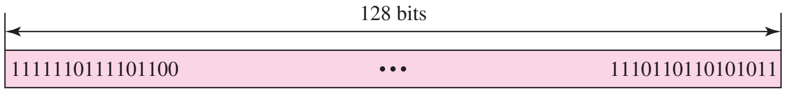
26.1 INTRODUCTION

An IPv6 address is 128 bits or 16 bytes (octet) long as shown in Figure 26.1. The address length in IPv6 is four times of the length address in IPv4.

Figure 26.1 *IPv6 address*



Notations

A computer normally stores the address in binary, but is clear that 128 bits cannot easily be handled by humans. Several notations have been proposed to represent IPv6 addresses when they are handled by humans:

Dotted-Decimal Notation

To be compatible with IPv4 addresses, we are tempted to use dotted-decimal notation as shown for IPv4 addresses in Chapter 5. Although this notation is convenient for 4-byte IPv4 addresses, it seems too long for 16-byte IPv6 addresses as shown below:

221.14.65.11.105.45.170.34.12.234.18.0.14.0.115.255

This notation is rarely used except partially as we see shortly.

Colon Hexadecimal Notation

To make addresses more readable, IPv6 specifies **colon hexadecimal notation** (or *colon hex* for short). In this notation, 128 bits are divided into eight sections, each 2 bytes in length. Two bytes in hexadecimal notation require four hexadecimal digits. Therefore, the address consists of 32 hexadecimal digits, with every four digits separated by a colon. Figure 26.2 shows an IPv6 address in colon hexadecimal notation.

Figure 26.2 *Colon hexadecimal notation*

FDEC : BA98 : 7654 : 3210 : ADBF : BBFF : 2922 : FFFF

Although the IP address, even in hexadecimal format, is very long, many of the digits are zeros. In this case, we can abbreviate the address. The leading zeros of a section can be omitted. Using this form of abbreviation, 0074 can be written as 74, 000F as F, and 0000 as 0. Note that 3210 cannot be abbreviated.

Further abbreviation, often called **zero compression,** can be applied to colon hex notation if there are consecutive sections consisting of zeros only. We can remove all the zeros altogether and replace them with a double semicolon. Figure 26.3 shows the concept.

Figure 26.3 *Zero compression*

Note that this type of abbreviation is allowed only once per address. If there are two runs of zero sections, only one of them can be compressed.

Mixed Representation

Sometimes we see a mixed representation of an IPv6 address: colon hex and dotted-decimal notation. This is appropriate during the transition period in which an IPv4 address is embedded in an IPv6 address (as the rightmost 32 bits). We can use the colon hex notation for the leftmost six section and four bytes dotted-decimal notation instead of the rightmost two sections as shown below:

FDEC:14AB:2311:BBFE:AAAA:BBBB:130.24.24.18

However, this happens when all or most of the rightmost sections of the IPv6 address are 0s. For example, the following is a legitimate address in IPv6, in which the zero compression shows that all 96 leftmost bits of the address are all zeros:

::130.24.24.18

CIDR Notation

As we see shortly, IPv6 uses hierarchical addressing. For this reason, IPv6 allows classless addressing and CIDR notation. For example, Figure 26.4 shows how we can define a prefix of 60 bits using CIDR. We will later show how an IPv6 is divided into a prefix and a suffix.

Figure 26.4 *CIDR address*

Example 26.1

Show the unabbreviated colon hex notation for the following IPv6 addresses:

a. An address with 64 0s followed by 64 1s.

b. An address with 128 0s.

c. An address with 128 1s.

d. An address with 128 alternative 1s and 0s.

Solution

a. `0000 : 0000 : 0000 : 0000 : FFFF : FFFF : FFFF : FFFF`

b. `0000 : 0000 : 0000 : 0000 : 0000 : 0000 : 0000 : 0000`

c. `FFFF : FFFF : FFFF : FFFF : FFFF : FFFF : FFFF : FFFF`

d. `AAAA : AAAA : AAAA : AAAA : AAAA : AAAA : AAAA : AAAA`

Example 26.2

The following shows the zero contraction version of addresses in Example 26.1 (part c and d cannot be abbreviated)

a. `: : FFFF : FFFF : FFFF : FFFF`

b. `: :`

c. `FFFF : FFFF : FFFF : FFFF : FFFF : FFFF : FFFF : FFFF`

d. `AAAA : AAAA : AAAA : AAAA : AAAA : AAAA : AAAA : AAAA`

Example 26.3

Show abbreviations for the following addresses:

