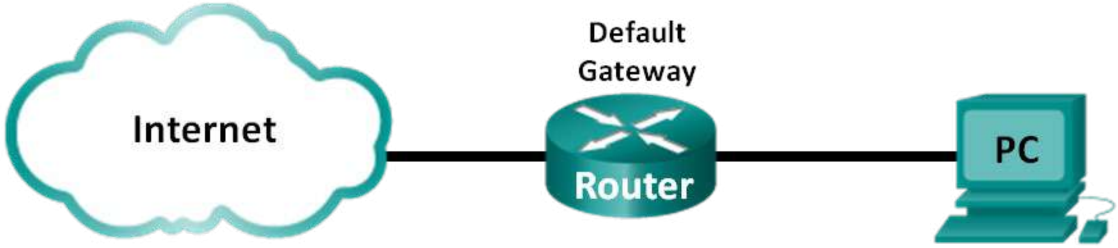
Identifying IPv6 Addresses

Topology



Objectives

Part 1: Identify the Different Types of IPv6 Addresses

Part 2: Examine a Host IPv6 Network Interface and Address

Part 3: Practice IPv6 Address Abbreviation

Background / Scenario

With the depletion of the Internet Protocol version 4 (IPv4) network address space and the adoption and transition to IPv6, networking professionals must understand how both IPv4 and IPv6 networks function. Many devices and applications already support IPv6. This includes extensive Cisco device Internetwork Operating System (IOS) support and workstation/server operating system support, such as that found in Windows and Linux.

This lab focuses on IPv6 addresses and the components of the address. In Part 1, you will identify the IPv6 address types, and in Part 2, you will view the IPv6 settings on a PC. In Part 3, you will practice IPv6 address abbreviation.

Required Resources

- 1 PC (Windows 7 or 8 with Internet access)

Part 1: Identify the Different Types of IPv6 Addresses

In Part 1, you will review the characteristics of IPv6 addresses to identify the different types of IPv6 addresses.

Step 1: Review the different types of IPv6 addresses.

An IPv6 address is 128 bits long. It is most often presented as 32 hexadecimal characters. Each hexadecimal character is the equivalent of 4 bits (4 x 32 = 128). A non-abbreviated IPv6 host address is shown here:

2001:0DB8:0001:0000:0000:0000:0000:0001

A hextet is the hexadecimal, IPv6 version of an IPv4 octet. An IPv4 address is 4 octets long, separated by dots. An IPv6 address is 8 hextets long, separated by colons.

An IPv4 address is 4 octets and is commonly written or displayed in decimal notation.

255.255.255.255

An IPv6 address is 8 hextets and is commonly written or displayed in hexadecimal notation.

FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF

In an IPv4 address, each individual octet is 8 binary digits (bits). Four octets equals one 32-bit IPv4 address.

11111111 = 255

11111111.11111111.11111111.11111111 = 255.255.255.255

In an IPv6 address, each individual hextet is 16 bits long. Eight hextets equals one 128-bit IPv6 address.

1111111111111111 = FFFF

1111111111111111.1111111111111111.1111111111111111.1111111111111111.
1111111111111111.1111111111111111.1111111111111111.1111111111111111 =
FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF

If we read an IPv6 address starting from the left, the first (or far left) hextet identifies the IPv6 address type. For example, if the IPv6 address has all zeros in the far left hextet, then the address is possibly a loopback address.

0000:0000:0000:0000:0000:0000:0000:0001 = loopback address (An address that sends outgoing signals back to the same computer for testing.)

::1 = loopback address abbreviated

As another example, if the IPv6 address has FE80 in the first hextet, then the address is a link-local address.

FE80:0000:0000:0000:C5B7:CB51:3C00:D6CE = link-local address

FE80::C5B7:CB51:3C00:D6CE = link-local address abbreviated

Study the chart below to help you identify the different types of IPv6 address based on the numbers in the first hextet.

First Hextet (Far Left)	Type of IPv6 Address
0000 to 00FF	Loopback address, any address, unspecified address, or IPv4-compatible
2000 to 3FFF	Global unicast address (a routable address in a range of addresses that is currently being handed out by the Internet Assigned Numbers Authority [IANA])
FE80 to FEBF	Link-local (a unicast address which identifies the host computer on the local network)
FC00 to FCFF	Unique-local (a unicast address which can be assigned to a host to identify it as being part of a specific subnet on the local network)
FF00 to FFFF	Multicast address

There are other IPv6 address types that are either not yet widely implemented, or have already become deprecated, and are no longer supported. For instance, an **anycast address** is new to IPv6 and can be used by routers to facilitate load sharing and provide alternate path flexibility if a router becomes unavailable. Only routers should respond to an anycast address. Alternatively, **site-local addresses** have been deprecated and replaced by unique-local addresses. Site-local addresses were identified by the numbers FEC0 in the initial hextet.

In IPv6 networks, there are no network (wire) addresses or broadcast addresses as there are in IPv4 networks.

Step 2: Match the IPv6 address to its type.

Match the IPv6 addresses to their corresponding address type. Notice that the addresses have been compressed to their abbreviated notation and that the slash network prefix number is not shown. Some answer choices must be used more than once.

IPv6 Address	Answer
2001:0DB8:1:ACAD::FE55:6789:B210	1. b
::1	2. a
FC00:22:A:2::CD4:23E4:76FA	3. d
2033:DB8:1:1:22:A33D:259A:21FE	4. b
FE80::3201:CC01:65B1	5. c
FF00::	6. e
FF00::DB7:4322:A231:67C	7. e
FF02::2	8. e

Answer Choices

a. Loopback address

b. Global unicast address

c. Link-local address

d. Unique-local address

e. Multicast address

Part 2: Examine a Host IPv6 Network Interface and Address

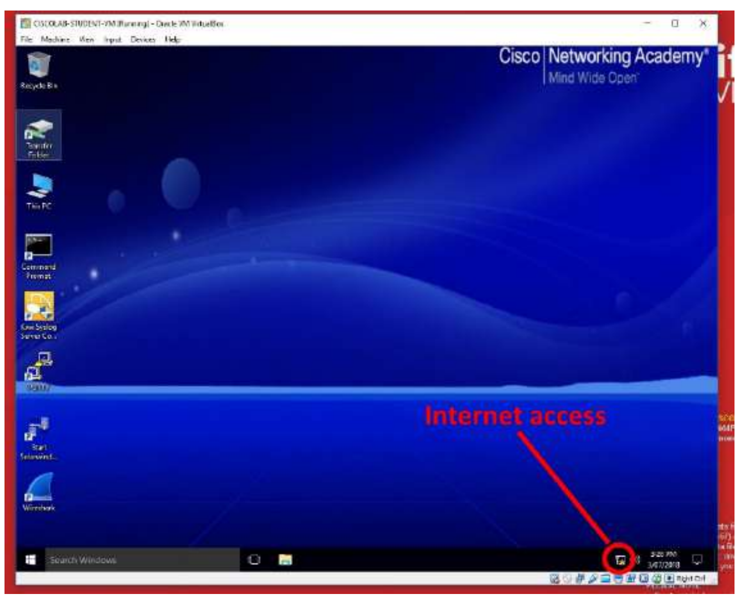
In Part 2, you will check the IPv6 network settings of your PC to identify your network interface IPv6 address.

Step 1: Check your PC IPv6 network address settings.

