

# IP Addressing

# Objectives

## The objectives of this chapter are to:

- Understand the format of IP addresses and the use of dotted decimal notation
- Understand the difference between Class A,B,C, and D addresses and how computers distinguish between them
- Describe the reason for subnetting and understand the use of subnet masks
- Given an IP address and subnet mask, work out the subnet address, broadcast address, and host range applicable.
- Understand VLSM and the use of summarization
- Understand IPv6 Addressing

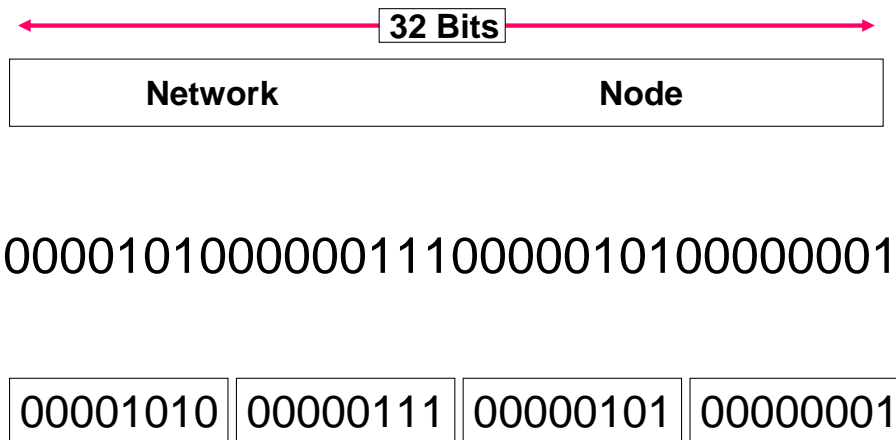
9 - 2

Copyright © Pancho Training & Consultancy Ltd. 2007

The objectives of this chapter are to:

- Understand the format of IP addresses and the use of dotted decimal notation
- Understand the difference between Class A,B,C, and D addresses and how computers distinguish between them
- Describe the reason for subnetting and understand the use of subnet masks
- Given an IP address and subnet mask, work out the subnet address, broadcast address, and host range applicable.

# IP Address Format



9 - 3

Copyright © Pancho Training & Consultancy Ltd. 2007

The IP address is 32 bits in length and has two parts:

Network number

Host number

The address format is known as dotted decimal notation

Example address: 10.7.5.1

Each bit in the octet has a binary weight, such as (128,...4, 2, 1).

The minimum value for an octet is 0; it contains all 0s.

The maximum value for an octet is 255; it contains all 1s.

The allocation of addresses is managed by a central authority, the Internet Assigned Numbers Authority.

# IP Address Format

00001010 00000111 00000101 00000001

128 64 32 16 8 4 2 1  
0 0 0 0 1 0 1 0  
= 10 decimal

128 64 32 16 8 4 2 1  
0 0 0 0 0 1 1 1  
= 7 decimal

9 - 4

Copyright © Pancho Training & Consultancy Ltd. 2007

The 32 bit address is split into 4 x 8 bit “chunks,” each one of which is represented by a decimal number equating to the 8 bit value.

The first “chunk” or octet in this address is represented by the value decimal 10 while the second is represented by decimal 7.

In order to differentiate between the octets, we use a dot as a separator i.e 10.7

# IP Address Format

00001010 00000111 00000101 00000001

128 64 32 16 8 4 2 1  
0 0 0 0 0 1 0 1  
= 5 decimal

128 64 32 16 8 4 2 1  
0 0 0 0 0 0 0 1  
= 1 decimal

9 - 5

Copyright © Pancho Training & Consultancy Ltd. 2007

To continue the example, the third and fourth octet are represented by decimal 5 and 1 respectively.

The full decimal notation for these 32 bits is therefore 10.7.5.1

## Dotted Decimal Notation

00001010	00000111	00000101	00000001
----------	----------	----------	----------

**10 . 7 . 5 . 1**

9 - 6

Copyright © Pancho Training & Consultancy Ltd. 2007

The full decimal notation for these 32 bits is therefore 10.7.5.1

## Other Layer 3 Protocols

	<u>Network</u>	<u>Node</u>
<b>Novell IPX</b>	<b>32 Bits</b>	<b>48 Bits</b>
<b>Appletalk</b>	<b>16 bits</b>	<b>8 Bits</b>
<b>DECNet</b>	<b>6 Bits</b>	<b>10 Bits</b>

9 - 7

Copyright © Pancho Training & Consultancy Ltd. 2007

The concept of network and host portion of an address is not unique to TCP/IP.

All other routable protocols use the same idea. Above we can see just three examples of other layer 3 protocols.

Novell IPX

32 bits network, 48 bits node (made up of the MAC address)

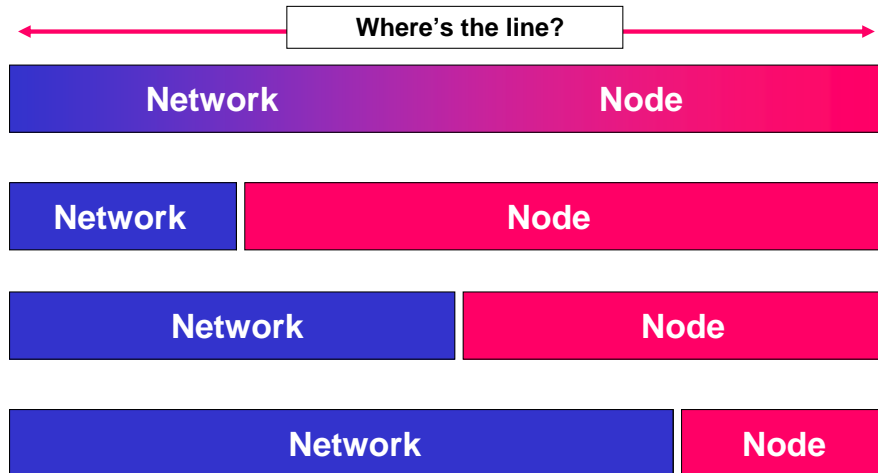
Appletalk

16 bits network, 8 bits node

DECnet

6 bits network, 10 bits node

# IP Address Format



9 - 8

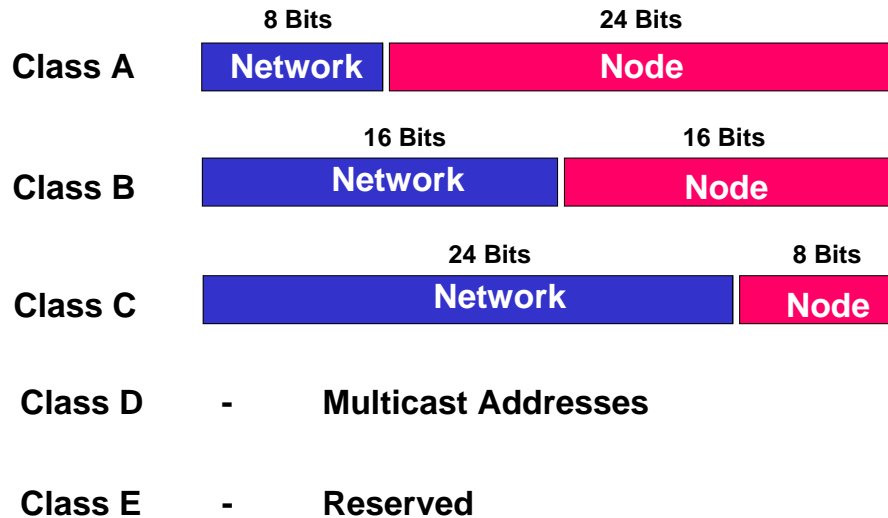
Copyright © Pancho Training & Consultancy Ltd. 2007

The major difference between TCP/IP and the other protocols is the fact that the line that divides the network portion of the address from the node is movable while the other protocols have fixed length network and node formats.

TCP/IP allows us to have an IP address that has a network portion of 8 bits, and a node of 24 bits giving us potentially 256 networks, each one of which can have 16.7 million hosts.

On the other hand, we could have 24 bits of network space and only 8 bits of node. This would create 16.7 million networks, each of which would be able to support 256 hosts.

