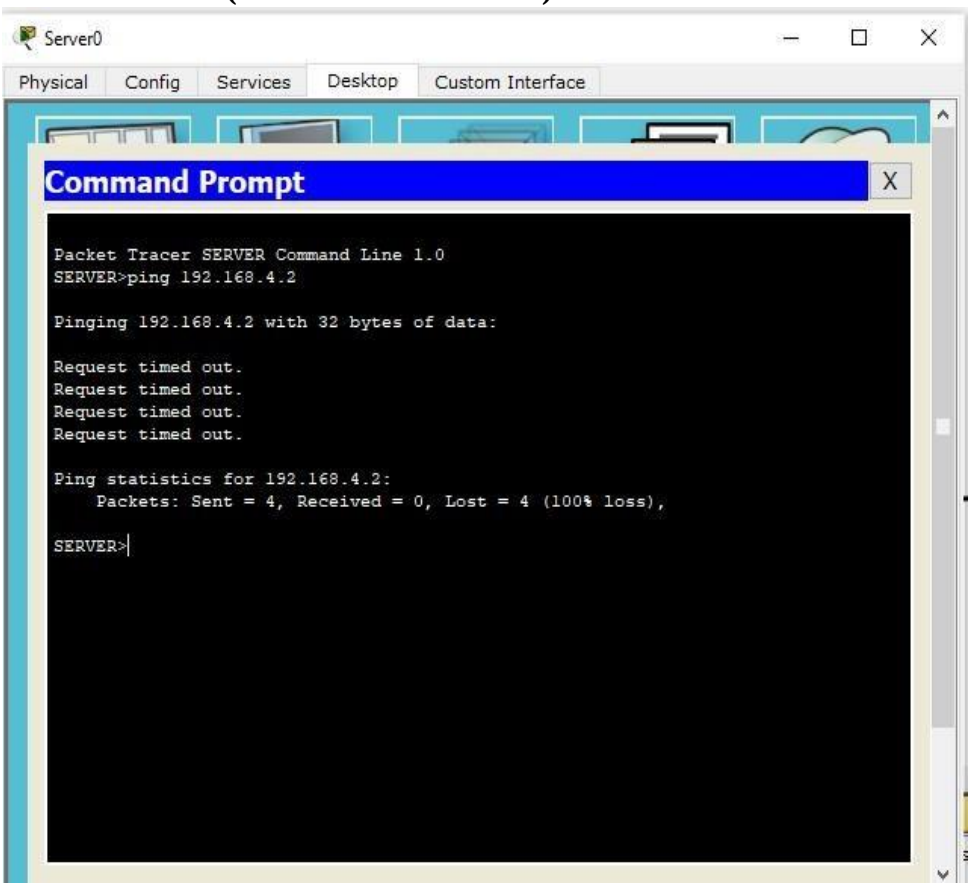


Step 4: We close the SSH connection and open the web browser and access the server address (192.168.1.2) and get the following



Part 5: Testing the Firewall Functionality (from out-zone to in-zone) by the following step: -

Ping PC0 from the SERVER (it will result in Failure)

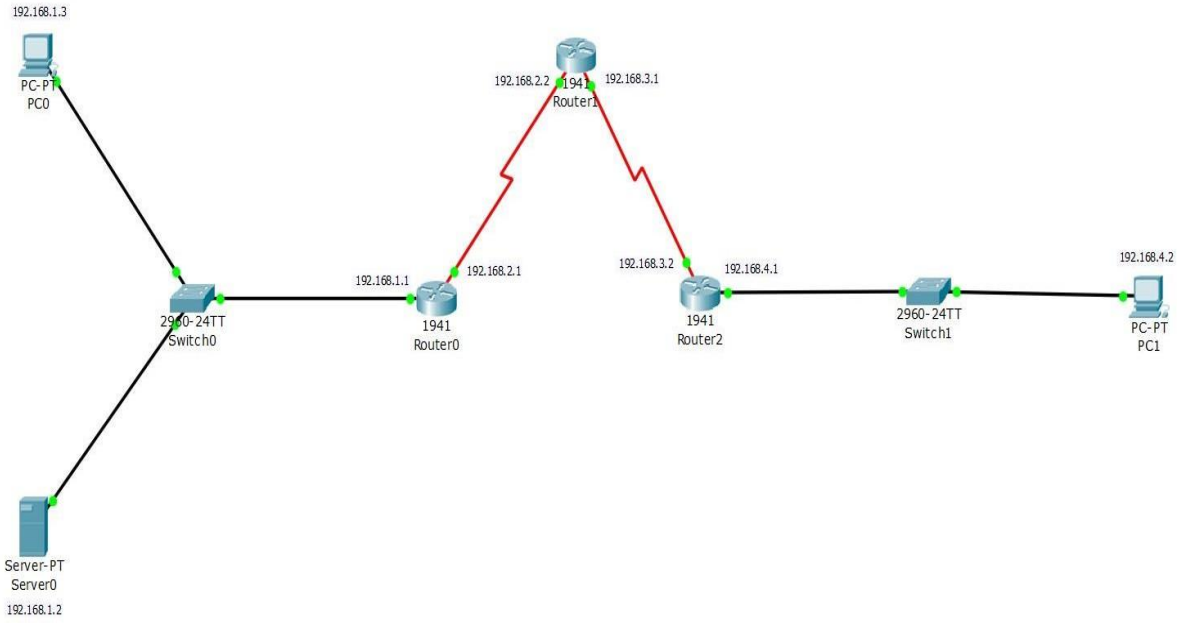


Hence the Firewall functionality has been verified.

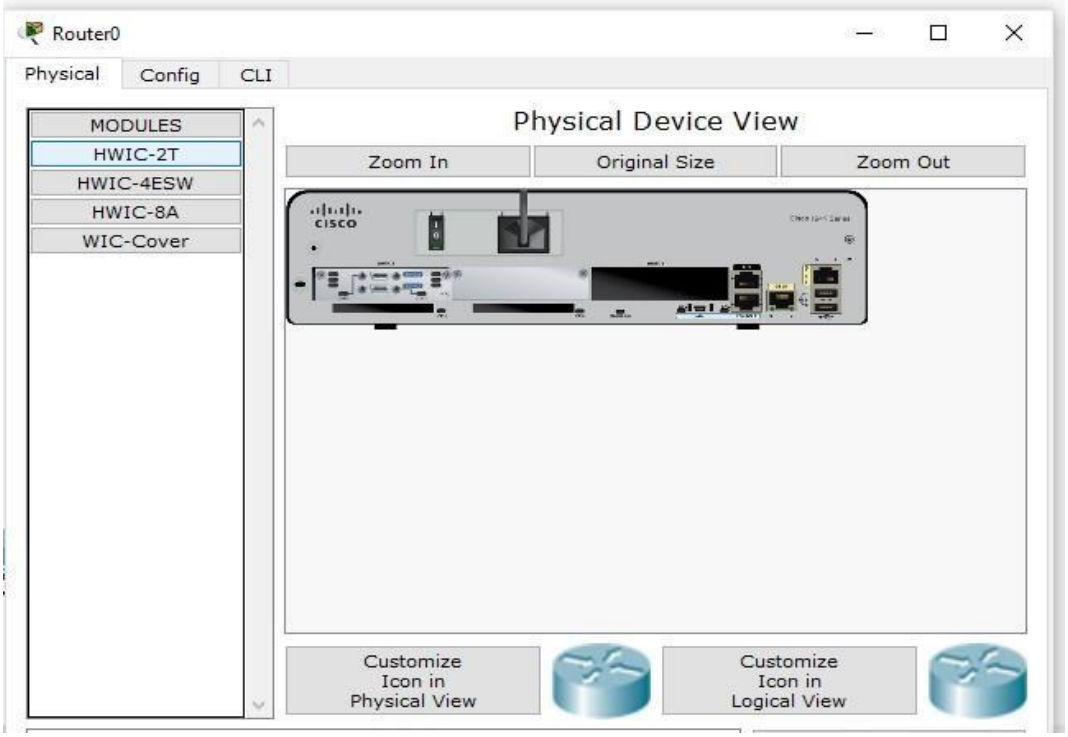
Practical 7

Configuring IOS Intrusion Prevention System (IPS) Using the CLI :
a) Enable IOS IPS b) Modify an IPS signature

