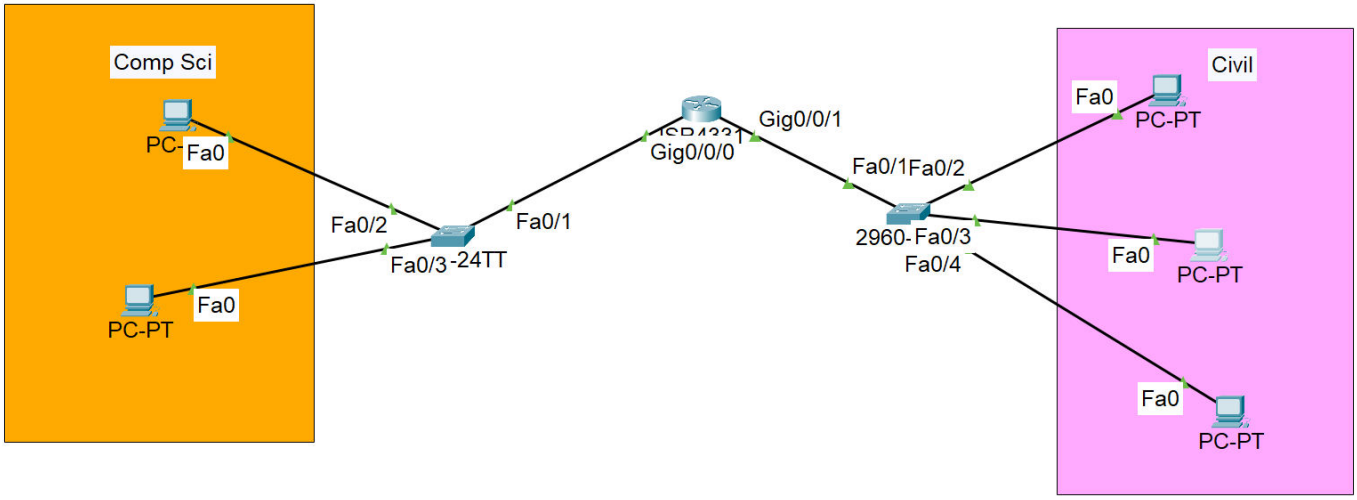
Configuring DHCP using Router

Devices:

- 5 x PCs
- 2 x 2960 Switch
- 1 x 4331 Router

Step1:

- Setup aforementioned devices like this.



Step 2:

- Configure IP addresses of Gig0/0/0 as 192.168.1.1 and Gig0/0/1 as 192.168.2.1 and turn on the both the ports.

Step 3:

- Go to Router CLI and type the following commands:

IOS Command Line Interface

```
Router>
Router>
Router>
Router>
Router>
Router>
Router>en
Router#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#ip dhcp excluded-address 192.168.1.0 192.168.1.10
Router(config)#ip dhcp excluded-address 192.168.2.0 192.168.2.10
Router(config)#ip dhcp pool compsci
Router(dhcp-config)#network 192.168.1.0 255.255.255.0
Router(dhcp-config)#default-router 192.168.1.1
Router(dhcp-config)#exit
Router(config)#ip dhcp pool civil
Router(dhcp-config)#network 192.168.2.0
% Incomplete command.
Router(dhcp-config)#network 192.168.2.0 255.255.255.0
Router(dhcp-config)#default-router 192.168.2.1
Router(dhcp-config)#exit
Router(config)#do wr
Building configuration...
[OK]
Router(config)#
```

Step 5:

- That’s it, now the router will distribute IP addresses for all computers in both LAN compsci and LAN civil within the defined range excluding the addresses we manually specified it not it.

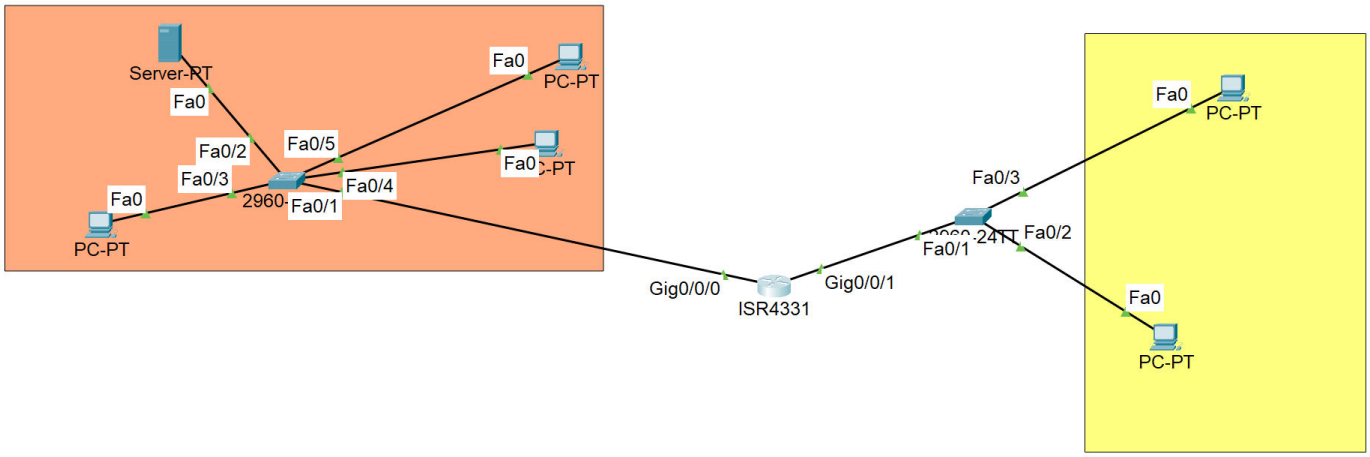
Configure DHCP using Server

Devices:

- 5 x PCs
- 2 x 2960 Switch
- 1 x 4331 Router
- 1 x Server-PT

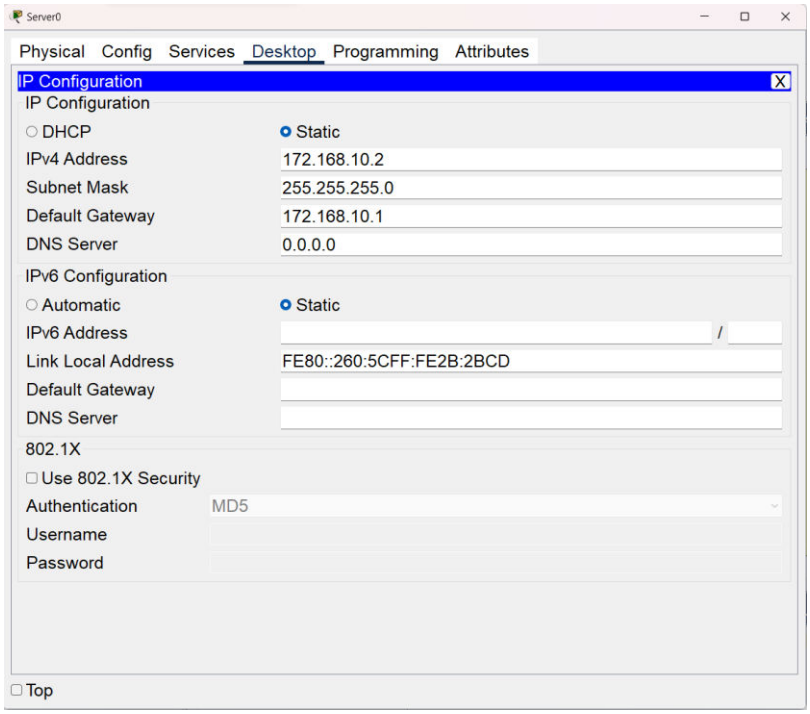
Step1:

- Setup aforementioned devices like this.



Step 2:

- Assign IP address and gateway for Server as follows:

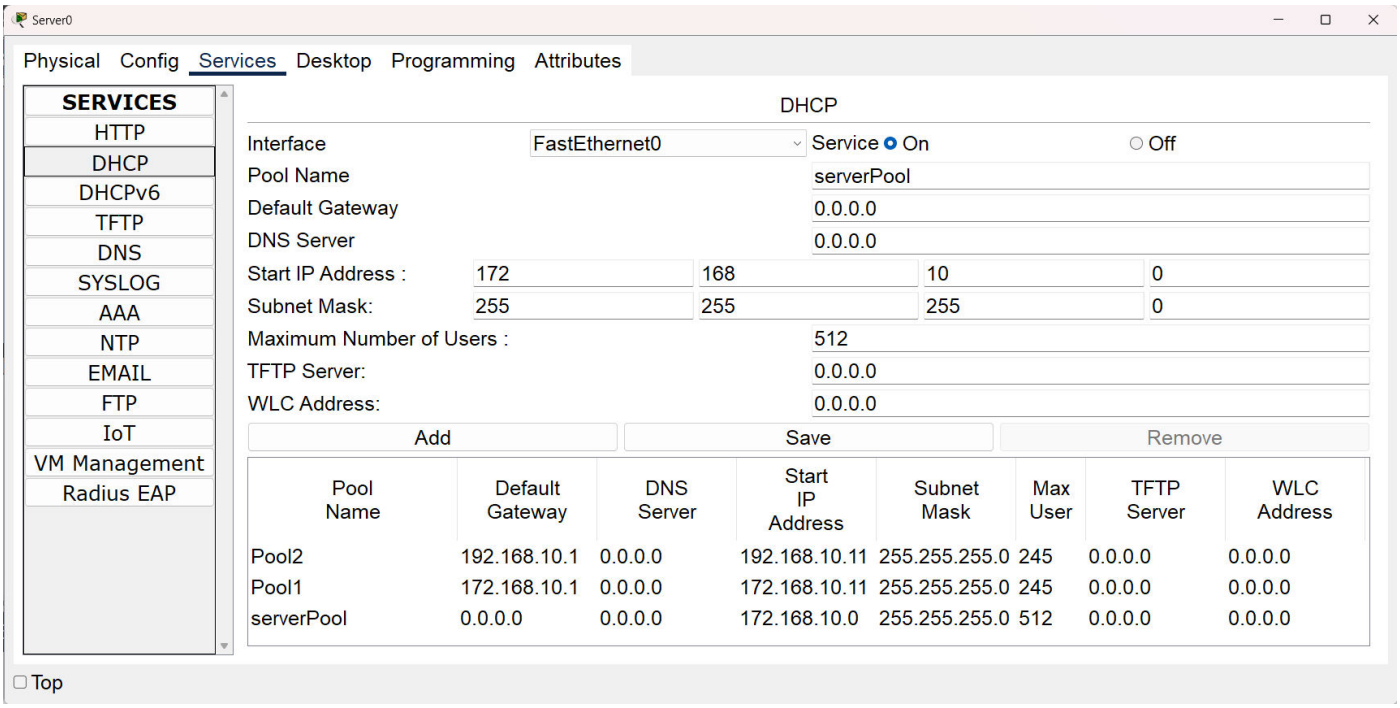


Step 3:

- Assign IP address 172.168.10.1 to Gig 0/0/0 and 192.168.10.1 to Gig 0/0/1 of the router.