# IP Address Format



9 - 9

Copyright © Pancho Training & Consultancy Ltd. 2007

It was decided to create five classes of address, A –E, each with a set number of network bits and a set number of host bits.

Classes A,B, and C would be for general use, class D would be for multicast (group) addressing and E would be reserved

The most significant bit pattern determines the class of the address, as well as how many bits make up the network portion of the address

Class A addresses include

- Range of network numbers: 1.0.0.0 to 126.0.0.0
- Number of host addresses: 16,777,214

Class B addresses include

- Range of network numbers: 128.1.0.0 to 191.254.0.0
- Number of host addresses: 65,534

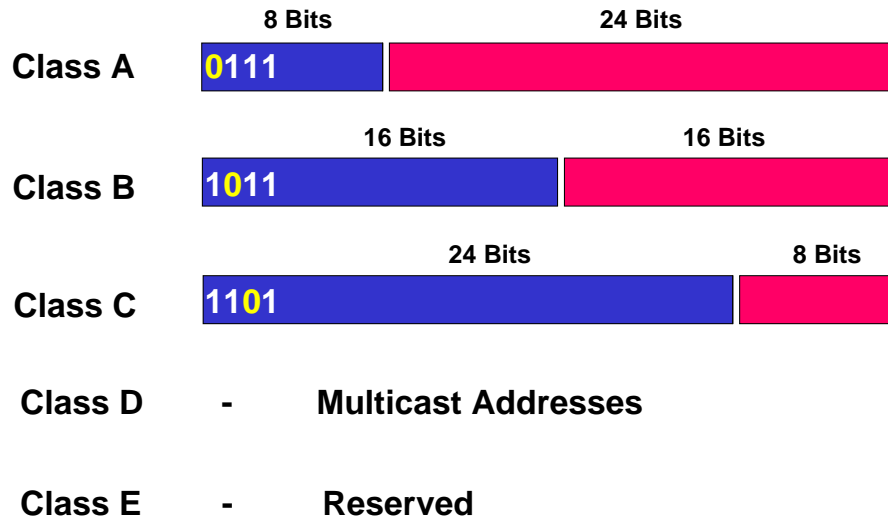
Class C addresses include

- Range of network numbers: 192.0.1.0 to 223.255.254.0
- Number of host addresses: 254

Class D addresses include

- Range of network numbers: 224.0.0.0 to 239.255.255.254

# IP Address Format



9 - 10

Copyright © Pancho Training & Consultancy Ltd. 2007

The problem was how would a computer device be able to distinguish between a class A address compared to a class B or C.

The answer was to get the computer to look for the most significant zero in the address.

If the zero was in the most significant position, the computer would recognise that as a class A address and know that the network/host line was after 8 bits.

If the zero was in the second position, the computer would recognise that as a class B address and know that the network/host line was after 16 bits.

If the zero was in the third position, the computer would recognise that as a class C address and know that the network/host line was after 24 bits.

That meant that for each of those classes of address, the first, second, or third bit must be set to zero which limits the address range for each of those classes.

# IP Address Format



From

128	64	32	16	8	4	2	1
0	0	0	0	0	0	0	0
= 0 decimal							

To

128	64	32	16	8	4	2	1
0	1	1	1	1	1	1	1
= 127 decimal							

**0 = Reserved**  
**127 = Loopback address**

**Range = 1 -126**

9 - 11

Copyright © Pancho Training & Consultancy Ltd. 2007

If this address is a class A address the first bit is set to zero but by doing this we lose 128 of our bit combinations in the first octet.

Instead of having a range of 0 – 256, we now have a range of 0 –127

The Internet authority also reserved 0 and 127 leaving us with a range of 1 –126

This does mean however that any address beginning with a decimal number between 1 and 126 can be recognised by us humans as a class A address.

# IP Address Format



From

128	64	32	16	8	4	2	1
1	0	0	0	0	0	0	0
= 128 decimal							

To

128	64	32	16	8	4	2	1
1	0	1	1	1	1	1	1
= 191 decimal							

**Range = 128 - 191**

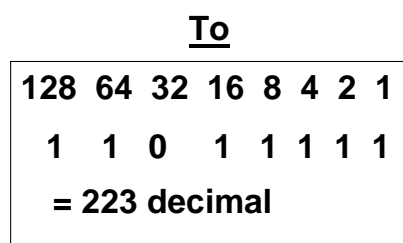
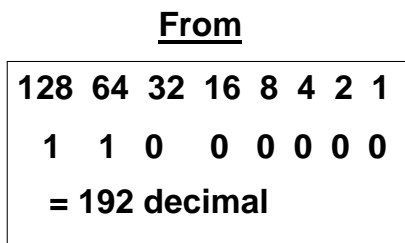
9 - 12

Copyright © Pancho Training & Consultancy Ltd. 2007

Similarly, a class B address has the second bit set to zero leaving us with a range of addresses beginning 128 – 191.

Any IP address that starts with a high order octet value of 128 – 191 is a class B address.

# IP Address Format



**Range = 192 - 223**

9 - 13

Copyright © Pancho Training & Consultancy Ltd. 2007

A class C address has the third bit set to zero leaving us with a range of addresses beginning 192 -223

Any IP address that starts with a high order octet value of 192 - 223 is a class C address.

# IP Address Format

Class D

1110

Multicast Group

Multicast Group

Multicast Group

From

128	64	32	16	8	4	2	1
1	1	1	0	0	0	0	0
= 224 decimal							

To

128	64	32	16	8	4	2	1
1	1	1	0	1	1	1	1
= 239 decimal							

**Range = 224 - 239**

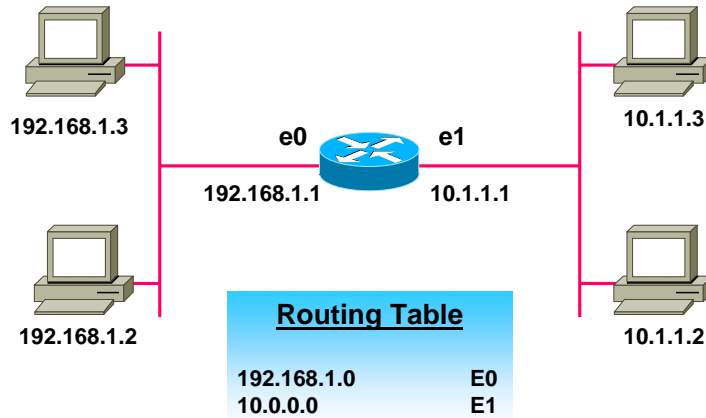
9 - 14

Copyright © Pancho Training & Consultancy Ltd. 2007

Finally, a class D address has the fourth bit set to zero leaving us with a range of addresses beginning 224 -239

Any IP address that starts with a high order octet value of 224 -239 is a class D address.

# IP Address Format



9 - 15

Copyright © Pancho Training & Consultancy Ltd. 2007

Each device or interface must have a nonzero host number.

The routing table contains entries for network or wire addresses; it usually does not contain any information about hosts.

An IP address and subnet address on an interface achieves three purposes:

- It enables the system to process the receipt and transmission of packets.
- It specifies the device's local address.
- It specifies a range of addresses that share the cable with the device.

# RFC 1918 Private Addresses

**Class A – 10.0.0.0 – 10.255.255.255**

**10.0.0.0/8**

**Class B – 172.16.0.0 – 172.31.255.255**

**172.16.0.0/12**

**Class C – 192.168.0.0 – 192.169.255.255**

**192.168.0.0/16**

9 - 16

Copyright © Pancho Training & Consultancy Ltd. 2007

Many applications require connectivity only within one enterprise and do not need external (outside the enterprise) connectivity for the majority of internal hosts. In larger enterprises it is often easy to identify a substantial number of hosts using TCP/IP that do not need network layer connectivity outside the enterprise.

The Internet Assigned Numbers Authority (IANA) has reserved the following three blocks of the IP address space for private internets:

10.0.0.0 - 10.255.255.255 (10/8 prefix)

172.16.0.0 - 172.31.255.255 (172.16/12 prefix)

192.168.0.0 - 192.168.255.255 (192.168/16 prefix)

An enterprise that decides to use IP addresses out of the address space defined in this document can do so without any coordination with IANA or an Internet registry. The address space can thus be used by many enterprises. Addresses within this private address space will only be unique within the enterprise, or the set of enterprises which choose to cooperate over this space so they may communicate with each other in their own private internet.

