
**Department of Computer Science and Engineering**
**Islamic University of Technology (IUT)**
A subsidiary organ of OIC


# Laboratory Report

CSE 4512: Computer Networks Lab


**Name:** Md Farhan Ishmam
**Student ID:** 180041120
**Section:** CSE-1
**Semester:** Fifth
**Academic Year:** 2021


**Date of Submission:** 09-08-21

**Title:** Configuring IPv6 addressing scheme in a network topology and configuration of Inter VLAN routing with IPv6.

# Objective:

**Task - 1**

- Configure IPv6 Addressing on the Router
- Configure IPv6 Addressing on Servers
- Configure IPv6 Addressing on Clients
- Test and Verify Network Connectivity

**Task - 2**

- Configure Layer 3 Switching
- Configure IPv6 Inter-VLAN Routing

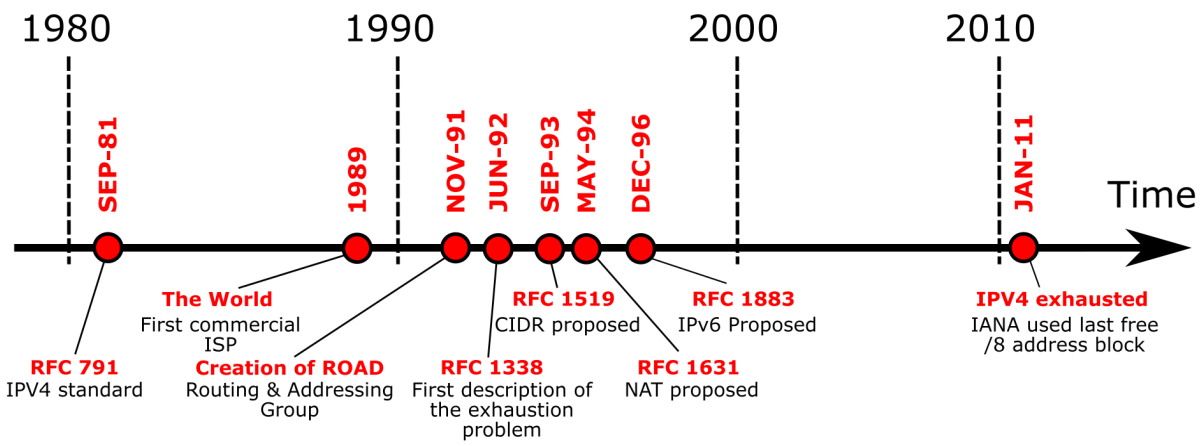
# Devices/ software Used:

1. Device: Windows PC
2. Software: Cisco Packet Tracer 7.3.0

# Theory:

## IPv6:

IPv6 stands for **Internet Protocol Version 6** is the most recent version of IP and also the successor to IPv4 or Internet Protocol Version 4. IPv6 was developed by the Internet Engineering Task Force (IETF) to solve the problems of IPv4, in particular, **IPv4 Address Exhaustion**. In the 1990s, it became clear that IPv4 didn't have enough space to allocate all the devices connected to the internet. This problem of depletion of IPv4 addresses is known as IPv4 Address Exhaustion. The problem occurred because devices connected to the internet had to be provided unique IP addresses. During the inception of the internet, it was thought that the IPv4 address space would be sufficient to provide unique identification to all the devices connected to the internet. But in the 1990s, the dot-com bubble occurred and the number of devices connected to the internet increased sharply in both household and commercial spaces. Such a sharp growth is the reason behind the development and deployment of IPv6. In the modern internet, IPv6 coexists with IPv4.




Theoretically, IPv4 can allocate $2^{32}$ unique addresses. So, it has a 32 bit address space. IPv6 increased this number to $2^{128}$ or a **128 bit** address space. This means to represent a single IP address in IPv6, we need 128 bits in total. Although, theoretically, $2^{128}$ is possible but in reality, the number is quite lesser due to some reserved address spaces. Even with such reservations, the remaining address space is large enough to allocate IP addresses for the whole internet for the upcoming decades.

The IPv6 addresses can be divided into 3 categories: **Unicast Address**, **Anycast Address** and **Multicast Address**. A Unicast Address defines the address of a single interface. This interface can be a computer or a router. A packet sent to a Unicast address will be delivered to the intended interface it was sent to. On the other hand, an Anycast Address defines the address of a group of interfaces, which can be a group of computers or routers. This group will share the same address. When a packet is sent to such an address, it will be delivered to the most reachable member of the group. A Multicast Address is similar to Anycast Address but with one key difference. In multicasting, each member of the group will receive a copy of the packet. This is why there is a dedicated block for Multicast Addresses in IPv6 but Unicast, and Anycast Addresses share the same block.

Similar to IPv4, an IPv6 address can be divided into 2 parts: the **Network Identifier** and the **Interface Identifier**. The most significant 64 bits represent the Network Identifier, and the least significant 64-bits represent the Interface Identifier. But, such division of the IP address is valid for unicast and anycast addresses, not for multicast addresses. The Network Identifier is similar to the Network Portion in IPv4. This portion determines which network the interface belongs to and is responsible for routing the packet to the correct network. The interface identifier is similar to the Host ID in IPv4. But in IPv4 there was no relationship between the host ID and the physical address (MAC address). That is because the MAC address is of 48 bits and the host address has to be less than 32 bits. So, the host address space isn't big enough to allocate the MAC address. On the other hand, in IPv6, the interface ID is much larger, which is 64 bits. Any physical address whose length is lesser than or equal to 64 bits can be embedded within this interface ID. The common physical addressing scheme used here is the 64 bit extended unique identifier or EUI-64 defined by IEEE, and the 48 bit physical address defined by Ethernet, which is also called MAC address.

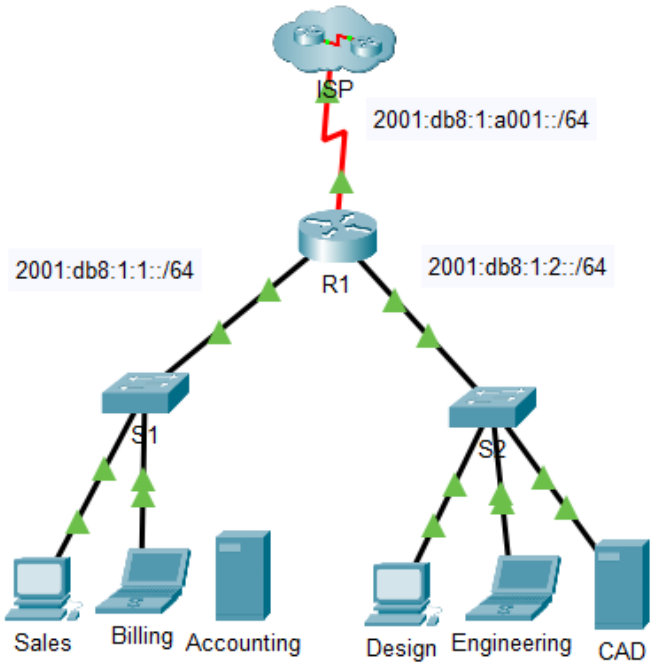
As discussed previously, an IPv6 address is 128 bits long. The whole address is divided into 8 groups, each group having 4 hexadecimal digits. Each group consists of 16 bits. Similar to IPv4's terminology, each group is formed of 2 octets. There are many kinds of notation to represent an IPv6 address. As it is a 128 bit binary number, humans don't use the binary notation to represent an IPv6 address. The dotted decimal notation, which was widely used in IPv4, can also be an approach but is rarely used. The most readable form of notation is the **colon hexadecimal notation** where the hexadecimal values of the groups are shown and each group is separated by a colon. An example of IPv6 address in colon hexadecimal notation is given below:

2001:0db8:85a3:0000:0000:8a2e:0370:7334

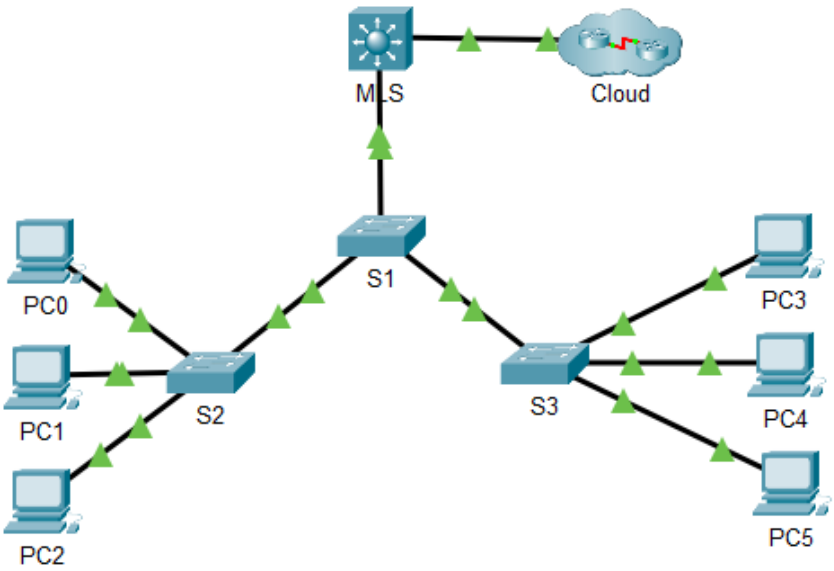
IPv6 follows two rules to keep its representation concise. If there is one or more leading zeros in a group then they can be omitted. For example - if the group has *00db*, it will be changed to *db*. The second rule is that consecutive groups of zeros can be replaced with a double colon (::) But, this replacement can only be done once in an address or else, it will be very difficult for the reader to interpret the address. An example of both of these rules can be seen with the address *2001:0db8:0000:0000:0000:ff00:0042:8329* After removing the leading zeroes this address becomes *2001:db8:0:0:0:ff00:42:8329*. Then removing consecutive sections of zeros with a double colon, it becomes *2001:db8::ff00:42:8329.* We can see that the final address after using our shortening techniques is quite smaller and more readable than the initial address. An extreme example of the above rules can be shown while representing the IPv6 Loopback Address. Initially, the Loopback Address can be written as *0000:0000:0000:0000:0000:0000:0000:0001*. After applying the two rules, it becomes just *::1*.
IPv6 has a special address called **link-local address** which has to be assigned to every networking interface that has IPv6 enabled. Therefore, an IPv6 host has more than one IPv6 address assigned to its IPv6 enabled interface. The link-local address is only valid for communicating within the network segment or the broadcast domain of the connected host, which is why it is called "link-local". The link-local address of the router is used as the default gateway for the connected hosts. Link-local address is also used as part of the neighbor discovery protocol and automatic address configuration. Link-local address uses the prefix *fe80::/10*. Of the 64 bits of a link-local addresses' network component, the most significant 10 bits (1111111010) correspond to the IANA-reserved "global routing prefix " for link-local addresses, while the other 54 bits or the "subnet ID" is zero. Rest of the 64-bits is used as the interface identifier.