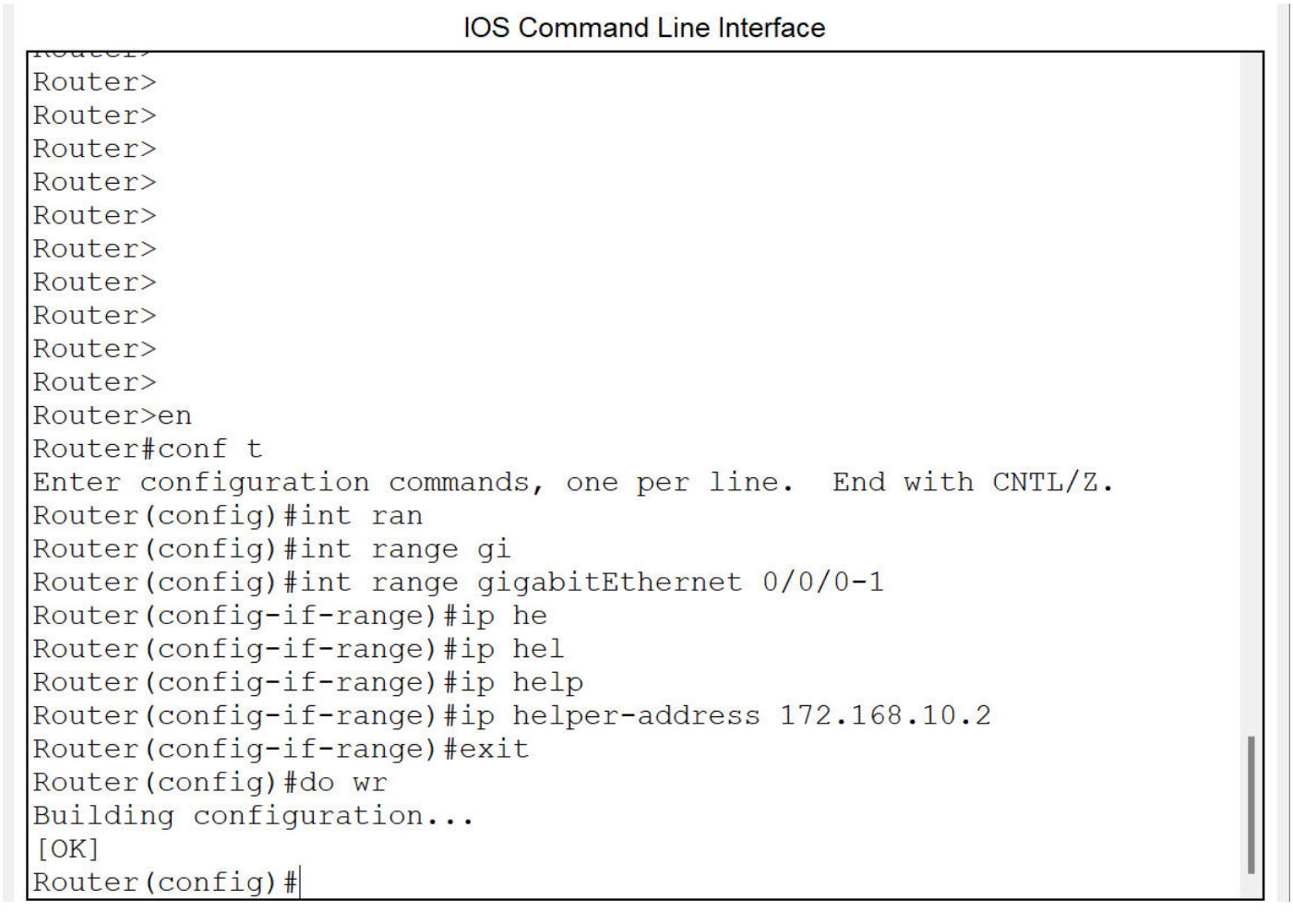
Step 5:

- Go to the Services section of the Server and configure the following DHCP pools, (make sure to update the default serverPool as well)



Step 6:

- Go to Router's CLI and configure as follows:



Step 7:

- That’s it, now go to each PC’s FA0 port and simply select DHCP instead of static IP configuration, the new IP address should be assigned automatically within a few seconds.

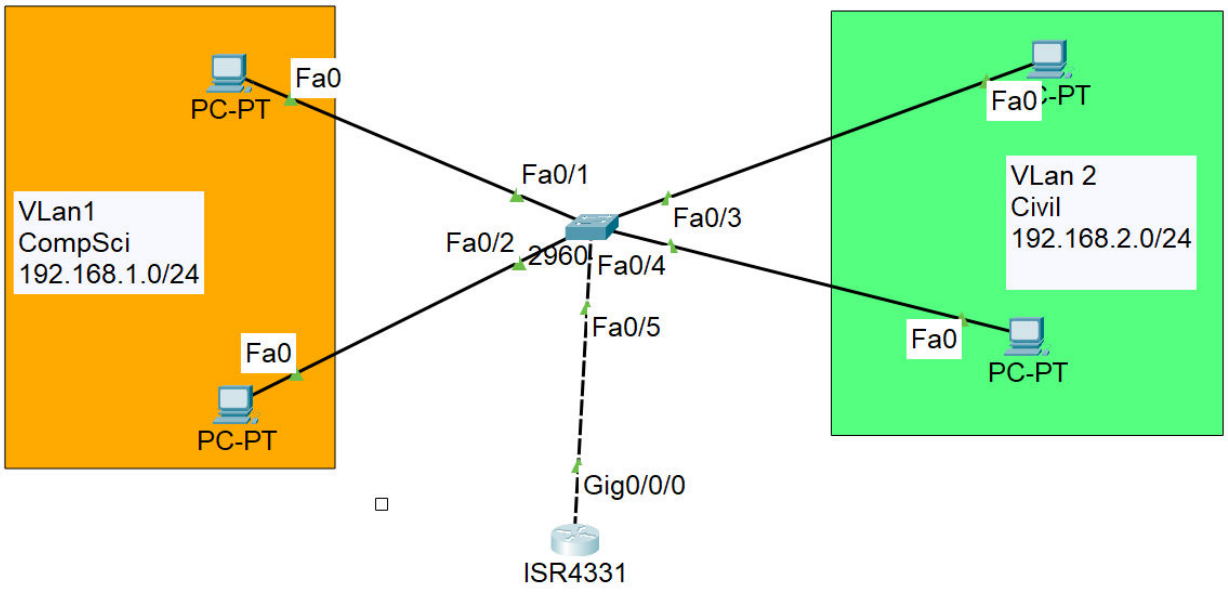
Configuring two VLANs using switch and communicating between them using Router

Devices:

- 4 x PCs
- 1 x 2960 Switch
- 1 x 4331 Router

Step1:

- Setup aforementioned devices like this.



Step 2:

- Assign IP address and gateway for all PCs of both CompSci and Civil VLAN with their respective networks addresses provided.

Step 3:

- Go to Switch CLI and type the following commands:

IOS Command Line Interface

```
Switch>
Switch>en
Switch#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)#vl
Switch(config)#vlan 10
Switch(config-vlan)#name compsci
Switch(config-vlan)#vlan 20
Switch(config-vlan)#name civil
Switch(config-vlan)#
Switch#
%SYS-5-CONFIG_I: Configured from console by console

Switch#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)#int r
Switch(config)#int range fa 0/1-2
Switch(config-if-range)#switchport mode access
Switch(config-if-range)#switchport access  vlan 10
Switch(config-if-range)#int range fa 0/3-4
Switch(config-if-range)#switchport mode access
Switch(config-if-range)#switchport access  vlan 20
Switch(config-if-range)#int fa 0/5
Switch(config-if)#switchport mode trunk
Switch(config-if)#
```

Step 4:

- Go to Router CLI and type the following commands

IOS Command Line Interface

```
Router#
Router#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#int gi
Router(config)#int gigabitEthernet 0/0/0.10
Router(config-subif)#
%LINK-5-CHANGED: Interface GigabitEthernet0/0/0.10, changed state
to up

Router(config-subif)#encapsulation dot1q 10
Router(config-subif)#ip address 192.168.1.1
% Incomplete command.
Router(config-subif)#ip address 192.168.1.1 255.255.255.0
Router(config-subif)#int gigabitEthernet 0/0/0.20
Router(config-subif)#
%LINK-5-CHANGED: Interface GigabitEthernet0/0/0.20, changed state
to up

Router(config-subif)#encapsulation dot1q 20
Router(config-subif)#ip address 192.168.2.1 255.255.255.0
Router(config-subif)#
Router(config-subif)#
Router#
%SYS-5-CONFIG_I: Configured from console by console
```

Step 5:

That's it, now PCs of both VLANs will communicate via the router as if they were two different physical LANs.

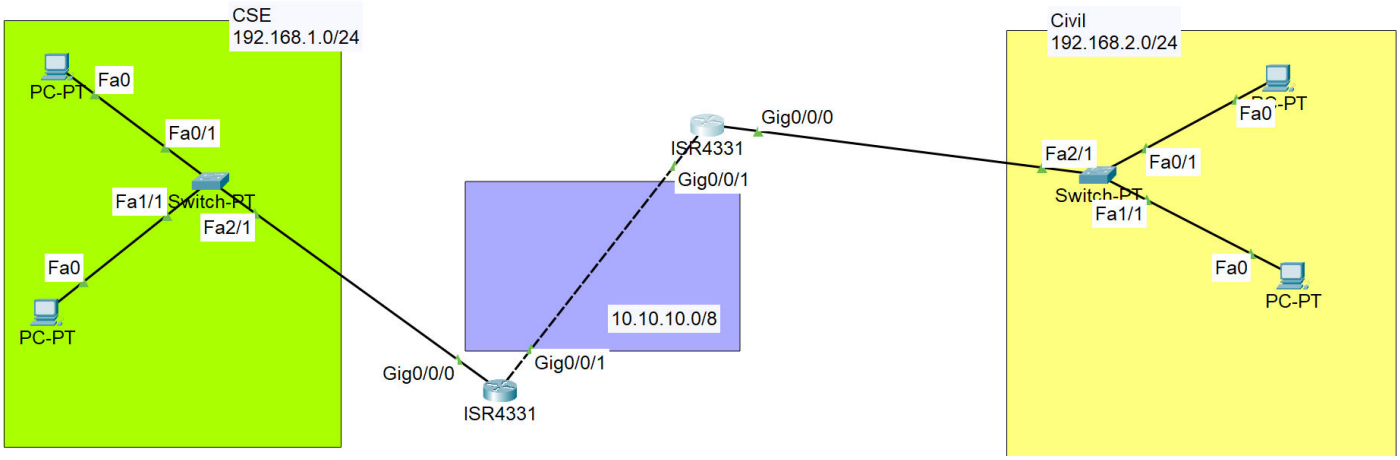
Using Extended Access List to block communication between certain hosts and certain packet types on different network

Devices:

- 4 x PCs
- 2 x 2960 Switch
- 2 x 4331 Routers

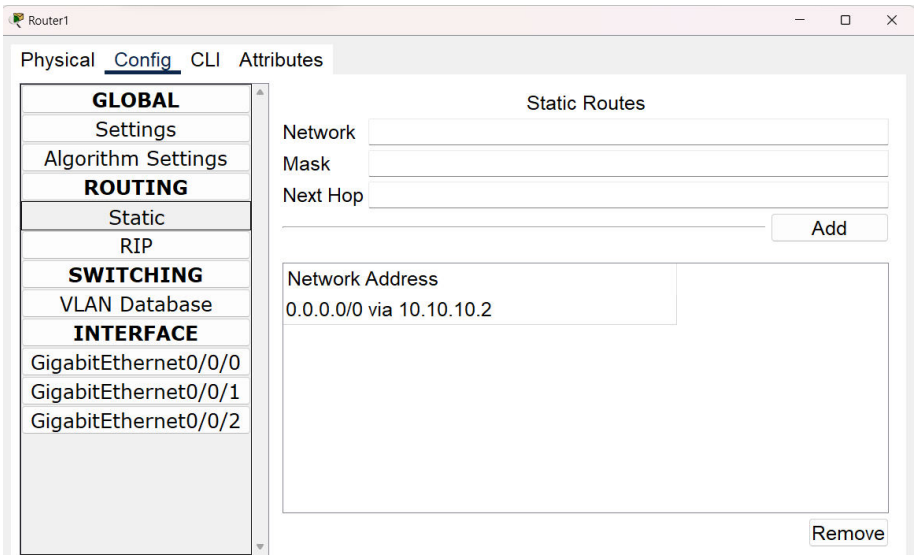
Step1:

- Setup aforementioned devices like this as well as assign the respective IP addresses.