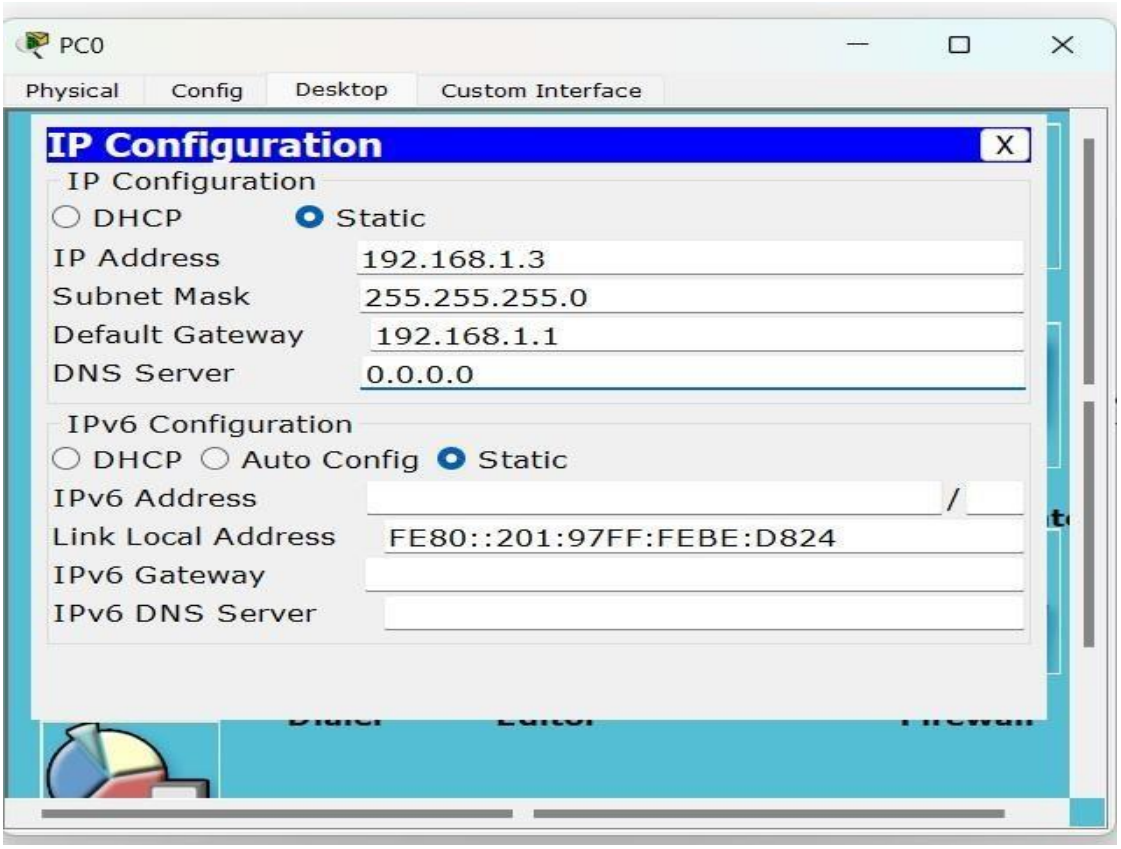
Topology



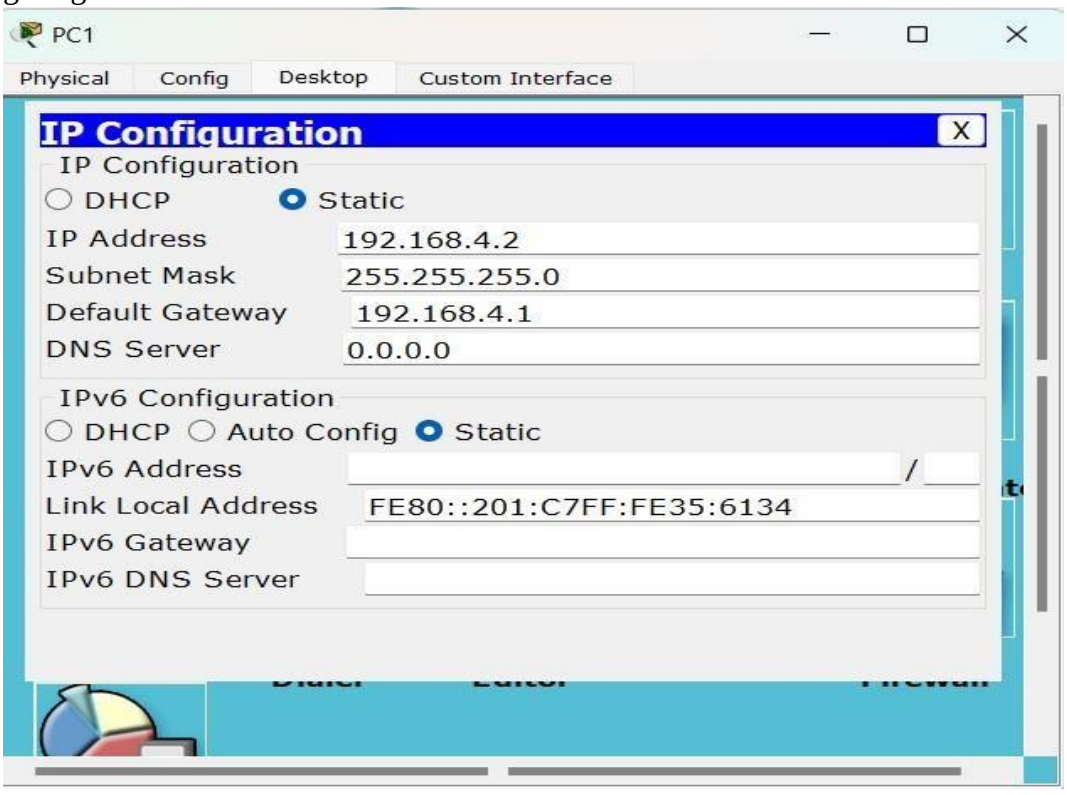
Serial Interface must be added in each Router before configuring it The serial interface in each Router is added as follows