# Diagram of the experiment:

**Task #01:**




**Task #02:**



# Working Procedure:

## TASK #01:

### Part 1: Configure IPv6 Addressing on the Router

### Step 1: Enable the router to forward IPv6 packets

- We click on the router R1 and then go to the CLI tab
- After going to the CLI, we press Enter. Then we access the privileged execution mode, and then go to the terminal. The following commands are used:

```
R1>en
R1#conf terminal
```

- Now, we enable ipV6 unicast routing inorder to allow the router to forward IPv6 Packets. The following command is to be entered:

```
R1(config)#ipv6 unicast-routing
```

```
R1>en
R1#conf terminal
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#
R1(config)#ipv6 unicast-routing
R1(config)#int g0/0
R1(config-if)#ipv6 address 2001:db8:1:1::1/64
R1(config-if)#ipv6 address fe80::1 link-local
R1(config-if)#no shutdown

R1(config-if)#
%LINK-5-CHANGED: Interface GigabitEthernet0/0, changed state to
up

%LINEPROTO-5-UPDOWN: Line protocol on Interface
GigabitEthernet0/0, changed state to up
```

### Step 2: Configure IPv6 addressing on GigabitEthernet0/0

- We will now set the Gigabit Ethernet 0/0 of Router R1. We enter the following command to access the Gigabit Ethernet 0/0:

```
R1(config)#int g0/0
```

- Now we configure the IP address of the interface using this command:

```
R1(config-if)#ipv6 address 2001:db8:1:1::1/64
```

- Then, we configure the link local IP address of the interface using the command:

```
R1(config-if)#ipv6 address fe80::1 link-local
```

- Finally, we activate the interface using the command:

```
R1(config-if)#no shutdown
```

### Step 3: Configure IPv6 addressing on GigabitEthernet0/1

- We will now set the Gigabit Ethernet 0/1 of Router R1. We enter the following command to access the Gigabit Ethernet 0/1:

```
R1(config-if)#int g0/1
```

- Now we configure the IP address of the interface using this command:

```
R1(config-if)#ipv6 address 2001:db8:1:2::1/64
```

- Then, we configure the link local IP address of the interface using the command:

```
R1(config-if)#ipv6 address fe80::1 link-local
```

- Finally, we activate the interface using the command:

```
R1(config-if)#no shutdown
```

```
R1(config-if)#int g0/1
R1(config-if)#ipv6 address 2001:db8:1:2::1/64
R1(config-if)#ipv6 address fe80::1 link-local
R1(config-if)#no shutdown

R1(config-if)#
%LINK-5-CHANGED: Interface GigabitEthernet0/1, changed state to
up

%LINEPROTO-5-UPDOWN: Line protocol on Interface
GigabitEthernet0/1, changed state to up
```

### Step 4: Configure IPv6 addressing on Serial0/0/0

- We will now set the Serial 0/0/0 of Router R1. We enter the following command to access the Serial0/0/0:

```
R1(config-if)#int s0/0/0
```

- Now we configure the IP address of the interface using this command:

```
R1(config-if)#ipv6 address 2001:db8:1:a001::2/64
```

- Then, we configure the link local IP address of the interface using the command:

```
R1(config-if)#ipv6 address fe80::1 link-local
```

- Finally, we activate the interface using the command:

```
R1(config-if)#no shutdown
```

```
R1(config-if)#int s0/0/0
R1(config-if)#ipv6 address 2001:db8:1:a001::2/64
R1(config-if)#ipv6 address fe80::1 link-local
R1(config-if)#no shutdown

R1(config-if)#
%LINK-5-CHANGED: Interface Serial0/0/0, changed state to up

R1(config-if)#
```

### Step 5: Verify IPv6 addressing on R1

- After configuring all the interfaces, we will now exit the configuration mode on the Router R1. The following commands are to be entered:

```
R1(config-if)#exit
R1(config)#exit
```

- Then we verify the addressing configured by enter the following command:

```
R1#show ipv6 interface brief
```

- We will check the configuration with the given table. If any address is incorrect, then we will remove the incorrect address first and then we enter the correct address. If we don't do so, then both the correct and incorrect addresses will remain in the router.

- Finally, after all the changes are verified, we will save the router configuration to the NVRAM. By doing so, even if we turn off the router, the changes will remain. That is because NVRAM is a form of non-volatile memory and can store the configuration even when we turn off the router. The following command is to be entered to save the changes:

```
R1#copy running-config startup-config
```

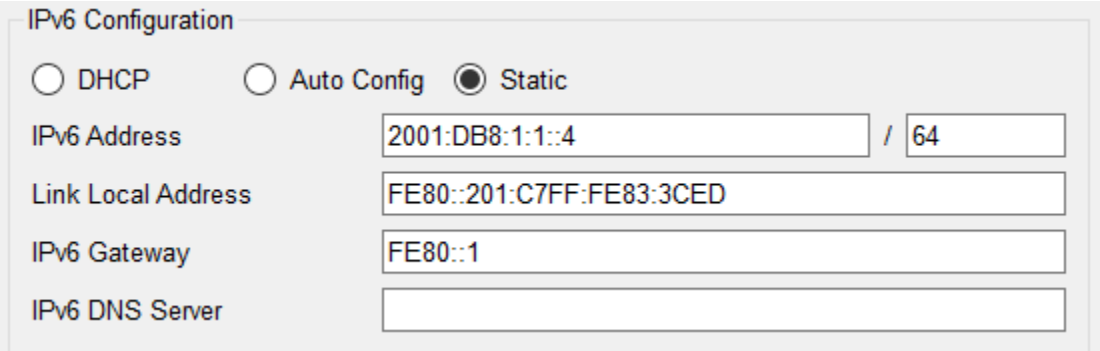
```
R1(config-if)#exit
R1(config)#exit
R1#
%SYS-5-CONFIG_I: Configured from console by console

R1#show ipv6 interface brief
GigabitEthernet0/0         [up/up]
    FE80::1
    2001:DB8:1:1::1
GigabitEthernet0/1         [up/up]
    FE80::1
    2001:DB8:1:2::1
GigabitEthernet0/2         [administratively down/down]
    unassigned
Serial0/0/0                [up/up]
    FE80::1
    2001:DB8:1:A001::2
Serial0/0/1                [administratively down/down]
    unassigned
Vlan1                      [administratively down/down]
    unassigned
R1#copy running-config startup-config
Destination filename [startup-config]?
Building configuration...
[OK]
```

## Part 2: Configure IPv6 Addressing on the Servers

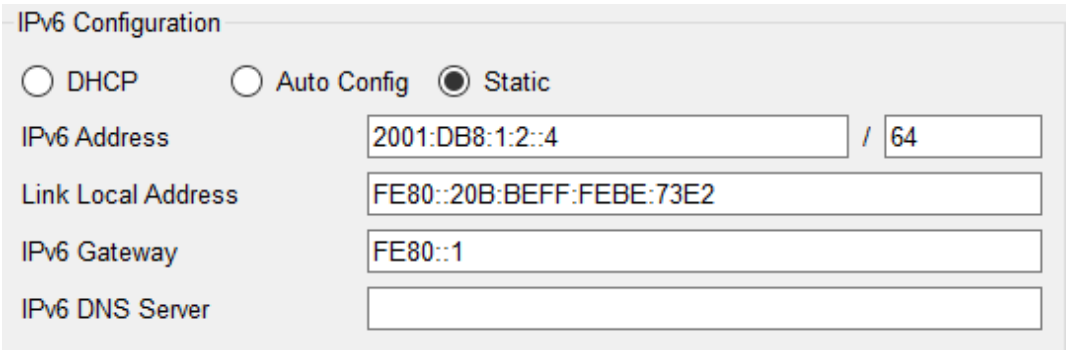
### Step 1: Configure IPv6 addressing on the Accounting Server

- First, we click on the Accounting Server. Then we click on the Desktop tab > IP Configuration
- Then, we set the IPv6 Address to 2001:db8:1:1::4 with a prefix of /64
- Finally, we set the IPv6 Gateway to the link-local address, fe80::1




### Step 2: Configure IPv6 addressing on the CAD Server

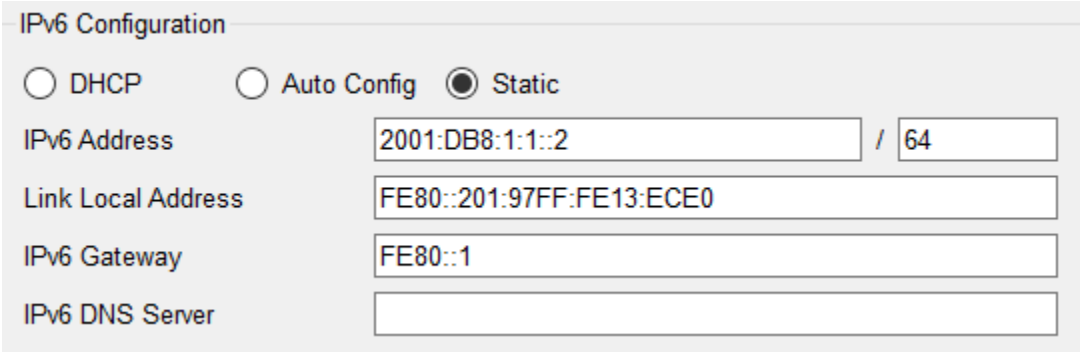
- First, we click on the CAD Server. Then we click on the Desktop tab > IP Configuration.
- Then, we set the IPv6 Address to 2001:db8:1:2::4/64 with a prefix of /64
- Finally, we set the IPv6 Gateway to the link-local address, fe80::1




### Part 3: Configure IPv6 Addressing on the Clients

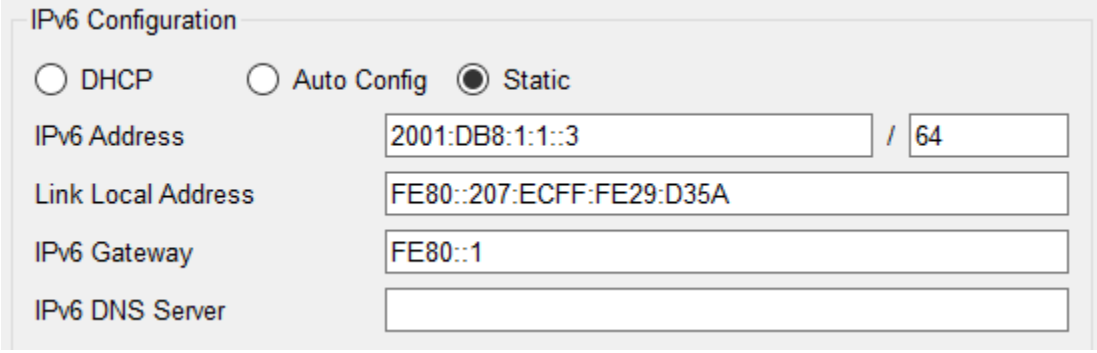
### Step 1: Configure IPv6 addressing on the Sales Clients

- First, we click on the Sales Client. Then we click on the Desktop tab > IP Configuration.
- Then, we set the IPv6 Address to 2001:db8:1:1::2/64 with a prefix of /64
- Finally, we set the IPv6 Gateway to the link-local address, fe80::1



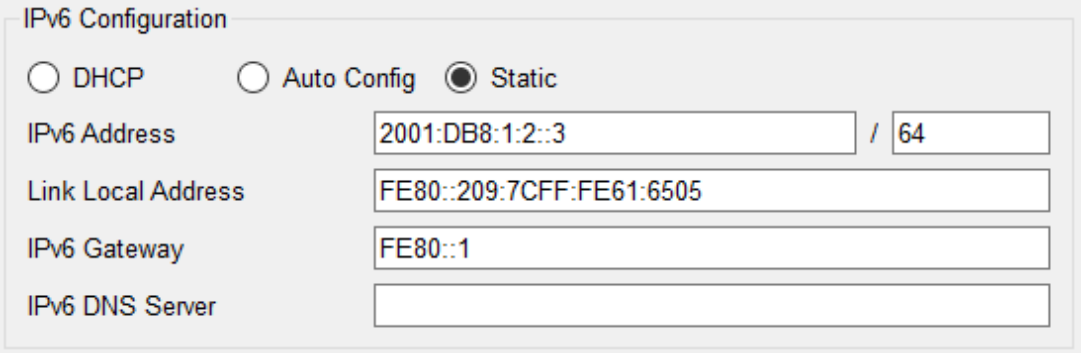

### Step 2: Configure IPv6 addressing on the Billing Clients

- First, we click on the Billing Client. Then we click on the Desktop tab > IP Configuration.
- Then, we set the IPv6 Address to 2001:db8:1:1::3/64 with a prefix of /64
- Finally, we set the IPv6 Gateway to the link-local address, fe80::1