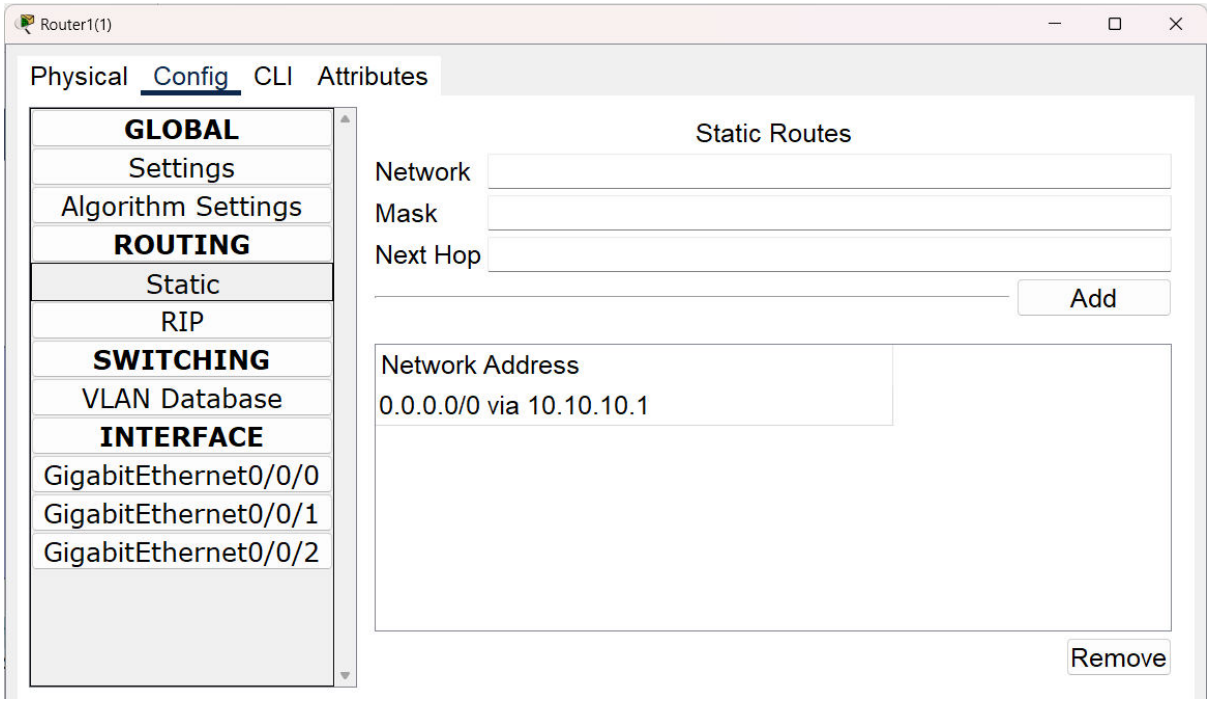


Step 2:

- Go to left Router, Config>Static and setup the following default static route.

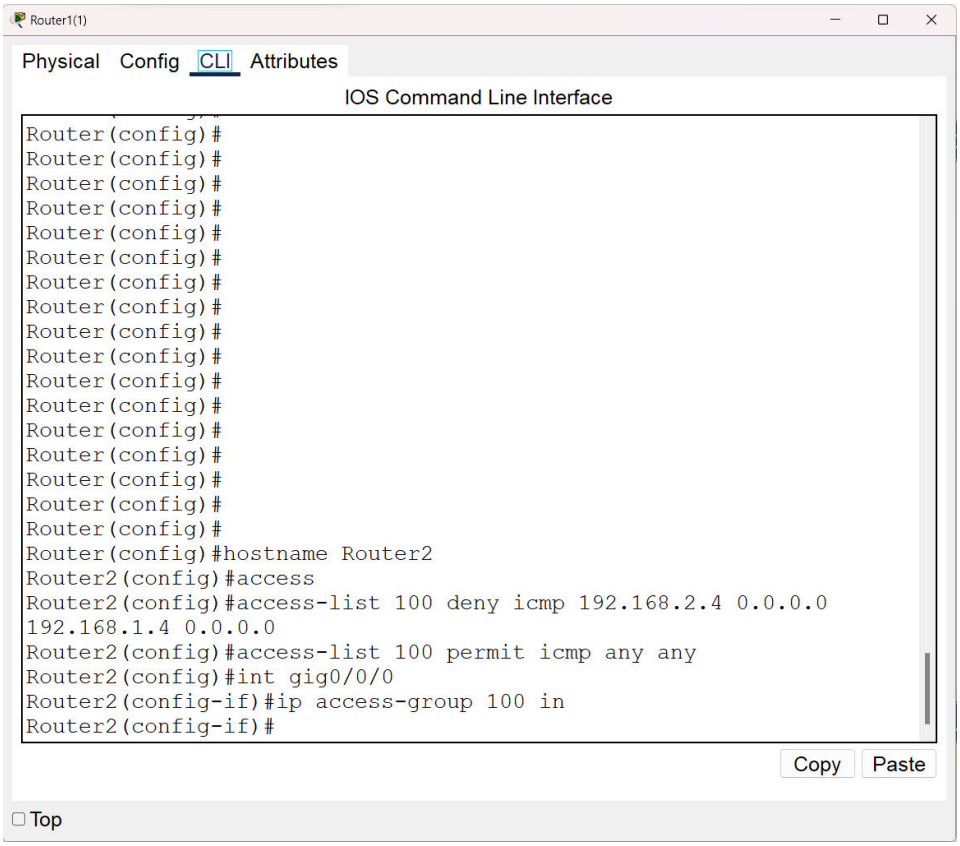


- Similarly do the same for right router like this:



Step 3:

- Go to the right Router CLI and configure the access list as follows:



Step 4:

- Now ICMP communication between host 192.168.2.4 and host 192.168.1.4 should be forbidden.

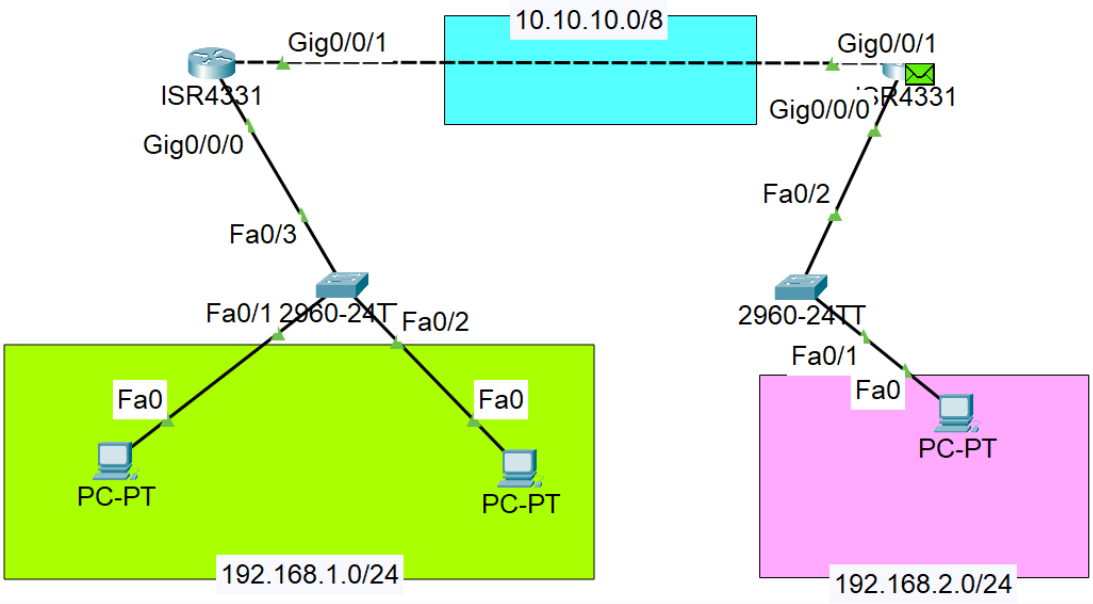
Using Static NAT to configure local and global IP addresses

Devices:

- 3 x PCs
- 2 x 2960 Switch
- 2 x 4331 Routers

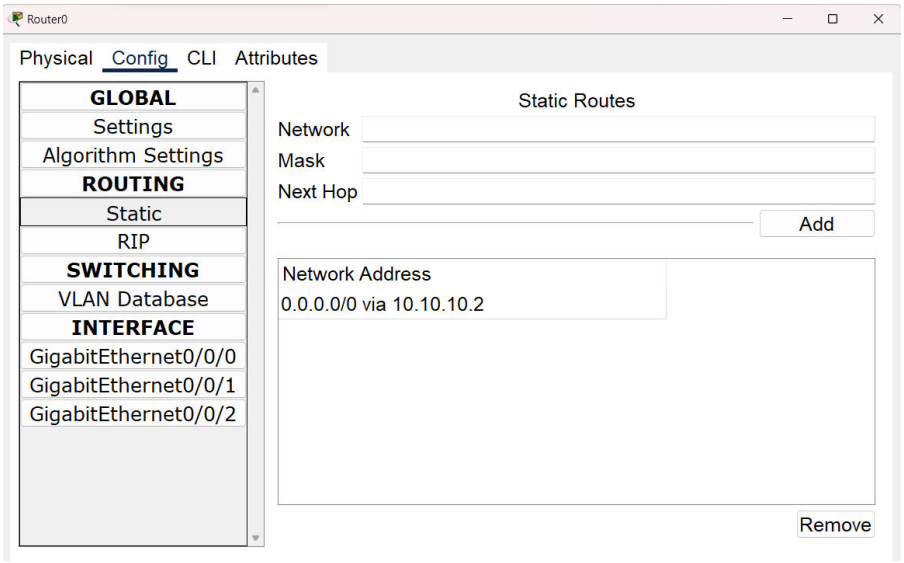
Step1:

- Setup aforementioned devices like this as well as assign the respective IP addresses.



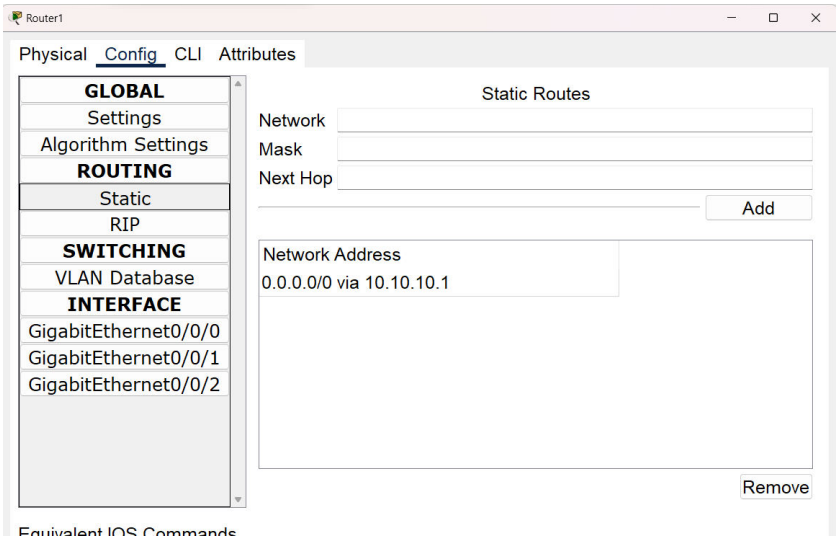
Step 2:

- Go to left Router > Config > Static Routing and add the following hop:



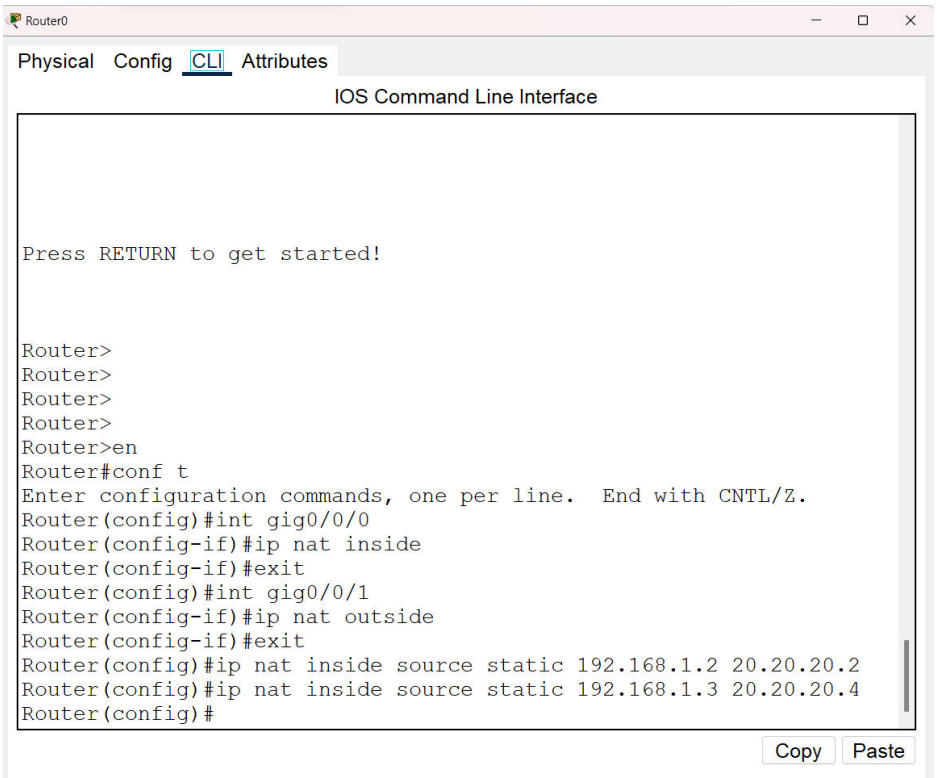
Step 2:

- Do the same for right Router as follows:



Step 5: Go to left Router CLI and type the following:

.