# Host Addressing

	24 Bits Network								8 Bits Host							
<b>Class C</b>	11000000		10101000				00000001									
	192		168				1									
									128 64 32 16 8 4 2 1							
<b>Network</b>	<del>0</del>	<del>0</del>	<del>0</del>	<del>0</del>	<del>0</del>	<del>0</del>	<del>0</del>	<del>0</del>	•	0						
	0	0	0	0	0	0	0	1	•	1						
	0	0	0	0	0	0	1	0	•	2						
	.....								•	..						
	0	1	1	1	1	1	1	1	•	254						
<b>Broadcast</b>	<del>1</del>	<del>1</del>	<del>1</del>	<del>1</del>	<del>1</del>	<del>1</del>	<del>1</del>	<del>1</del>	•	255						

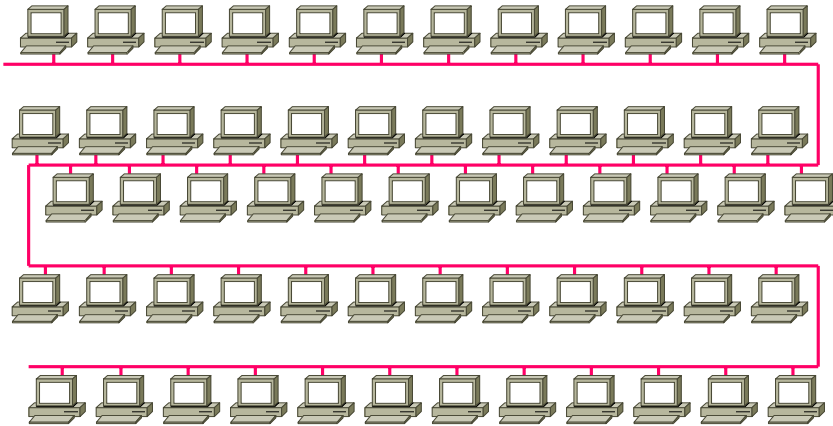
9 - 17

Copyright © Pancho Training & Consultancy Ltd. 2007

A host address of all ones is reserved for an IP broadcast into that network.

A value of zero means "this network" or "the wire itself (for example, 192.168.1.0). It was also used for IP broadcasts in some early TCP/IP implementations, although it is rarely found now.

## IP Addressing without Subnets



**172.16.0.0 ?**

9 - 18

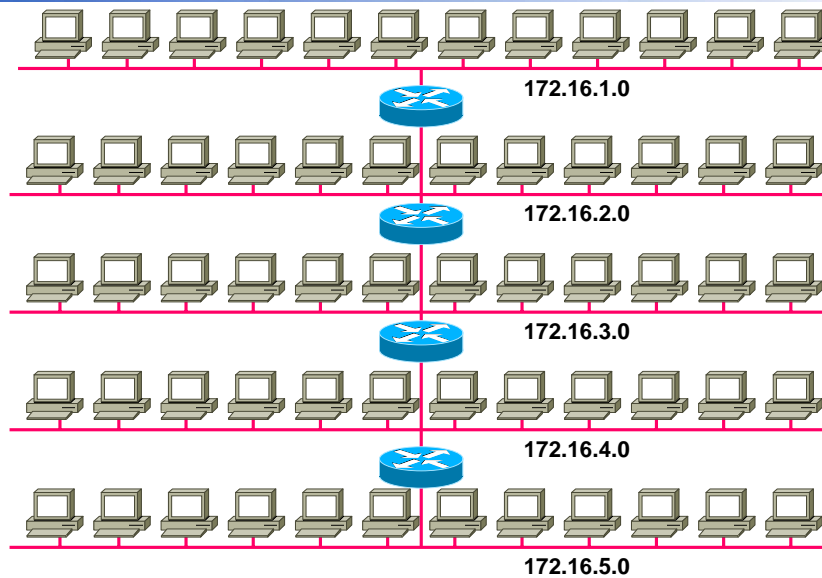
Copyright © Pancho Training & Consultancy Ltd. 2007

For an address without subnets, the outside world sees the organization as a single network, and no detailed knowledge of our internal structure is required. All datagrams addressed to 172.16 are treated the same way, regardless of the third and fourth octet of the address. A benefit from this can be the relatively short routing tables that routers can use.

Network addressing with the scheme we have set up so far has no way of distinguishing individual segments (wires) within the network. Inside the cloud having no subnets we have a single large broadcast domain—all systems on the network encounter all the broadcasts on the network. This can result in relatively poor network performance.

By default, this Class B address space defines one wire with 65,000 workstations on it. What is needed is a way to divide this wire into segments.

# IP Addressing with Subnets

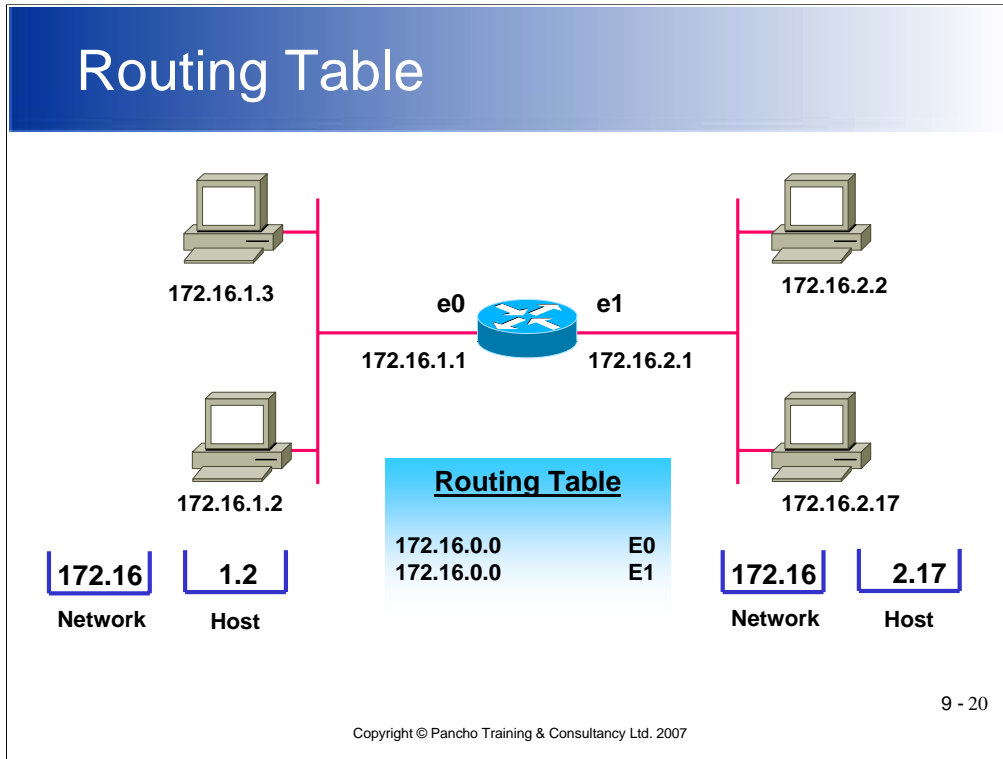


9 - 19

Copyright © Pancho Training & Consultancy Ltd. 2007

With subnets, the network address use is more efficient. There is no change to how the outside world sees the network, but within the organization, there is additional structure. In the example, the network 172.16.0.0 is subdivided or broken up into five subnets, 172.16.1.0, 172.16.2.0, 172.16.3.0, 172.16.4.0, and 172.16.5.0. Routers determine the destination network using the subnet address, limiting the amount of traffic on the other network segments.