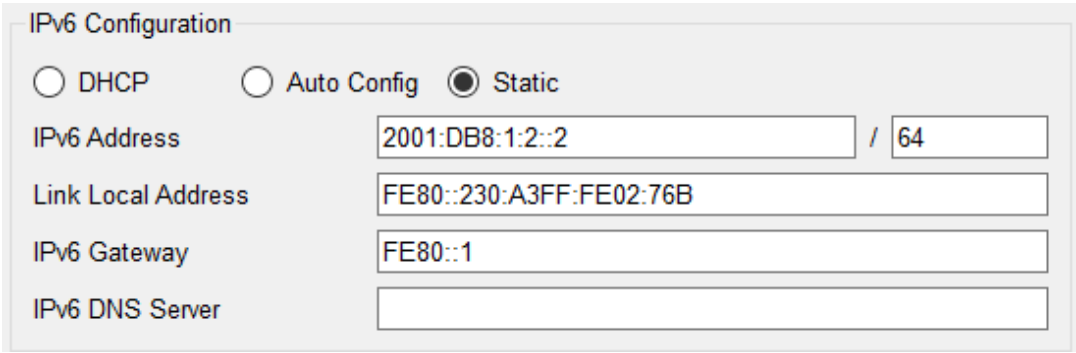
### Step 3: Configure IPv6 addressing on the Engineering Clients

- First, we click on the Engineering Client. Then we click on the Desktop tab > IP Configuration.
- Then, we set the IPv6 Address to 2001:db8:1:2::3  with a prefix of /64
- Finally, we set the IPv6 Gateway to the link-local address, fe80::1




### Step 4: Configure IPv6 addressing on the Design Clients

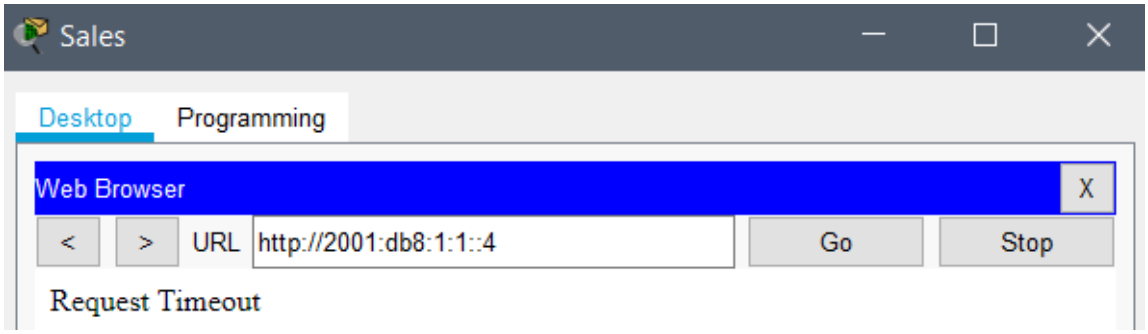
- First, we click on the Design Client. Then we click on the Desktop tab > IP Configuration.
- Then, we set the IPv6 Address to 2001:db8:1:2::2/64  with a prefix of /64
- Finally, we set the IPv6 Gateway to the link-local address, fe80::1



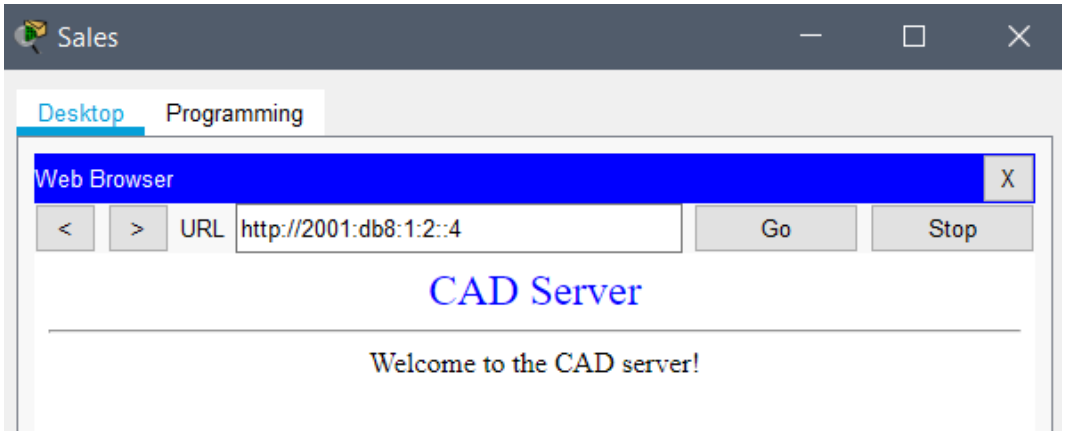

## Part 4: Test and Verify Network Connectivity

### Step 1: Open the server web pages from Sales

- We click Sales and go to the Desktop tab.
- We click the Web Browser. Enter 2001:db8:1:1::4 in the URL box and then, click Go. The Accounting website didn't appear because the Accounting Server isn't connected to the rest of the network. It shows Request Timeout.

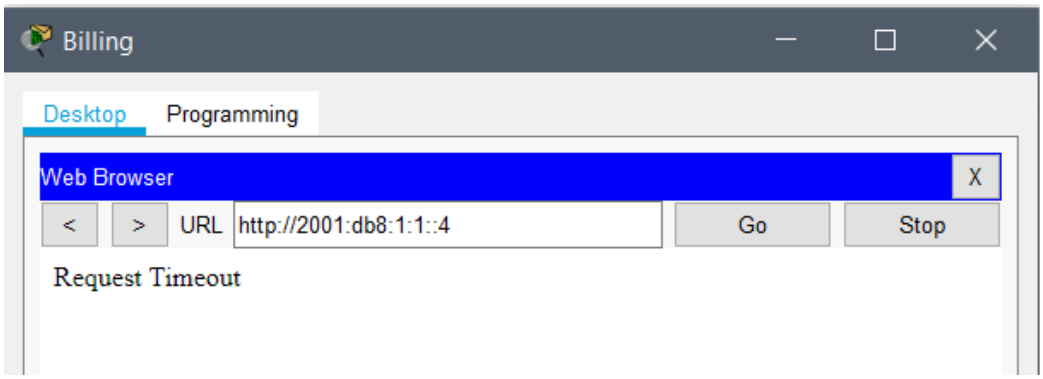



- Then we enter 2001:db8:1:2::4 in the URL box and click Go. The CAD website would appear

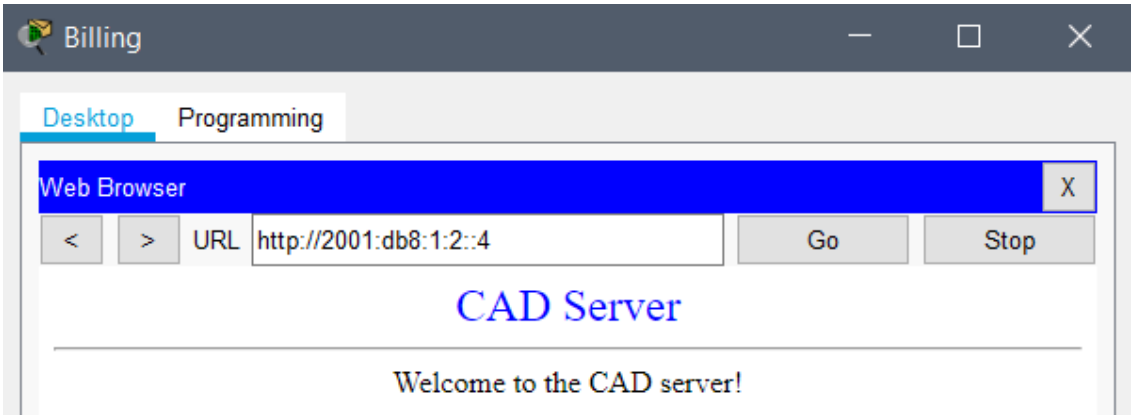



**Step 2: Open the server web pages from Billing**

- We click Billing and go to the Desktop tab.
- We click the Web Browser. Enter 2001:db8:1:1::4 in the URL box and then, click Go. The Accounting website didn't appear because the Accounting Server isn't connected to the rest of the network. It shows Request Timeout.

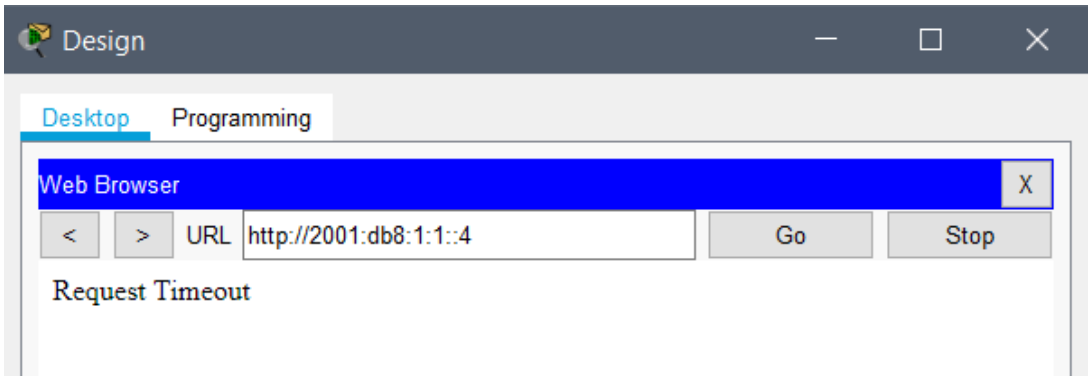


- Then we enter 2001:db8:1:2::4 in the URL box and click Go. The CAD website would appear

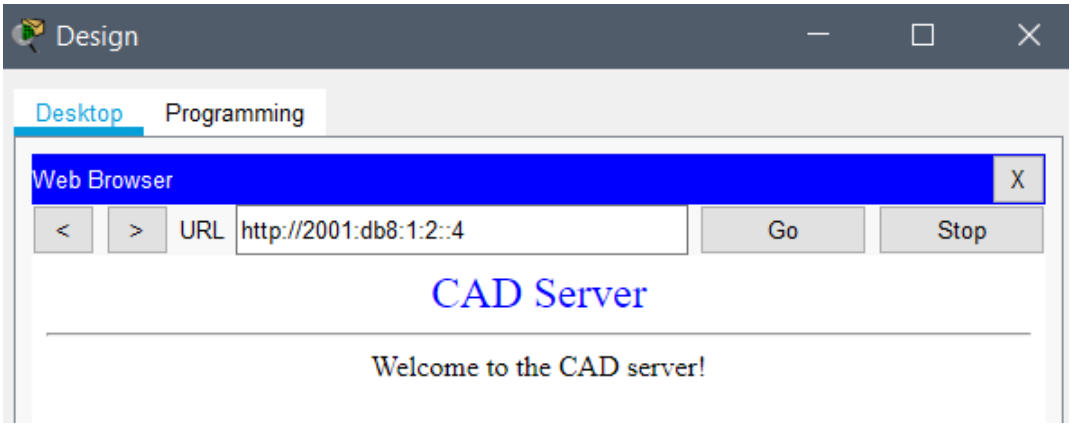



**Step 3: Open the server web pages from Design**

- We click Design and go to the Desktop tab.
- We click the Web Browser. Enter 2001:db8:1:1::4 in the URL box and then, click Go. The Accounting website didn't appear because the Accounting Server isn't connected to the rest of the network. It shows Request Timeout

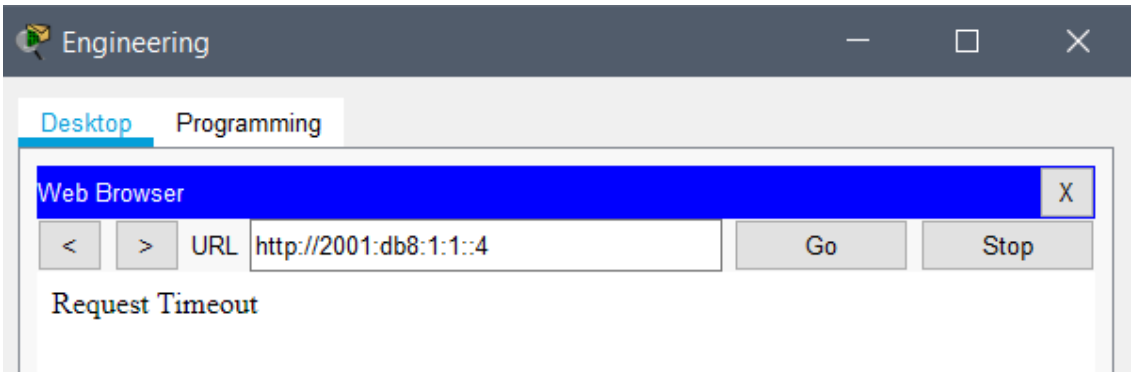



- Then we enter 2001:db8:1:2::4 in the URL box and click Go. The CAD website would appear.