That's it, now the left router will convert source IP address from local to global before further transmitting any packets.

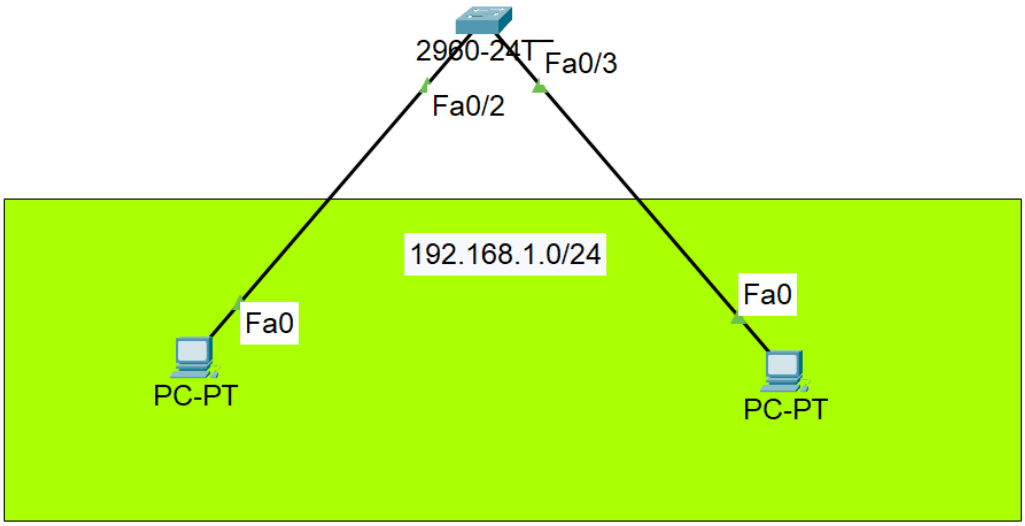
Using PORT security to prevent rogue hosts from impersonating original hosts on a switch

Devices:

- 3 x PCs
- 1 x 2960 Switch

Step1:

- Setup aforementioned devices like this as well as assign the respective IP addresses.

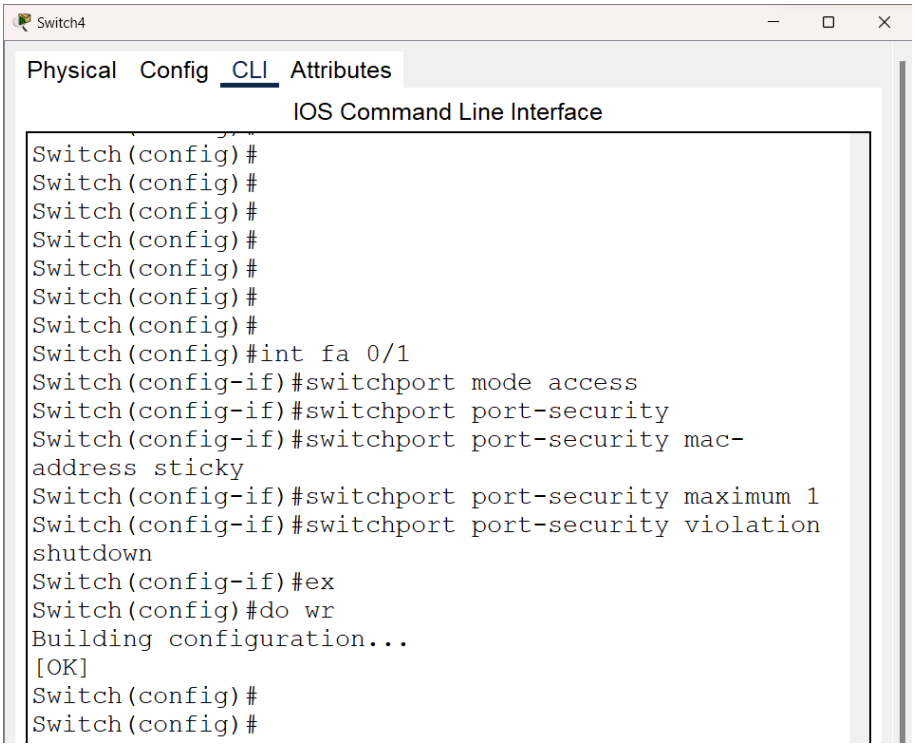


Step 2:

- Ping from one PC to another

Step 3:

Go to switch CLI and type the following commands:



Step 4:

- Bring a new rogue PC and configure it's IP address to impersonate left host (the one connected to fa0/1 port)
- Now disconnect original PC and connect rogue PC to switch's fa0/1 port.
- Try communication between the new pair of PCs

Step 5:

- The fa0/1 port should immediately shutdown if any rogue PC tries to impersonate the original host PC.

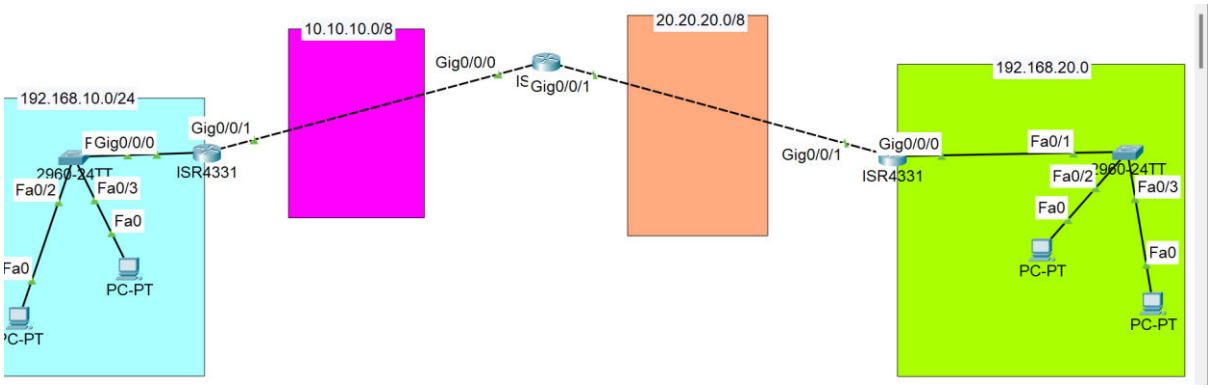
Using RIP Routing to allow routers to communicate with each other

Devices:

- 4 x PCs
- 2 x 2960 Switch
- 2 x 4331 Routers

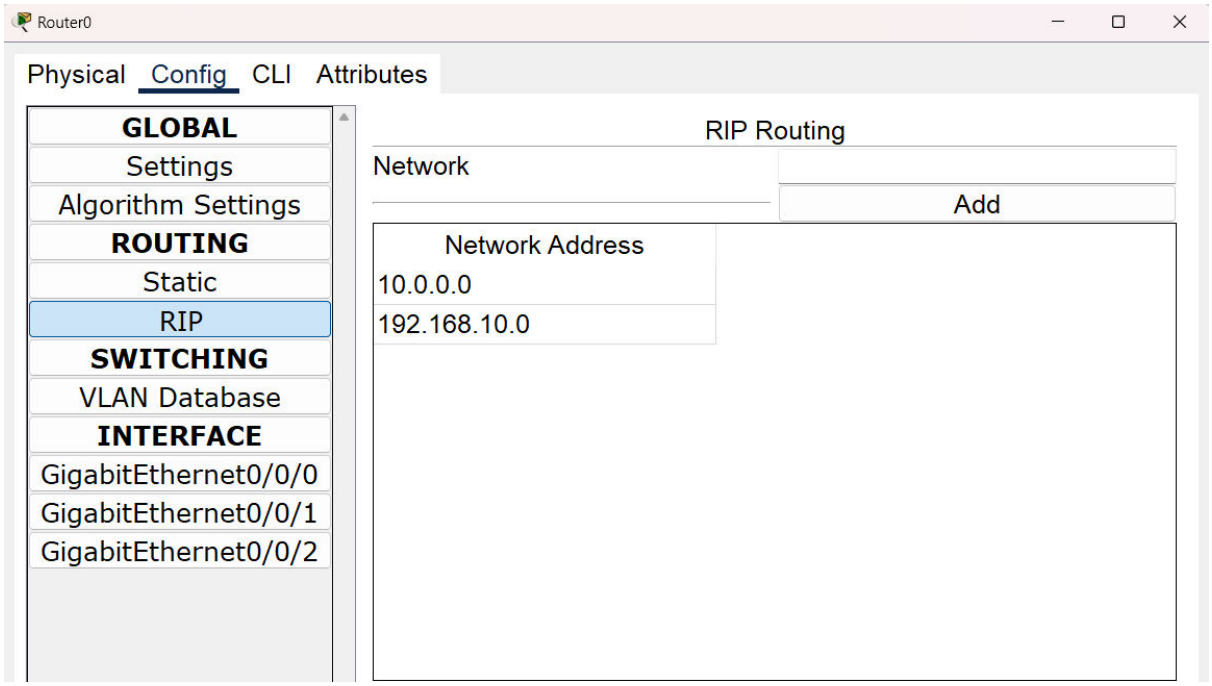
Step1:

- Setup aforementioned devices like this as well as assign the respective IP addresses.