a. `0000 : 0000 : FFFF : 0000 : 0000 : 0000 : 0000 : 0000`

b. `1234 : 2346 : 0000 : 0000 : 0000 : 0000 : 0000 : 1111`

c. `0000 : 0001 : 0000 : 0000 : 0000 : 0000 : 1200 : 1000`

d. `0000 : 0000 : 0000 : 0000 : 0000 : FFFF : 24.123.12.6`

Solution

a. `0 : 0 : FFFF : :`

b. `1234 : 2346 : : 1111`

c. `0 : 1 : : 1200 : 1000`

d. `: : FFFF : 24.123.12.6`

Example 26.4

Decompress the following addresses and show the complete unabbreviated IPv6 address:

a. `1111 : : 2222`

b. `: :`

c. `0 : 1 : :`

d. `AAAA : A : AA : : 1234`

Solution

a. `1111 : 0000 : 0000 : 0000 : 0000 : 0000 : 0000 : 2222`

b. `0000 : 0000 : 0000 : 0000 : 0000 : 0000 : 0000 : 0000`

c. `0000 : 0001 : 0000 : 0000 : 0000 : 0000 : 0000 : 0000`

d. `AAAA : 000A : 00AA : 0000 : 0000 : 0000 : 0000 : 1234`

Address Space

The address space of IPv6 contains 2^{128} addresses as shown below. This address space is 2^{96} times of the IPv4 address—definitely no address depletion.

340,282,366,920,938,463,374,607,431,768,211,456

Example 26.5

To give some idea about the number of addresses, let us assume that the number of people on the planet earth is soon to be 2^{34} (more than 16 billion). Each person can have 2^{94} addresses to use.

Example 26.6

If we assign 2^{60} addresses to the users each year (almost one billion each second), it takes 2^{68} years to deplete addresses.

Example 26.7

If we can build a high-rise building over the land and sea to accommodate 2^{68} computers in each square meter of the earth, still there are enough addresses to connect all computers to the Internet (the planet earth is approximately 2^{60} square meters).

Three Address Types

In IPv6, a destination address can belong to one of three categories: unicast, anycast, and multicast.

Unicast Address

A unicast address defines a single interface (computer or router). The packet sent to a unicast address will be routed to the intended recipient. As we see shortly, IPv6 has designated a large block from which unicast addresses can be assigned to interfaces.

Anycast Address

An anycast address defines a group of computers that all share a single address. A packet with an anycast address is delivered to only one member of the group, the most reachable one. An anycast communication is used, for example, when there are several servers that can respond to an inquiry. The request is sent to the one that is most reachable. The hardware and software generate only one copy of the request; the copy reaches only one of the servers. IPv6 does not designate a block for anycasting; the addresses are assigned from the unicast block.

Multicast Address

A multicast address also defines a group of computers. However, there is a difference between anycasting and multicasting. In multicasting, each member of the group receives a copy. As we will see shortly, IPv6 has designated a block for multicasting from which the same address is assigned to the members of the group.

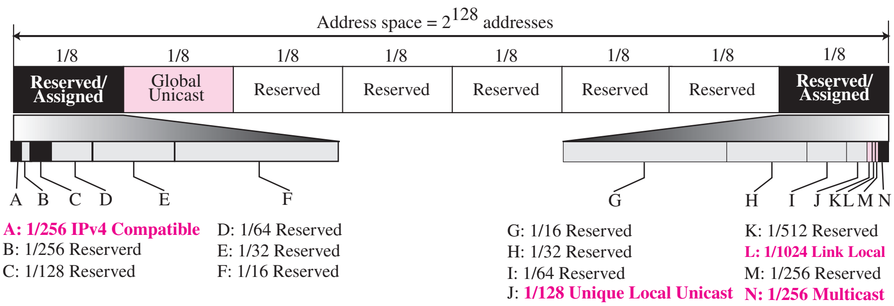
Broadcasting and Multicasting

It is interesting that IPv6 does not define broadcasting, even in a limited version, as IPv4 does. In Chapter 5, we discussed that some addresses in a block can be used for limited broadcasting. As we will see, IPv6 considers broadcasting as a special case of multicasting.

26.2 ADDRESS SPACE ALLOCATION

Like the address space of IPv4, the address space of IPv6 is divided into several blocks of varying size and each block is allocated for special purpose. Most of the blocks are still unassigned and have been left aside for future use. To better understand the allocation and the location of each block in address space, we first divide the whole address space into eight equal ranges. This division does not show the block allocation, but we believe it shows where each actual block is located (Figure 26.5).