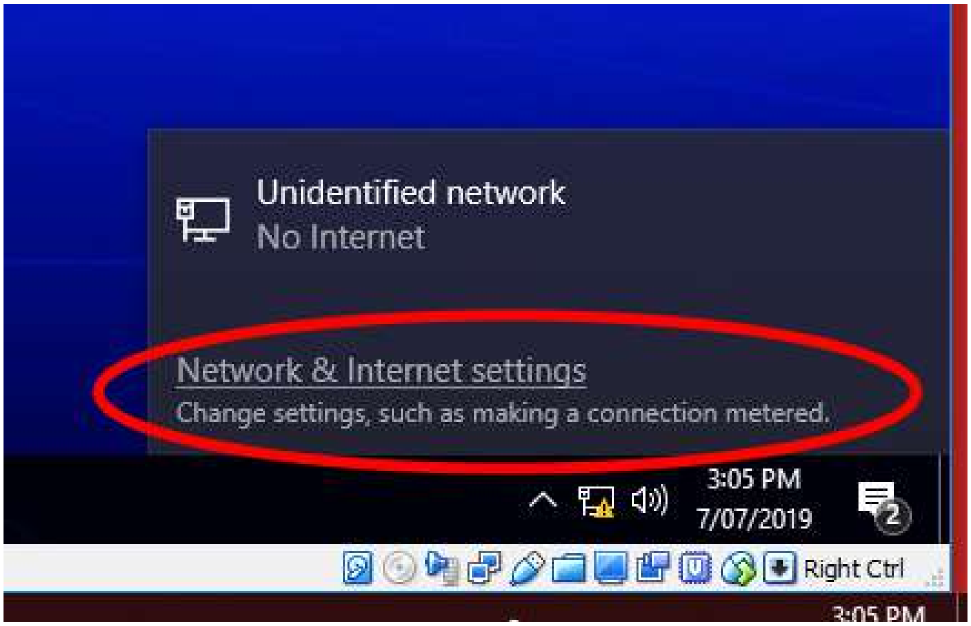
a. Verify that the IPv6 protocol is installed and active on your PC-A (check your Local Area Connection settings).

b. On a windows machine right-click the Network Internet access icon, Open Network and Sharing Center:

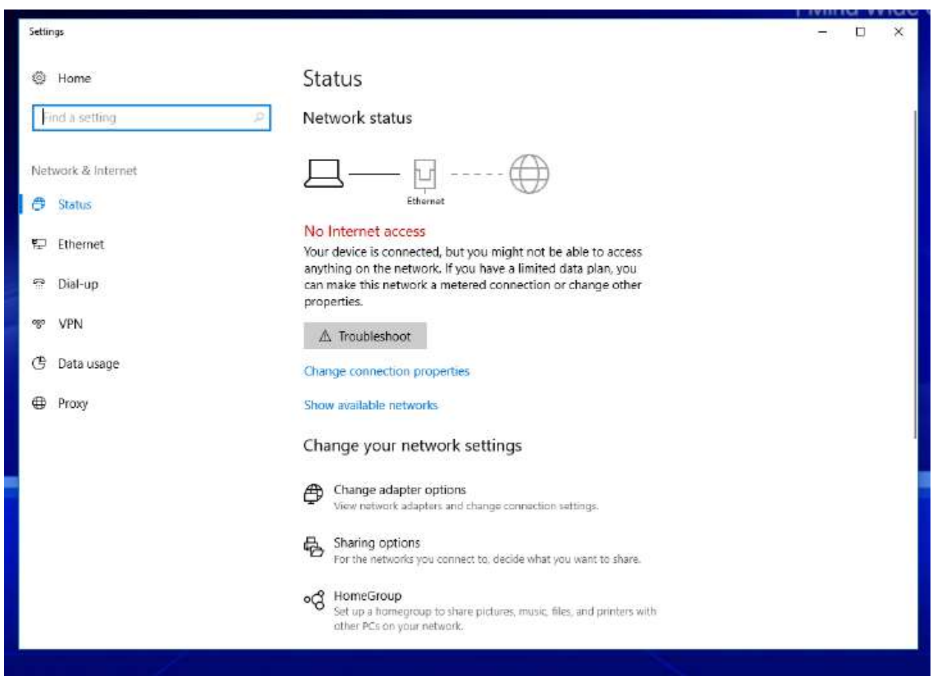
   Note: The following example is illustrative only and the steps required may be slightly different on your machine. Please experiment and adjust the process as necessary.



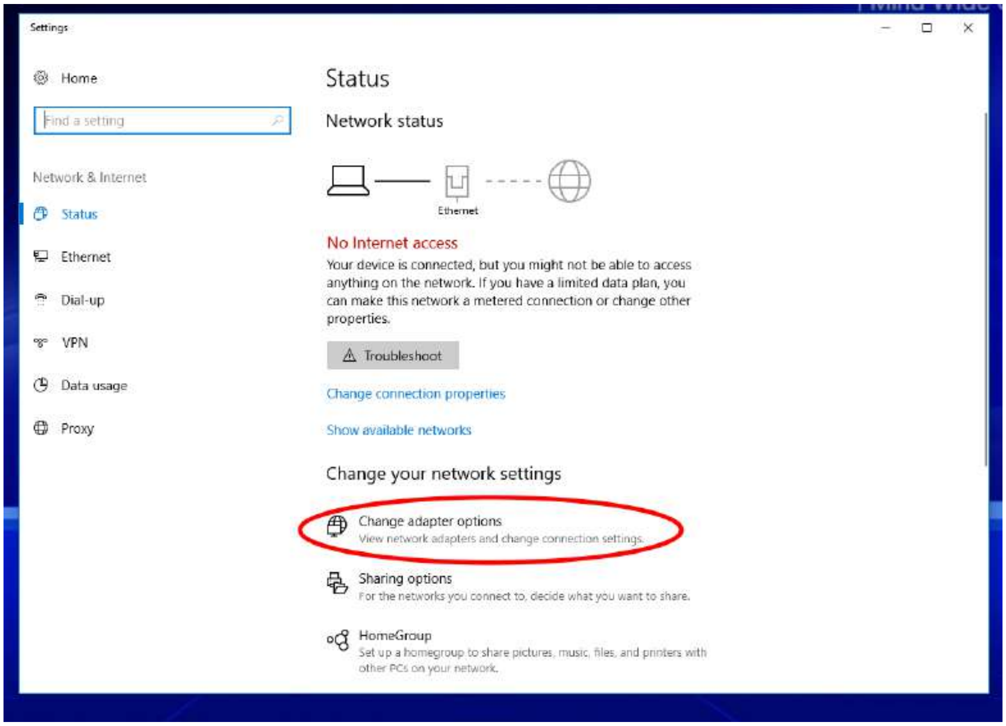
c. Click on the Network & Internet Settings popup.



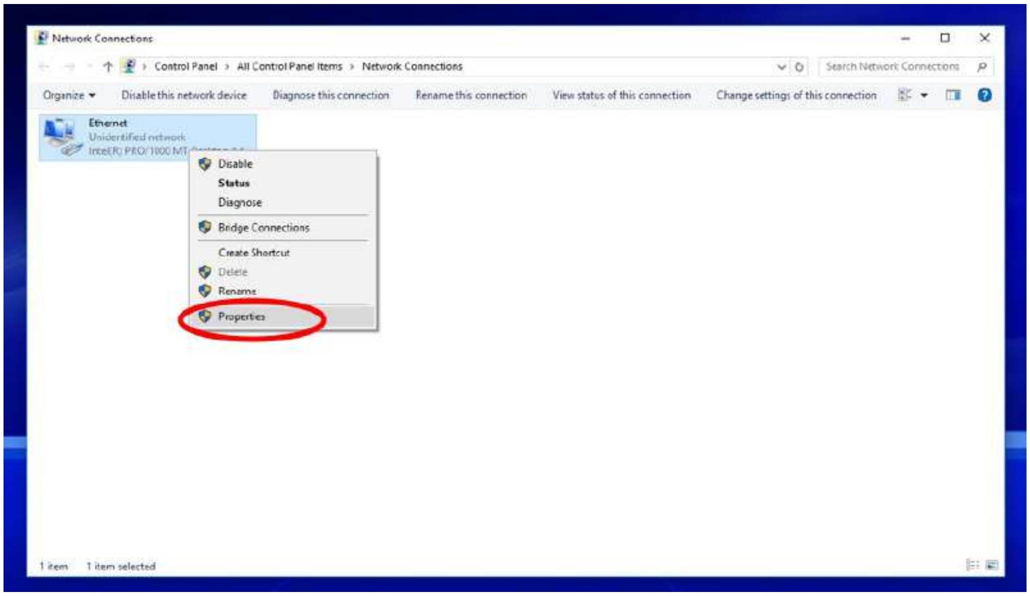
d. You should now see something like the following screen:



e. At the bottom of the middle pane of this example status tab, click the **Change adapter options** link.