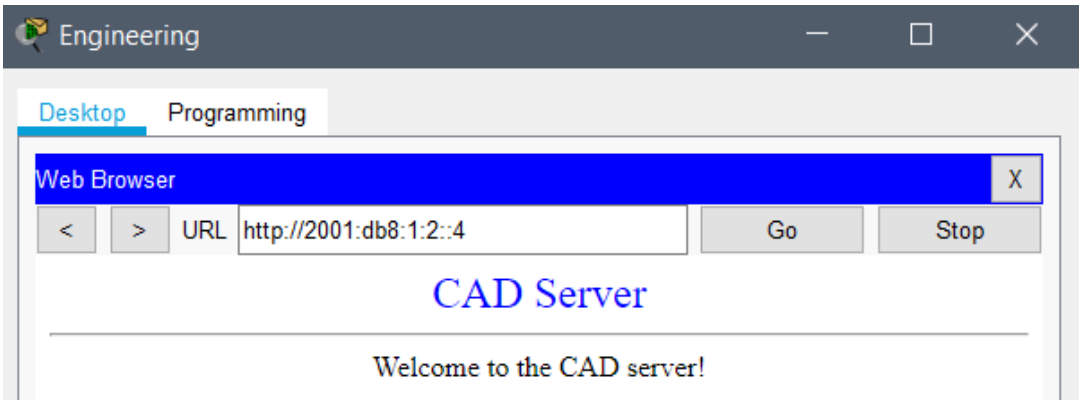


### Step 4: Open the server web pages from Engineering

- We click Engineering and go to the Desktop tab.
- We click the Web Browser. Enter 2001:db8:1:1::4 in the URL box and then, click Go. The Accounting website didn't appear because the Accounting Server isn't connected to the rest of the network. It shows Request Timeout




- Then we enter 2001:db8:1:2::4 in the URL box and click Go. The CAD website would appear



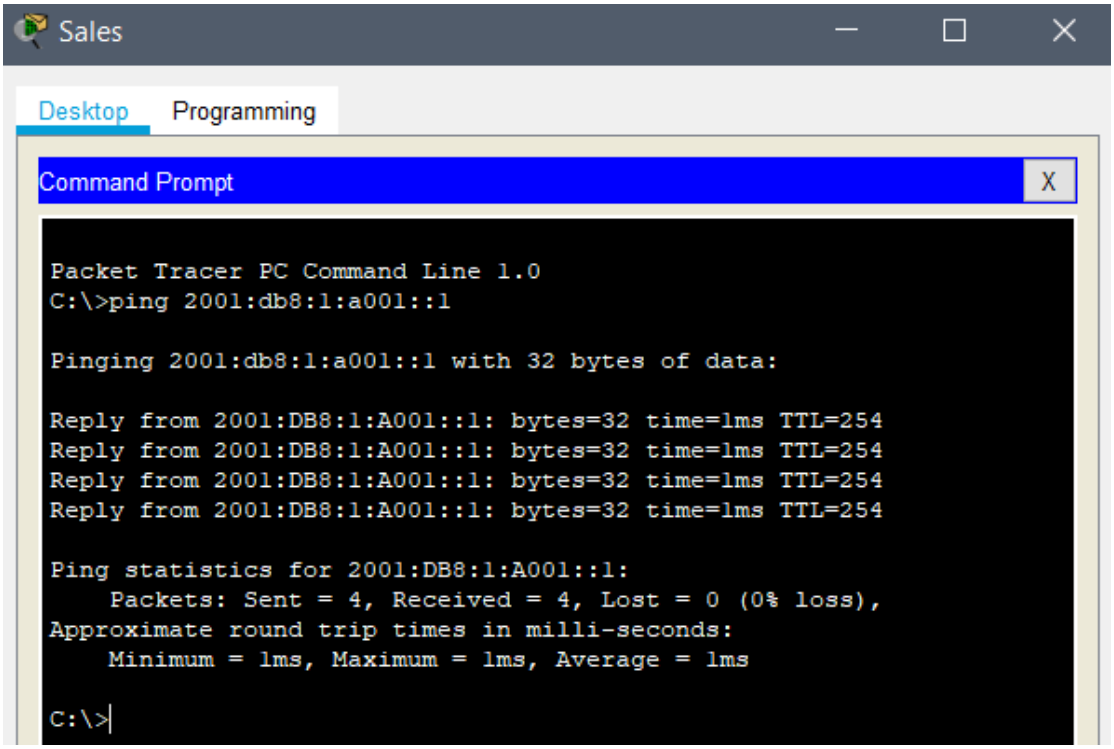

### Step 5: Ping the ISP

- We click on any client and go to the Desktop tab > Command Prompt
- Then, we test connectivity to the ISP by entering the following command:

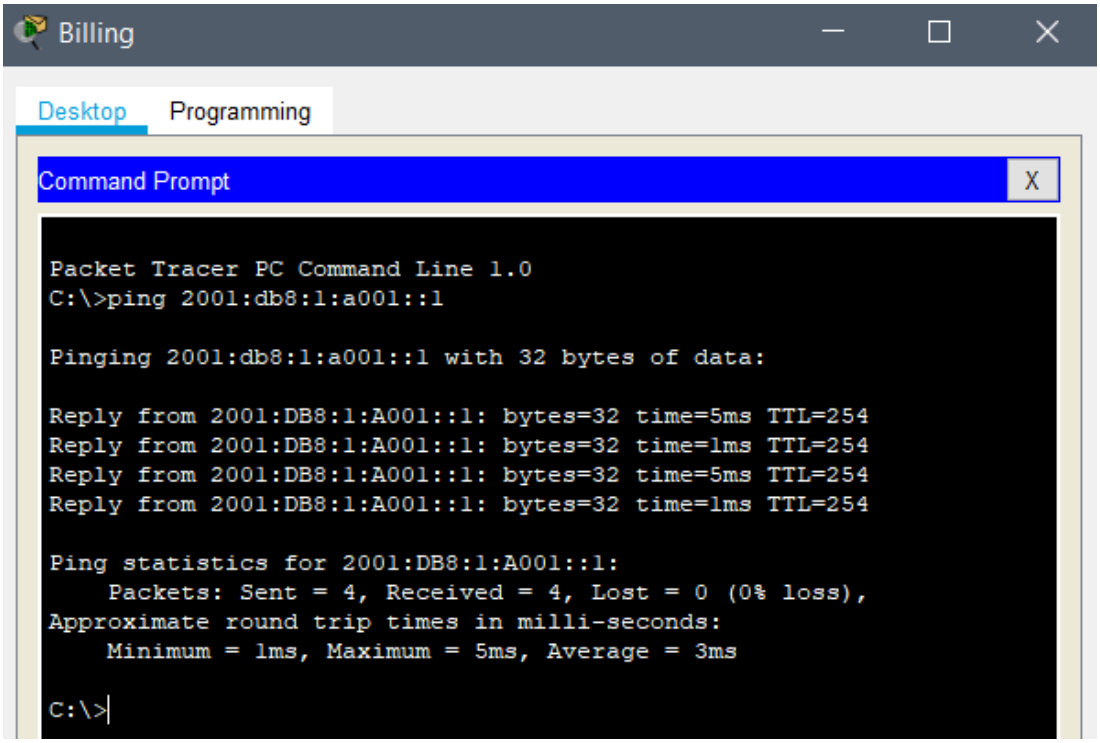
```
C:\> ping 2001:db8:1:a001::1
```

- We repeat the ping commands for the other clients until full connectivity is verified