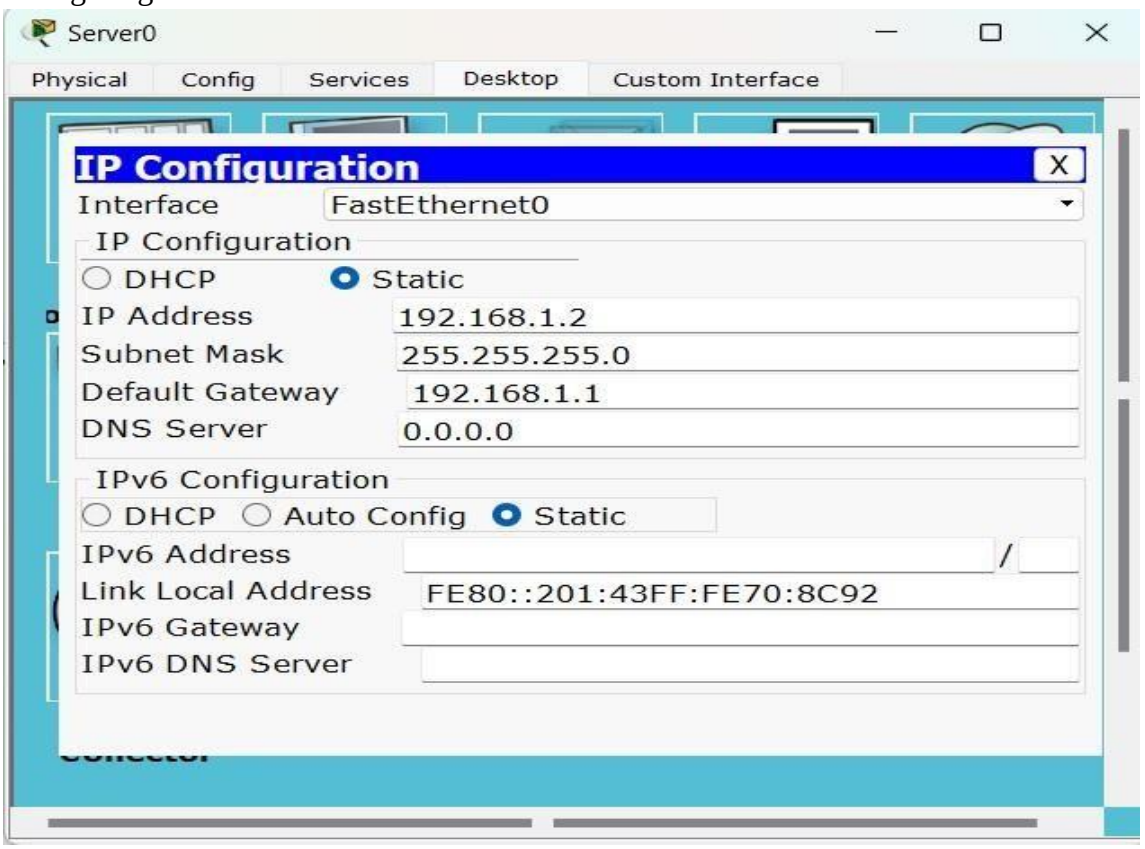
Configuring PC0



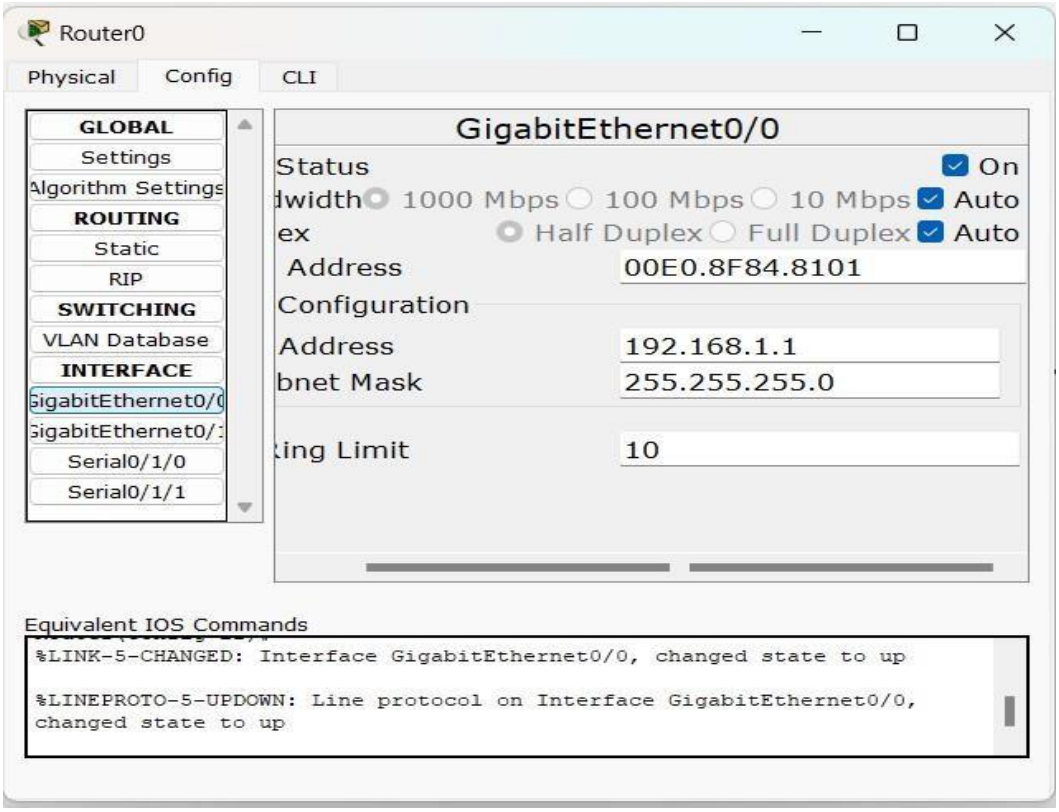
Configuring PC1

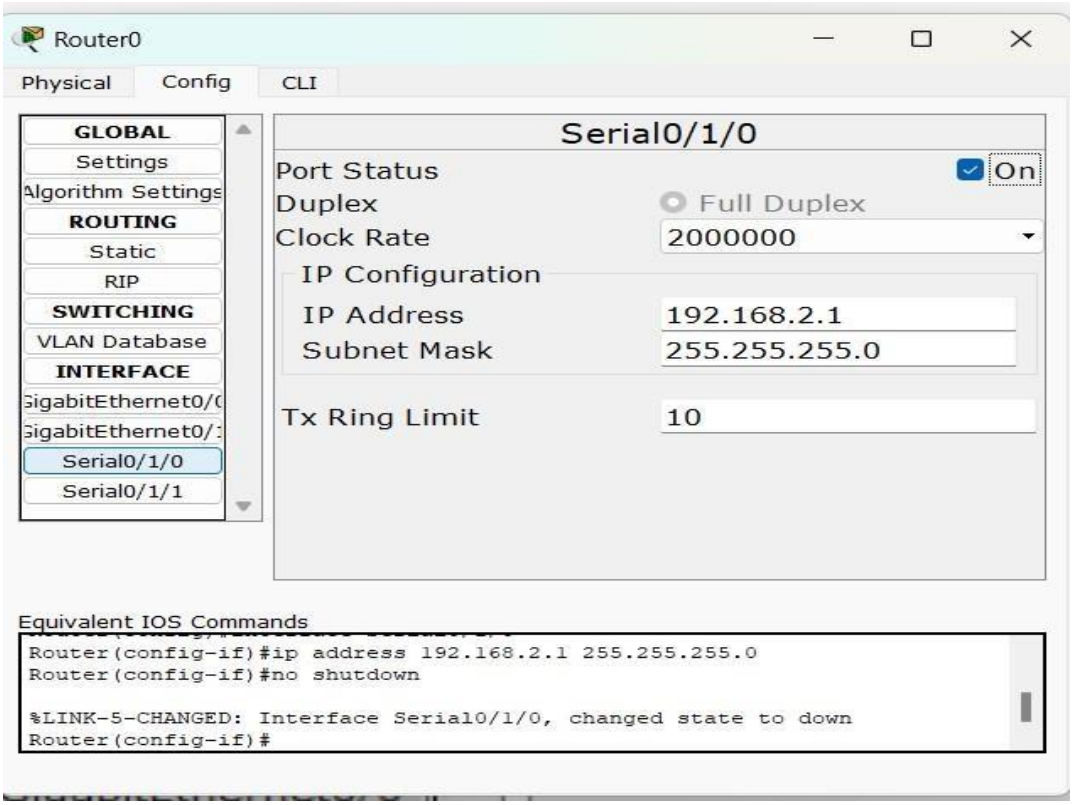


Configuring Server0

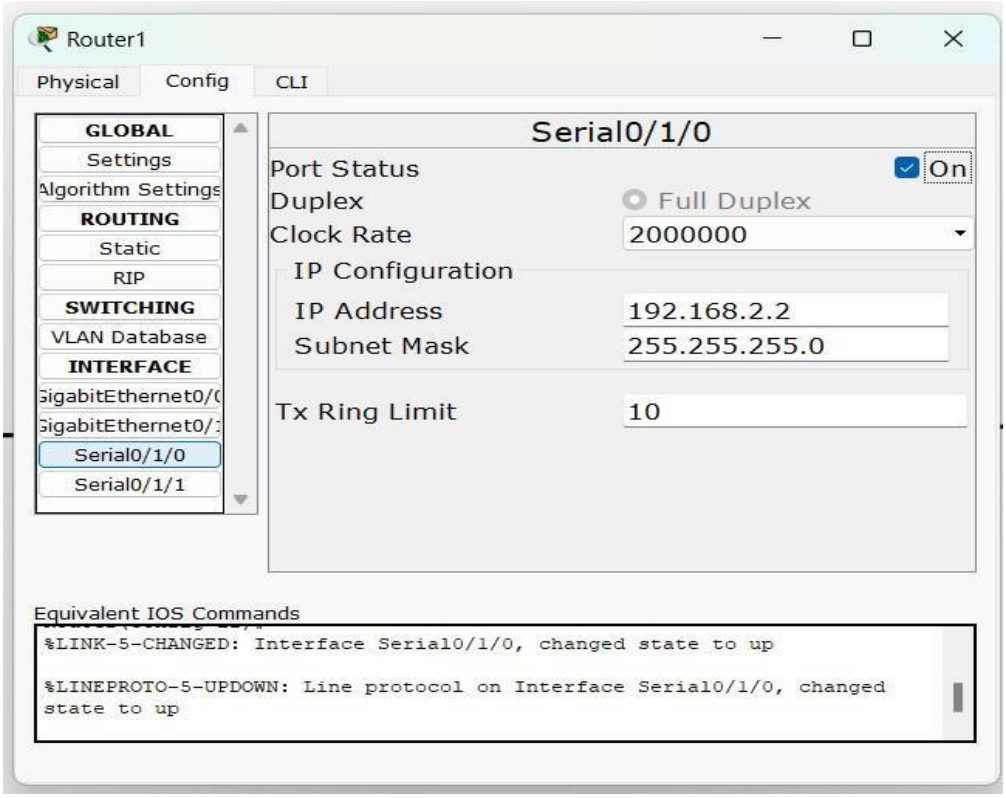


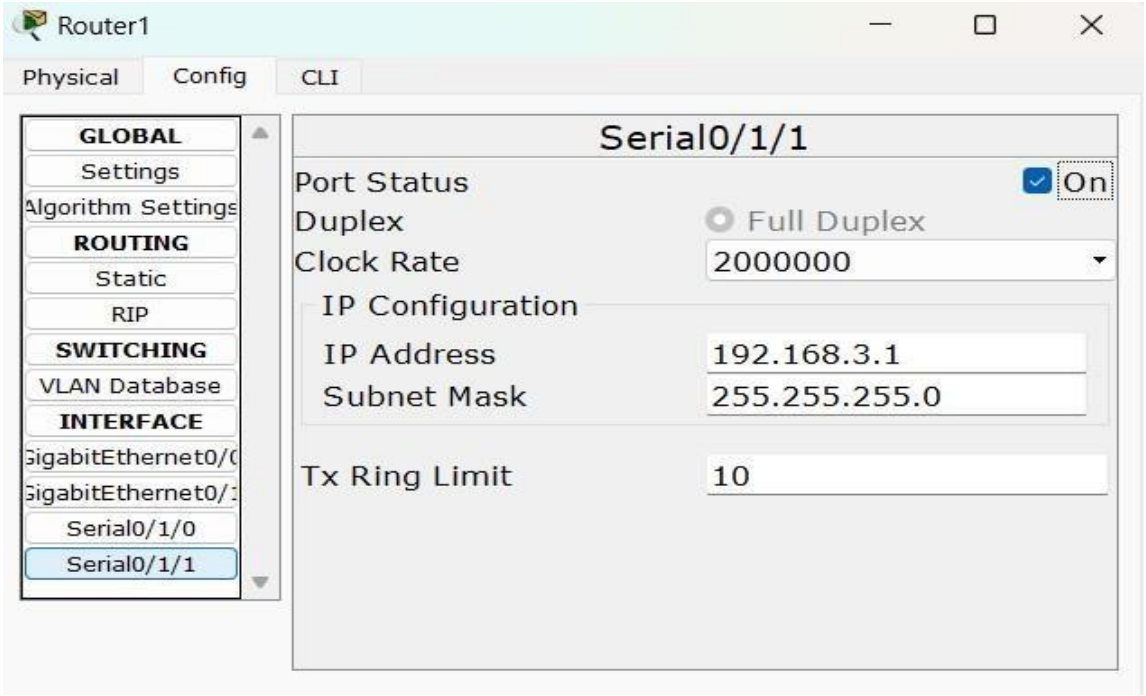
Configuring Router0



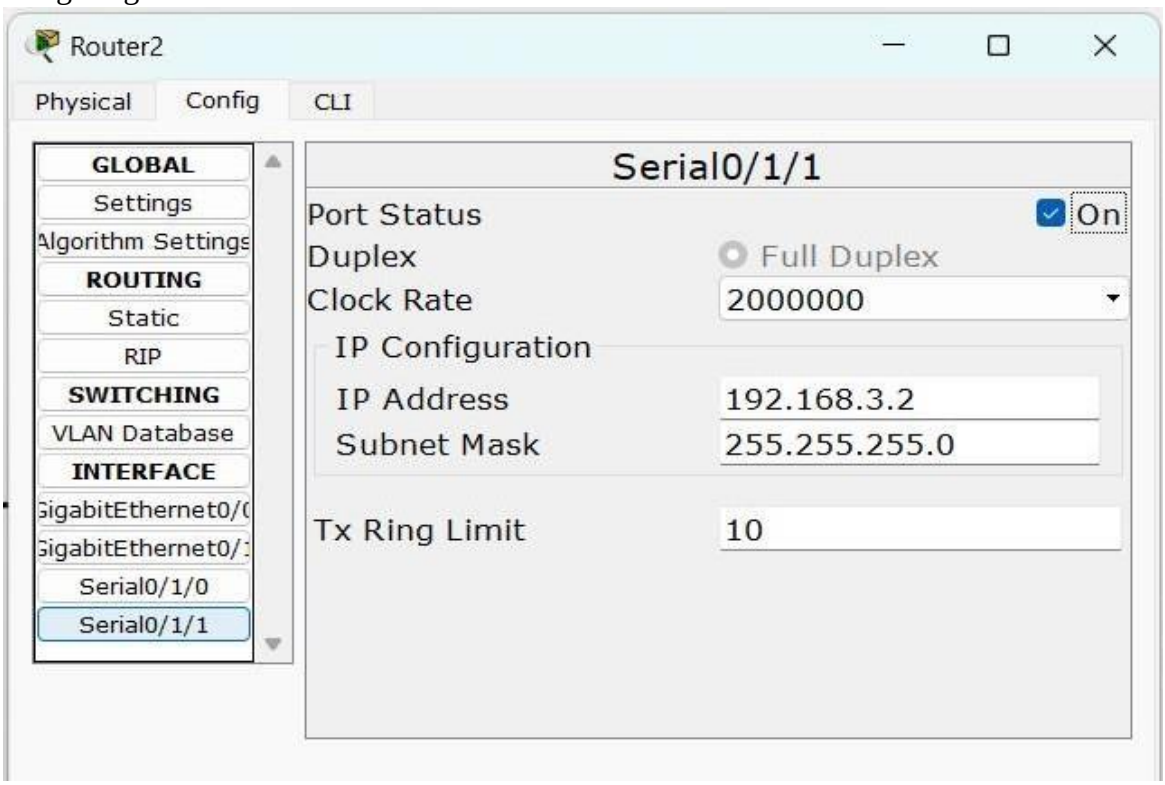