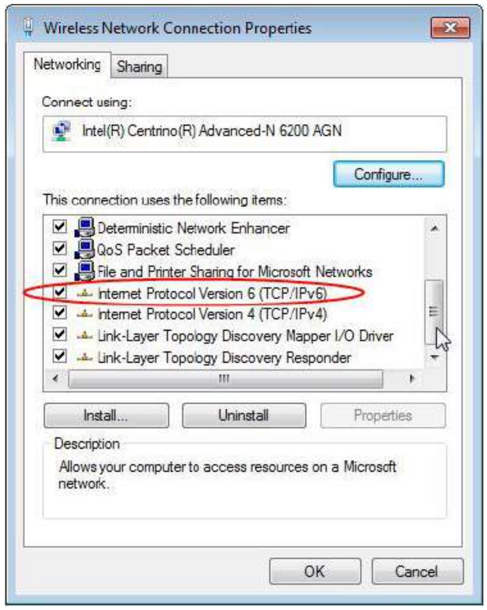


f. The Network Connections window displays the available interfaces on the PC. Right-click the appropriate network interface for your network and select **Properties**.



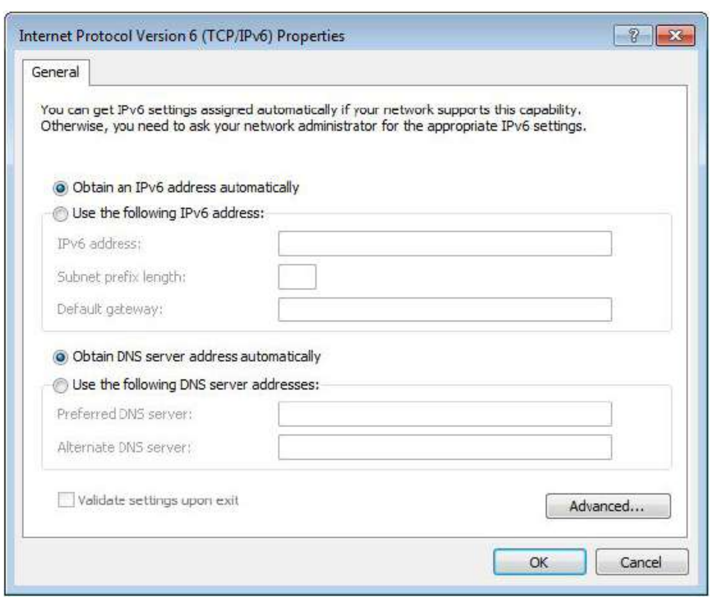
g. Select the **Internet Protocol Version 6 (TCP/IPv6)** option and then click **Properties**.

You should now see your Network Connection Properties window. Scroll through the list of items to determine whether IPv6 is present, which indicates that it is installed, and if it is also check marked, which indicates that it is active.



h. Select the item **Internet Protocol Version 6 (TCP/IPv6)** and click **Properties**. You should see the IPv6 settings for your network interface. Your IPv6 properties window is likely set to **Obtain an IPv6 address automatically**. This does not mean that IPv6 relies on the Dynamic Host Configuration Protocol (DHCP). Instead of using DHCP, IPv6 looks to the local router for IPv6 network information and then autoconfigures its own IPv6 addresses. To manually configure IPv6, you must provide the IPv6 address, the subnet prefix length, and the default gateway.

Note: The local router can refer host requests for IPv6 information, especially Domain Name System (DNS) information, to a DHCPv6 server on the network.



i. After you have verified that IPv6 is installed and active on your PC, you should check your IPv6 address information. To do this, click the **Start** button, type **cmd** in the *Search programs and files* form box, and press Enter. This opens a Windows command prompt window.

j. Type **ipconfig /all** and press Enter. Your output should look similar to this:

```
C:\Users\user> ipconfig /all

Windows IP Configuration

<output omitted>

Wireless LAN adapter Wireless Network Connection:

   Connection-specific DNS Suffix  . :
   Description . . . . . . . . . . . : Intel(R) Centrino(R) Advanced-N 6200 AGN
   Physical Address. . . . . . . . . : 02-37-10-41-FB-48
   DHCP Enabled. . . . . . . . . . . : Yes
   Autoconfiguration Enabled . . . . : Yes
   Link-local IPv6 Address . . . . . : fe80::8d4f:4f4d:3237:95e2%14(Preferred)
   IPv4 Address. . . . . . . . . . . : 192.168.2.106(Preferred)
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Lease Obtained. . . . . . . . . . : Sunday, January 06, 2013 9:47:36 AM
   Lease Expires . . . . . . . . . . : Monday, January 07, 2013 9:47:38 AM
   Default Gateway . . . . . . . . . : 192.168.2.1
   DHCP Server . . . . . . . . . . . : 192.168.2.1
   DHCPv6 IAID . . . . . . . . . . . : 335554320
   DHCPv6 Client DUID. . . . . . . . : 00-01-00-01-14-57-84-B1-1C-C1-DE-91-C3-5D

   DNS Servers . . . . . . . . . . . : 192.168.1.1
                                       8.8.4.4
   <output omitted>
```

k. You can see from the output that the client PC has an IPv6 link-local address with a randomly generated interface ID. What does it indicate about the network regarding IPv6 global unicast address, IPv6 uniquelocal address, or IPv6 gateway address?

It indicates that there is no IPv6 enabled gateway router providing global address, local address,or subnet information on the network

l. What kind of IPv6 addresses did you find when using **ipconfig /all**?

link-local addresses

Part 3: Practice IPv6 Address Abbreviation

In Part 3, you will study and review rules for IPv6 address abbreviation to correctly compress and decompress IPv6 addresses.

Step 1: Study and review the rules for IPv6 address abbreviation.

Rule 1: In an IPv6 address, a string of four zeros (0s) in a hextet can be abbreviated as a single zero.

2001:0404:0001:1000:**0000:0000**:0EF0:BC00

2001:0404:0001:1000:**0**:**0**:0EF0:BC00 (abbreviated with single zeros)

Rule 2: In an IPv6 address, the leading zeros in each hextet can be omitted, trailing zeros cannot be omitted.

2001:**0**404:**000**1:1000:0000:0000:**0**EF0:BC00

2001:404:1:1000:0:0:EF0:BC00 (abbreviated with leading zeros omitted)

Rule 3: In an IPv6 address, a single continuous string of four or more zeros can be abbreviated as a double colon (::). The double colon abbreviation can only be used one time in an IP address.

2001:0404:0001:1000**:0000:0000:**0EF0:BC00

2001:404:1:1000**::**EF0:BC00 (abbreviated with leading zeroes omitted and continuous zeros replaced with a double colon)

The image below illustrates these rules of IPv6 address abbreviation:

```
  FF01:0000:0000:0000:0000:0000:0000:1
= FF01:0:0:0:0:0:0:1
= FF01::1
```

```
  E3D7:0000:0000:0000:51F4:00C8:C0A8:6420
= E3D7::51F4:C8:C0A8:6420
```

```
  3FFE:0501:0008:0000:0260:97FF:FE40:EFAB
= 3FFE:501:8:0:260:97FF:FE40:EFAB
= 3FFE:501:8::260:97FF:FE40:EFAB
```

Step 2: Practice compressing and decompressing IPv6 addresses.