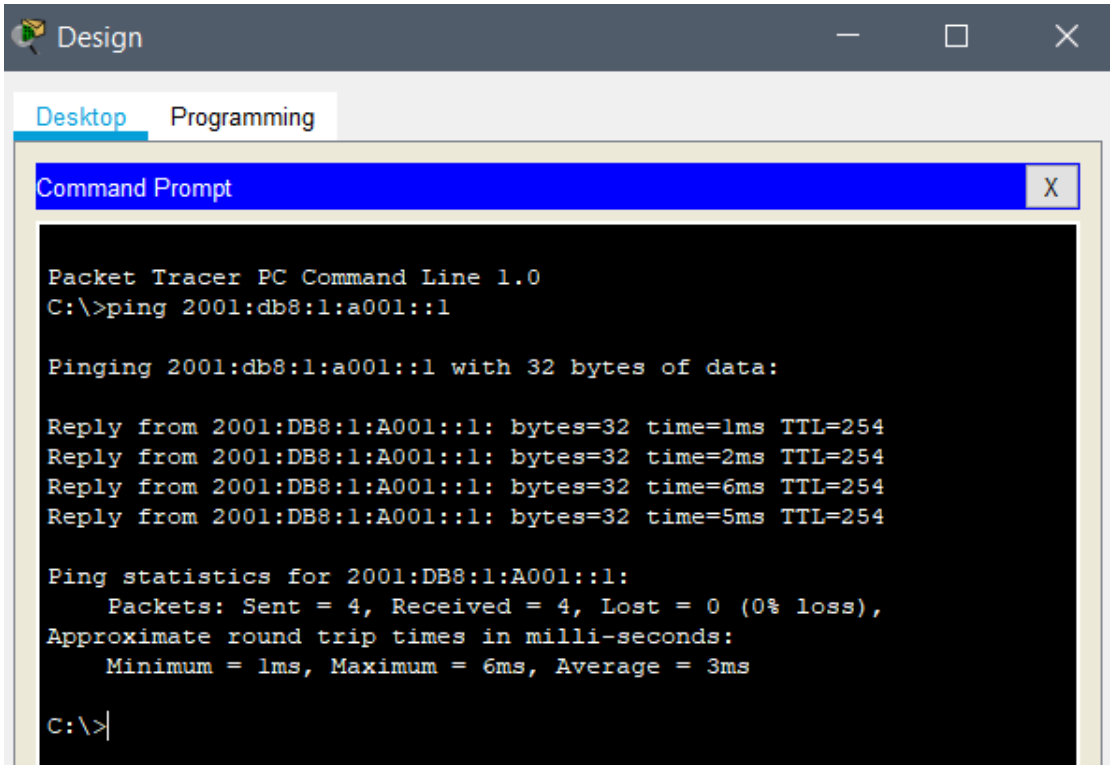
**Pinging from Sales:**



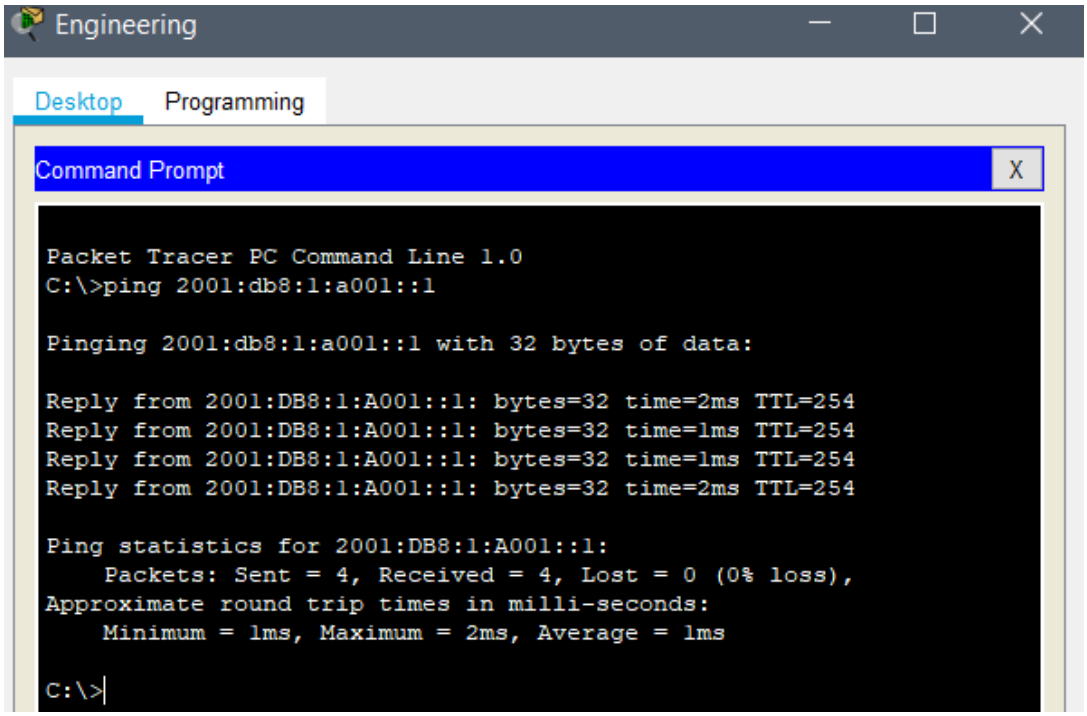

**Pinging from Billing:**



**Pinging from Design:**




**Pinging from Engineering:**



## TASK #02: Configure IPv6 Inter-VLAN Routing

### Step 1: Enable IPv6 routing

- We enter the privileged mode and go to the configuration terminal. The following commands are used:

```
MLS>en
MLS#config terminal
```

- We enter the ipv6 unicast-routing command to enable IPv6 routing:

```
MLS(config)#ipv6 unicast-routing
```

### Step 2: Configure SVI for IPv6 on MLS

- We configure IPv6 addressing on SVI for VLANs 10, 20, and 30 according to the Addressing Table. The configuration for VLAN 10 is shown below:

```
MLS(config)#interface vlan 10
MLS(config-if)#ipv6 address 2001:db8:acad:10::1/64
```

- The configuration for VLAN 20 is given below:

```
MLS(config)#interface vlan 20
MLS(config-if)#ipv6 address 2001:db8:acad:20::1/64
```

- The configuration for VLAN 30 is given below:

```
MLS(config)#interface vlan 30
MLS(config-if)#ipv6 address 2001:db8:acad:30::1/64
```

```
MLS>en
MLS#config terminal
Enter configuration commands, one per line.  End with CNTL/Z.
MLS(config)#ipv6 unicast-routing
MLS(config)#interface vlan 10
MLS(config-if)#ipv6 address 2001:db8:acad:10::1/64
MLS(config-if)#interface vlan 20
MLS(config-if)#ipv6 address 2001:db8:acad:20::1/64
MLS(config-if)#interface vlan 30
MLS(config-if)#ipv6 address 2001:db8:acad:30::1/64
MLS(config-if)#interface G0/2
MLS(config-if)#ipv6 address 2001:db8:acad:a::1/64
```

### Step 3: Configure G0/2 with IPv6 on MLS

- Now, we configure IPv6 addressing on G0/2. We first enter the G0/2 interface and then assign it the IP address from the table using the following commands:

```
MLS(config)# interface G0/2
MLS(config-if)# ipv6 address 2001:db8:acad:a::1/64
```

- We use the show ipv6 route command to verify IPv6 connected networks

```
MLS# show ipv6 route
```

```
MLS#show ipv6 route
IPv6 Routing Table - 10 entries
Codes: C - Connected, L - Local, S - Static, R - RIP, B - BGP
       U - Per-user Static route, M - MIPv6
       I1 - ISIS L1, I2 - ISIS L2, IA - ISIS interarea, IS - ISIS summary
       O - OSPF intra, OI - OSPF inter, OE1 - OSPF ext 1, OE2 - OSPF ext 2
       ON1 - OSPF NSSA ext 1, ON2 - OSPF NSSA ext 2
       D - EIGRP, EX - EIGRP external
S   ::/0 [1/0]
     via 2001:DB8:ACAD:A::2, GigabitEthernet0/2
C   2001:DB8:ACAD:A::/64 [0/0]
     via ::, GigabitEthernet0/2
L   2001:DB8:ACAD:A::1/128 [0/0]
     via ::, GigabitEthernet0/2
C   2001:DB8:ACAD:10::/64 [0/0]
     via ::, Vlan10
L   2001:DB8:ACAD:10::1/128 [0/0]
     via ::, Vlan10
C   2001:DB8:ACAD:20::/64 [0/0]
     via ::, Vlan20
L   2001:DB8:ACAD:20::1/128 [0/0]
     via ::, Vlan20
C   2001:DB8:ACAD:30::/64 [0/0]
     via ::, Vlan30
L   2001:DB8:ACAD:30::1/128 [0/0]
     via ::, Vlan30
L   FF00::/8 [0/0]
     via ::, Null0
```

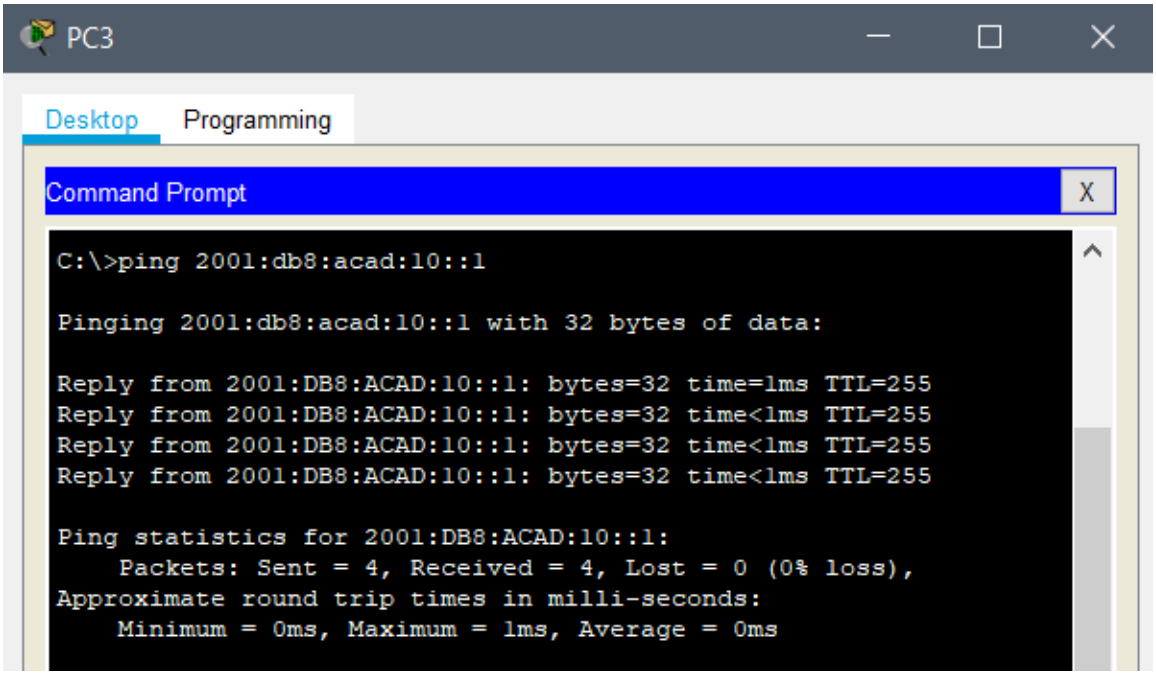
### Step 4: Verify IPv6 connectivity

Devices PC3, PC4, and PC5 have been configured with IPv6 addresses. Now, we verify IPv6 inter-VLAN routing and connectivity to Cloud. To do so, we click on the end-device, go to Desktop Tab> Command prompt. Then we write the following command:

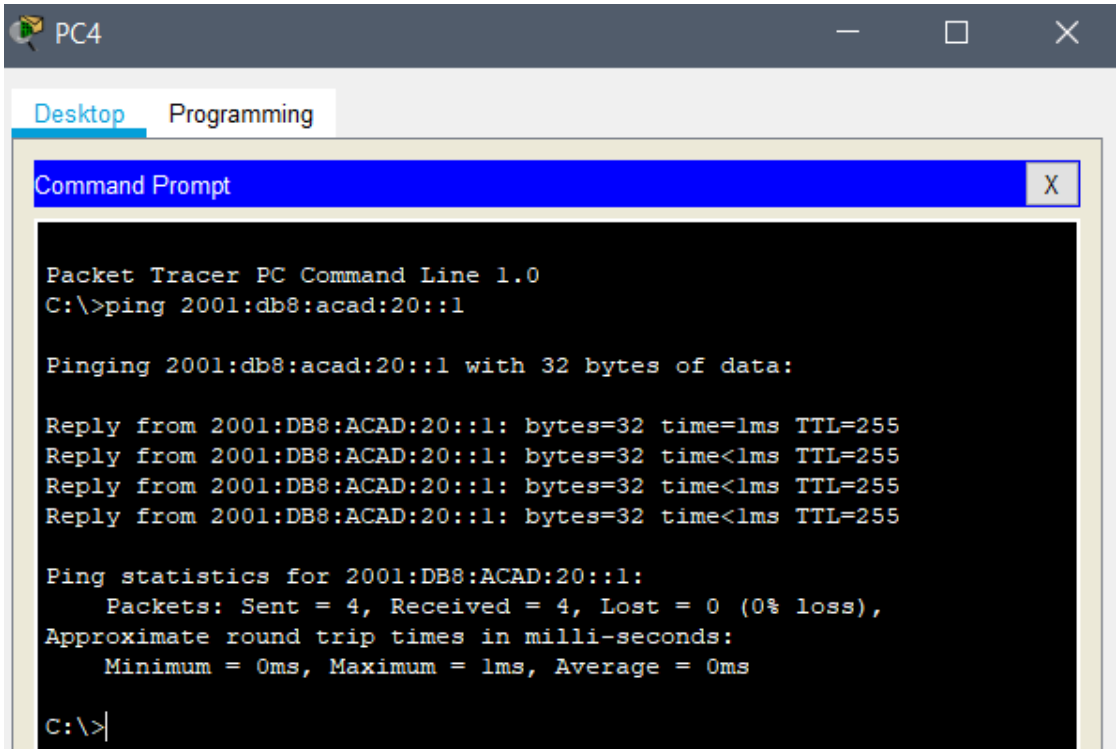
```
C:\> ping 2201:db8:acad:10::1
```

In this command we enter the address with which we want to check the connection after writing the ping command. In this example it is the IPv6 address of MLS from PC3.