# Routing Table

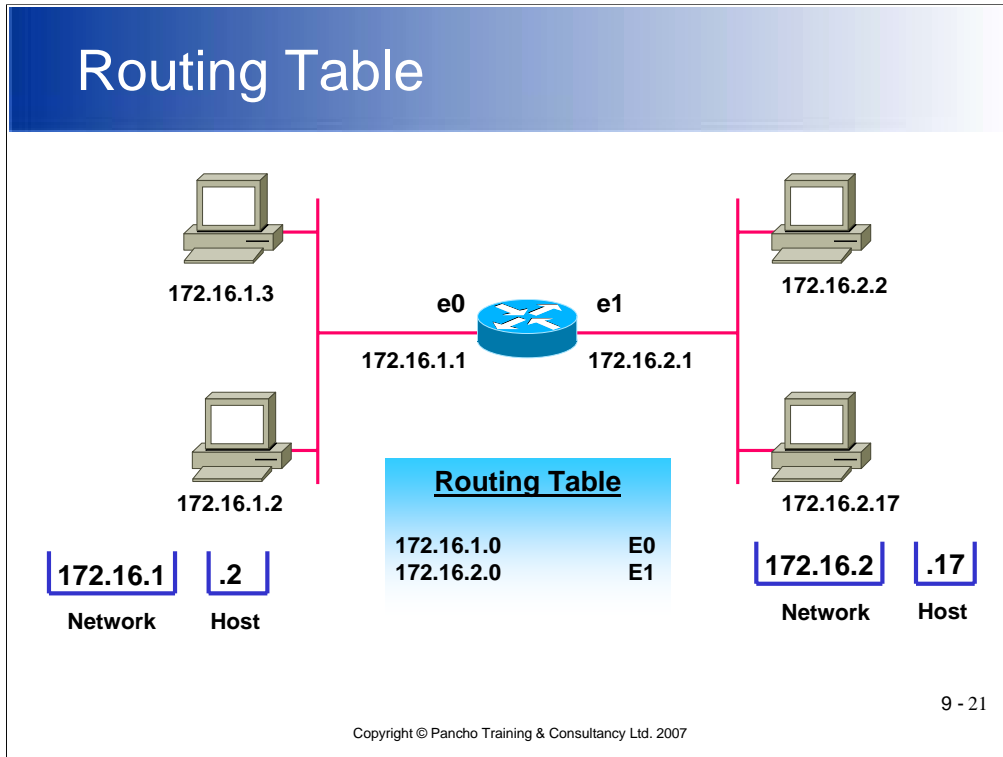


However, if we organise the network in that fashion, the routers will still use the high order bits to determine where the network/host line is drawn and, because the address is a class B, will divide the address into 16 bits network / 16 bits host.

If we looked in the routing table, we would see both segments represented as 172.16.0.0

We clearly need some other way of telling the router where to draw the network /host line apart from high order bits.

# Routing Table



We would prefer the routing table looked like this, distinguishing between the two subnets in the routing table.

This will mean using some method to tell the router that it should consider the first 24 bits as network and the last 8 bits as host, even though this is a class B address.

The tool used for this purpose is the subnet mask which simply identify to the device how much of the address should be considered network and how much host space.

# Subnet Masking

**1 = Network      0 = Host**

<b>172</b>	<b>16</b>	<b>2</b>	<b>17</b>
<b>255</b>	<b>255</b>	<b>0</b>	<b>0</b>
<b>11111111</b>	<b>11111111</b>	<b>00000000</b>	<b>00000000</b>

**Network** **Host**

9 - 22

Copyright © Pancho Training & Consultancy Ltd. 2007

An IP address is 32 bits in size, written as four octets. The subnet mask is 32 bits in size, written as four octets. The layout of the subnet mask field is as follows:

Binary 1 for the network bits

Binary 1 for the subnet bits

Binary 0 for the host bits

Subnet masks indicate which of the bits in the host field are used to specify different parts (subnets) of a particular network.

# Subnet Masking

**1 = Network      0 = Host**

<b>172</b>	<b>16</b>	<b>2</b>	<b>17</b>
<b>255</b>	<b>255</b>	<b>255</b>	<b>0</b>
<b>11111111</b>	<b>11111111</b>	<b>11111111</b>	<b>00000000</b>
<b>Network</b>			<b>Host</b>

9 - 23

Copyright © Pancho Training & Consultancy Ltd. 2007

Here we see that by extending the mask to the right we can tell the router that this address should be read as having 24 bits of network and only 8 bits of host space.

# Subnet Masking

	Network		Host	
<b>172.16.2.17</b>	10101100	00010000	00000010	00010001
<b>Logical AND</b>				
<b>255.255.0.0</b>	11111111	11111111	00000000	00000000
<b>Subnet</b>				
<b>172.16.0.0</b>	10101100	00010000	00000000	00000000

9 - 24

Copyright © Pancho Training & Consultancy Ltd. 2007

The router extracts the IP destination address from the packet and retrieves the internal subnet mask.

The router performs a logical AND operation to obtain the network number. During the logical AND operation, the host portion of the destination address is removed.

Routing decisions are then based on network number only.

In this example, with no subnetting, the network number "extracted" is 172.16.0.0.

# Subnet Masking

	Network			Host
<b>172.16.2.17</b>	10101100	00010000	00000010	00010001
<b>Logical AND</b>				
<b>255.255.255.0</b>	11111111	11111111	11111111	00000000
<b>Subnet</b>				
<b>172.16.2.0</b>	10101100	00010000	00000010	00000000

9 - 25

Copyright © Pancho Training & Consultancy Ltd. 2007

With eight bits of subnetting, the extracted network (subnet) number is 172.16.2.0.

This sample shows more bits turned on, extending the network portion and creating a secondary field extending from the end of the standard mask and using eight of the host bits. This secondary field is the subnet field and is used to represent wires (or subnetworks) inside the network.

# Subnet Masking

	Network			Host
172.16.2.17	10101100	00010000	00000010	00010001
<b>Logical AND</b>				
255.255.255.240	11111111	11111111	11111111	11110000
<b>Subnet</b>				
172.16.2.16	10101100	00010000	00000010	00010000

9 - 26

Copyright © Pancho Training & Consultancy Ltd. 2007

There is no logical reason why we cannot extend the subnet mask into the last octet as in this example.

We now have an address that consist of 28 bits of network and only 4 bits of host space creating a subnet that will support a maximum of 14 hosts.

## Subnet Mask Exercise

Address	Subnet Mask	Class	Subnet
172.16.5.33	255.255.255.0		
10.9.15.3	255.255.0.0		
198.17.23.44	255.255.255.0		
201.200.100.193	255.255.255.128		

9 - 27

Copyright © Pancho Training & Consultancy Ltd. 2007

# Broadcast Address

	Network		Host	
<b>172.16.2.17</b>	10101100	00010000	00000010	00010001
<b>Logical AND</b>				
<b>255.255.0.0</b>	11111111	11111111	00000000	00000000
<b>Subnet</b>				
<b>172.16.0.0</b>	10101100	00010000	00000000	00000000
<b>Broadcast</b>				
<b>172.16.255.255</b>			11111111	11111111
<b>Host Range</b>				
<b>172.16.0.1 – 172.16.255.254</b>				

9 - 28

Copyright © Pancho Training & Consultancy Ltd. 2007

We know we can discover the subnet address of any IP address given its mask by simply performing a logical AND between the address and mask.

It logically follows that if we can find the subnet, it is easy to determine the broadcast address by filling the host portion with zeros and converting back to decimal.

If we know the subnet and broadcast address then everything in between the two must be available host addresses.

# Broadcast Address

	Network			Host
<b>172.16.2.17</b>	10101100	00010000	00000010	00010001
<b>Logical AND</b>				
<b>255.255.255.0</b>	11111111	11111111	11111111	00000000
<b>Subnet</b>				
<b>172.16.2.0</b>	10101100	00010000	00000010	00000000
<b>Broadcast</b>				
<b>172.16.2.255</b>				11111111
<b>Host Range</b>				
<b>172.16.2.1 – 172.16.2.254</b>				

9 - 29

Copyright © Pancho Training & Consultancy Ltd. 2007

Here is another example.

# Broadcast Address

	Network			Host
<b>172.16.2.17</b>	10101100	00010000	00000010	00010001
<b>Logical AND</b>				
<b>255.255.255.240</b>	11111111	11111111	11111111	11110000
<b>Subnet</b>				
<b>172.16.2.16</b>	10101100	00010000	00000010	00010000
<b>Broadcast</b>				
<b>172.16.2.31</b>				1111
<b>Host Range</b>				
<b>172.16.2.17 – 172.16.2.30</b>				

9 - 30

Copyright © Pancho Training & Consultancy Ltd. 2007

Here is another example.