Using the rules of IPv6 address abbreviation, either compress or decompress the following addresses:

1) 2002:0EC0:0200:0001:0000:04EB:44CE:08A2

2002:EC0:200:1::4EB:44CE:8A2

2) FE80:0000:0000:0001:0000:60BB:008E:7402

FE80::1:0:60BB:8E:7402

3) FE80::7042:B3D7:3DEC:84B8
   FE80:0000:0000:0000:7042:B3D7:3DEC:84B8
4) FF00::
   FF00:0000:0000:0000:0000:0000:0000:0000
5) 2001:0030:0001:ACAD:0000:330E:10C2:32BF
   2001:30:1:ACAD::330E:10C2:32BF

Reflection

1. How do you think you must support IPv6 in the future?

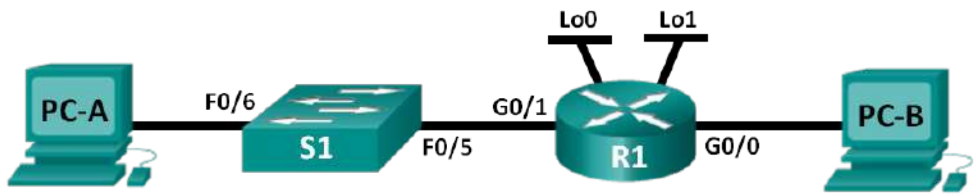
   Try not to waste IP address, only use the ones you really need, otherwise eventually IPv6 will probably end up like IPv4. Thou IPv6 got plenty of addresses at the moment, however if people waste too much addresses IPv6 addresses might eventually run out as well just like IPv4 addresses.

2. Do you think IPv4 networks continue on, or will everyone eventually switch over to IPv6? How long do you think it will take?

   Since we're already running out of IPv4 address, we probably will be all switch to IPv6 in near future. The future will be piling up with IoTs so it won't take long for the switching to happen.

Configuring Subnetted IPv4 Addressing

Topology



Addressing Table

Device	Interface	IP Address	Subnet Mask	Default Gateway
R1	G0/0	10.16.0.1	255.240.0.0	N/A
	G0/1	10.32.0.1	255.240.0.0	N/A
	Lo0	10.48.0.1	255.240.0.0	N/A
	Lo1	10.64.0.1	255.240.0.0	N/A
S1	VLAN 1	10.47.255.254	255.240.0.0	10.32.0.1
PC-A	NIC	10.32.0.2	255.240.0.0	10.32.0.1
PC-B	NIC	10.16.0.2	255.240.0.0	10.16.0.1

```
R1(config)#interface gigabit0/0
R1(config-if)#ip address 10.16.0.1 255.240.0.0
R1(config-if)#no shutdown
R1(config)#interface gigabitethernet0/1
R1(config-if)#ip address 10.32.0.1 255.240.0.0
R1(config-if)#no shutdown
R1(config)#interface lo0
R1(config-if)#ip address 10.48.0.1 255.240.0.0
R1(config-if)#no shutdown
R1(config)#interface loopback1
R1(config-if)#ip address 10.64.0.1 255.240.0.0
R1(config-if)#no shutdown
S1(config-if)#ip default-gateway 10.32.0.1
S1(config)#interface vlan1
S1(config-if)#ip address 10.47.255.254 255.240.0.0
S1(config-if)#no shutdown
```

Objectives

Part 1: Configure the Devices

Part 2: Test and Troubleshoot the Network

Background / Scenario

You will configure the host PCs, switch and router interfaces, including loopback interfaces. The loopback interfaces are created to simulate additional LANs attached to router R1.

After the network devices and host PCs have been configured, you will use the **ping** command to test for network connectivity.

Required Resources

Packet Tracer

Part 1: Configure the Devices

In Part 1, set up the network topology and configure settings on the PCs, Switch and router, such as the router Gigabit Ethernet interface IP addresses, and the PC's IP addresses, subnet masks, and default gateways. Refer to the Addressing Table for device names and address information:

a. Fully configure the switch as per the standard configuration steps discussed in previous labs and save the configuration:

1) Correct device names as per the topology

2) DNS lookup turned off `S1(config)#no ip domain-lookup`

3) IP address as listed in Addressing Table

4) Configure the default gateway for the Switch

```
interface Vlan1
 ip address 10.47.255.254 255.240.0.0
!
ip default-gateway 10.32.0.1
```

5) Clear text passwords encrypted. `S1(config)#service password-encryption`

6) **cisco** as the console and vty passwords with login enabled

```
S1(config)#line console 0
S1(config-line)#password cisco
S1(config-line)#login
S1(config-line)#exit
```

```
S1(config)#line vty 0 15
S1(config-line)#password cisco
S1(config-line)#login
S1(config-line)#exit
```

7) **class** as the privileged EXEC password `S1(config)#enable secret class`

8) Banner that warns anyone accessing the device that unauthorized access is prohibited. With the following text:

```
S1(config)#banner motd #Unauthorised access is prohibited and will be
strictly prosecuted.#
```

Unauthorised access is prohibited and will be strictly prosecuted.

9) Save the configuration

```
S1#copy running-config startup-config
Destination filename [startup-config]?
Building configuration...
[OK]
```

b. Configure the router:

Configure the routers as per the standard configuration in previous labs:

1) Assign a device name to the router.

2) Disable DNS lookup to prevent the router from attempting to translate incorrectly entered commands as though they were host names. `R1(config)#no ip domain-lookup`

3) Assign **class** as the privileged EXEC encrypted password. `R1(config)#enable secret class`

4) Assign **cisco** as the console password and enable login.

```
R1(config)#line console 0
R1(config-line)#password cisco
R1(config-line)#login
R1(config-line)#exit
```

5) Assign **cisco** as the VTY password and enable login. Verify the number of vty lines are on the router.

```
R1(config)#line vty 0 4
R1(config-line)#password cisco
R1(config-line)#login
R1(config-line)# exit
```

6) Encrypt the clear text passwords. `R1(config)#service password-encryption`

7) Create a banner that warns anyone accessing the device:

```
R1(config)#banner motd #Unauthorised access is prohibited and you will be
strictly prosecuted.#
```

Unauthorised access is prohibited and you will be strictly prosecuted.

8) Assign *IPv4* addresses to all interfaces on Router as per the addressing table and enable them.

```
interface Loopback0
 ip address 10.48.0.1 255.240.0.0
!
interface Loopback1
 ip address 10.64.0.1 255.240.0.0
!
interface GigabitEthernet0/0
 ip address 10.16.0.1 255.240.0.0
 duplex auto
 speed auto
!
interface GigabitEthernet0/1
 ip address 10.32.0.1 255.240.0.0
 duplex auto
 speed auto
!
```

9) Save the running configuration to the startup configuration.

```
R1#copy running-config startup-config
Destination filename [startup-config]?
Building configuration...
[OK]
```

Step 2: Configure the PC interfaces.

a. Configure the IP address, subnet mask, and default gateway settings on PC-A.