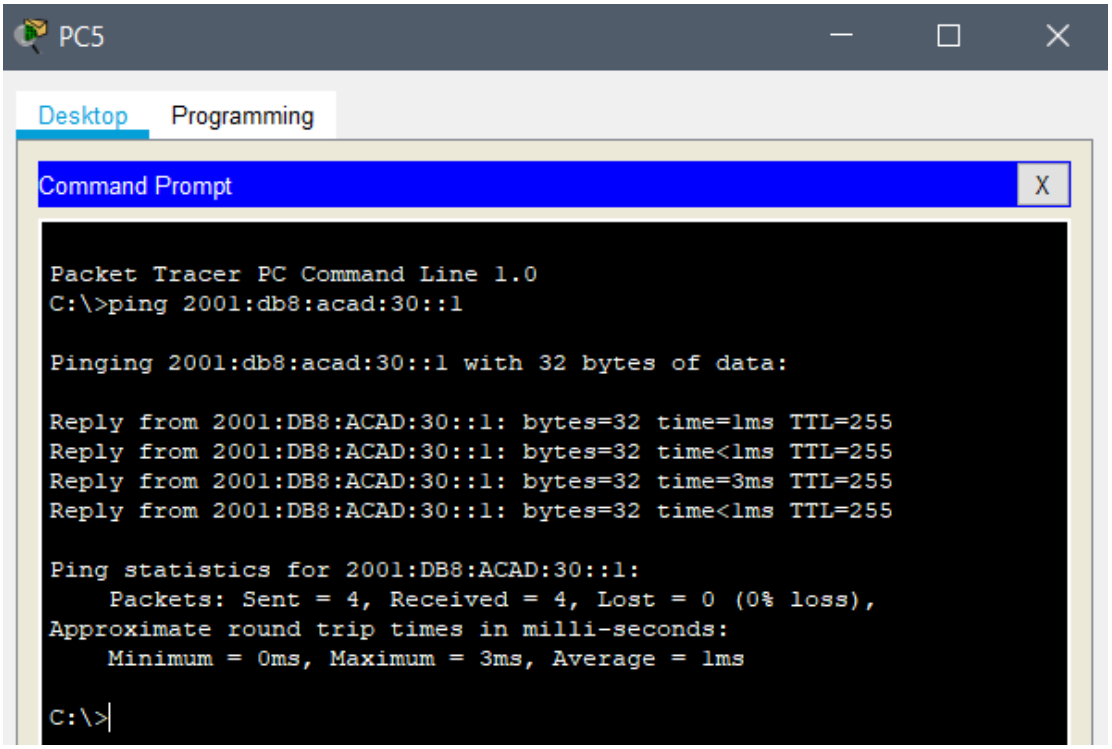
- **From PC3, we ping MLS to verify connectivity within VLAN 10**



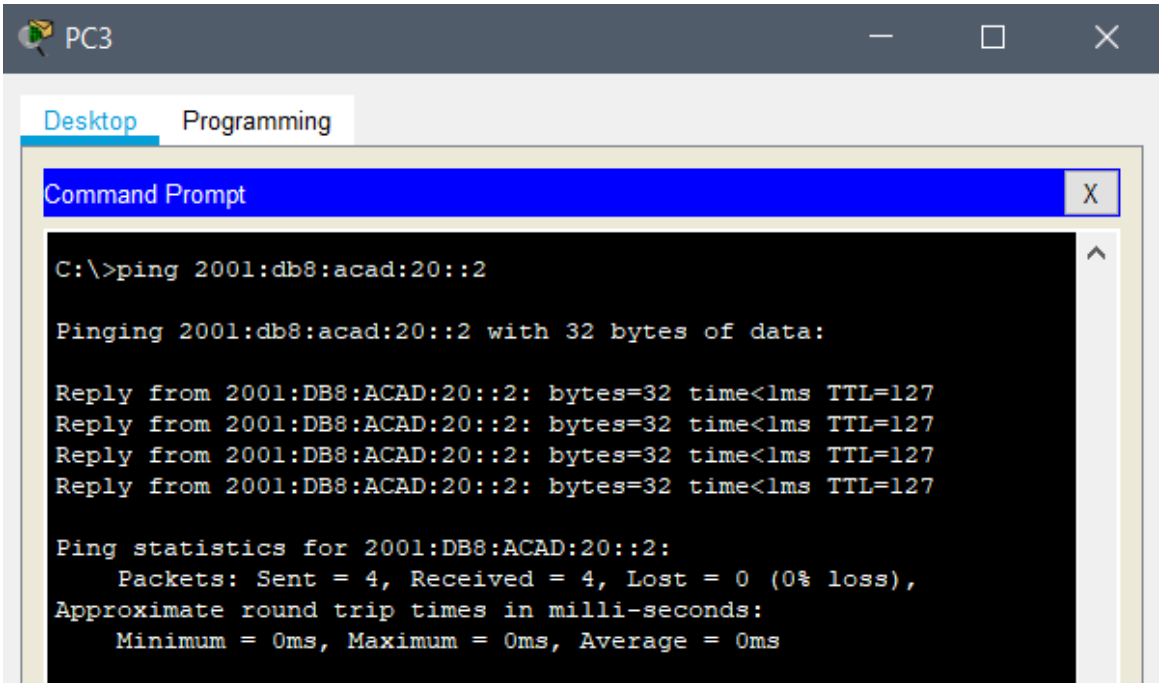

- **From PC4, we ping MLS to verify connectivity within VLAN 20**



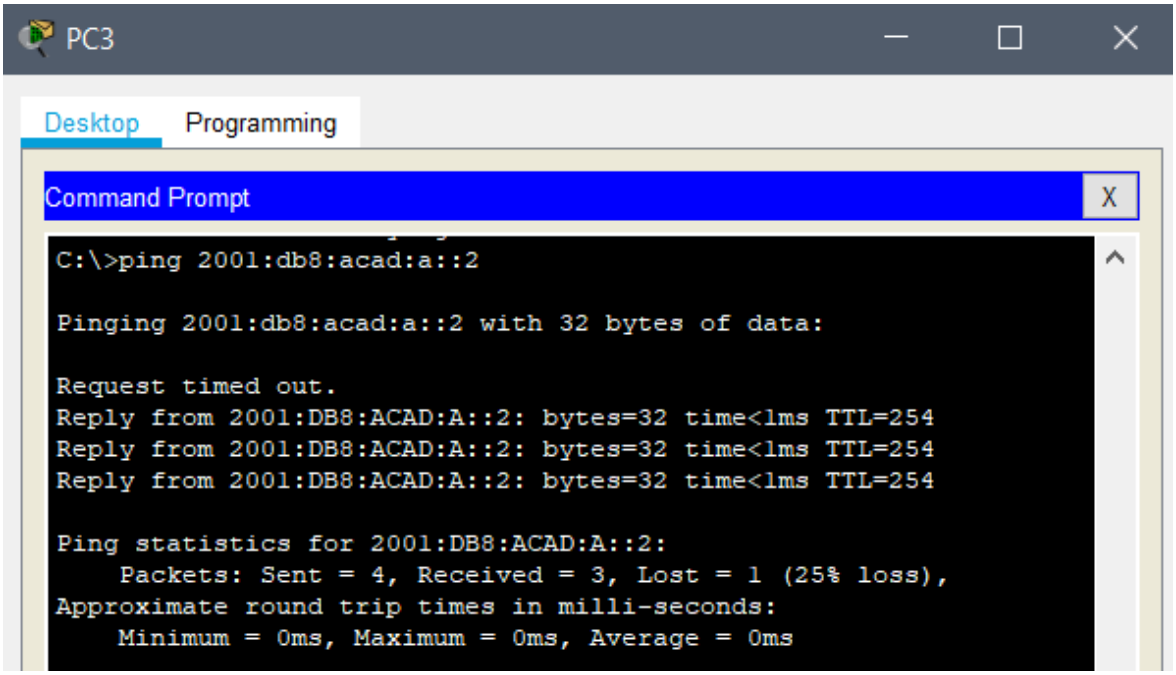
- **From PC5, we ping MLS to verify connectivity within VLAN 30**




- To verify inter-VLAN routing, we ping between devices PC3, PC4, and PC5. **Here we pinged from PC3 to PC4**



- **From PC3, we ping the address inside Cloud, 2001:db8:acad:a::2**




# Observation:

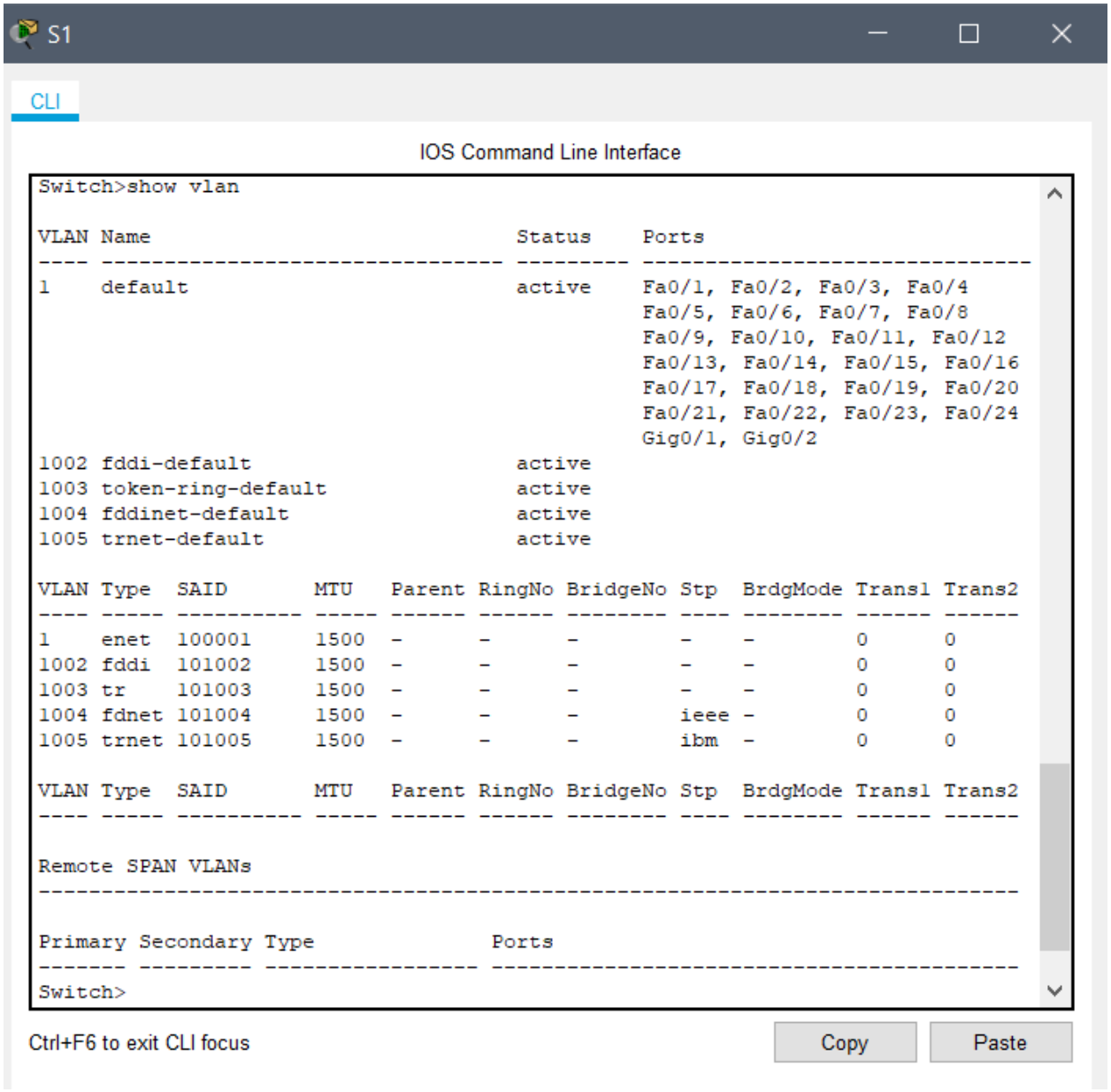
## Task-1:

We use the show VLAN command in the S1 and S2 switches by going to the configuration tab and entering the following command:

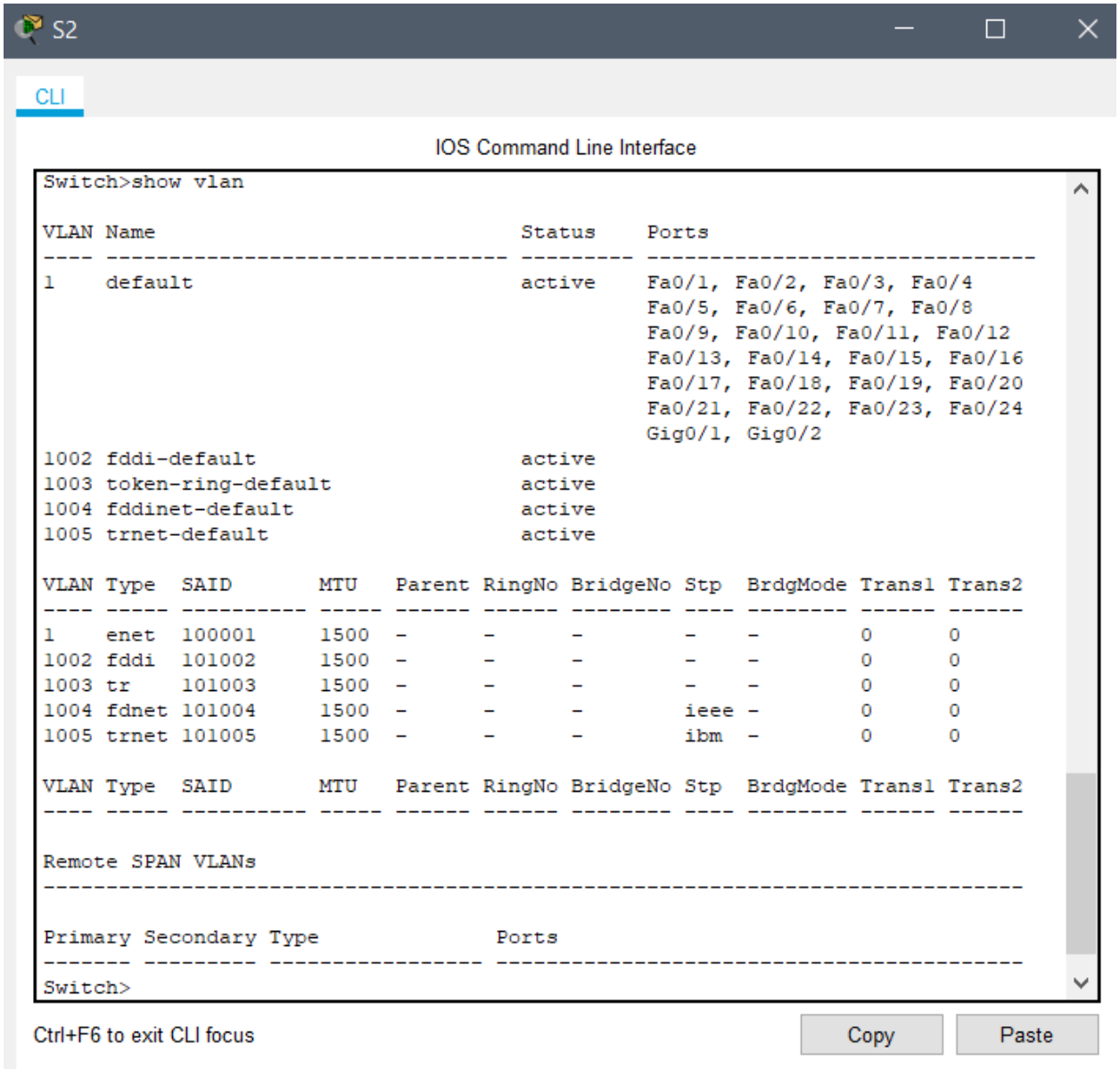
```
s1> show vlan
```

In this task, we didn't configure VLAN. Hence, the show VLAN command displayed that all the ports belong to the default VLAN. This means that no VLAN has been configured for both of these switches.

**Switch S1:**


S1
CLI
IOS Command Line Interface
Switch>show vlan
VLAN Name Status Ports
1 default active Fa0/1, Fa0/2, Fa0/3, Fa0/4
Fa0/5, Fa0/6, Fa0/7, Fa0/8
Fa0/9, Fa0/10, Fa0/11, Fa0/12
Fa0/13, Fa0/14, Fa0/15, Fa0/16
Fa0/17, Fa0/18, Fa0/19, Fa0/20
Fa0/21, Fa0/22, Fa0/23, Fa0/24
Gig0/1, Gig0/2
1002 fddi-default active
1003 token-ring-default active
1004 fddinet-default active
1005 trnet-default active
VLAN Type SAID MTU Parent RingNo BridgeNo Stp BrdgMode Trans1 Trans2
1 enet 100001 1500 - - - - - 0 0
1002 fddi 101002 1500 - - - - - 0 0
1003 tr 101003 1500 - - - - - 0 0
1004 fdnet 101004 1500 - - - ieee - 0 0
1005 trnet 101005 1500 - - - ibm - 0 0
VLAN Type SAID MTU Parent RingNo BridgeNo Stp BrdgMode Trans1 Trans2
Remote SPAN VLANs
Primary Secondary Type Ports
Switch>
Ctrl+F6 to exit CLI focus
Copy
Paste

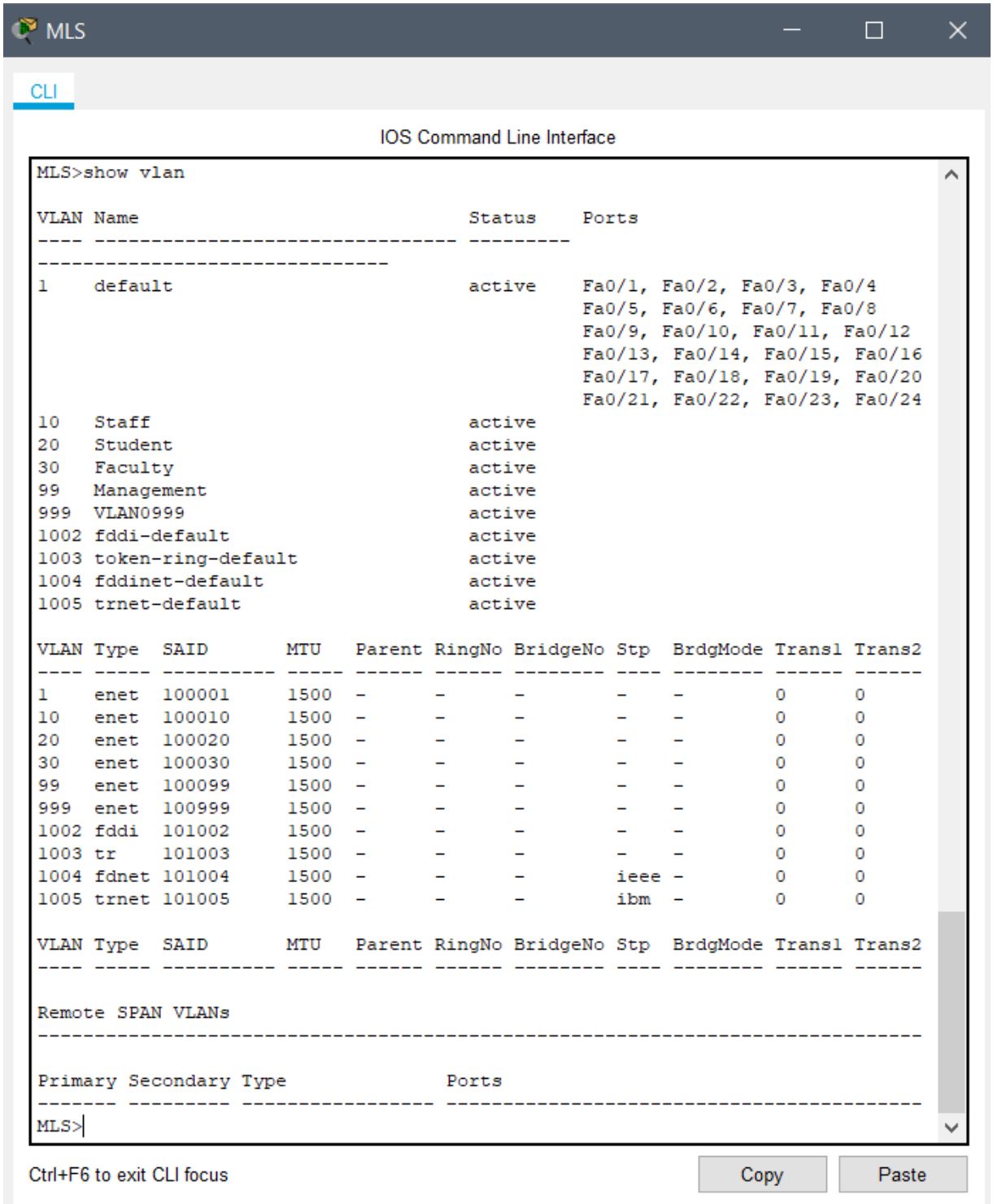
**Switch S2:**


S2
CLI
IOS Command Line Interface
Switch>show vlan
VLAN Name Status Ports
---- -------------------------------- --------- -------------------------------
1 default active Fa0/1, Fa0/2, Fa0/3, Fa0/4
Fa0/5, Fa0/6, Fa0/7, Fa0/8
Fa0/9, Fa0/10, Fa0/11, Fa0/12
Fa0/13, Fa0/14, Fa0/15, Fa0/16
Fa0/17, Fa0/18, Fa0/19, Fa0/20
Fa0/21, Fa0/22, Fa0/23, Fa0/24
Gig0/1, Gig0/2
1002 fddi-default active
1003 token-ring-default active
1004 fddinet-default active
1005 trnet-default active
VLAN Type SAID MTU Parent RingNo BridgeNo Stp BrdgMode Trans1 Trans2
---- ----- ---------- ----- ------ ------ -------- ---- -------- ------ ------
1 enet 100001 1500 - - - - - 0 0
1002 fddi 101002 1500 - - - - - 0 0
1003 tr 101003 1500 - - - - - 0 0
1004 fdnet 101004 1500 - - - ieee - 0 0
1005 trnet 101005 1500 - - - ibm - 0 0
VLAN Type SAID MTU Parent RingNo BridgeNo Stp BrdgMode Trans1 Trans2
---- ----- ---------- ----- ------ ------ -------- ---- -------- ------ ------
Remote SPAN VLANs
Primary Secondary Type Ports
Switch>
Ctrl+F6 to exit CLI focus
Copy
Paste