## CCNA Type Question

**Given the IP address 172.16.12.54, with a mask of 255.255.255.240, which of the following are valid host addresses on the same subnet?**

- A. 172.16.12.64**
- B. 172.16.12.57**
- C. 172.16.12.49**
- D. 172.16.12.48**
- E. 172.16.12.63**
- F. 172.16.12.45**

9 - 31

Copyright © Pancho Training & Consultancy Ltd. 2007

# The Trick

$$\begin{array}{r}
 172.16.12.54 \quad 255.255.255.240 \quad - \\
 \hline
 \phantom{172.16.12.54} \phantom{255.255.255.240} \quad 16
 \end{array}$$

- |                 |             |     |     |
|-----------------|-------------|-----|-----|
| A. 172.16.12.64 | = Network   | 0   | 192 |
| B. 172.16.12.57 | = Host      | 16  | 208 |
| C. 172.16.12.49 | = Host      | 32  | 224 |
| D. 172.16.12.48 | = Network   | 48  | 240 |
| E. 172.16.12.63 | = Broadcast | 64  |     |
| F. 172.16.12.45 | = Host      | 80  |     |
|                 |             | 96  |     |
|                 |             | 112 |     |
|                 |             | 128 |     |
|                 |             | 144 |     |
|                 |             | 160 |     |
|                 |             | 176 |     |

9 - 32

Copyright © Pancho Training & Consultancy Ltd. 2007

## CCNA Type Question 2

**How many hosts will each segment support if the Class B address 132.2.0.0 is subnetted with a 29 bit mask?**

- A. 10**
- B. 14**
- C. 15**
- D. 6**
- E. 8**

9 - 33

Copyright © Pancho Training & Consultancy Ltd. 2007

## Creating a lookup Table

	128	64	32	16	8	4	2	1	
128 =	1	0	0	0	0	0	0	0	/1, /9, /17, /25
192 =	1	1	0	0	0	0	0	0	/2, /10, /18, /26
224 =	1	1	1	0	0	0	0	0	/3, /11, /19, /27
240 =	1	1	1	1	0	0	0	0	/4, /12, /20, /28
248 =	1	1	1	1	1	0	0	0	/5, /13, /21, /29
252 =	1	1	1	1	1	1	0	0	/6, /14, /22, /30
254 =	1	1	1	1	1	1	1	0	/7, /15, /23, /31
255 =	1	1	1	1	1	1	1	1	/8, /16, /24, /32

9 - 34

Copyright © Pancho Training & Consultancy Ltd. 2007

## Creating a lookup Table

	128	64	32	16	8	4	2	1	
255 =	1	1	1	1	1	1	1	1	<b>No Host Bits</b>
254 =	1	1	1	1	1	1	1	0	<b>2 - 2 = No Hosts</b>
252 =	1	1	1	1	1	1	0	0	<b>4 - 2 = 2 Hosts</b>
248 =	1	1	1	1	1	0	0	0	<b>8 - 2 = 6 Hosts</b>
240 =	1	1	1	1	0	0	0	0	<b>16 - 2 = 14 Hosts</b>
224 =	1	1	1	0	0	0	0	0	<b>32 - 2 = 30 Hosts</b>
192 =	1	1	0	0	0	0	0	0	<b>64 - 2 = 62 Hosts</b>
128 =	1	0	0	0	0	0	0	0	<b>128 - 2 = 126 Hosts</b>

9 - 35

Copyright © Pancho Training & Consultancy Ltd. 2007

## Creating a lookup Table

**128 192 224 240 248 252 254 255**

**2 - 2 = 0**

**4 - 2 = 2**

**8 - 2 = 6**

**16 - 2 = 14**

**32 - 2 = 30**

**64 - 2 = 62**

**128 - 2 = 126**

**256 - 2 = 254**

.....

Copyright © Pancho Training & Consultancy Ltd. 2007

9 - 36

## Using a lookup Table

**128 192 224 240 248 252 254 255**

**2 - 2 = 0**

**4 - 2 = 2**

**8 - 2 = 6**

**16 - 2 = 14**

**32 - 2 = 30**

**64 - 2 = 62**

**128 - 2 = 126**

**256 - 2 = 254**

.....

**How many hosts will each segment support if the Class B address 132.2.0.0 is subnetted with a 29 bit mask?**

**24 bits = 255.255.255.0**

**Extra 5 bits – Count from left to right**

**29 bits = 255.255.255.248**

9 - 37

Copyright © Pancho Training & Consultancy Ltd. 2007

## Using a lookup Table

128 192 224 240 248 252 254 255

**2 - 2 = 0**

**4 - 2 = 2**

**8 - 2 = 6**

**16 - 2 = 14**

**32 - 2 = 30**

**64 - 2 = 62**

**128 - 2 = 126**

**256 - 2 = 254**

.....

**How many hosts will each  
segment support if the Class  
B address 132.2.0.0 is  
subnetted with a 29 bit mask?**

**Leaves 3 host bits**

**3 Host bits – Count down column**

**Supports 6 Hosts**

9 - 38

Copyright © Pancho Training & Consultancy Ltd. 2007

## CCNA Type Question 2

**How many hosts will each segment support if the Class B address 132.2.0.0 is subnetted with a 29 bit mask?**

- A. 10
- B. 14
- C. 15
- D. 6
- E. 8

**Answer is D - 6 Hosts**

9 - 39

Copyright © Pancho Training & Consultancy Ltd. 2007

## CCNA Type Question 3

**Network 172.16.0.0 has a 20 bit mask.  
Which of the following are broadcast addresses?**

- A. 172.16.82.255**
- B. 172.16.95.255**
- C. 172.16.64.255**
- D. 172.16.32.255**
- E. 172.16.47.255**
- F. 172.16.79.255**

9 - 40

Copyright © Pancho Training & Consultancy Ltd. 2007

## Find the Decimal Mask

**128 192 224 240 248 252 254 255**

**2 - 2 = 0**

**4 - 2 = 2**

**8 - 2 = 6**

**16 - 2 = 14**

**32 - 2 = 30**

**64 - 2 = 62**

**128 - 2 = 126**

**256 - 2 = 254**

.....

**Network 172.16.0.0 has a 20 bit mask. Which of the following are broadcast addresses?**

**16 bits = 255.255.0.0**

**Extra 4 bits – Count from left to right**

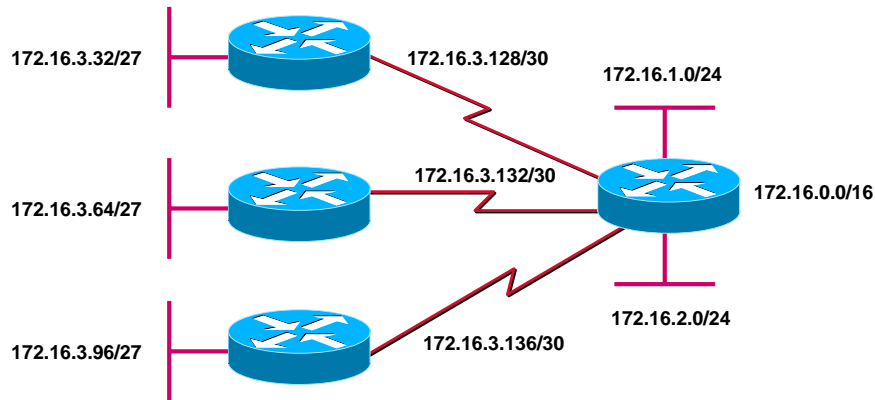
**20 bits = 255.255.240.0**

9 - 41

Copyright © Pancho Training & Consultancy Ltd. 2007



# VLSM



**Subnet 172.16.3.0/24 is divided into smaller subnets:**

**Subnet with one mask (/27)**

**Then further subnet one of the unused /27 subnets into multiple /30 subnets**

9 - 43

Copyright © Pancho Training & Consultancy Ltd. 2007

As IP subnets have grown, administrators have looked for ways to use their address space more efficiently.