Figure 26.5 *Address space allocation*



Each section is one-eighth of the whole address space (2^{125} addresses). The first section contains six variable-size blocks; three of these blocks are reserved and three unassigned. The second section is considered one single block and is used for global unicast addresses, which we discuss later in the chapter. The next five sections are unassigned addresses. The last section is divided into eight blocks. Some of these blocks are still unassigned and some are reserved for special purposes. The figure shows that more than five-eighths of the address space is still unassigned. Only one-eighth of the address space is used for unicast communication between the users.

Table 26.1 shows the prefix for each type of address. The third column shows the fraction of each type of address relative to the whole address space. The left-most column is not part of the standard; it shows only the section described in Figure 26.5.

Table 26.1 *Prefixes for IPv6 Addresses*

	Block Prefix	*CIDR*	*Block Assignment*	*Fraction*
1	0000 0000	0000::/8	Reserved (IPv4 compatible)	1/256
	0000 0001	0100::/8	Reserved	1/256
	0000 001	0200::/7	Reserved	1/128
	0000 01	0400::/6	Reserved	1/64
	0000 1	0800::/5	Reserved	1/32
	0001	1000::/4	Reserved	1/16
2	**001**	**2000::/3**	**Global unicast**	**1/8**
3	010	4000::/3	Reserved	1/8
4	011	6000::/3	Reserved	1/8
5	100	8000::/3	Reserved	1/8
6	101	A000::/3	Reserved	1/8
7	110	C000::/3	Reserved	1/8
8	1110	E000::/4	Reserved	1/16
	1111 0	F000::/5	Reserved	1/32
	1111 10	F800::/6	Reserved	1/64
	1111 110	FC00::/7	Unique local unicast	1/128
	1111 1110 0	FE00::/9	Reserved	1/512
	1111 1110 10	FE80::/10	Link local addresses	1/1024
	1111 1110 11	FEC0::/10	Reserved	1/1024
	1111 1111	FF00::/8	Multicast addresses	1/256

Example 26.8

Figure 26.5 shows that only a portion of the address space can be used for global unicast communication. How many addresses are in this block?

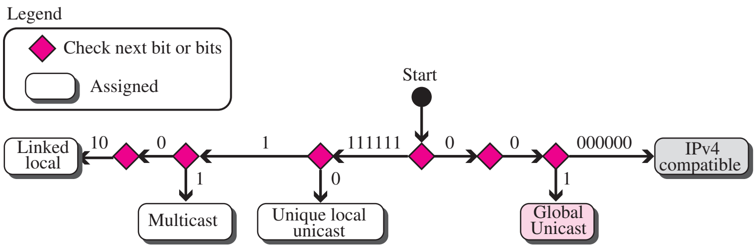
Solution

This block occupies only one-eighth of the address spaces. To find the number of addresses, we can divide the total address space by 8 or 2^3. The result is $(2^{128})/(2^3) = 2^{125}$—a huge block.

Algorithm

To show that the prefixes in Table 26.1 unambiguously find the block to which an IPv6 belongs to, we have created the diagram in Figure 26.6. The algorithm can be used to write a program to find the block when an address is given. The algorithm has to check only a maximum of 10 bits to find the block of the address. Note that the reserved

Figure 26.6 *Algorithm for finding the allocated blocks*



blocks (with the exception of IPv4-compatible addresses) are not shown to make the diagram simpler.

Assigned and Reserved Blocks

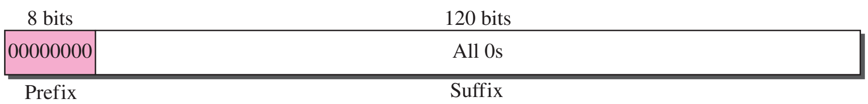
In this section, we discuss the characteristics and purposes of assigned and reserved blocks starting with the first row of Table 26.1.

IPv4 Compatible Addresses

Addresses that use the prefix (00000000) are reserved, but part of it is used to define some IPv4 compatible addresses. This block occupies 1/256 of the total address space, which means that there are 2^{120} addresses in this block. In CIDR notation, this block can be defined as 0000::**/8**. This block is further divided into several subblocks that are discussed later.

Unspecified Address The unspecified address is a subblock containing only one single address, which is defined by letting all suffix bits to 0s. In other words, the entire address consists of zeros. The unspecified address is used during bootstrap when a host does not know its own address and wants to send an inquiry to find it. Since any IPv6 packet needs a source address, the host uses this address for this purpose. Note that the unspecified address cannot be used as a destination address. The CIDR notation for this one-address subblock is ::/**128**. The unspecified address format is shown in Figure 26.7.