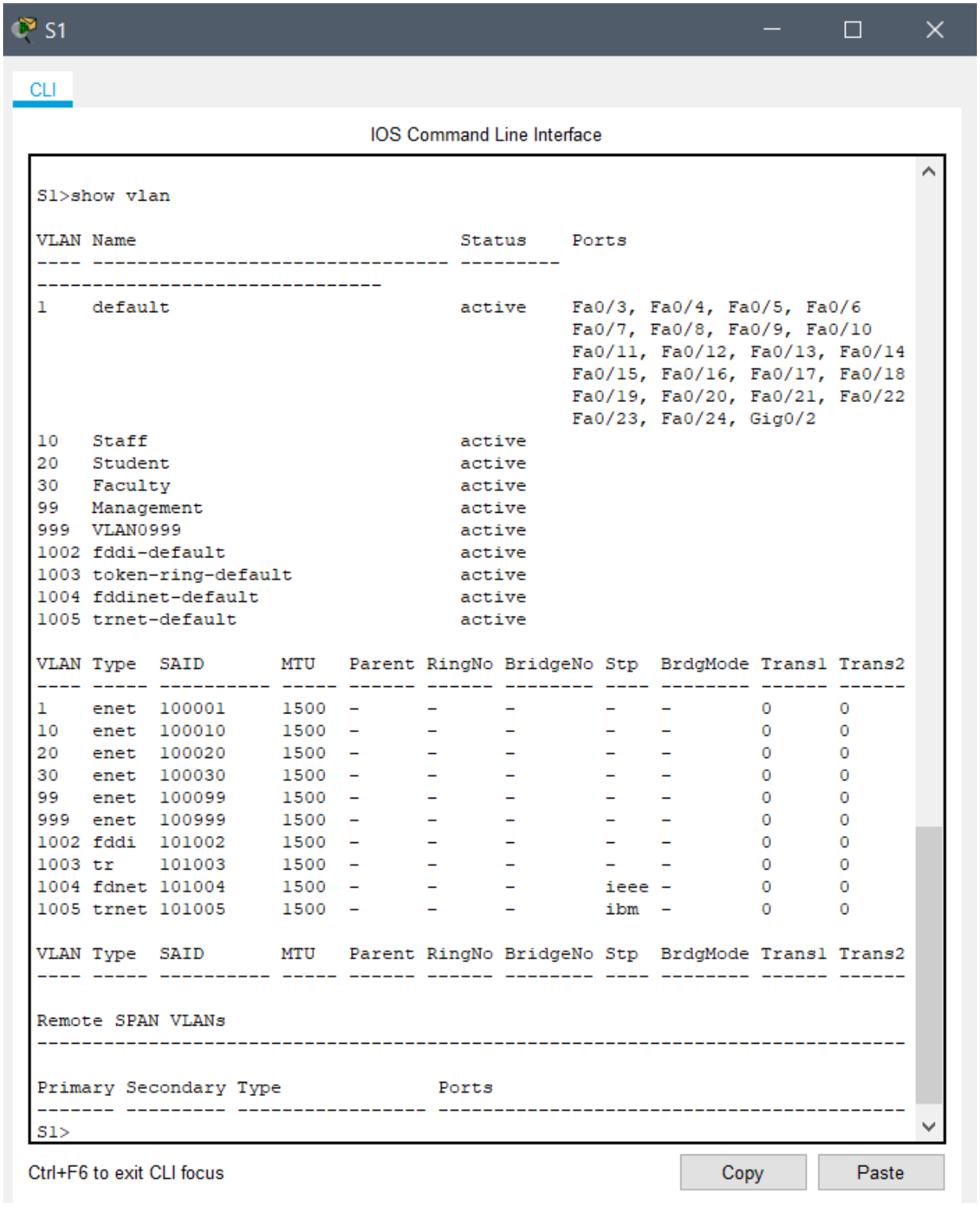
## Task - 2:

Similar to task-1, we use the show VLAN command in all four switches, the MLS, S1, S2 and S3

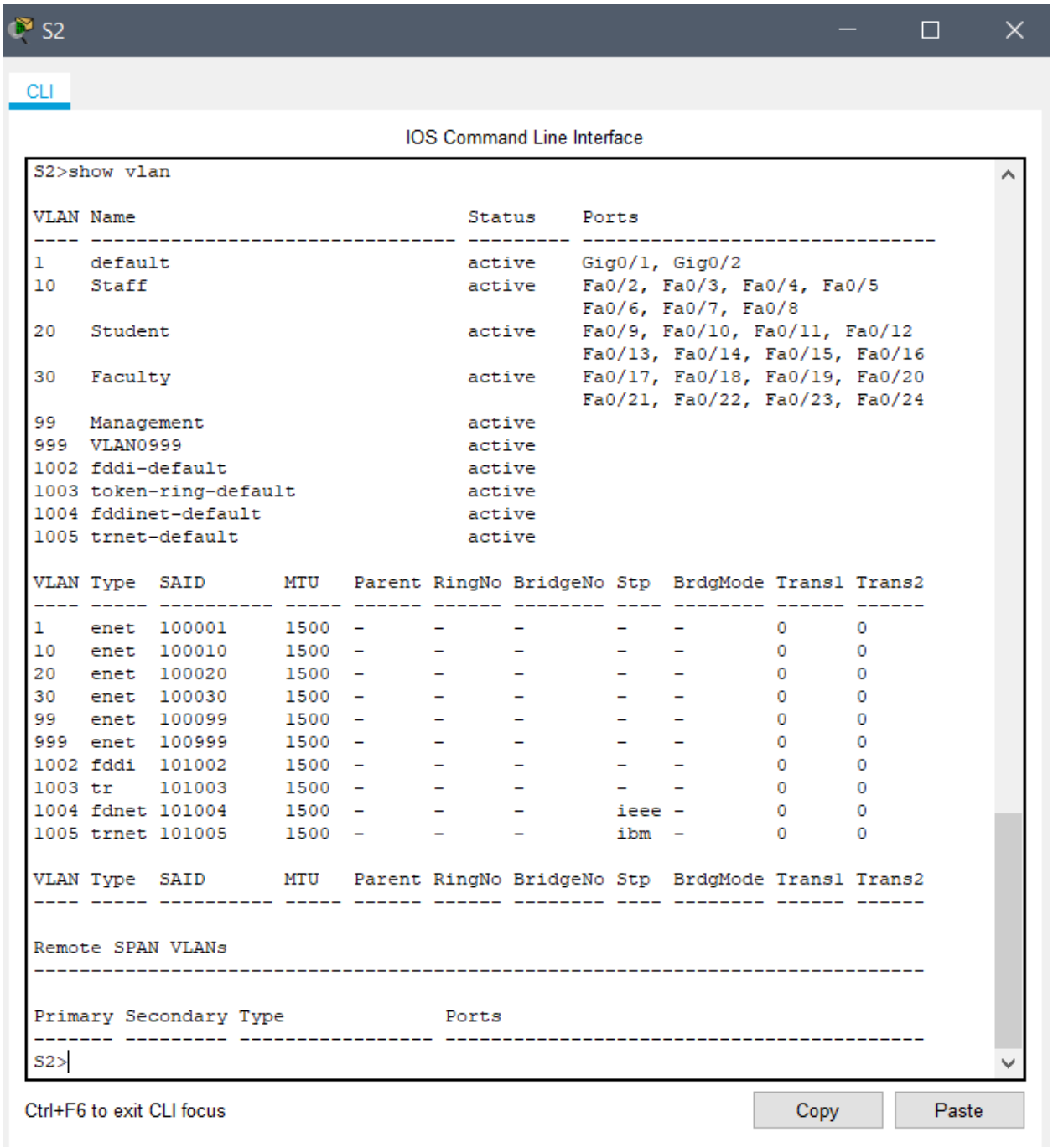
**MLS:** The default VLAN was assigned to all the ports. Hence, no VLAN was created in the MLS. The missing ports are used to connect to Switch-1 and Cloud



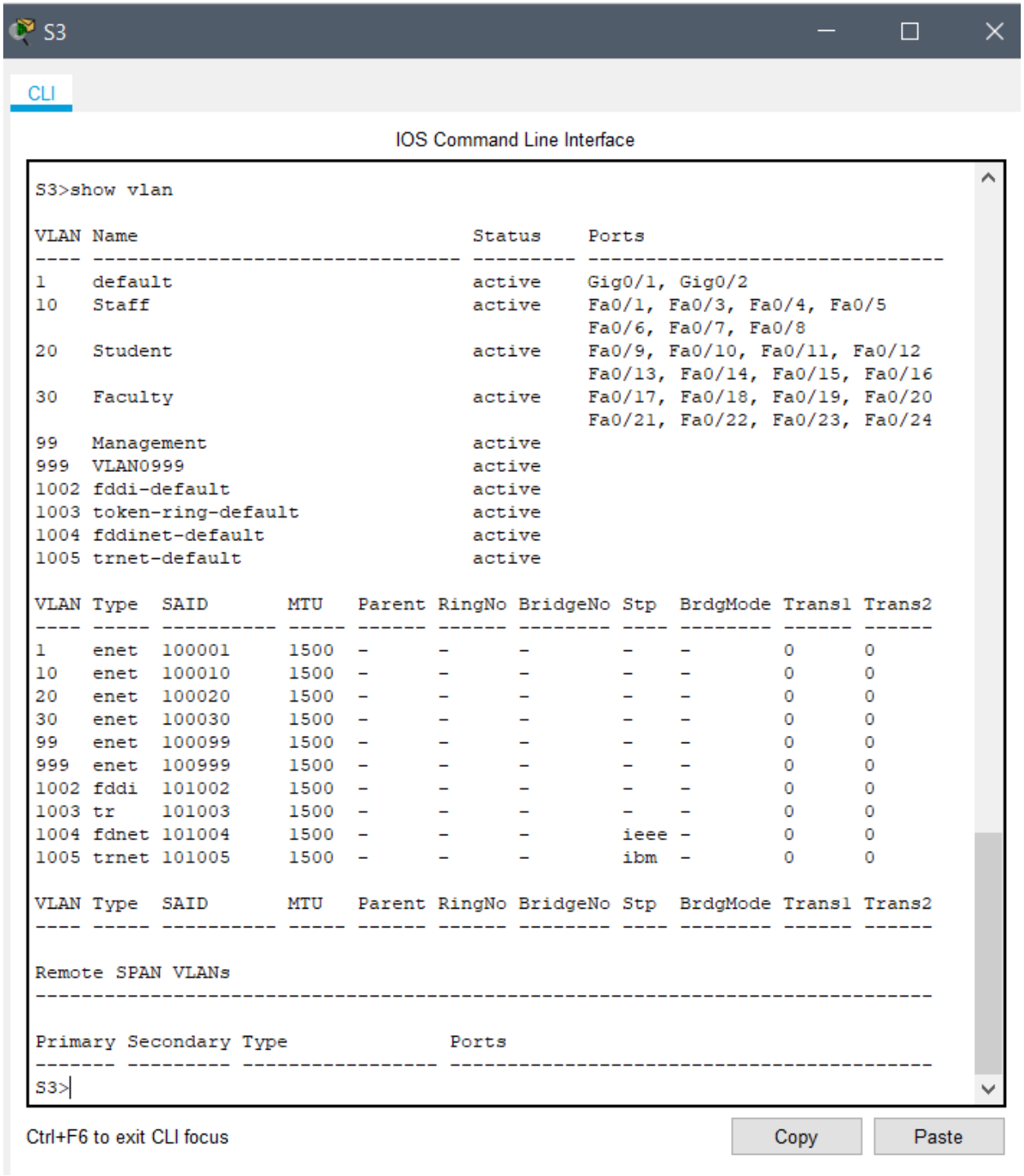
**Switch S1:** Here, VLAN 1,which is the default VLAN, is assigned to all the ports. Port Fa0/1, Fa0/2, and Gig0/1 are missing which were used to connect with switches and MLS. Since, all the ports have the default VLAN assigned to them, VLAN wasn't created in this switch.



**For Switch S2:** We assigned VLANs to ports as mentioned in the table. All the VLANs are active. Fa0/1 was used to connect switch-2 with switch-1 and is missing in the ports section.



**For Switch S3:** All the VLANs were assigned to ports following the table. Only Fa0/2 doesn't have a vlan since it was used for connecting to switch-1, and that is why this port is missing in the ports section.



## Challenges (if any):

- The Accounting Server isn't connected to the topology which is why the server's web page can not be loaded. The connections are locked, so we can not establish a connection with the Accounting Server and the Switch.