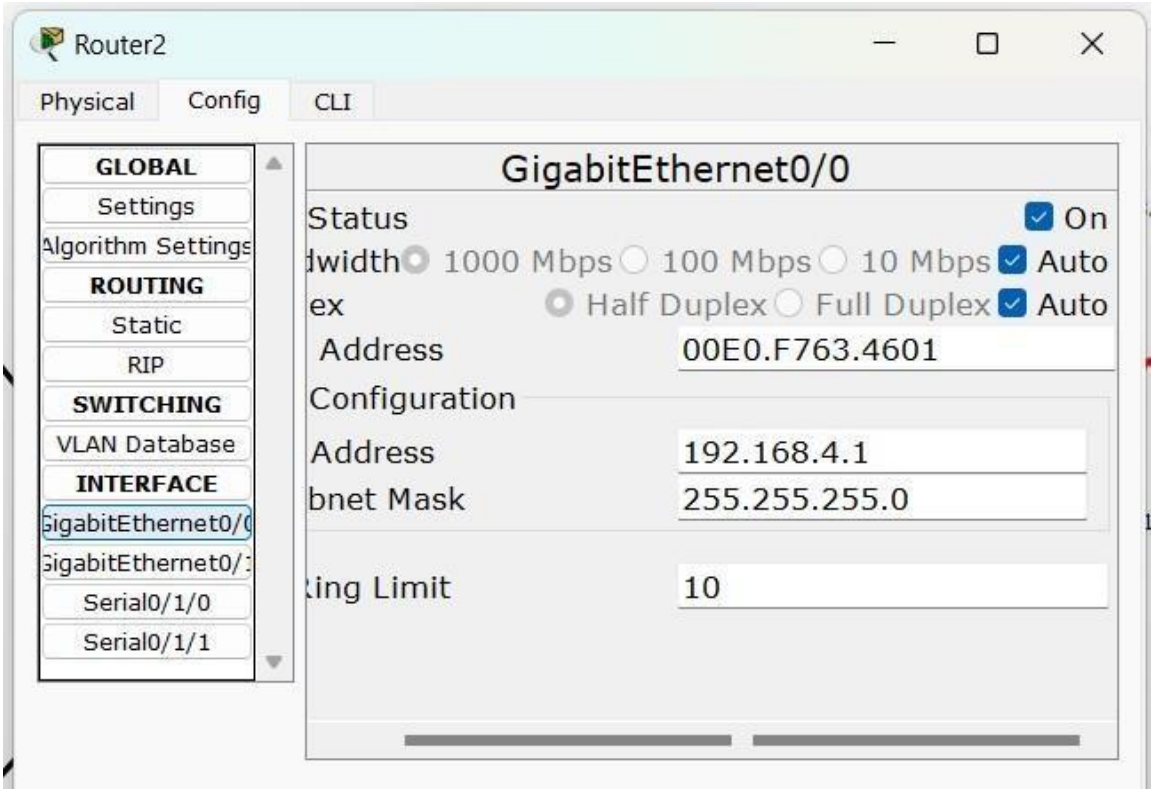
Configuring Router1





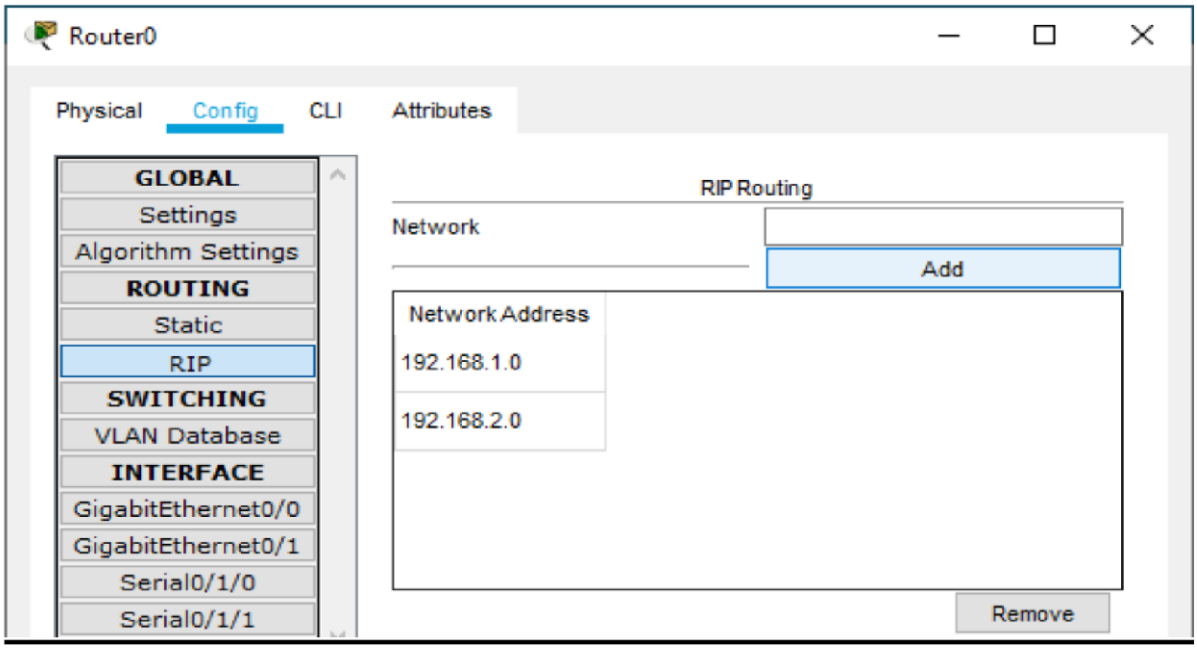
Configuring Router2



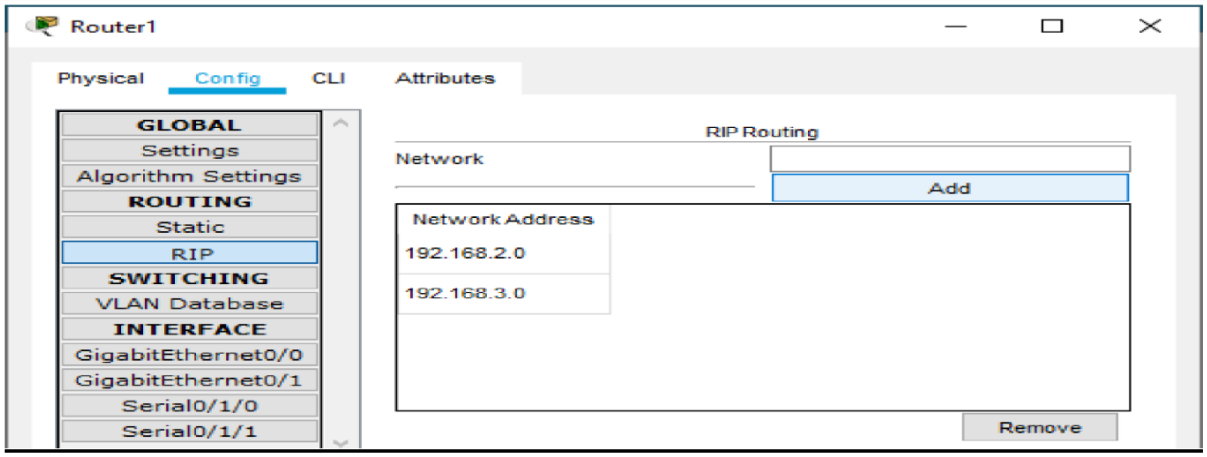


We need to set the Routing table in all the Routers so that each node could send and receive packets from others (RIP is set in all the Routers as follows)