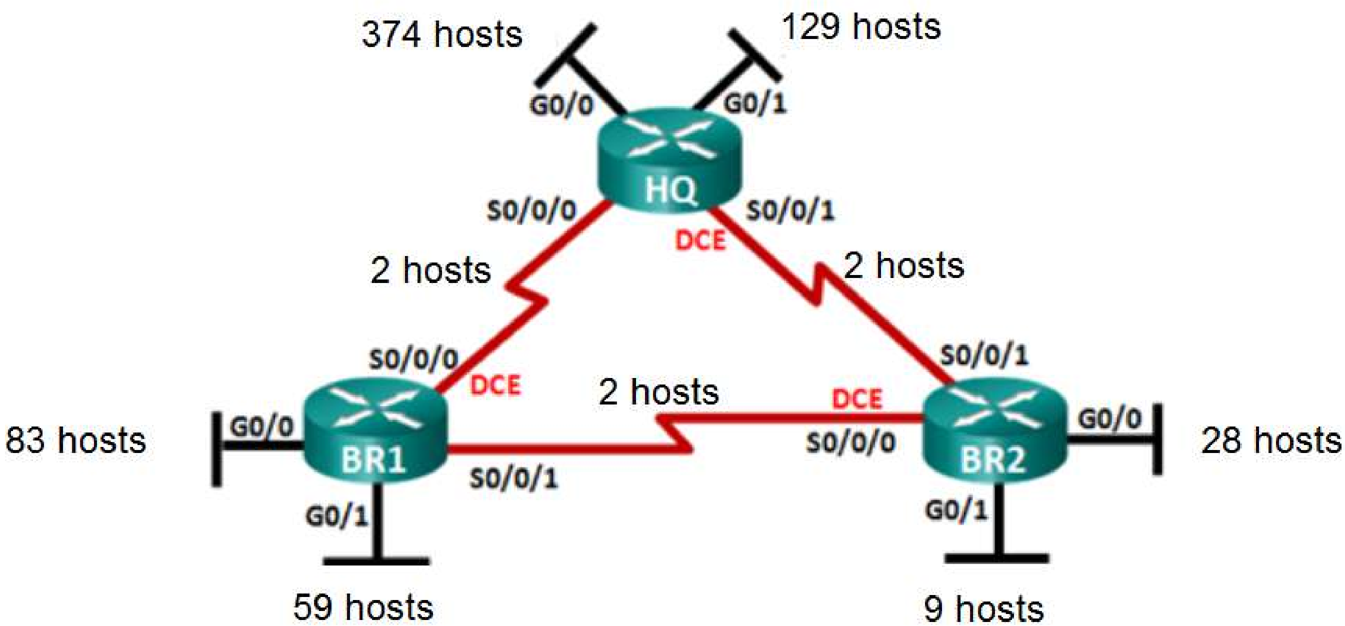
b. Configure the IP address, subnet mask, and default gateway settings on PC-B.

Part 2: Test and Troubleshoot the Network:

Use appropriate commands to verify the configuration

Design VLSM Addressing Scheme

Topology



Objectives

Part 1: Examine Network Requirements

Part 2: Design the VLSM Address Scheme

Background / Scenario

Variable Length Subnet Mask (VLSM) was designed to avoid wasting IP addresses. With VLSM, a network is subnetted and then re-subnetted. This process can be repeated multiple times to create subnets of various sizes based on the number of hosts required in each subnet. Effective use of VLSM requires address planning.

In this lab, use the 172.16.4.0/22 network address to develop an address scheme for the network displayed in the topology diagram. Imagine that this network has been provided to you by your ISP. VLSM is used to meet the IPv4 addressing requirements. After you have designed the VLSM address scheme, you will configure the interfaces on the routers with the appropriate IP address information.

Note: Serial cables have two ends, one of which is called DCE and the other is called DTE. The DCE end will provide the clock and other control signals to the DTE end.

Part 1: Examine Network Requirements

In Part 1, you will examine the network requirements to develop a VLSM address scheme for the network displayed in the topology diagram using the 172.16.4.0/22 network address.

Step 1: Determine how many host addresses and subnets are available.

How many host addresses are available in a /22 network? 1022 (2^10 -2 = 1024 – 2)

How many networks are available in the 172.16.4.0/22 network if only the 3rd octet is used? 4

With a /22 prefix the subnet mask is: 11111111 . 11111111 . 11111100 . 00000000

The magic number is: 4

The starting network is: 172.16.4.0 The next network starts at: 172.16.8.0

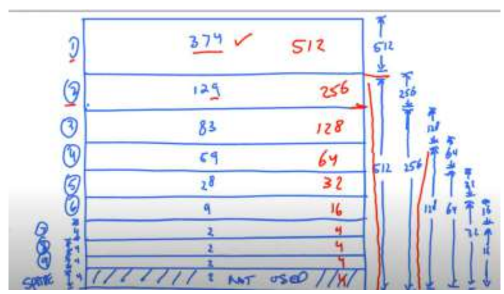
This leaves the following 4 possible networks:

172.16.4.0	10101100 . 00010000 . 00000100 . 00000000
172.16.5.0	10101100 . 00010000 . 00000101 . 00000000
172.16.6.0	10101100 . 00010000 . 00000110 . 00000000
172.16.7.0	10101100 . 00010000 . 00000111 . 00000000

How many hosts can each of these networks support? 256

What is the total number of host addresses needed in the topology diagram? 688

HQ-G0/0: 374 +
HQ-G0/1: 129 +
BR1-G0/0: 83 +
BR1-G0/1: 59 +
BR2-G0/1: 9 +
BR2-G0/0: 28 +
HQ_To_BR1: 2 +
BR1_To_BR2: 2 +
BR2_To_HQ: 2



What is the minimum number of bits needed to support all the hosts? 10

Given that many host-bits, how many bits are left for the network and subnets? 22

What is the subnet mask for a /22 network? 255.255.252.0

What is the binary for the /22 subnet mask? 11111111 . 11111111 . 11111100 . 00000000

What is the binary for IP address 172.16.4.0: 10101100 . 00010000 . 00000100 . 00000000

How many subnets are needed in the network topology? 9

HQ-G0/0:
HQ-G0/1
BR1-G0/0
BR1-G0/1
BR2-G0/1
BR2-G0/0
HQ-S0/0/0_To_BR1_S/0/0/0
BR1-S0/0/1_To_BR2-S/0/0/0
BR2-S0/0/1_To_HQ-S0/0/1

Step 2: Determine the largest subnet.

What is the subnet name of the largest subnet?	HQ-G0/0
How many IP addresses are required in the largest subnet?	374
How many bits are required to support that many hosts?	9 (2^9 = 512)
How many hosts can that many bits actually support?	510 (2^9 – 2)
What subnet mask can support that many host addresses?	/23

```
/23        11111111 . 1111111 . 11111110 . 00000000
                                       ^  (2^9 = 512)
```

How many actual host addresses can that subnet mask support?	510 (512 – 2)
Can you subnet the 172.16.4.0/22 network address to support this subnet?	Yes

What are the two network addresses that would result from this subnetting?

Network address 1:	Decimal:	172.16.4.0/23
	Binary:	10101100 . 00010000 . 00000100 . 00000000
	Subnet mask:	11111111 . 11111111 . 11111110 . 00000000
	First host address:	10101100 . 00010000 . 00000100 . 00000001
	Decimal:	172.16.4.1
	Last host address:	10101100 . 00010000 . 00000101 . 11111110
	Decimal:	172.16.5.254
	Broadcast address:	10101100 . 00010000 . 00000101 . 11111111
	Decimal:	172.16.5.255
Network address 2:	Decimal:	172.16.6.0/23
	Binary:	10101100 . 00010000 . 00000110 . 00000000
	Subnet mask:	11111111 . 11111111 . 11111110 . 00000000
	First host address:	10101100 . 00010000 . 00000110 . 00000001
	Decimal:	172.16.6.1
	Last host address:	10101100 . 00010000 . 00000111 . 11111110
	Decimal:	172.16.7.254
	Broadcast address:	10101100 . 00010000 . 00000111 . 11111111
	Decimal:	172.16.7.255

Use the first network address (172.16.4.0/23) for this subnet. That leaves the second network (172.16.6.0/23) with up to 510 available IP addresses for further subnetting.

Step 3: Determine the second largest subnet.

What is the subnet name for the next largest subnet? HQ-G0/1

How many IP addresses are required in the next largest subnet? 129

How many bits are required to support that many hosts? 8 (2^8 = 256)

How many hosts can that many bits actually support? 254 (2^8 – 2)

What subnet mask can support that many host addresses? /24

```
/24      11111111 . 1111111 . 11111111 . 00000000
                                         ^ (2^8 = 256)
```

How many actual host addresses can that subnet mask support? 254 (256 – 2)

Can you subnet the remaining network from Step 2 (172.16.6.0/23) to support this subnet?

Yes, the remaining subnet (/23) can support 510 possible hosts. Splitting this in two gives us two further subnets with up to 254 hosts each.