One of the techniques that has resulted is called Variable Length Subnet Masks (VLSM). With VLSM, a network administrator can use a long mask on networks with few hosts and a short mask on subnets with many hosts. However, this technique is more complex than making them all one size, and addresses must be assigned carefully.

In order to use VLSM, a network administrator must use a routing protocol that supports it. Cisco routers support VLSM with Open Shortest Path First (OSPF), Integrated Intermediate System to Intermediate System (Integrated IS-IS), Enhanced Interior Gateway Routing Protocol (Enhanced IGRP), RIPv2 and static routing all support VLSM.

# Calculating VLSM

<b>Subnetted Address:</b>	172.16.3.128/27
<b>Binary Equivalent:</b>	10101110. 00010000. 00000011.10000000
<b>New VLSM Mask:</b>	172.16.3.128/30
<b>Binary Equivalent:</b>	10101110. 00010000. 00000011.10000000
<b>1<sup>st</sup> Subnet:</b>	10101110. 00010000. 00000011.10000000
<b>Decimal:</b>	172.16.3.128/30
<b>2<sup>nd</sup> Subnet</b>	10101110. 00010000. 00000011.10000100
<b>Decimal:</b>	172.16.3.132/30
<b>3<sup>rd</sup> Subnet</b>	10101110. 00010000. 00000011.10001100
<b>Decimal:</b>	172.16.3.136/30

9 - 44

Copyright © Pancho Training & Consultancy Ltd. 2007

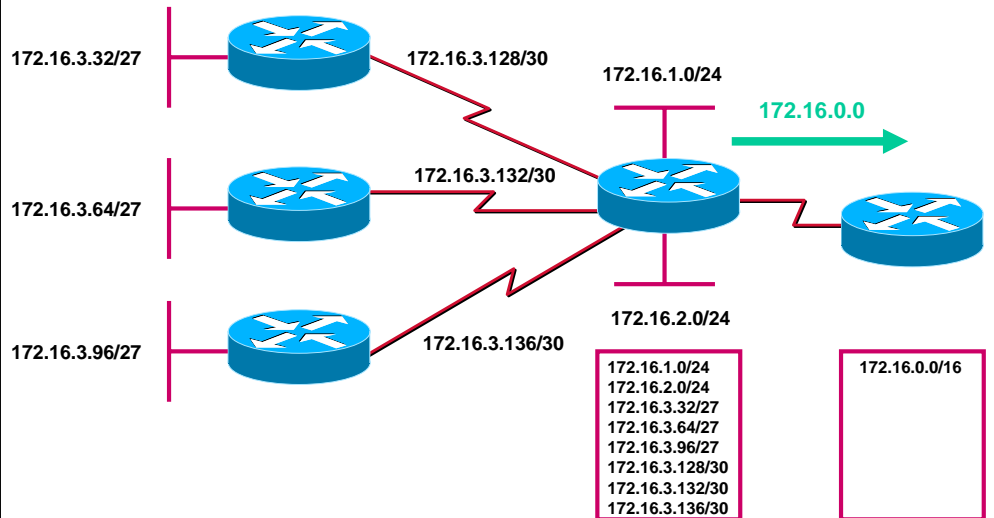
Many people make the mistake of making Variable Length Subnet Masking (VLSM) more complicated than it actually is. The key to VLSM is taking a network address that has been subnetted, and then further subnetting those subnets.

In the example, one of the available 27 bit subnets of 172.16.0.0, namely 172.16.128.0/27 has been divided into a series of 30 bit subnets by moving the network/host line to the right.

As you can see, this gives us additional networking space shown in the figure in blue. We have gained another 3 bits of subnet space. The first subnet is that space set to all zeros, the second with the last bit to one, and so on.

When we convert these values back to decimal we end up with the subnets 172.16.3.128, first host 172.16.3.129, second 172.16.3.130, broadcast address 172.16.3.131, next subnet 172.16.3.132, etc.

# Route Aggregation/Summarization



9 - 45

Copyright © Pancho Training & Consultancy Ltd. 2007

Route Aggregation is the ability to summarize routes. Route Aggregation allows routing information to be shared more efficiently in properly designed networks. When possible, ranges of addresses within an area can be aggregated (or summarized) into a single, much smaller route entry. To derive the maximum benefit from Route Aggregation, it is important that IP network addresses be assigned in a structured, hierarchical manner.

The benefits of summarization are derived through efficient resource utilization. Since multiple addresses can be summarized as a single routing entry, the number of Link State Packets can be reduced, thus minimizing the amount of bandwidth required for routing updates.

Also, changes to explicit routes that have been subsumed by an aggregated route do not usually generate LSP traffic, thus further reducing update traffic in the network. And, a single summarized route entry can replace multiple explicit network addresses in the link state database. This can substantially reduce the size of the database, leading to reduced memory requirements.

# Calculating Summarization

172.168.1.0/16	10101110.00010000.00000001.00000000
172.168.2.0/16	10101110.00010000.00000010.00000000
172.168.3.32/27	10101110.00010000.00000100.00000000
172.168.3.64/27	10101110.00010000.00000110.00000000
172.168.3.96/27	10101110.00010000.00001000.00000000
172.16.3.128/30	10101110.00010000.00001100.00000000
172.16.3.132/30	10101110.00010000.00001101.00000000
172.16.3.136/30	10101110.00010000.00001110.00000000

**1<sup>st</sup> Summarization point is at the commonality of bits = 172.16.0.0/22**

9 - 46

Copyright © Pancho Training & Consultancy Ltd. 2007

This time we need to move the mask from left to right until we find a commonality of bits i.e. a point where all the bits in the addresses at a certain position are the same.

This is the first point that we can summarize that block of addresses, in the case above, we can summarize the block with a network address of 172.16.0.0 with a 22 bit mask.

## Implementation Considerations

**Multiple IP addresses must have the same highest-order bits.**

**Routing decisions are made based on the entire address.**

**Routing protocols must carry the prefix (subnet mask) length.**

9 - 47

Copyright © Pancho Training & Consultancy Ltd. 2007

Some considerations must be taken into account when attempting to aggregate or summarize a group of routes.

Summarization can only occur where there is a commonality of high order bits.

Routing decisions are always made according to the longest match in the routing table.

Routing protocols must be capable of carrying the subnet mask in their updates for VLSM and consequently, route summaries.

# Name lookup

```
Router(config)# ip host Tokyo 10.1.1.1 192.168.1.1
```

- Creates a local name lookup table on the router

```
Router(config)# ip name-server 10.99.1.15
```

- Specifies one or more hosts that supply DNS lookup

9 - 48

Copyright © Pancho Training & Consultancy Ltd. 2007

The **ip host** command makes a static name-to-address entry in the router's configuration file.

## **ip host Command**

## **Description**

*name*

Any name you prefer to describe the destination.

*tcp-port-number*

Optional number that identifies TCP port to use when using the host name with an EXEC connect or Telnet command. The default is port 23 for Telnet.

*address*

IP address or addresses where the device can be reached.

In the example:

**ip host Tokyo 10.1.1.1 192.168.1.1** Defines two addresses to the host **Tokyo**.

**The ip name-server** command defines which hosts can provide the name service.

A maximum of six IP addresses can be specified as name servers in a single command. To map domain names to IP addresses, you must identify the host names, then specify a name server, and enable the Domain Name System (DNS). Any time the operating system software receives a command or address it does not recognize, it refers to DNS for the IP address of that device.

# Name lookup

```
Router(config)# ip domain-lookup
```

- Enabled by default – Always attempt DNS lookup

```
Router(config)# no ip domain-lookup
```

- Never attempt DNS lookup

9 - 49

Copyright © Pancho Training & Consultancy Ltd. 2007

Each unique IP address can have a host name associated with it. The Cisco IOS software maintains a cache of host name-to-address mappings for use by EXEC commands. This cache speeds the process of converting names to addresses.

IP defines a naming scheme that allows a device to be identified by its location in IP. A name such as ftp.cisco.com identifies the domain of the File Transfer Protocol for Cisco. To keep track of domain names, IP identifies a name server that maintains the name cache.

DNS is enabled by default with a server address of 255.255.255.255, which is a local broadcast.

The **no ip domain-lookup** command turns off name-to-address translation in the router. This means the router will not forward name system broadcast packets.