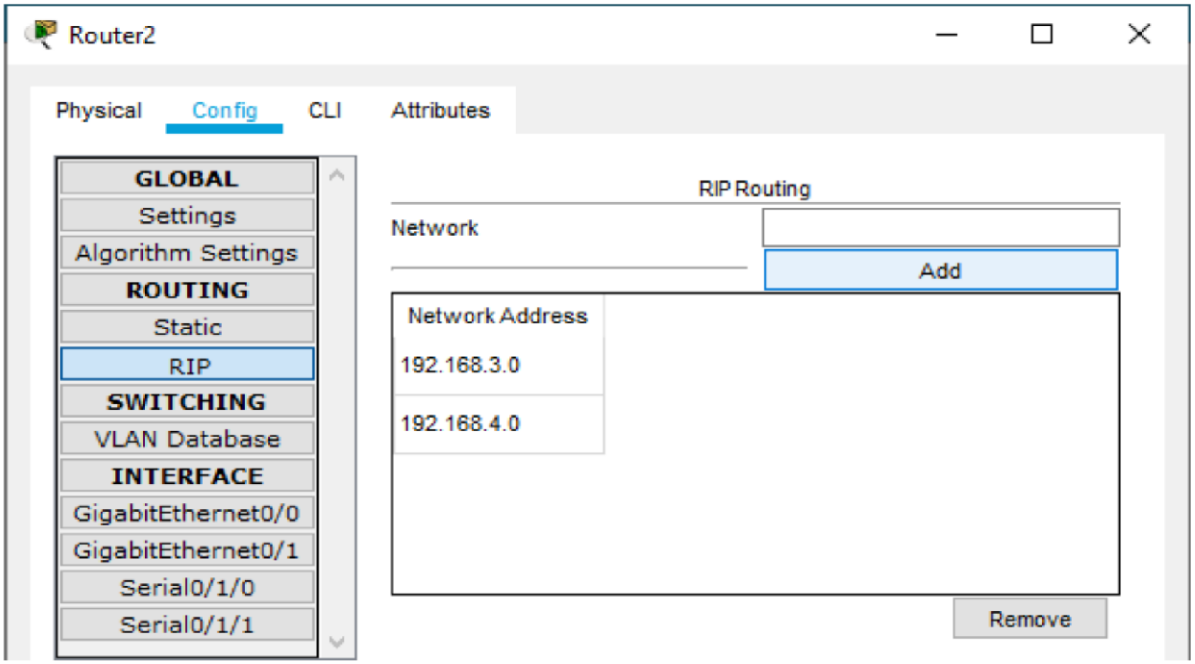
Router0



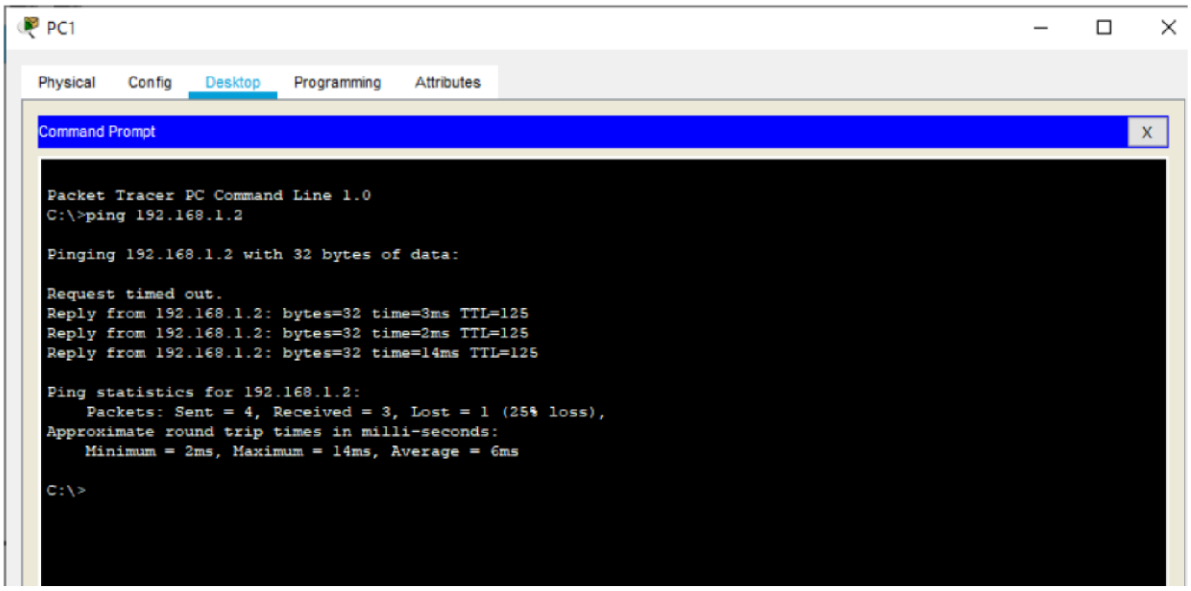
Router1



Router2



Now we can check the connectivity by sending ping commands from any node to any other node



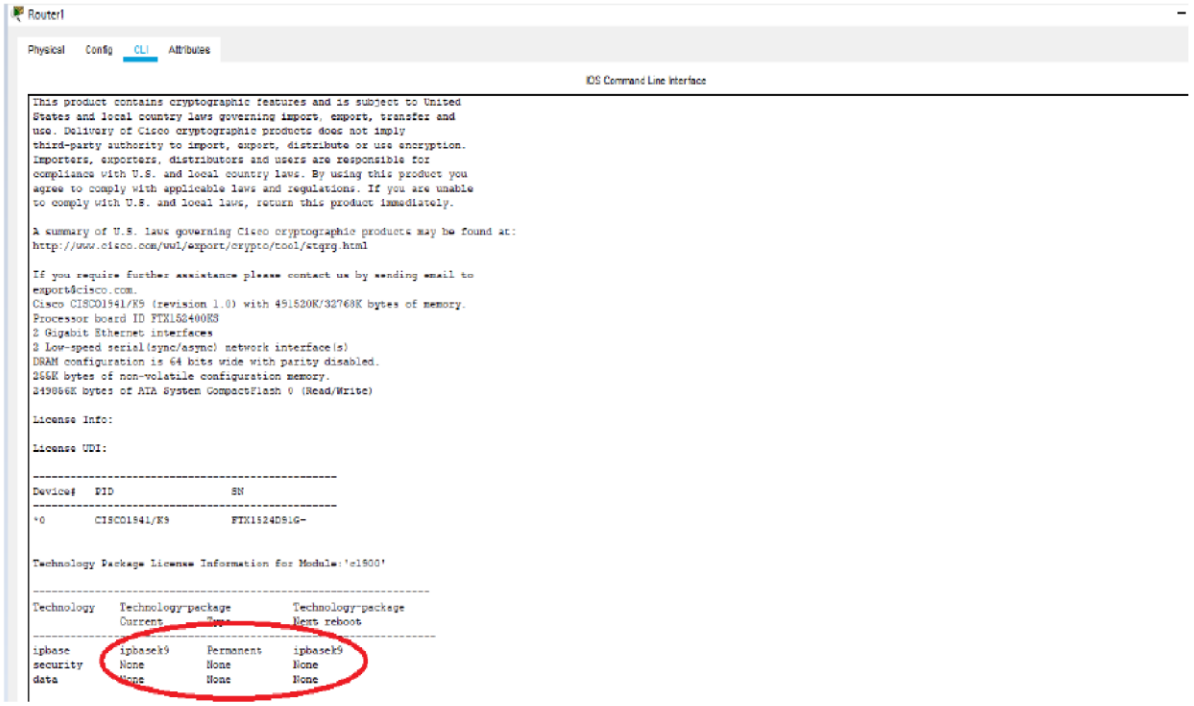
So, we conclude that the connectivity has been established

Part 1: Enable the IOS IPS (on Router1)

Type the following command in the CLI mode of Router1

Router#show version

We will get a message informing whether the security Package is enabled or not



As seen above the security package is not enabled, to enable the security feature, type the following command in Router1

Router>en
Router#conf t
Router(config)#license boot module c1900 technology-package securityk9
ACCEPT? [yes/no]: y
Router(config)#exit
Router#copy run start
Press enter when prompted
Router#reload
Continue with configuration dialog? [yes/no]: n
Router#show version

We will get a message informing whether the security package is enabled or not.

```
If you require further assistance please contact us by sending email to
export@cisco.com.
Cisco CISCO1941/K9 (revision 1.0) with 491520K/32768K bytes of memory.
Processor board ID FTX152400KS
2 Gigabit Ethernet interfaces
2 Low-speed serial(sync/async) network interface(s)
DRAM configuration is 64 bits wide with parity disabled.
255K bytes of non-volatile configuration memory.
249856K bytes of ATA System CompactFlash 0 (Read/Write)

License Info:

License UDI:

-------------------------------------------------
Device#   PID                 SN
-------------------------------------------------
*0        CISCO1941/K9        FTX1524D91G-


Technology Package License Information for Module:'c1900'

-----------------------------------------------------------------
Technology    Technology-package          Technology-package
              Current       Type          Next reboot
-----------------------------------------------------------------
ipbase        ipbasek9      Permanent     ipbasek9
security      securityk9    Evaluation    securityk9
data          disable       None          None

Configuration register is 0x2102
```

As seen above now the security package has been enabled

Now, type the following commands in the CLI mode of Router1

```
Router>en
Router#mkdir dalmia
Create directory filename [dalmia]?
Created dir flash:dalmia
Router#conf t
Router(config)#ip ips config location flash:dalmia
Router(config)#ip ips name iosips
Router(config)#ip ips notify log
Router(config)#ip ips signature-category
Router(config-ips-category)#category all
Router(config-ips-category-action)#retired true
Router(config-ips-category-action)#exit


Router(config-ips-category)#category ios_ips basic
Router(config-ips-category-action)#retired false
Router(config-ips-category-action)#exit
Router(config-ips-category)#exit
Do you want to accept these changes? [confirm]
Applying Category configuration to signatures ...
%IPS-6-ENGINE_BUILDING: atomic-ip - 288 signatures - 6 of 13 engines
%IPS-6-ENGINE_READY: atomic -ip - build time 30 ms - packets for this engine will be
scanned

Router(config)#int Se0/1/0
Router(config-if)#ip ips iosips out

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

Part 2: Modify the Signature

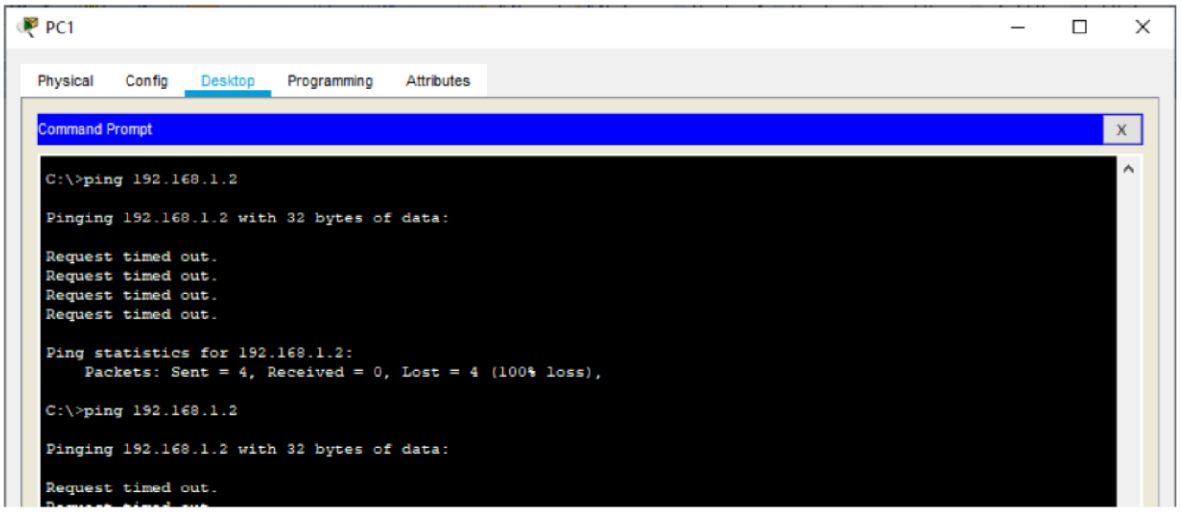
Type the following commands in the CLI mode of Router1