What are the two network addresses that would result from this subnetting?

Network address 1:	Decimal:	172.16.6.0/24
	Binary:	10101100 . 00010000 . 00000110 . 00000000
	Subnet mask:	11111111 . 11111111 . 11111111 . 00000000
	First host address:	10101100 . 00010000 . 00000110 . 00000001
	Decimal:	172.16.6.1
	Last host address:	10101100 . 00010000 . 00000110 . 11111110
	Decimal:	172.16.6.254
	Broadcast address:	10101100 . 00010000 . 00000110 . 11111111
	Decimal:	172.16.6.255
Network address 2:	Decimal:	172.16.7.0/24
	Binary:	10101100 . 00010000 . 00000111 . 00000000
	Subnet mask:	11111111 . 11111111 . 11111111 . 00000000
	First host address:	10101100 . 00010000 . 00000111 . 00000001
	Decimal:	172.16.7.1
	Last host address:	10101100 . 00010000 . 00000111 . 11111110
	Decimal:	172.16.7.254
	Broadcast address:	10101100 . 00010000 . 00000111 . 11111111
	Decimal:	172.16.7.255

Use the first network address (172.16.6.0/24) for this subnet. That leaves the second network (172.16.7.0/24) with up to 510 available IP addresses for further subnetting.

Step 4: Determine the next (third) largest subnet.

What is the subnet name for the next largest subnet?	BR1-G0/0
How many IP addresses are required in the next largest subnet?	83
How many bits are required to support that many hosts?	7 (2^7 = 128)
How many hosts can that many bits actually support?	126 (2^7 – 2)
What subnet mask can support that many host addresses?	/25

```
/25        11111111 . 1111111 . 11111111 . 10000000
                                           ^ (2^7 = 128)
```

How many actual host addresses can that subnet mask support? 126 (128 – 2)

Can you subnet the remaining network from Step 3 (172.16.7.0/24) to support this subnet?

Yes, the remaining subnet (/24) can support 256 possible hosts. Splitting this in two gives us two further subnets with up to 126 hosts each.

What are the two network addresses that would result from this subnetting?

Network address 1:	Decimal:	172.16.7.0/25
	Binary:	`10101100 . 00010000 . 00000111 . 00000000`
	Subnet mask:	`11111111 . 11111111 . 11111111 . 10000000`
	First host address:	`10101100 . 00010000 . 00000111 . 00000001`
	Decimal:	172.16.7.1
	Last host address:	`10101100 . 00010000 . 00000111 . 01111110`
	Decimal:	172.16.7.126
	Broadcast address:	`10101100 . 00010000 . 00000111 . 01111111`
	Decimal:	172.16.7.127
Network address 2:	Decimal:	172.16.7.128/25
	Binary:	`10101100 . 00010000 . 00000111 . 10000000`
	Subnet mask:	`11111111 . 11111111 . 11111111 . 10000000`
	First host address:	`10101100 . 00010000 . 00000111 . 10000001`
	Decimal:	172.16.7.129
	Last host address:	`10101100 . 00010000 . 00000111 . 11111110`
	Decimal:	172.16.7.254
	Broadcast address:	`10101100 . 00010000 . 00000111 . 11111111`
	Decimal:	172.16.7.255

Use the first network address (172.16.7.0/25) for this subnet. That leaves the second network (172.16.7.128/25) with up to 128 available IP addresses for further subnetting.

Step 5: Determine the next (fourth) largest subnet.

What is the subnet name for the next largest subnet?	BR1-G0/1
How many IP addresses are required in the next largest subnet?	59
How many bits are required to support that many hosts?	6 (2^6 = 64)
How many hosts can that many bits actually support?	62 (2^6 – 2)
What subnet mask can support that many host addresses?	/26

```
/26        11111111 . 1111111 . 11111111 . 11000000
                                            ^ (2^6 = 64)
```

How many actual host addresses can that subnet mask support? 62 (64 – 2)

Can you subnet the remaining network from Step 4 (172.16.7.128/25) to support this subnet?

Yes, the remaining subnet (/25) can support 128 possible hosts. Splitting this in two gives us two further subnets with up to 64 hosts each.

What are the two network addresses that would result from this subnetting?

Network address 1:	Decimal:	172.16.7.128/26
	Binary:	10101100 . 00010000 . 00000111 . 10000000
	Subnet mask:	11111111 . 11111111 . 11111111 . 11000000
	First host address:	10101100 . 00010000 . 00000111 . 10000001
	Decimal:	172.16.7.129
	Last host address:	10101100 . 00010000 . 00000111 . 10111110
	Decimal:	172.16.7.190
	Broadcast address:	10101100 . 00010000 . 00000111 . 10111111
	Decimal:	172.16.7.191
Network address 2:	Decimal:	172.16.7.192/26
	Binary:	10101100 . 00010000 . 00000111 . 11000000
	Subnet mask:	11111111 . 11111111 . 11111111 . 11000000
	First host address:	10101100 . 00010000 . 00000111 . 11000001
	Decimal:	172.16.7.193
	Last host address:	10101100 . 00010000 . 00000111 . 11111110
	Decimal:	172.16.7.254
	Broadcast address:	10101100 . 00010000 . 00000111 . 11111111
	Decimal:	172.16.7.255

Use the first network address (172.16.7.128/26) for this subnet. That leaves the second network (172.16.7.192/26) with up to 64 available IP addresses for further subnetting.

Step 6: Determine the next (fifth) largest subnet.

What is the subnet name for the next largest subnet? BR2-G0/0

How many IP addresses are required in the next largest subnet? 28

How many bits are required to support that many hosts? 5 (2^5 = 32)

How many hosts can that many bits actually support? 30 (2^5 – 2)

What subnet mask can support that many host addresses? /27

```
/27        11111111 . 1111111 . 11111111 . 11100000
                                              ^ (2^5 = 32)
```

How many actual host addresses can that subnet mask support? 30 (32 – 2)

Can you subnet the remaining network from Step 5 (172.16.7.192/26) to support this subnet?

Yes, the remaining subnet (/26) can support 64 possible hosts. Splitting this in two gives us two further subnets with up to 32 hosts each.

What are the two network addresses that would result from this subnetting?

Network address 1:

Decimal:	172.16.7.192/27	
Binary:	10101100 . 00010000 . 00000111 . 11000000	
Subnet mask:	11111111 . 11111111 . 11111111 . 11100000	
First host address:	10101100 . 00010000 . 00000111 . 11000001	
Decimal:	172.16.7.193	
Last host address:	10101100 . 00010000 . 00000111 . 11011110	
Decimal:	172.16.7.222	
Broadcast address:	10101100 . 00010000 . 00000111 . 11011111	
Decimal:	172.16.7.223	

Network address 2:

Decimal:	172.16.7.224/27	
Binary:	10101100 . 00010000 . 00000111 . 11100000	
Subnet mask:	11111111 . 11111111 . 11111111 . 11100000	
First host address:	10101100 . 00010000 . 00000111 . 11100001	
Decimal:	172.16.7.225	
Last host address:	10101100 . 00010000 . 00000111 . 11111110	
Decimal:	172.16.7.254	
Broadcast address:	10101100 . 00010000 . 00000111 . 11111111	
Decimal:	172.16.7.255	

Use the first network address (172.16.7.192/27) for this subnet. That leaves the second network

Step 7: Determine the next (sixth) largest subnet.

What is the subnet name for the next largest subnet? BR2-G0/1

How many IP addresses are required in the next largest subnet? 9

How many bits are required to support that many hosts? 4 (2^4 = 16)

How many hosts can that many bits actually support? 14 (2^4 – 2)

What subnet mask can support that many host addresses? /28

```
/28        11111111 . 1111111 . 11111111 . 11110000
                                               ^ (2^4 = 16)
```

How many actual host addresses can that subnet mask support? 14 (16 – 2)

Can you subnet the remaining network from Step 5 (172.16.7.224/27) to support this subnet?