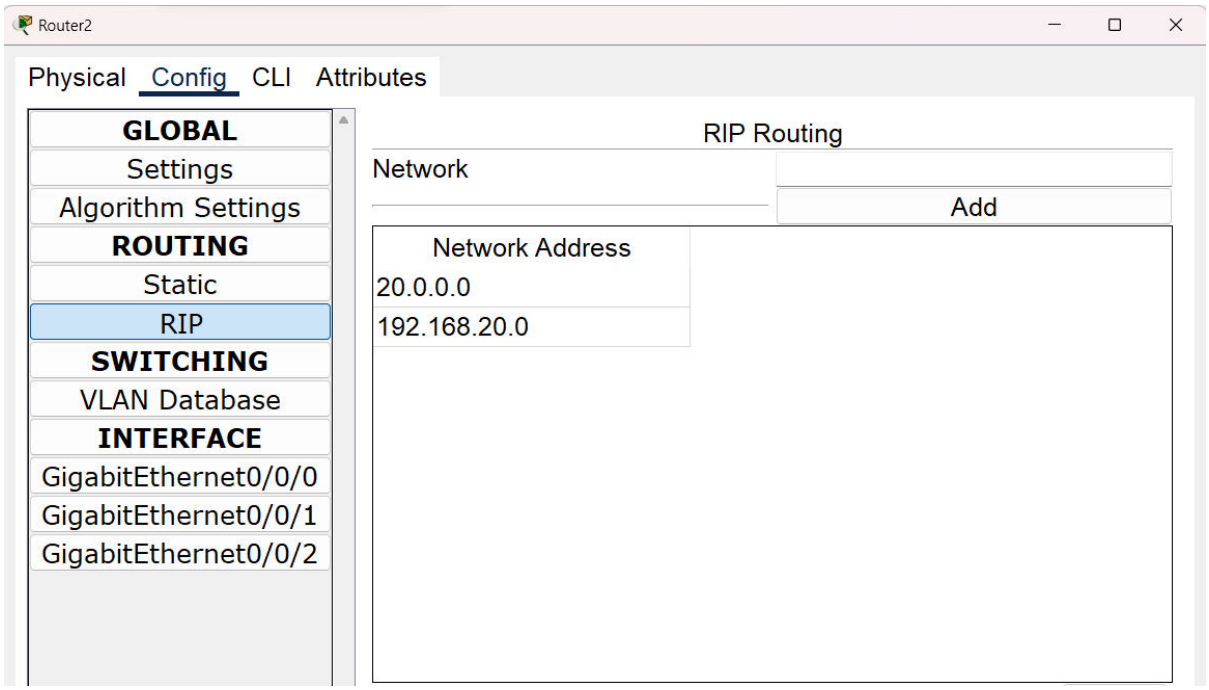


Step 2:

- Go to leftmost Router from LHS, go to Config>RIP
- Define the routing as follows:

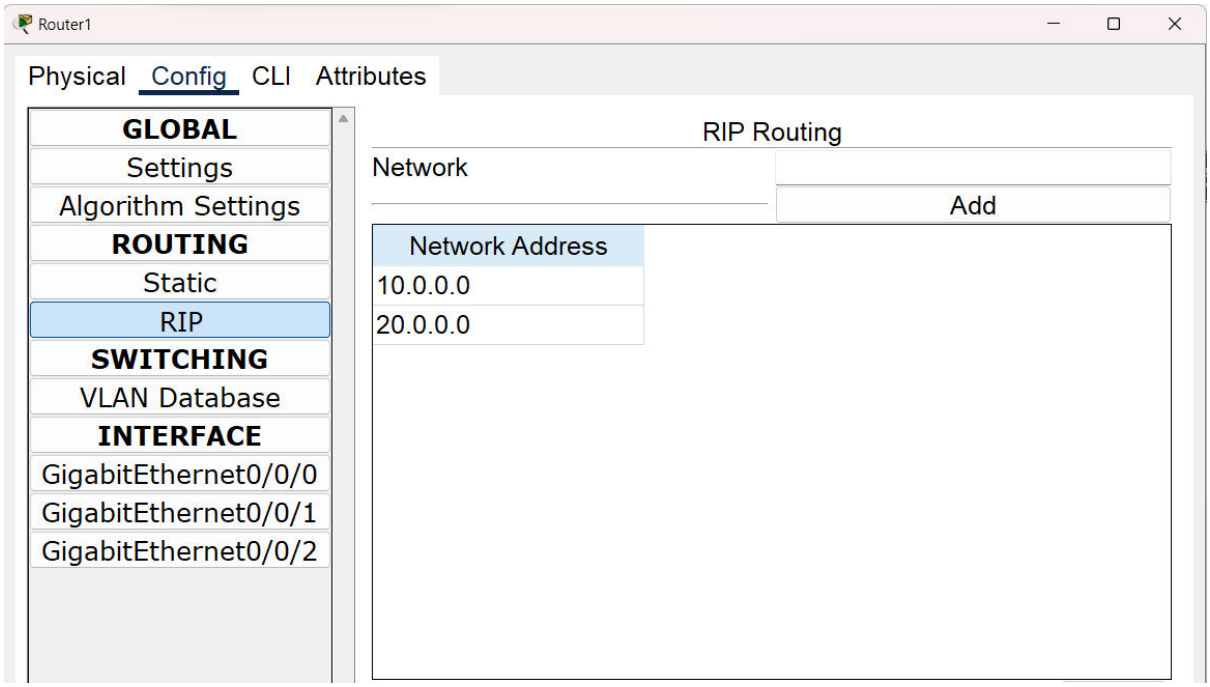


- Similarly do the same for rightmost router like this:



Step 3:

- Similarly go to the centremost router and configure it like this:



Step 4:

- That’s it, after 3-5 attempts, pings between any two PCs should work.

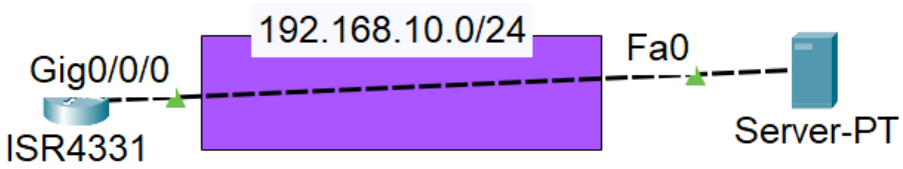
Using TFTP Server to backup and restore Router configuration

Devices:

- 1 x 4331 Routers
- 1 x Server-PT

Step1:

- Setup aforementioned devices like this as well as assign the respective IP addresses.



Step 2:

- Go to Router CLI and backup the running config to TFTP Server as follows:

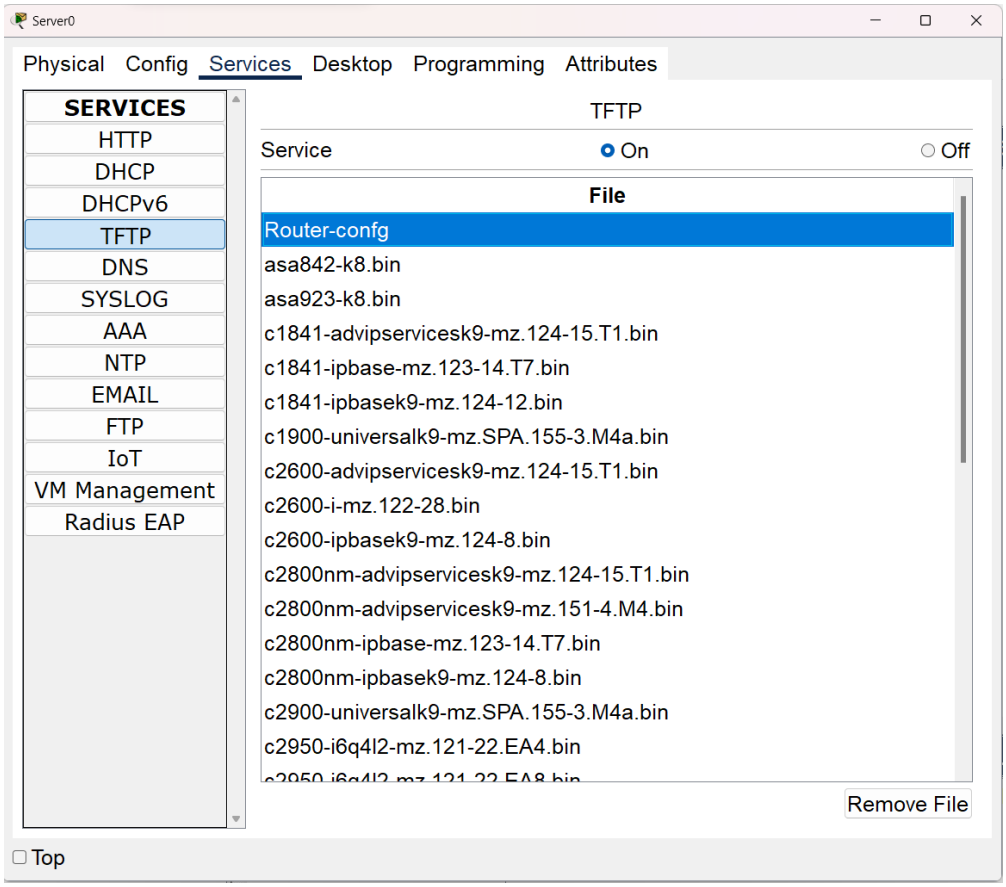
```
Router#
Router#
Router#
Router#copy startup-config tftp
Address or name of remote host []? 192.168.10.2
Destination filename [Router-confg]?

Writing startup-config...!!
[OK - 734 bytes]

734 bytes copied in 0 secs
Router#
```

Step 3:

- Go to Server > Services > TFTP and check the name of the config file uploaded.



Step 4:

- Erase the current Router config files as follows:

```
Router#
Router#
Router#
Router#erase startup-config
Erasing the nvram filesystem will remove all configuration files!
Continue? [confirm]
[OK]
Erase of nvram: complete
%SYS-7-NV_BLOCK_INIT: Initialized the geometry of nvram
Router#reload
Proceed with reload? [confirm]
Initializing Hardware ...
```

Step 5:

- Give any temporary IP address to Router so that TFTP request can be made
- Go back to Router CLI and restore config file as follows:

```
Router#copy tftp running-config
Address or name of remote host []? 192.168.10.2
Source filename []? Router-confg
Destination filename [running-config]?

Accessing tftp://192.168.10.2/Router-confg....
Loading Router-confg from 192.168.10.2: !
[OK - 734 bytes]

734 bytes copied in 3.011 secs (243 bytes/sec)
Router#
%SYS-5-CONFIG_I: Configured from console by console
```

That’s it, now the Router has restored the previously backup up configuration.

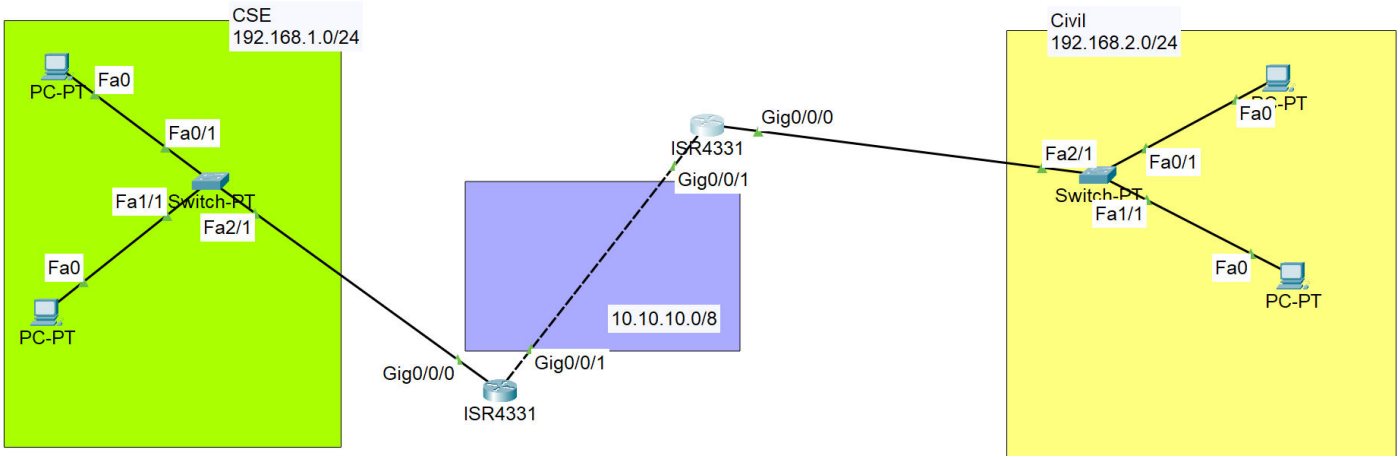
Using Standard Access List to block communication between certain hosts on different network

Devices:

- 4 x PCs
- 2 x 2960 Switch
- 2 x 4331 Routers

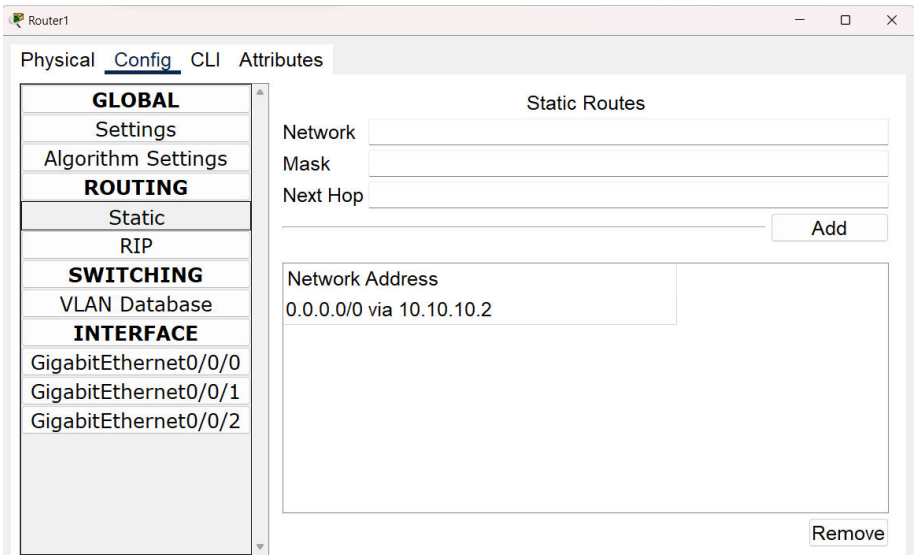
Step1:

- Setup aforementioned devices like this as well as assign the respective IP addresses.