# Show Hosts

```
Router#show hosts
Default domain is not set
Name/address lookup uses domain service
Name servers are 10.99.1.15

Host          Flags   Age Type  Address(es)
Tokyo        (perm, OK) 0 IP  10.1.1.1 192.168.1.1
```

9 - 50

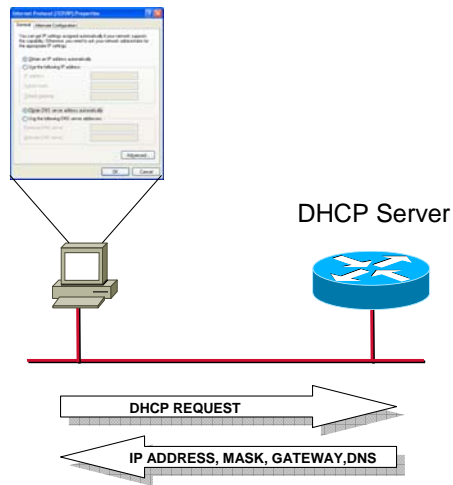
Copyright © Pancho Training & Consultancy Ltd. 2007

The **show hosts** command is used to display a cached list of host names and addresses.

## show hosts

Host	Names of learned hosts.
Flags status.	Descriptions of how information was learned and its current
perm	Manually configured in a static host table
Temp	Acquired from DNS use.
OK	Entry is current.
EX	Entry has aged-out, it has expired.
Age entry.	<i>Time</i> measured in hours since software referred to the
Type	Protocol field
Address	Logical addresses associated with the name of the host.

# What is DHCP?



9 - 51

Copyright © Pancho Training & Consultancy Ltd. 2007

The DHCP Server feature is a full DHCP Server implementation that assigns and manages IP addresses from specified address pools within the router to DHCP clients. If the DHCP Server cannot satisfy a DHCP request from its own database, it can forward the request to one or more secondary DHCP Servers defined by the network administrator.

# Configuring a DHCP Server

Enables the DHCP server

```
Router(config)# service dhcp
```

Excludes a range of addresses from the DHCP Pool

```
Router(config)# ip dhcp excluded-address low-address [high-address]
```

9 - 52

Copyright © Pancho Training & Consultancy Ltd. 2007

## **service dhcp**

Enables the DHCP server and relay features on your router.

## **ip dhcp excluded-address**

Specifies the IP addresses that the DHCP Server should not assign to DHCP clients.

# Configuring a DHCP Server

Creates a name for the DHCP pool and puts you into DHCP pool configuration mode

```
Router(config)# ip dhcp pool name
```

Specifies the network address and mask of the DHCP pool

```
Router(dhcp-config)# network network-number [mask]
```

9 - 53

Copyright © Pancho Training & Consultancy Ltd. 2007

## **ip dhcp pool**

Creates a name for the DHCP Server address pool and places you in DHCP pool configuration mode (identified by the dhcp-config# prompt).

## **network**

Specifies the subnet network number and mask of the DHCP address pool.

# Configuring a DHCP Server

Specifies the Default Gateway address

```
Router(dhcp-config)# default-router address
```

Specifies the DNS server address

```
Router(dhcp-config)# dns-server address
```

Specifies the domain name

```
Router(dhcp-config)# domain-name domain
```

9 - 54

Copyright © Pancho Training & Consultancy Ltd. 2007

## **default-router**

Specifies the IP address of the default router for a DHCP client. One IP address is required; however, you can specify up to eight addresses in one command line.

## **dns-server**

Specifies the IP address of a DNS server that is available to a DHCP client. One IP address is required; however, you can specify up to eight IP addresses in one command line.

## **domain-name**

Specifies the domain name for the client.

# Configuring a DHCP Server

Specifies the lease

```
Router(dhcp-config)# lease {days [hours][minutes] | infinite}
```

9 - 55

Copyright © Pancho Training & Consultancy Ltd. 2007

## **Lease**

Specifies the duration of the lease. The default is a one-day lease.

# DHCP Example

```
ip dhcp excluded-address 172.16.1.100 172.16.1.103
!  
ip dhcp pool FRED  
network 172.16.1.0 255.255.255.0  
domain-name yourname.com  
dns-server 172.16.2.102  
default-router 172.16.1.100  
interface ethernet 0/0  
ip address 172.16.1.100 255.255.255.0
```

9 - 56

Copyright © Pancho Training & Consultancy Ltd. 2007

# Verifying DHCP

```
Router# show ip dhcp binding
```

```
Router# show ip dhcp conflict
```

9 - 57

Copyright © Pancho Training & Consultancy Ltd. 2007

## **show ip dhcp binding**

Use the show ip dhcp binding to display the lease expiration time and date of the IP address of the host and the number. You can also use this command to display the IP addresses that have already been assigned.

## **show ip dhcp conflict**

Displays a list of all address conflicts recorded by a specific DHCP Server.

# Implementation Considerations

<b>172.16.3.33</b>	<b>/32</b>	<b>Host</b>
<b>172.16.3.32</b>	<b>/27</b>	<b>Subnet</b>
<b>172.16.3.0</b>	<b>/24</b>	<b>Network</b>
<b>172.16.0.0</b>	<b>/16</b>	<b>Block of Networks</b>
<b>0.0.0.0</b>	<b>/0</b>	<b>Default</b>

**Routers always forward traffic to the longest match in the routing table**

9 - 58

Copyright © Pancho Training & Consultancy Ltd. 2007

Routers always forward packets to the longest match in the routing table. The prefix number after the network address is the number of bits of the IP packets destination that must match to forward that packet out of that interface.

# Understanding IPv6

## Benefits of IPv6

**Larger Address Space**

**Unicast and Multicast Addressing**

**Address Aggregation**

**Auto-configuration – Plug and Play**

**Simple, efficient header**

**Security**

**Mobility**

**Transition from IPv4 to IPv6**

**IPv6 Routing Protocols**

9 - 59

Copyright © Pancho Training & Consultancy Ltd. 2007

So far we have looked at some of the methods available to alleviate the exhaustion of IPv4 addresses. We shall now look at a new addressing scheme altogether in IPv6.

IPv6 has been developed mainly to get round the address exhaustion problem but has a great many other benefits. The address space has been increased from 32 bits (IPv4) to 128 bits and, to give you some idea of what that means, this equates to 655,570,793,348,866,943,898,599 addresses for every square metre of the earth's surface.

Because we have so many addresses to play with, we can dispense with solutions like NAT, private addressing, and DHCP and every device can have its own permanent public address. With this directness come some enhancements including increased security and QoS.

There are a number of other benefits which are listed above and we will use the next few slides to discuss each one.

# IPv6 Address Space

**021:0000:210C:0000:0000:BAEE:1234:131**

← 128 bits →

**Expressed as groups of 16 bits in Hex**

**Hex numbers not case sensitive**

**Leading 0s in any 16 bit field can be dropped**

**Pair of colons ( :: ) indicates successive 16 bit fields of 0s dropped**

**One pair of colons only allowed in each address**

9 - 60

Copyright © Pancho Training & Consultancy Ltd. 2007

The IPv6 address is very different from the IPv4 address. It consists of 128 bits grouped in 16 bit “chunks” with each “chunk” represented by a hexadecimal number. The hex digits are not case sensitive to allow for easier administration.

Any leading zeros in a 16 bit field may be dropped and represented by a pair of colons, in fact, successive groups of 16 bits set to zero may be dropped and represented by a pair of colons. We can easily tell how many zeros have been dropped by adding zeros to the value until we have 128 bits again.

The use of a pair of colons is only valid once in an IPv6 address as we could not determine how many zeros should be in each location if multiple instances were allowed.

# IPv6 Address Space

021:0000:210C:0000:0000:BAEE:1234:131

=

021:0:210C::BAEE:1234:131

Copyright © Pancho Training & Consultancy Ltd. 2007

9 - 61

Although there can be only one instance of the double colon, those fields consisting of all zeros can be represented by a single zero as shown in the example above.