Yes, the remaining subnet (/27) can support 32 possible hosts. Splitting this in two gives us two further subnets with up to 16 hosts each.

What are the two network addresses that would result from this subnetting?

Network address 1:	Decimal:	172.16.7.224/28
	Binary:	10101100 . 00010000 . 00000111 . 11100000
	Subnet mask:	11111111 . 11111111 . 11111111 . 11110000
	First host address:	10101100 . 00010000 . 00000111 . 11100001
	Decimal:	172.16.7.224
	Last host address:	10101100 . 00010000 . 00000111 . 11101110
	Decimal:	172.16.7.238
	Broadcast address:	10101100 . 00010000 . 00000111 . 11101111
	Decimal:	172.16.7.239
Network address 2:	Decimal:	172.16.7.240/28
	Binary:	10101100 . 00010000 . 00000111 . 11110000
	Subnet mask:	11111111 . 11111111 . 11111111 . 11110000
	First host address:	10101100 . 00010000 . 00000111 . 11110001
	Decimal:	172.16.7.241
	Last host address:	10101100 . 00010000 . 00000111 . 11111110
	Decimal:	172.16.7.254
	Broadcast address:	10101100 . 00010000 . 00000111 . 11111111
	Decimal:	172.16.7.255

Use the first network address (172.16.7.224/28) for this subnet. That leaves the second network

Step 8: Determine the (three) subnets needed to support the serial links.

We will need to support three router to router networks (serial links). Each of these networks consists of 2 routers (actual hosts). What are the subnet names for the three remaining router to router networks?

HQ-S0/0/0_To_BR1_S/0/0/0

BR1-S0/0/1_To_BR2-S/0/0/0

BR2-S0/0/1_To_HQ-S0/0/1

How many IP addresses are required in each of these subnets? 2

How many bits are required to support that many hosts? 2 (2^2 = 4)

How many hosts can that many bits actually support? 2 (2^2 – 2)

What subnet mask can support that many host addresses? /30

```
/30        11111111 . 1111111 . 11111111 . 11111100
                                                ^ (2^2 = 4)
```

How many actual host addresses can that subnet mask support? 2 (4 – 2)

Can you subnet the remaining network from Step 5 (172.16.7.240/28) to support these three subnets?

Yes, the remaining subnet (/28) can support 16 possible hosts. We need 3 x 4 = 12 hosts to support the three router to router networks. The remaining /28 subnet could support up to 4 /30 subnets.

What are the four network addresses that would result from this subnetting?

Network address 1:	Decimal:	172.16.7.240/30
	Binary:	10101100 . 00010000 . 00000111 . 11111000
	Subnet mask:	11111111 . 11111111 . 11111111 . 11111100
	First host address:	10101100 . 00010000 . 00000111 . 11110001
	Decimal:	172.16.7.241
	Last host address:	10101100 . 00010000 . 00000111 . 11110010
	Decimal:	172.16.7.242
	Broadcast address:	10101100 . 00010000 . 00000111 . 11110011
	Decimal:	172.16.7.243
Network address 2:	Decimal:	172.16.7.244/30
	Binary:	10101100 . 00010000 . 00000111 . 11110100
	Subnet mask:	11111111 . 11111111 . 11111111 . 11111100
	First host address:	10101100 . 00010000 . 00000111 . 11110101
	Decimal:	172.16.7.245
	Last host address:	10101100 . 00010000 . 00000111 . 11110110
	Decimal:	172.16.7.246
	Broadcast address:	10101100 . 00010000 . 00000111 . 11110111
	Decimal:	172.16.7.247

Network address 3:	Decimal:	172.16.7.248/30
	Binary:	10101100 . 00010000 . 00000111 . 11111000
	Subnet mask:	11111111 . 11111111 . 11111111 . 11111100
	First host address:	10101100 . 00010000 . 00000111 . 11111001
	Decimal:	172.16.7.249
	Last host address:	10101100 . 00010000 . 00000111 . 11111010
	Decimal:	172.16.7.250
	Broadcast address:	10101100 . 00010000 . 00000111 . 11111011
	Decimal:	172.16.7.251
Network address 4:	Decimal:	172.16.7.252/30
	Binary:	10101100 . 00010000 . 00000111 . 11111100
	Subnet mask:	11111111 . 11111111 . 11111111 . 11111100
	First host address:	10101100 . 00010000 . 00000111 . 11111101
	Decimal:	172.16.7.253
	Last host address:	10101100 . 00010000 . 00000111 . 11111110
	Decimal:	172.16.7.254
	Broadcast address:	10101100 . 00010000 . 00000111 . 11111111

8a. Use the first network address (172.16.7.240/30) for HQ-S0/0/0_To_BR1_S/0/0/0.

8b. Use the second network address (172.16.7.248/30) for BR2-S0/0/1_To_HQ-S0/0/1.

8c. Use the third network address (172.16.7.244/30) for BR1-S0/0/1_To_BR2-S/0/0/0.

8d. The remaining (fourth) network address (172.16.7.252/30) is unused for now.

Part 2: Design the VLSM Address Scheme

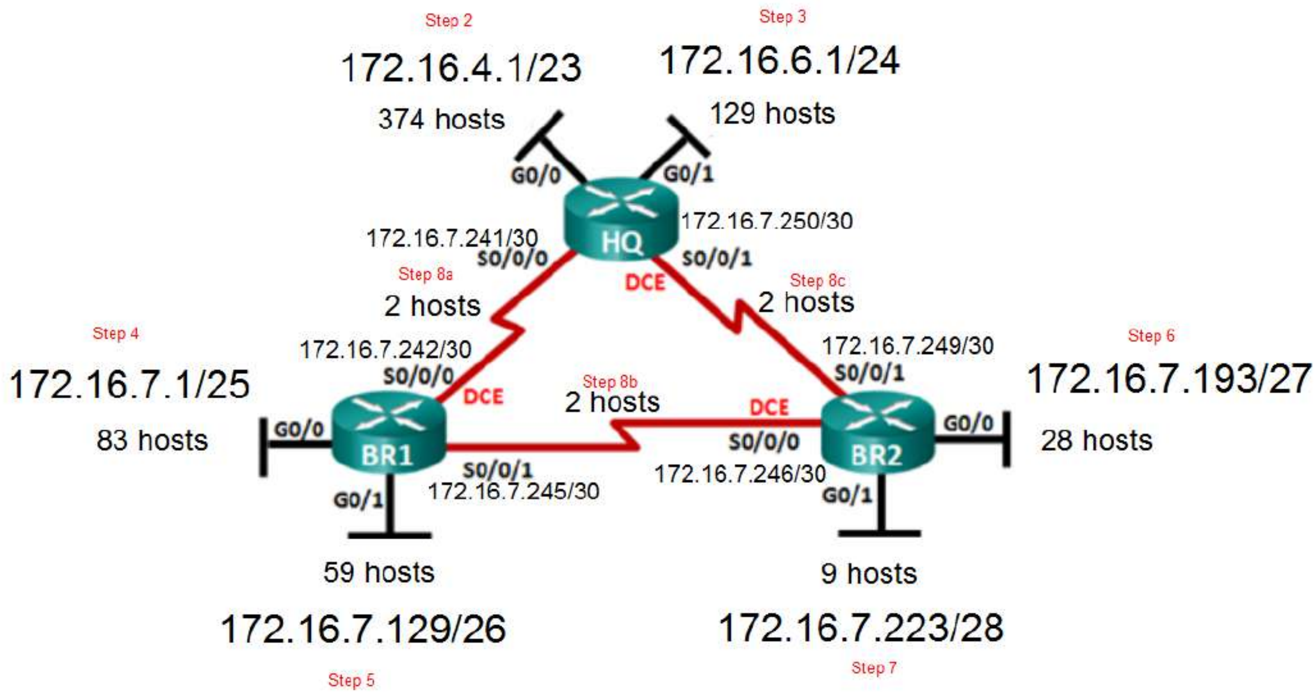
Step 1: Calculate the subnet information.