```
Router#conf t
Router(config)#ip ips signature-definition
Router(config-sigdef)#signature 2004 0
Router(config-sigdef-sig)#status
Router(config-sigdef-sig-status)#retired false
Router(config-sigdef-sig-status)#enabled true
Router(config-sigdef-sig-status)#exit
Router(config-sigdef-sig)#engine
Router(config-sigdef-sig-engine)#event-action produce-alert
Router(config-sigdef-sig-engine)#event-action deny-packet-inline
Router(config-sigdef-sig-engine)#exit
Router(config-sigdef-sig)#exit
Router(config-sigdef)#exit
```

Do you want to accept these changes? [confirm] <Enter>
Router(config)#

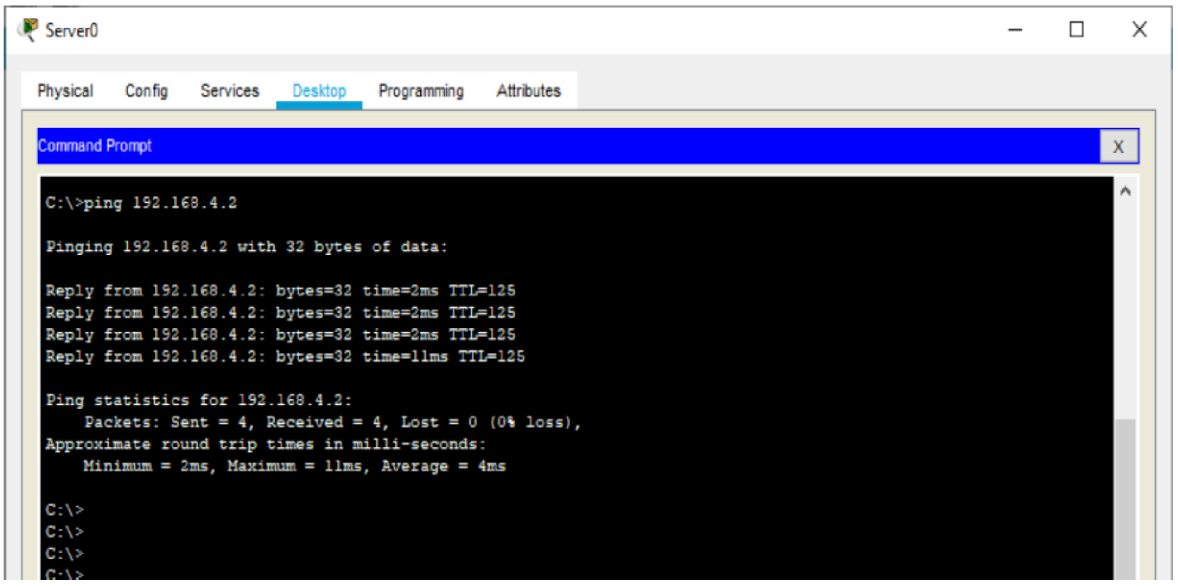
Now we need to verify the above IPS configuration, we do it first by pinging PC1 to SERVER and then from SERVER to PC1

PC1 to SERVER

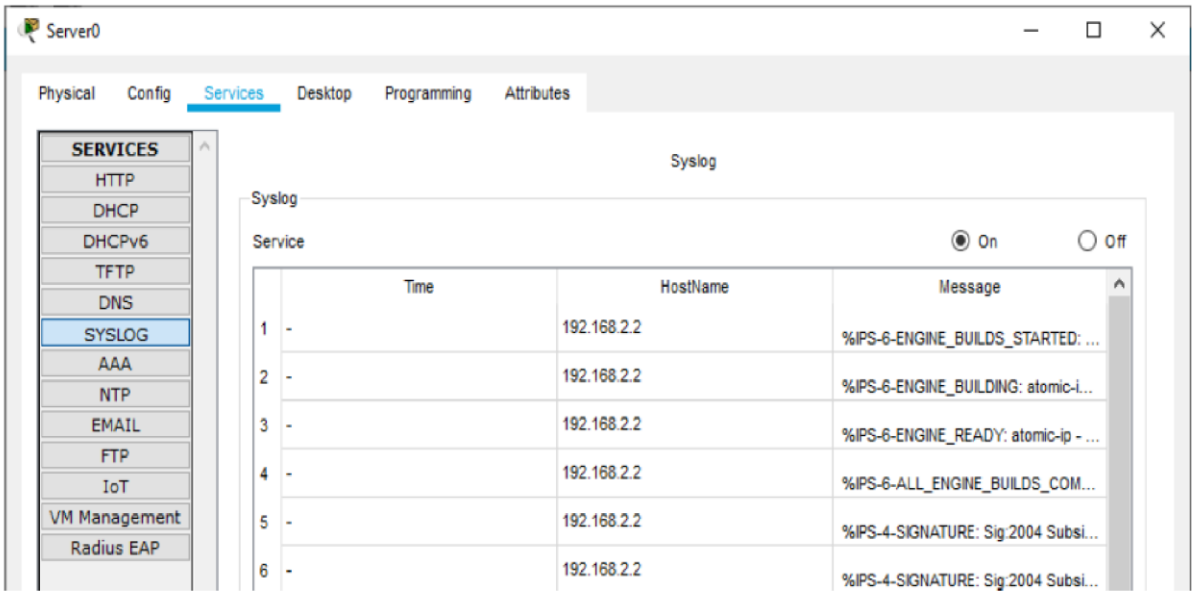


The ping FAILS

SERVER to PC1



Also, we can observe the Syslog service in the SERVER to check the log activities



Use show commands to verify IPS on Router1

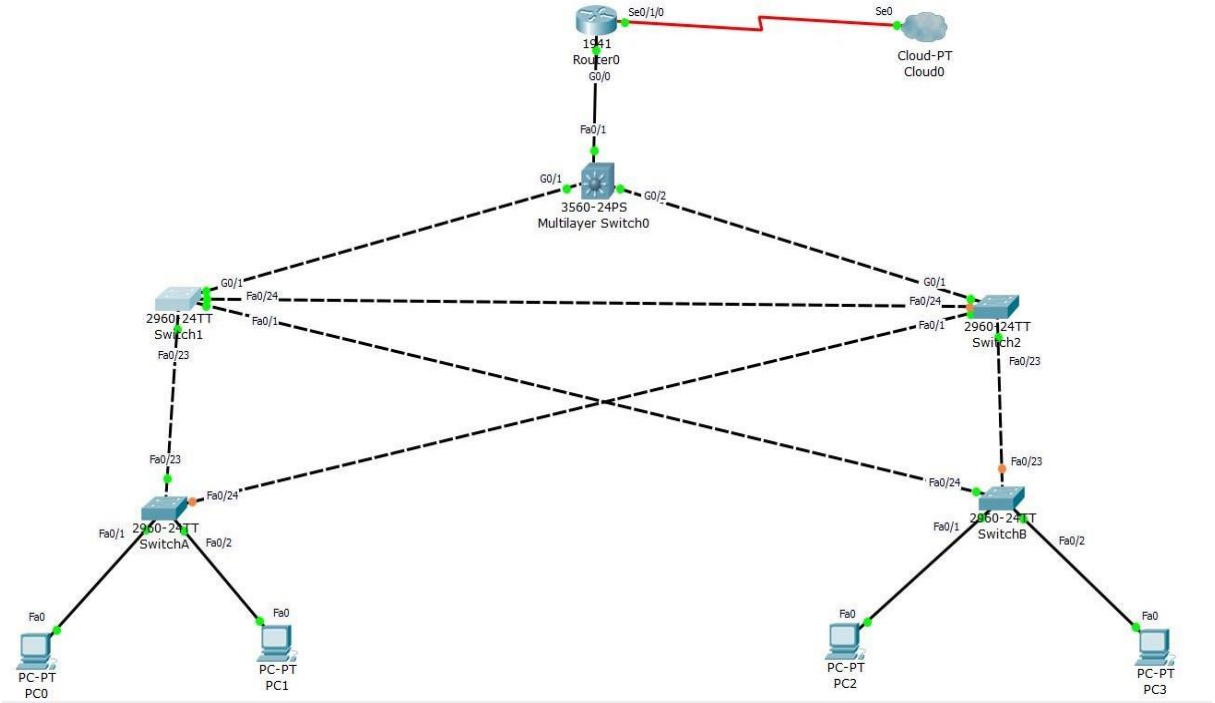
Router#show ip ips all

Hence we set the IPS and also verified it on Router1

Practical 8

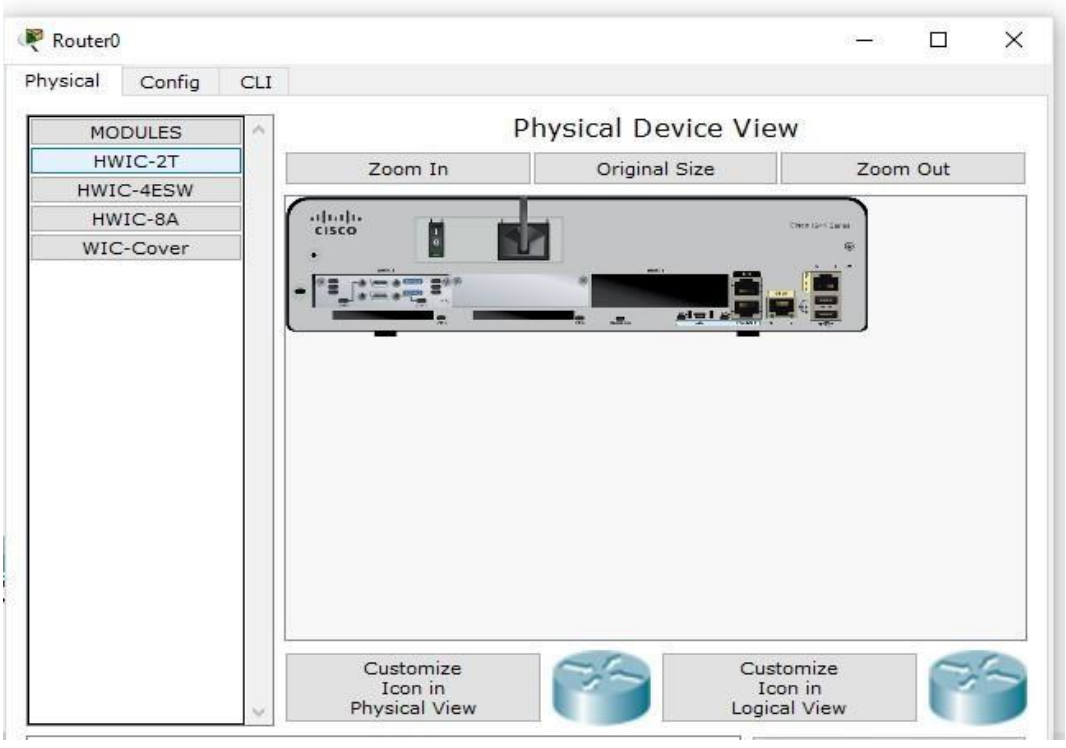
Layer 2 Security a) Assign the Central switch as the root bridge. b) Secure spanning-tree parameters to prevent STP manipulation attacks. c) Enable port security and disable unused ports.