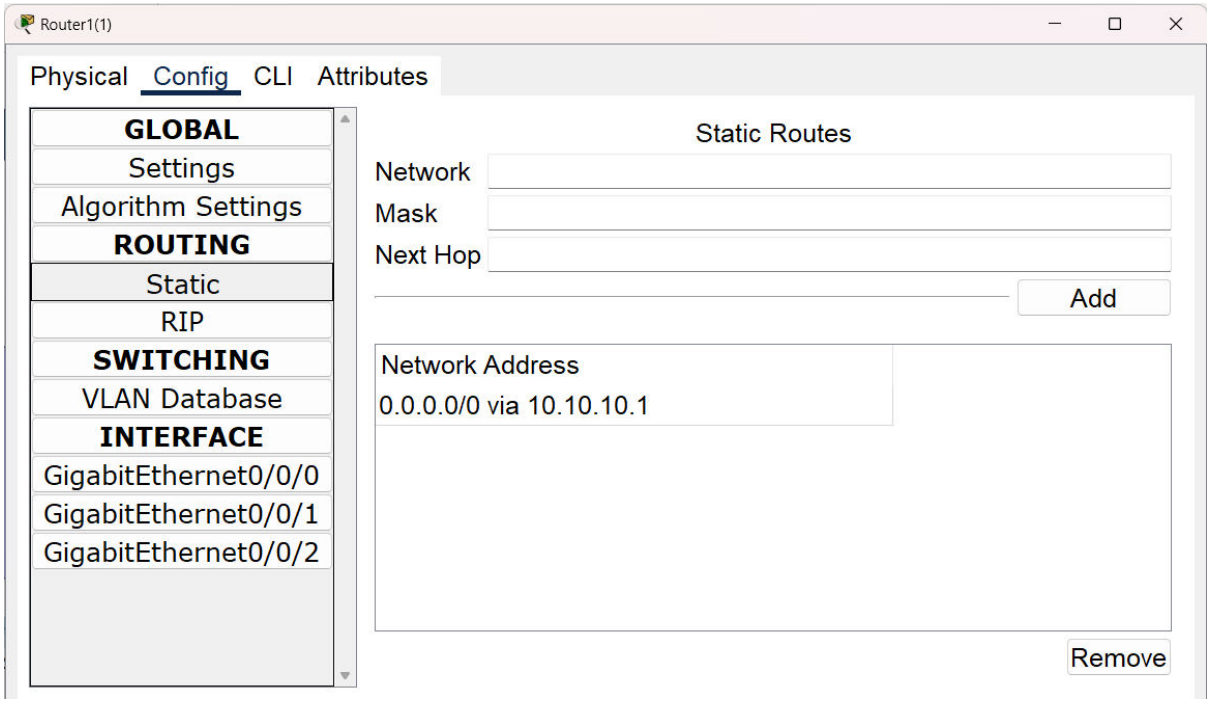


Step 2:

- Go to left Router, Config>Static and setup the following default static route.

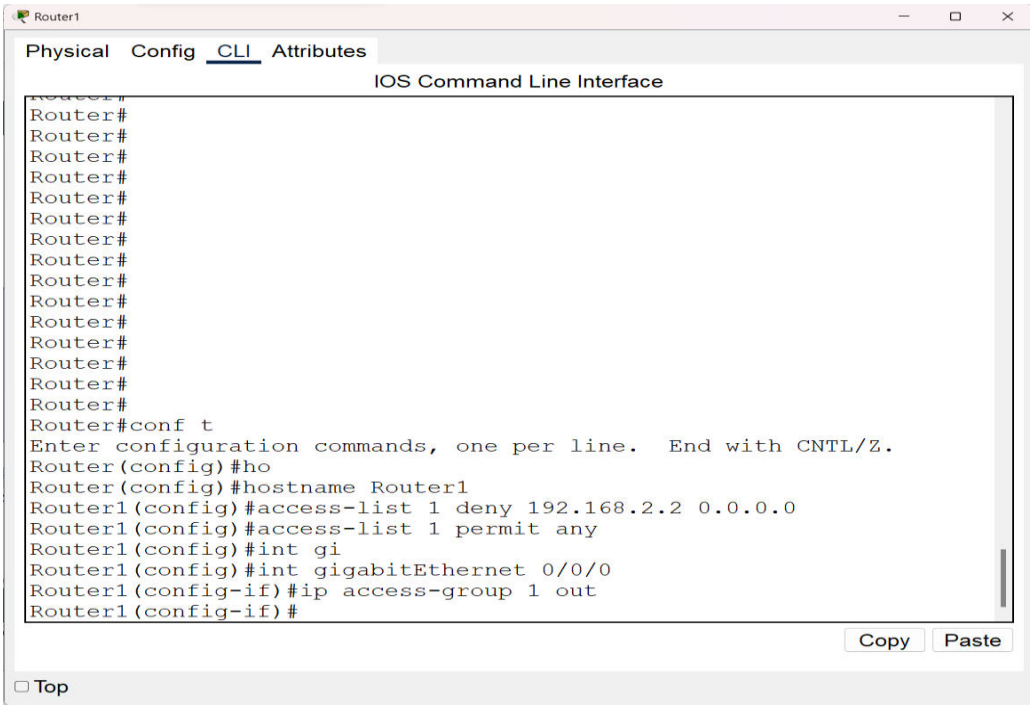


- Similarly do the same for right router like this:



Step 3:

- Go to the left Router CLI and configure the access list as follows:



Step 4:

- Now communication between CSE department and the host 192.168.2.2 should be forbidden.

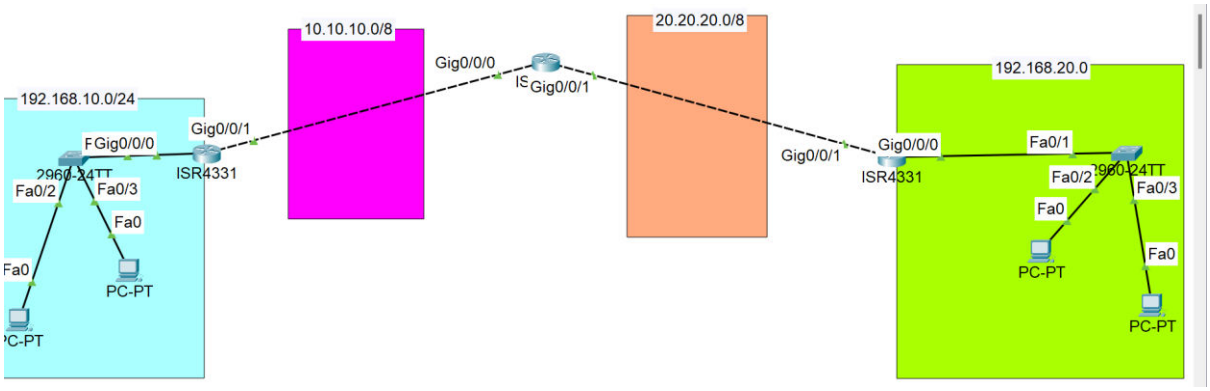
Using Static Routing to allow routers to communicate with each other

Devices:

- 4 x PCs
- 2 x 2960 Switch
- 2 x 4331 Routers

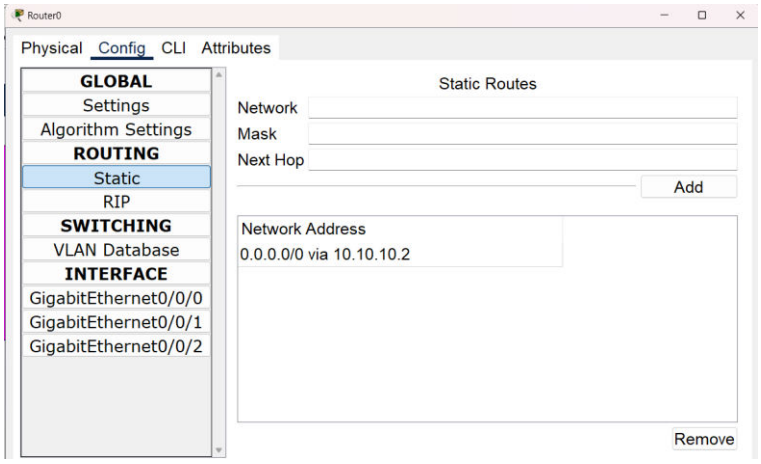
Step1:

- Setup aforementioned devices like this as well as assign the respective IP addresses.

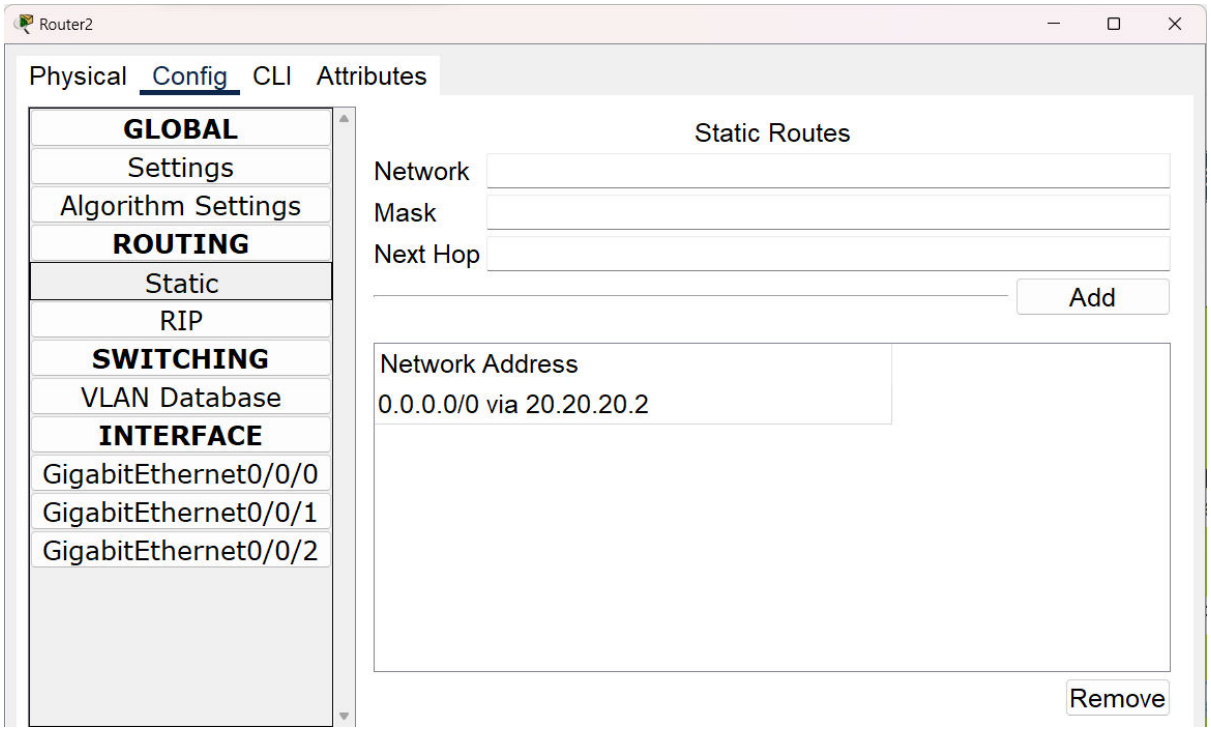


Step 2:

- Go to leftmost Router from LHS, go to Config>Static
- Define the default routing as follows:

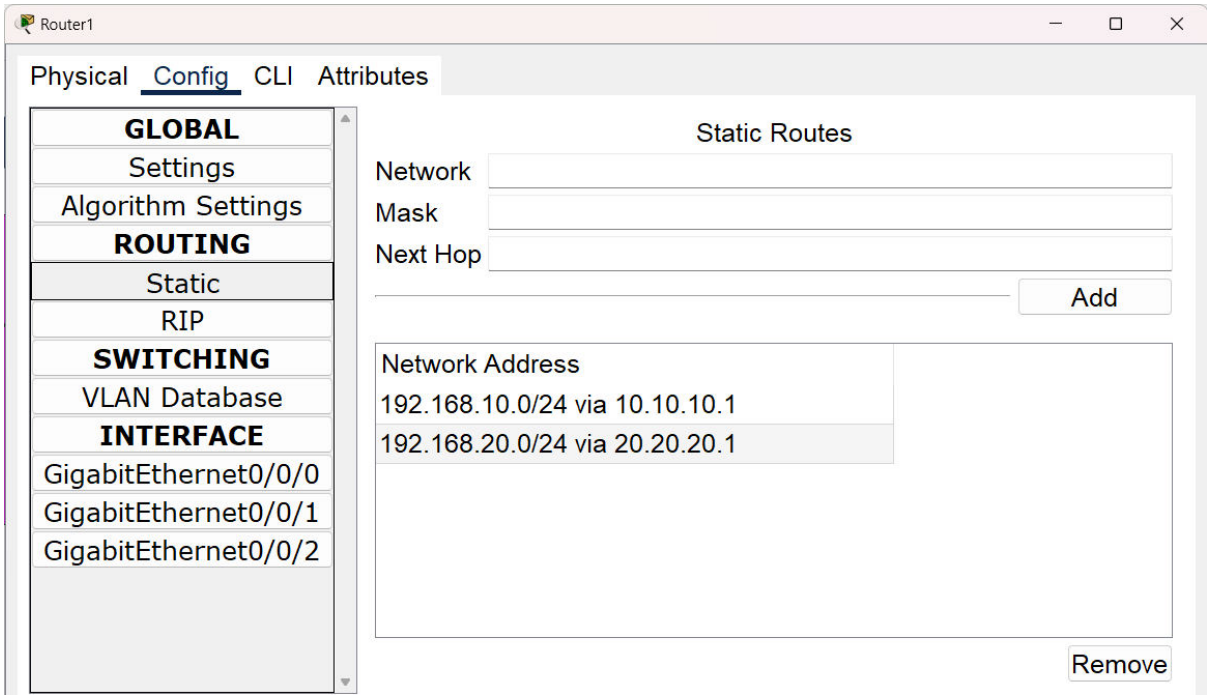


- Similarly do the same for rightmost router like this:



Step 3:

- Similarly go to the centremost router and configure it like this:



Step 4:

That’s it, after 3-5 attempts, pings between any two PCs should work. Make sure you don’t give up after initial failures :)

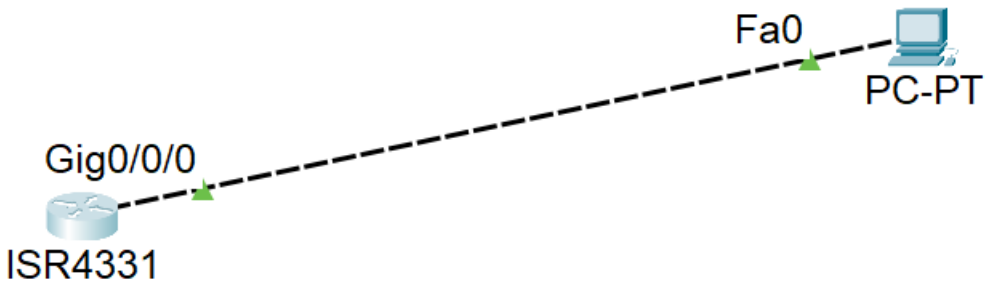
Using TELNET to access router CLI from a host PC

Devices:

- 1 x PCs
- 1 x 4331 Router

Step1:

- Setup aforementioned devices like this as well as assign the respective IP addresses.



Step 2:

Go to Router CLI and type the following

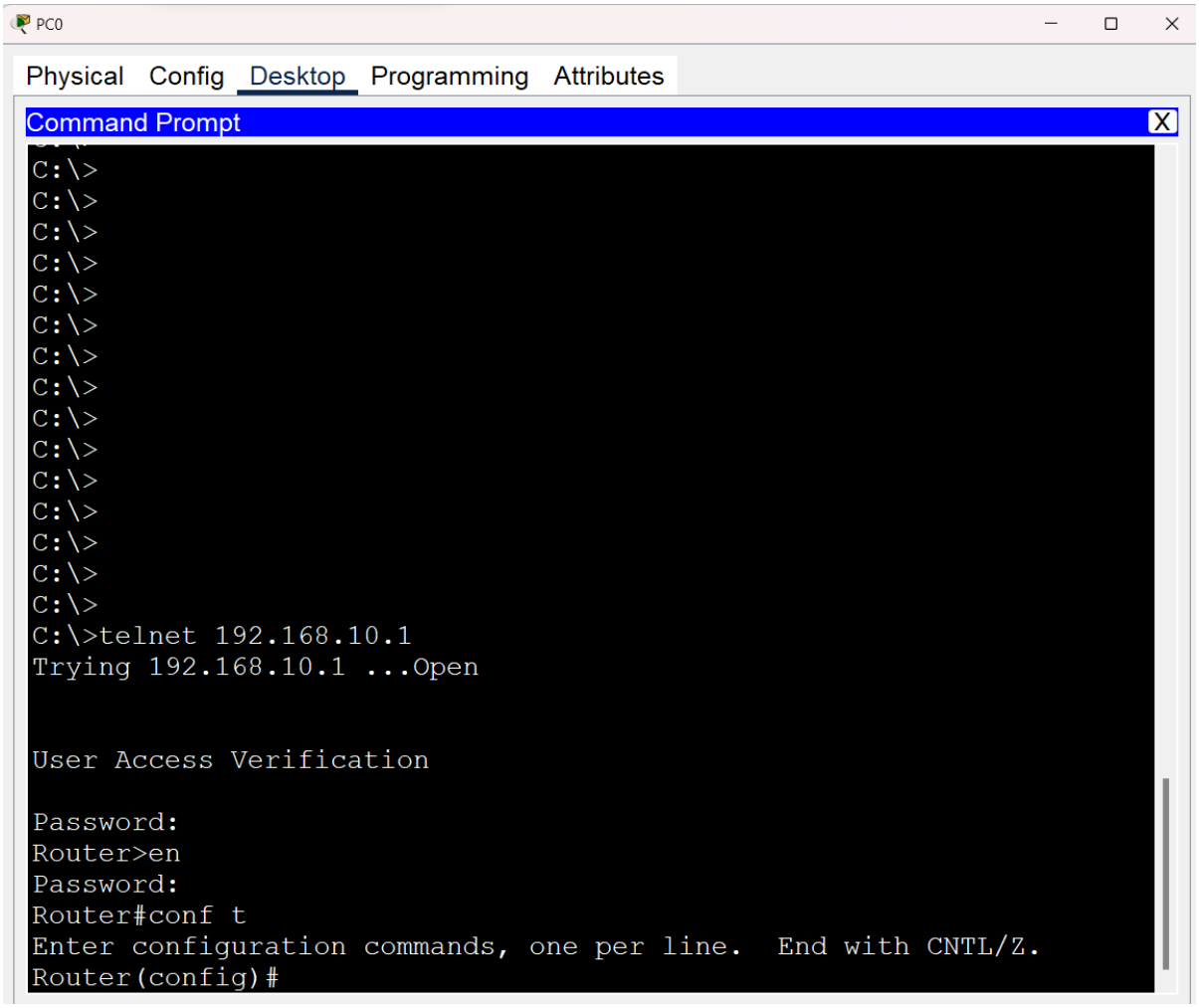
```
Router#
Router#
Router#
Router#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#enable secret hello123
Router(config)#int gig0/0/0
Router(config-if)#line vty 0 5
Router(config-line)#password hello123
Router(config-line)#exit
Router(config)#do wr
Building configuration...
[OK]
Router(config)#
```

Copy Paste

Top

Step 3:

Go to host PC, Desktop> cmd and type the following:



Step 5:

- You can now access and configure router from host PCs command line interface.

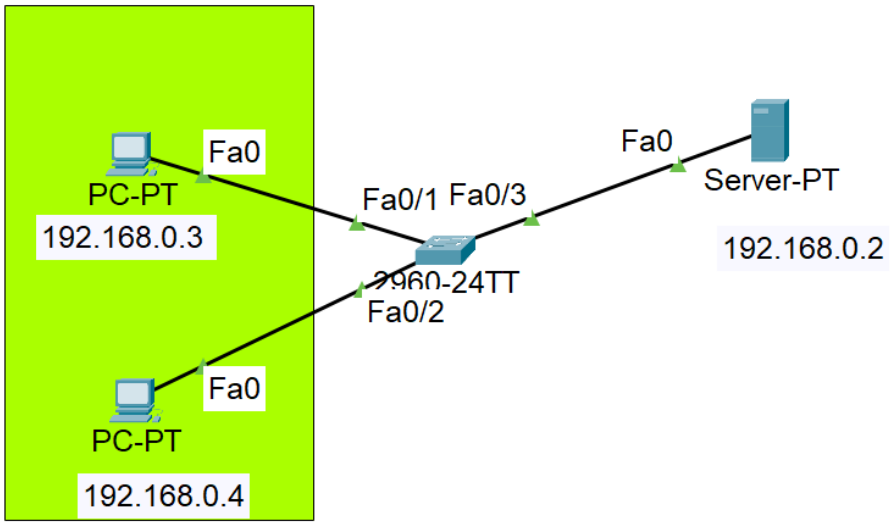
Using DNS server to map website URL to their IP address

Devices:

- 2 x PCs
- 1 x 2960 Switch
- 1 x Server-PT

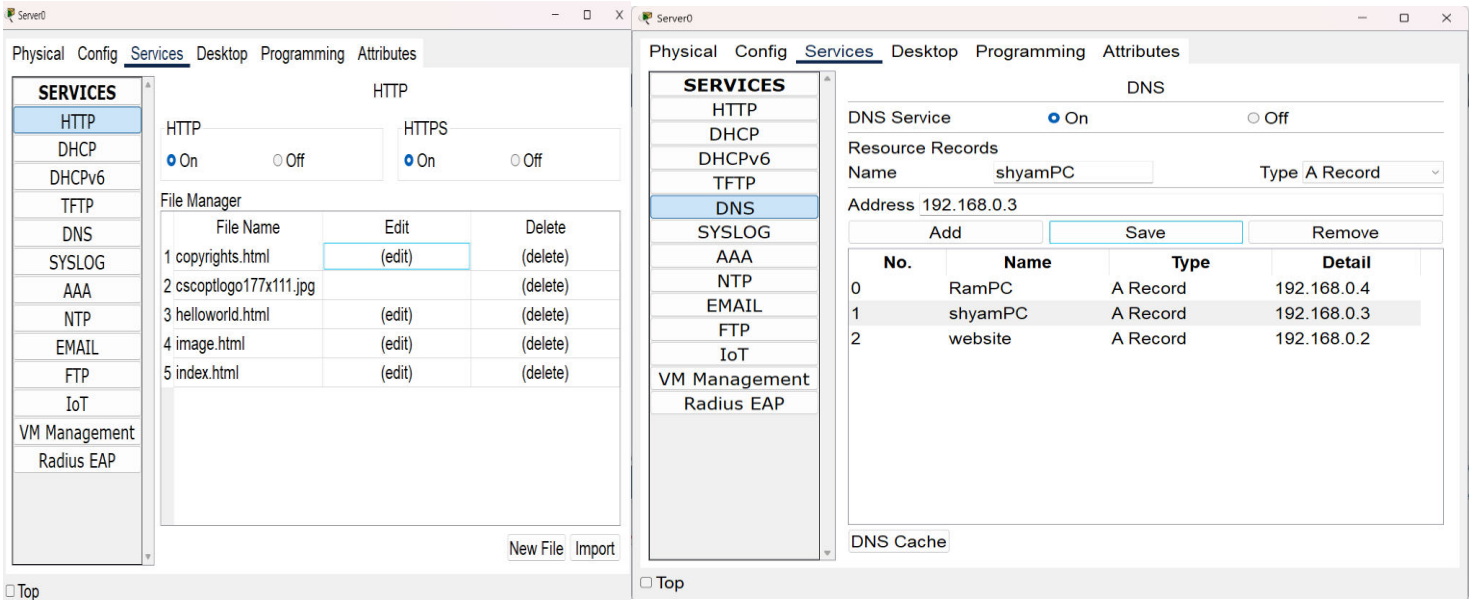
Step1:

- Setup aforementioned devices like this as well as assign the respective IP addresses.
- The DNS server of each PC should be same as IP address of the server.



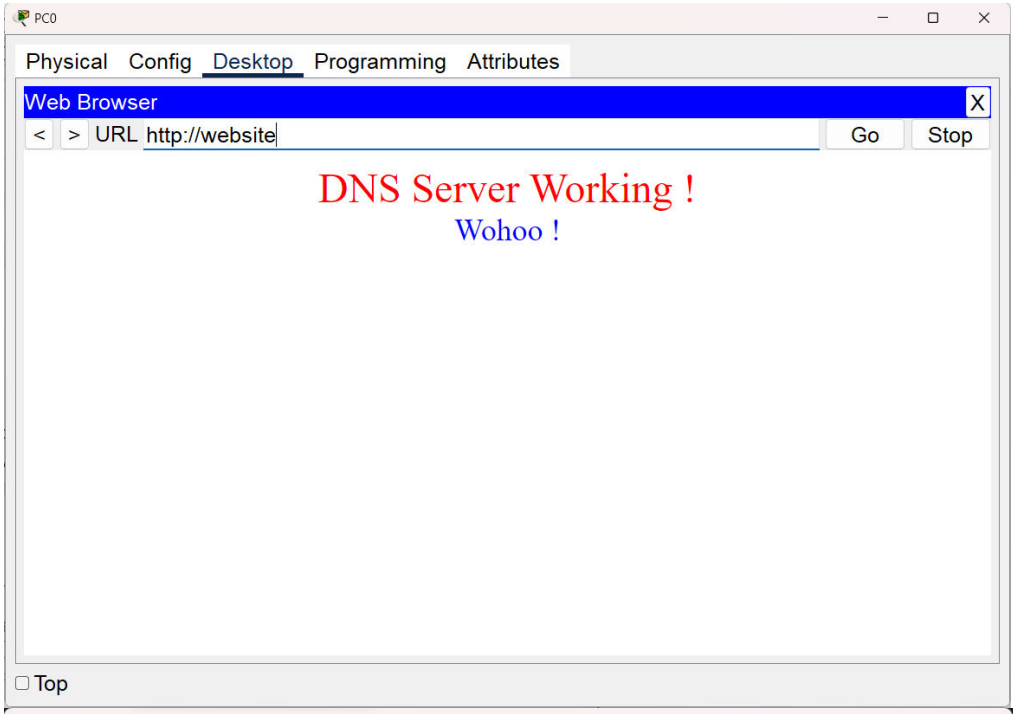
Step 2:

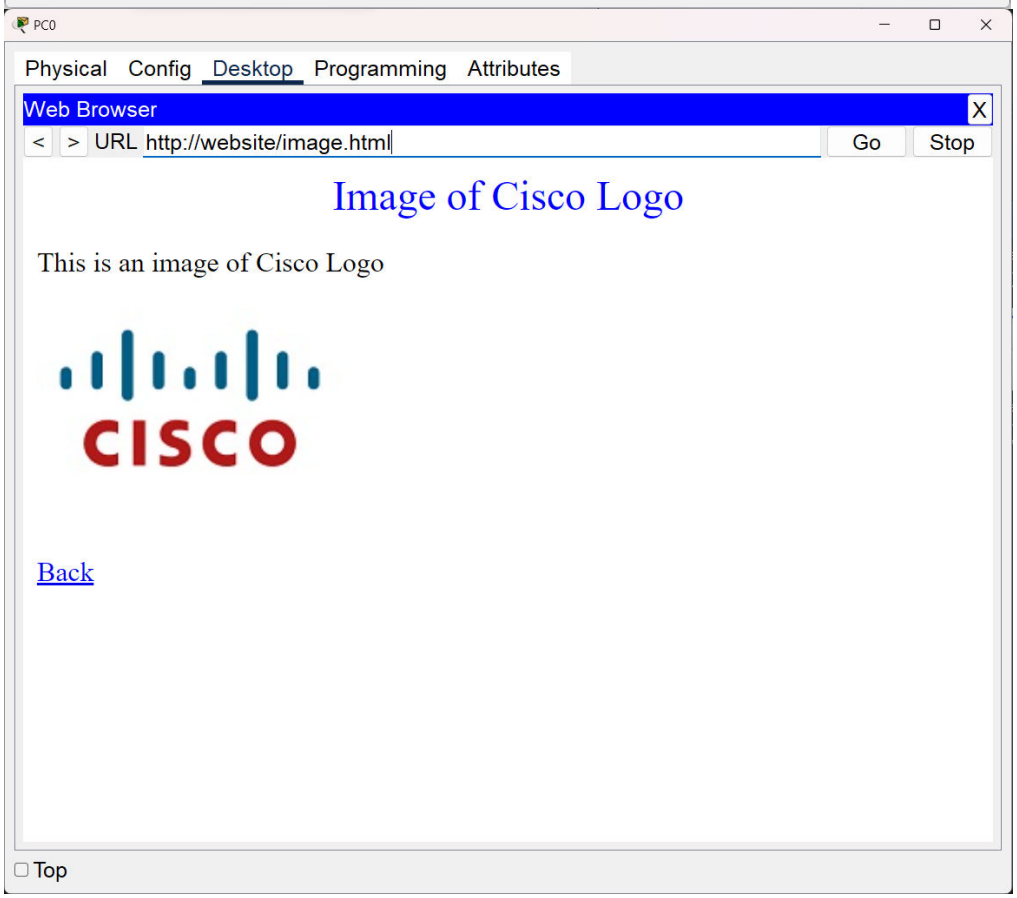
- Go to Server > Services > HTTP
- Edit any of these files(or create new ones) as per your wish in HTML-CSS. Name them what you want them to be fetched by from the browser.
- Go to DNS and turn DNS services on.
- Name your website homepage, and assign it's address the same as server's IP address and save it.
- (Optional) If you want DNS to work for inter-PC communication as well, add a new entry, with an address of PC and name it whatever you wish, repeat the same for all PCs for which you want this to work.



Step 3:

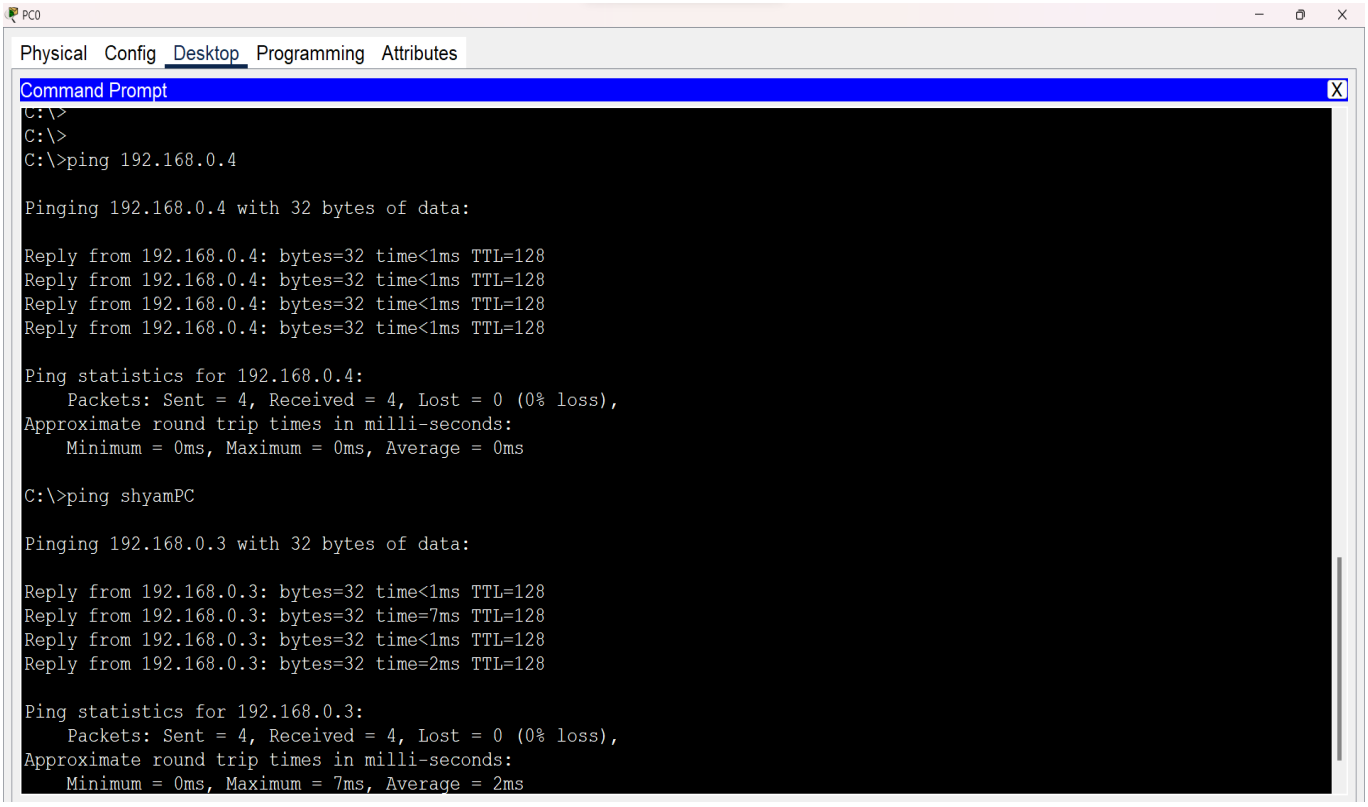
- Go to any PC > Desktop > Web Browser
- Type 192.168.0.2 or “website” (The name of your homepage in DNS record), your index.html file should be opened.
- Add a slash after the URL and type the name of any other HTML file you want to open.





Step 5: (Optional)

- Go to any PC, then Desktop > Command Prompt and ping any other PC in the network with it's IP address or it's name you entered in the DNS records.



That's it, now the DNS server should work for both applications.