Subnet Description	Number of Hosts Needed	Subnet size	Network Address /CIDR	First Host Address	Last Host Address	Broadcast Address	Step #
HQ G0/0	374	512	172.16.4.0/23	172.16.4.1	172.16.5.254	172.16.5.255	2
HQ G0/1	129	256	172.16.6.0/24	172.16.6.1	172.16.6.254	172.16.6.255	3
BR1 G0/0	83	128	172.16.7.0/25	172.16.7.1	172.16.7.126	172.16.7.127	4
BR1 G0/1	59	64	172.16.7.128/26	172.16.7.129	172.16.7.190	172.16.7.191	5
BR2 G0/0	28	32	172.16.7.192/27	172.16.7.193	172.16.7.222	172.16.7.223	6
BR2 G0/1	9	16	172.16.7.224/28	172.16.7.225	172.16.7.238	172.16.7.239	7
HQ S0/0/0 – BR1 S0/0/0	2	4	172.16.7.240/30	172.16.7.241	172.16.7.242	172.16.7.243	8a
HQ S0/0/1 – BR2 S0/0/1	2	4	172.16.7.244/30	172.16.7.245	172.16.7.246	172.16.7.247	8b
BR1 S0/0/1 – BR2 S0/0/0	2	4	172.16.7.248/30	172.16.7.249	172.16.7.250	172.16.7.251	8c

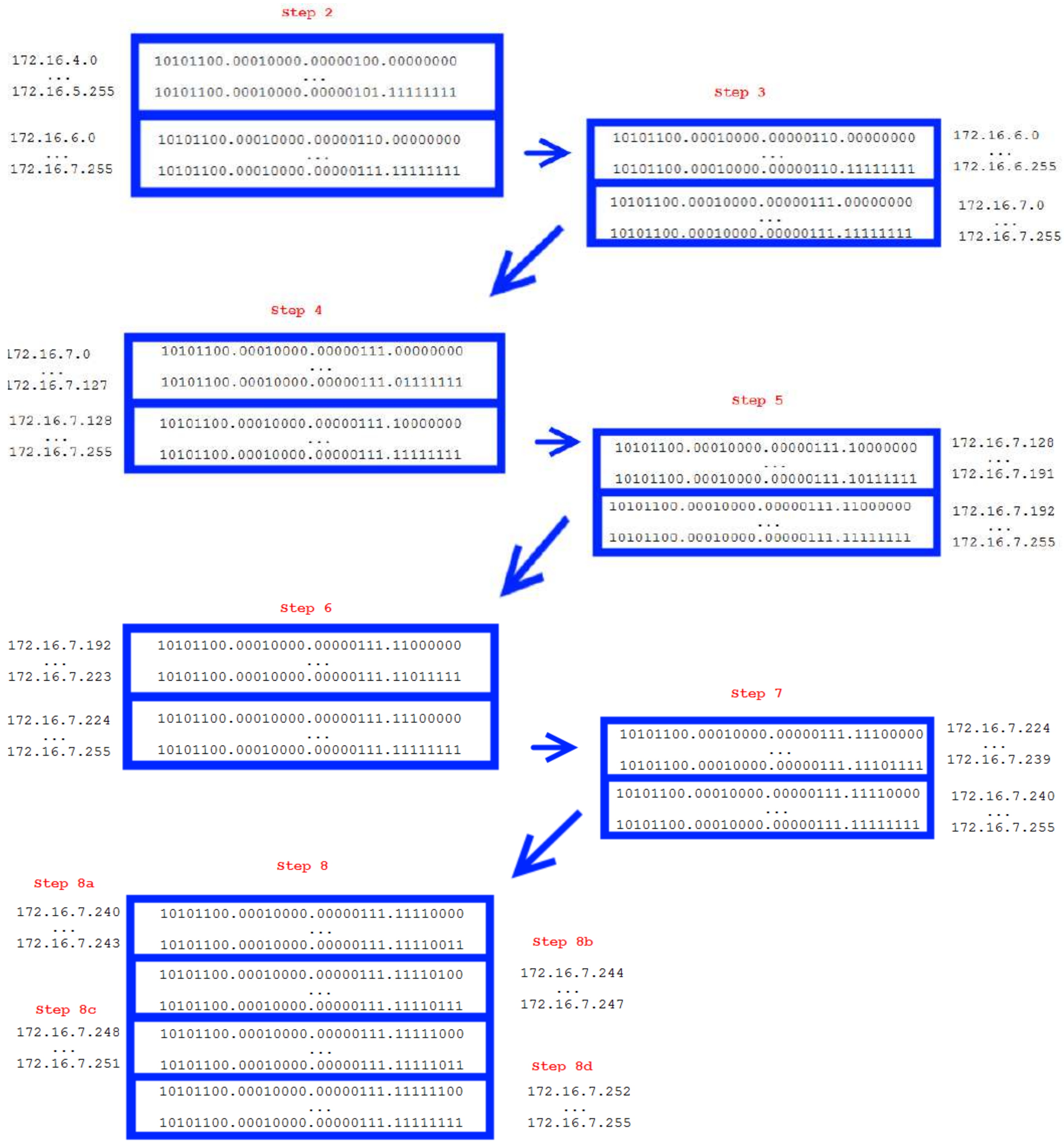
Step 2: Complete the device interface address table.

Assign the first host address in the subnet to the Ethernet interfaces.

Device	Interface	IP Address	Prefix	Subnet mask	Connects to	Step #
HQ	G0/0	172.16.4.1	/23	255.255.254.0	374 Host LAN	2
	G0/1	172.16.6.1	/24	255.255.255.0	129 Host LAN	3
	S0/0/0	172.16.7.241	/30	255.255.255.252	BR1 S0/0/0	8a
	S0/0/1	172.16.7.250	/30	255.255.255.252	BR2 S0/0/1	8c
BR1	G0/0	172.16.7.1	/25	255.255.255.128	83 Host LAN	4
	G0/1	172.16.7.129	/26	255.255.255.192	59 Host LAN	5
	S0/0/0	172.16.7.242	/30	255.255.255.252	HQ S0/0/0	8a
	S0/0/1	172.16.7.245	/30	255.255.255.252	BR2 S0/0/0	8b
BR2	G0/0	172.16.193	/27	255.255.255.224	28 Host LAN	6
	G0/1	172.16.7.225	/28	255.255.255.240	9 Host LAN	7
	S0/0/0	172.16.7.246	/30	255.255.255.252	BR1 S0/0/1	8b
	S0/0/1	172.16.7.249	/30	255.255.255.252	HQ S0/0/1	8c



Subnet work allocation diagram:



Troubleshooting practice

Ticket number	Problem as described	Actual problem	How to diagnose	How to fix
1	PC5 cannot telnet to PC6	S2 has a shutdown link connecting to S1	checked connections on both switches using show ip interface brief command	On S2, no shut-down on f0/1.
2	PC7 cannot ping G0/1	Interface g0/1 on R4 is manually shutdown	On R4, using show ip interface brief command to check connections	On R4, no shut-down on g0/1.
3	PC5 Cannot ping PC7	ip address on PC7 incorrectly configured	check PC7's ip configuration	Configure PC7 IP address correctly
4	PC6 cannot Telnet to S1	vlan1's interface of S1 is not configured.	On S1, use show ip interface brief	configure vlan1: interface vlan1 -> ip address 192.168.50.30 255.255.255.0 -> exit -> ip default-gateway 192.168.50.1
5	PC5 cannot login to router via telnet	misconfigure password for vty lines	consoled into the device okay. checked vty password	configured the vty line 04 with password cisco -> login
6	PC6 cannot ping PC7	default gateway on PC6 not set	checked default gateway configuration on PC6	configured default gateway on PC6 correctly
Other things to note:	There may be other problems with the topology but unless they are assigned a ticket number they are not considered as a problem to solve at this time.			