Topology

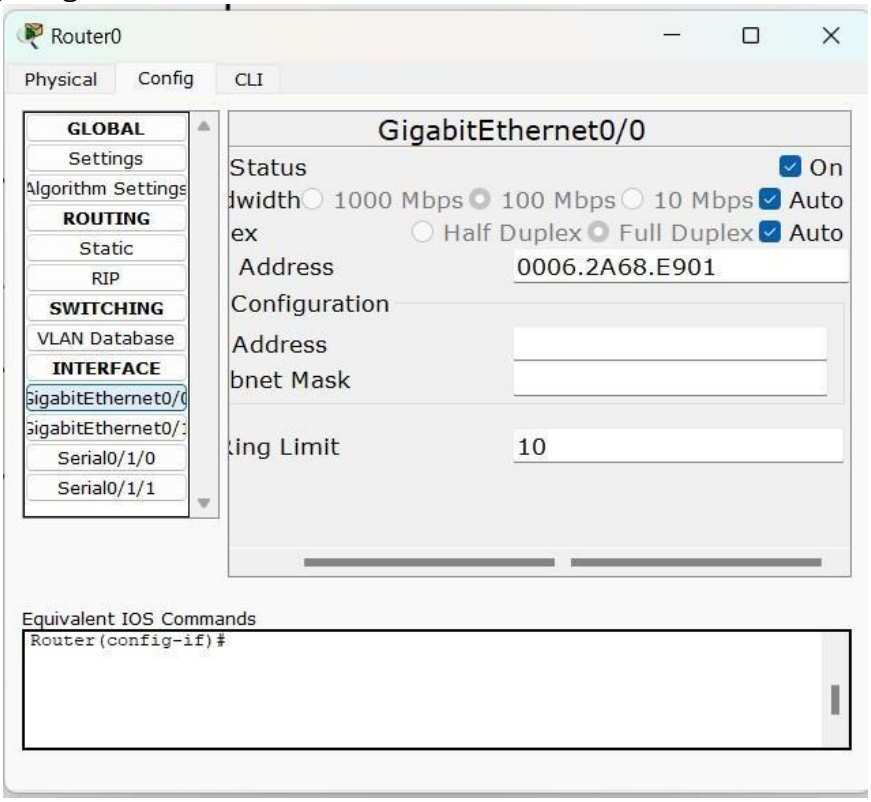


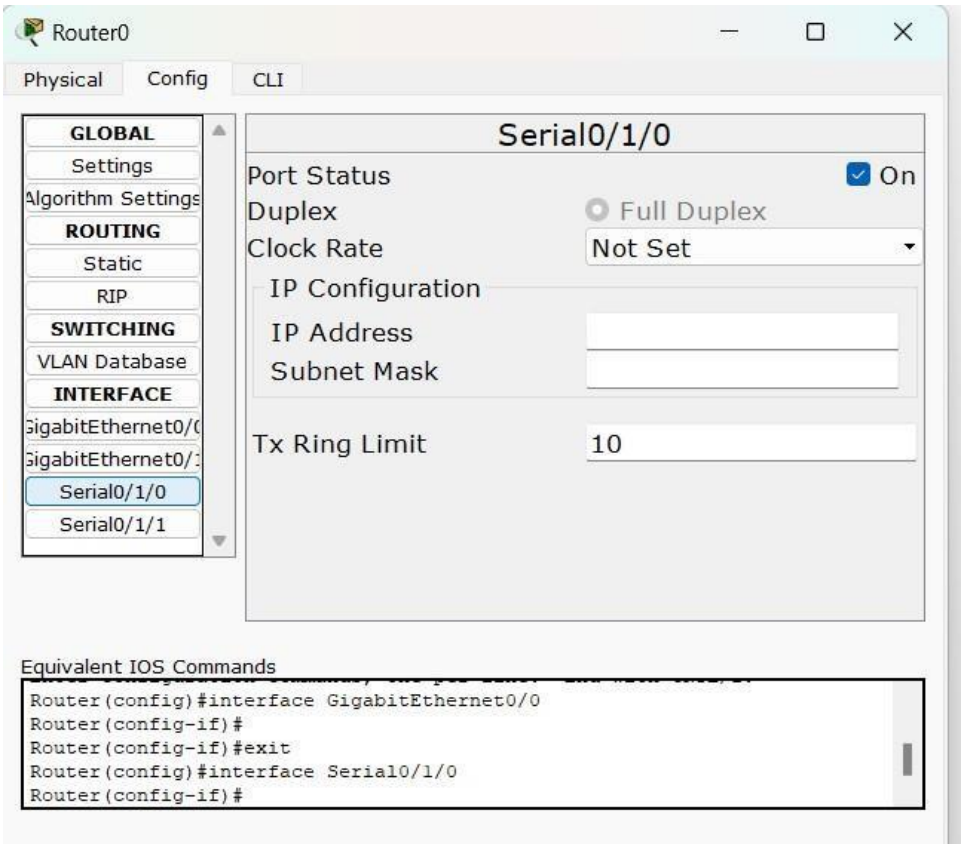
Serial Interface must be added in the Router0 before configuring it

The serial interface in Router0 is added as follows



Configuring Router0

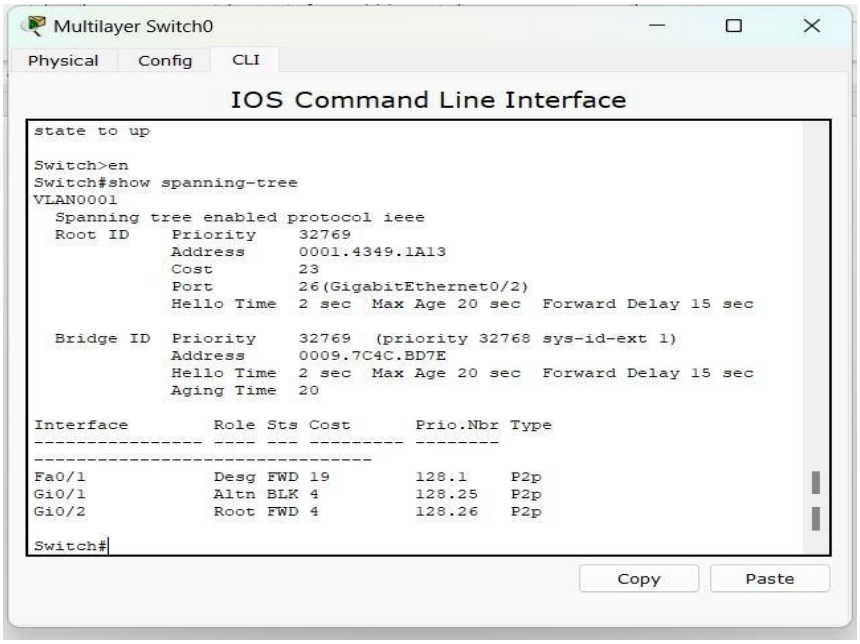


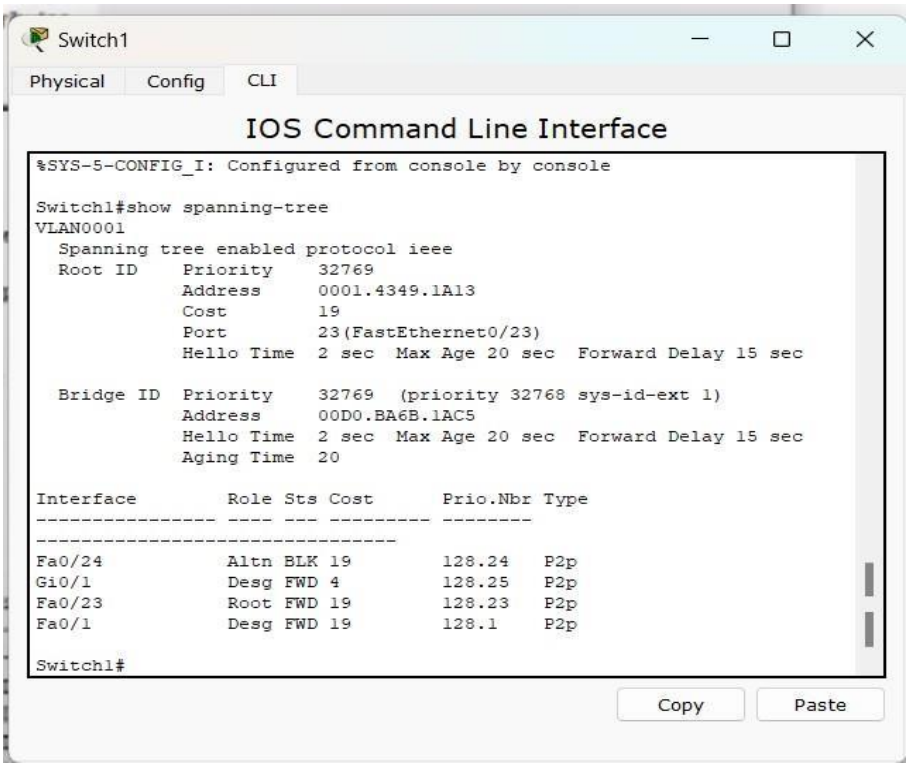


Part 1: Configure Root Bridge Step 1: Determine the current root bridge.

From **Multilayer Switch0**, issue the **show spanning-tree** command to determine the current root bridge, to see the ports in use, and to see their status.

switch>en
switch# show spanning -tree





Step 2: Assign Multilayer Switch0 as the primary root bridge. Using the **spanning-tree vlan 1 root primary** command, and assign **Multilayer switch0** as the root bridge.

switch#conf t

switch(config)#spanning-tree vlan 1 root primary
switch(config)#do show span

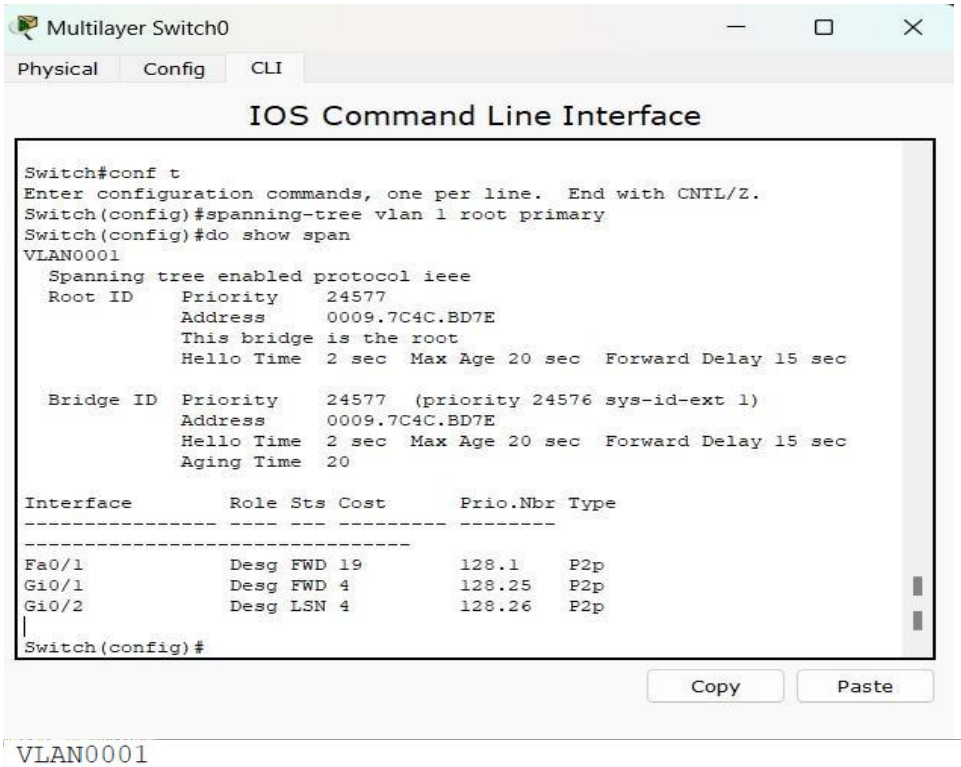
Step 3: Assign Switch1 as a secondary root bridge. Assign SW-1 as the secondary root bridge using the spanning-tree vlan 1 root secondary command.

switch#conf t

switch(config)#spanning-tree vlan 1 root secondary

Step 4: Verify the spanning-tree configuration. Issue the show spanning-tree command to verify that Multi-layer Switch0 is the root bridge.

switch# show spanning-tree