# IPv6 Unicast Addresses

Link Local (FE80::

Site Local (FEC0::

Global Aggregate (2000::

Unspecified or Loopback (::1)

IPv4-Compatible Address

9 - 62

Copyright © Pancho Training & Consultancy Ltd. 2007

IPv6 unicast addresses are divided up according to their function. The scope (or the allowed range) of the unicast address is clearly defined in IPv6. IPv6 unicast addresses come in the following flavours:

## Link Local

This is an address used on individual links or segments specific to that link. They are typically used by discovery protocols, routing protocols, etc. that only need to pass information across a single link. Link local addresses are auto-configured and all start with FE80::

## Site Local

These addresses are specific to a site or group of devices under the same administration.

## Aggregate Global

These are globally recognised Internet addresses

## Unspecified and Loopback

These are test addresses and place holders used when requesting a real address. The loopback address is 0000.0000.0000.0000.0000.0000.0000.0001 or ::1

# IPv6 Multicast Addresses

## RFC 2357 defined Multicast Addresses

### Node Local Scope

FF01:0:0:0:0:0:0:1 All Nodes Address  
FF01:0:0:0:0:0:0:2 All Routers Address

### Link Local Scope

FF02:0:0:0:0:0:0:1 All Nodes Address  
FF02:0:0:0:0:0:0:2 All Routers Address  
FF02:0:0:0:0:0:0:5 OSPF  
FF02:0:0:0:0:0:0:6 OSPF Designated Routers  
FF02:0:0:0:0:0:0:9 RIP Routers  
FF02:0:0:0:0:0:0:A EIGRP Routers

9 - 63

Copyright © Pancho Training & Consultancy Ltd. 2007

Multicast addresses identify a group of devices, packets are delivered to all devices that are part of the multicast group. Using multicast addressing is much more efficient than using broadcasts which are sent to all devices whether they need to process the packet or not. If a system is not part of a multicast address group, it discards the packet at layer 2 of the protocol stack.

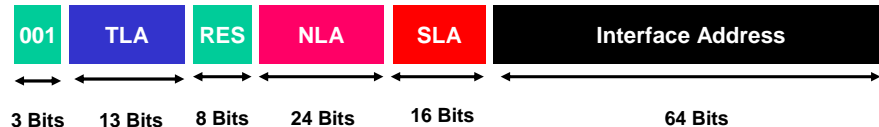
IPv6 does not use broadcast addresses at all, relying on multicast addresses to do the job that broadcasting achieved in IPv4.

All IPv6 multicast addresses start with the first eight bits set to 1. This means all IPv6 multicast addresses start with hex FF.

The multicast address range is FF00::/8, that is, 1111 1111. The second octet following the prefix defines the lifetime and scope of the multicast address. A permanent multicast address has a lifetime parameter equal to 0; a temporary multicast address has a lifetime parameter equal to 1. A multicast address that has the scope of an interface, link, subnet, admin, site, organization, or a global scope has a scope parameter of 1, 2, 3, 4, 5, 8, or E, respectively. IPv6, therefore, has millions of multicast addresses available.

A full list of defined IPv6 multicast addresses can be found in RFC 2357

# Global Aggregate Address



9 - 64

Copyright © Pancho Training & Consultancy Ltd. 2007

Wherever possible is necessary to summarize IPv6 addresses. Routing tables simply can cope with full details of every one of the addresses and so a hierarchical system must be employed to make them manageable.

As in IPv4, the left most bits of the address are used to summarize networks that appear lower in the hierarchy.

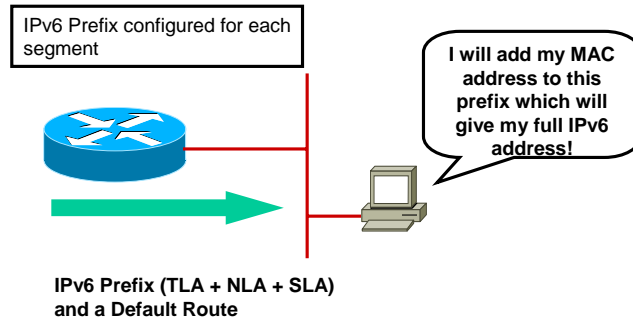
In IPv6, the address structure is very long and is split up into different parts, each part serving a specific function.

The first 48 bits of the address are used to route within the Internet. This part must be assigned by the Internet addressing authority (IANA) and creates an Aggregate Global Address. The most significant 3 bits of this part are always set to 001 to indicate a global address. If this first 48 bits is not used, the address becomes a private address similar to the private addresses used in IPv4.

The next 16 bits make up the Site Level Aggregator (SLA) which is used for routing inside an autonomous system. It can be used without the Aggregator Global Address part forming a private address.

The last 64 bits is the node or interface identifier. This is usually auto-configured using the MAC address to form this part of the address.

# Auto-Configuration



9 - 65

Copyright © Pancho Training & Consultancy Ltd. 2007

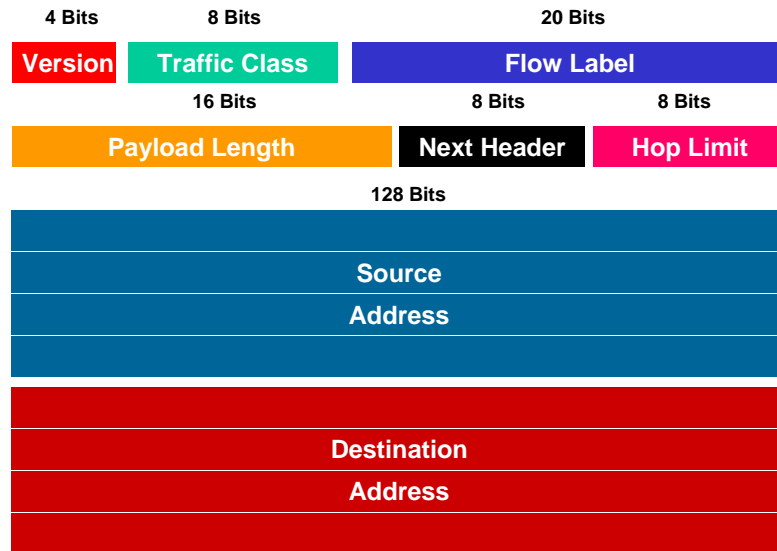
One of the main advantages to IPv6 addressing is the way that individual addresses are allocated to devices in the network. A router connected to a segment is allocated a prefix for that segment which contains the Aggregator Global Address (if used) and SLA.

The router sends the prefix plus a default route to all nodes on the wire, allowing them to add their own MAC address to the prefix and thus auto-configure their own IPv6 address.

The ability to simply plug in a device without any configuration or need for DHCP allows new devices to be added to the network extremely easily in a “plug and play” fashion.

Many of the administrative nightmares are now a thing of the past. Renumbering a network is simply a case of changing the prefixes allocated to the routers.

# Simple Header



9 - 66

Copyright © Pancho Training & Consultancy Ltd. 2007

**Version.** 4 bits.

IPv6 version number.

**Traffic Class.** 8 bits.

Internet traffic priority delivery value.

**Flow Label.** 20 bits.

Used for specifying special router handling from source to destination(s) for a sequence of packets.

**Payload Length.** 16 bits, unsigned.

Specifies the length of the data in the packet. When cleared to zero, the option is a hop-by-hop Jumbo payload.

**Next Header.** 8 bits.

Specifies the next encapsulated protocol. The values are compatible with those specified for the IPv4 protocol field.

**Hop Limit.** 8 bits, unsigned.

For each router that forwards the packet, the hop limit is decremented by 1. When the hop limit field reaches zero, the packet is discarded. This replaces the TTL field in the IPv4 header that was originally intended to be used as a time based hop limit.

**Source address.** 16 bytes.

The IPv6 address of the sending node.

**Destination address.** 16 bytes.

The IPv6 address of the destination node.

# Summary

## The objectives of this chapter were to:

- Understand the format of IP addresses and the use of dotted decimal notation
- Understand the difference between Class A,B,C, and D addresses and how computers distinguish between them
- Describe the reason for subnetting and understand the use of subnet masks
- Given an IP address and subnet mask, work out the subnet address, broadcast address, and host range applicable.
- Understand VLSM and the use of summarization
- Understand IPv6 Addressing

9 - 67

Copyright © Pancho Training & Consultancy Ltd. 2007