```
VLAN0001
   Spanning tree enabled protocol ieee
   Root ID  Priority      24577
            Address       00D0.D31C.634C
            This bridge is the root
            Hello Time  2 sec  Max Age  20 sec   Forward Delay  15 sec
```

Part 2: Protect Against STP Attacks Secure the STP parameters to prevent STP manipulation attacks.

Step 1: Enable PortFast on all access ports.
PortFast is configured on access ports that connect to a single workstation or server to enable them to become active more quickly. On the connected access ports of the SwitchA and SwitchB, use the spanning-tree portfast command.

SwitchA>en

SwitchA#conf t

SwitchA(config)#int range f0/1-2

SwitchA(config-if-range)#spanning-tree portfast

SwitchB>en

SwitchB#conf t

SwitchB(config)#int range f0/1-2

SwitchB(config-if-range)#spanning-tree portfast

Step 2: Enable BPDU guard on all access ports.
BPDU guard is a feature that can help prevent rogue switches and spoofing on access ports. Enable BPDU guard on SwitchA and SwitchB access ports.

SwitchA(config)#int range f0/1-2

SwitchA(config-if-range)#spanning-tree bpduguard enable

SwitchB(config)#int range f0/1-2

SwitchB(config-if-range)#spanning-tree bpduguard enable

Step 3: Enable root guard.
Root guard can be enabled on all ports on a switch that are not root ports. It is best deployed on ports that connect to other non-root switches. Use the show spanning-tree command to determine the location of the root port on each switch.

On Switch1, enable root guard on ports F0/23 and F0/24. On Switch2, enable root guard on ports F0/23 and F0/24.

Switch1>en

Switch1#conf t

Switch1(config)#int range f0/23-24

Switch1(config-if-range)#spanning-tree guard root

Switch2>en

Switch2#conf t

Switch2(config)#int range f0/23-24

Switch2(config-if-range)#spanning-tree guard root

Part 3: Configure Port Security and Disable Unused Ports Step 1:
Configure basic port security on all ports connected to host devices.
This procedure should be performed on all access ports on SwitchA and SwitchB. Set the maximum number of learned MAC addresses to 2, allow the MAC address to be learned dynamically, and set the violation to **shutdown**. Note: A switch port must be configured as an access port to enable port security.

SwitchA>en

SwitchA#conf t

SwitchA(config)#int range f0/1-2

SwitchA(config-if-range)#switchport mode access

SwitchA(config-if-range)#switchport port-security

SwitchA(config-if-range)#switchport port-security maximum 2

SwitchA(config-if-range)#switchport port-security violation shutdown

SwitchA(config-if-range)#switchport port-security mac-address sticky

SwitchB>en

SwitchB#conf t

SwitchB(config)#int range f0/1-2

SwitchB(config-if-range)#switchport mode access

SwitchB(config-if-range)#switchport port-security

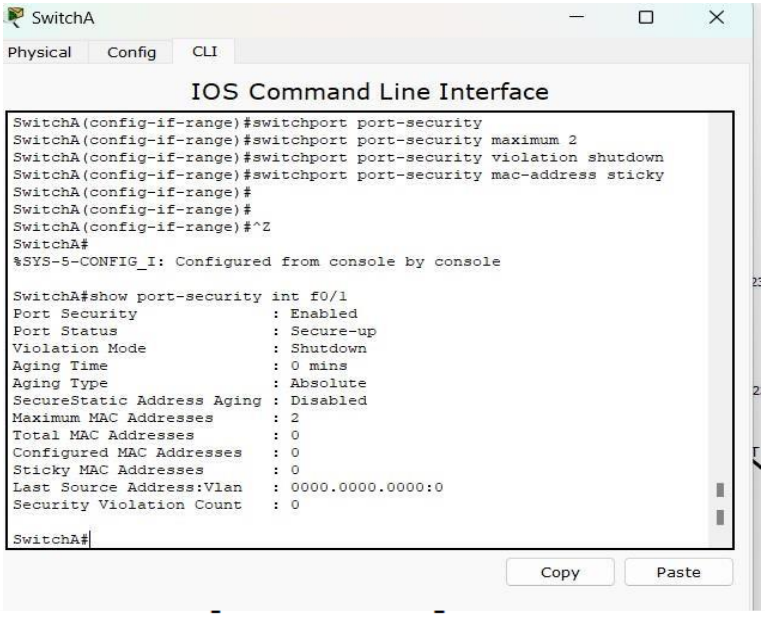
SwitchB(config-if-range)#switchport port-security maximum 2

SwitchB(config-if-range)#switchport port-security violation shutdown

SwitchB(config-if-range)#switchport port-security mac-address sticky

Step 2: Verify port security.
On SwitchA, issue the command show port-security int f0/1 to verify that port security has been configured.

SwitchA#show port -security int f0/1




```
Port Security              : Enabled
Port Status                : Secure-up
Violation Mode             : Shutdown
Aging Time                 : 0 mins
Aging Type                 : Absolute
SecureStatic Address Aging : Disabled
Maximum MAC Addresses      : 2
Total MAC Addresses        : 0
Configured MAC Addresses   : 0
Sticky MAC Addresses       : 0
Last Source Address:Vlan   : 0000.0000.0000:0
Security Violation Count   : 0
```

Step 3: Disable unused ports.
Disable all ports that are currently unused.

SwitchA(config)#int range f0/3-22

SwitchA(config-if-range)#shutdown

SwitchB(config)#int range f0/3-22

SwitchB(config-if-range)#shutdown

Hence the Port security has been enabled.