Figure 26.7 *Unspecified address*



The unspecified address in IPv6 is ::/128.
It should never be used as a destination address.

Example 26.9

Comparing the unspecified address in IPv4 to the unspecified addresses in IPv6.

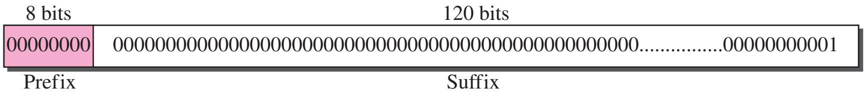
Solution

In both architectures, an unspecified address is an all-zero address. In IPv4 this address is part of class A address; in IPv6 this address is part of the reserved block.

Loopback Address This subblock also consists of one single address. We discussed loopback addresses in Chapter 5. This is an address used by a host to test itself without going into the network. In this case, a message is created in the application layer, sent to the transport layer, and passed to the network layer. However, instead of going to the physical network, it returns to the transport layer and then passes to the application layer.

This is very useful for testing the functions of software packages in these layers before even connecting the computer to the network. The loopback address as shown in Figure 26.8 consists of the prefix 00000000 followed by 119 0s and one 1. The CIDR notation for this one-address single block is ::1**/128**.

Figure 26.8 *Loopback address*



The loopback address in IPv6 is ::1/128.
It should never be used as a destination address.

Example 26.10

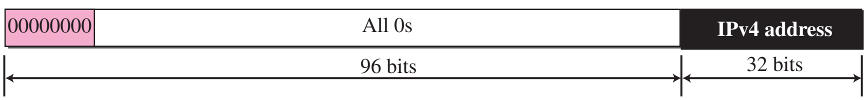
Compare the loop addresses in IPv4 to the loopback address in IPv6.

Solution

There are two differences in this case. In classful addressing, a whole block is allocated for loopback addresses; in IPv6 only one address is allocated as the loopback address. In addition, the loopback block in classful addressing is part of the class A block. In IPv6, it is only one single address in the reserved block.

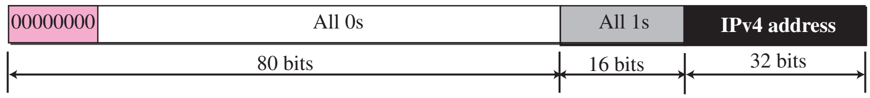
Embedded IPv4 Addresses As we will see in Chapter 27, during the transition from IPv4 to IPv6, hosts can use their IPv4 addresses embedded in IPv6 addresses. Two formats have been designed for this purpose: compatible and mapped. A **compatible address** is an address of 96 bits of zero followed by 32 bits of IPv4 address. It is used when a computer using IPv6 wants to send a message to another computer using IPv6. However, suppose the packet passes through a region where the networks are still using IPv4. The sender then must use the IPv4-compatible address to facilitate the passage of the packet through the IPv4 region. For example, the IPv4 address 2.13.17.14 (in dotted decimal format) becomes 0::2.13.17.14 (in mixed format). The IPv4 address is prepended with 96 zeros to create a 128-bit IPv6 address (see Figure 26.9). This subblock is a reservation that can contain up to 2^{32} addresses. The CIDR notation for this subblock is ::**/96**. We will discuss more about this address in Chapter 27.

Figure 26.9 *Compatible address*



A **mapped address** comprises 80 bits of zero, followed by 16 bits of one, followed by the 32-bit IPv4 address. It is used when a computer that has migrated to IPv6 wants to send a packet to a computer still using IPv4. The packet travels mostly through IPv6 networks but is finally delivered to a host that uses IPv4. For example, the IPv4 address 2.13.17.14 (in dotted decimal format) becomes 0::FFFF:2.13.17.14 (in hexadecimal colon format). The IPv4 address is prepended with 16 ones and 80 zeros to create a 128-bit IPv6 address (see Section 27.3 on Transition Strategies). Figure 26.10 shows a mapped address.

Figure 26.10 *Mapped address*



A very interesting point about mapped and compatible addresses is that they are designed such that, when calculating the checksum, one can use either the embedded address or the total address because extra 0s or 1s in multiples of 16 do not have any effect in checksum calculation. This is important for UDP and TCP, which use a pseudoheader to calculate the checksum because the checksum calculation is not affected if the address of the packet is changed from IPv6 to IPv4 by a router.

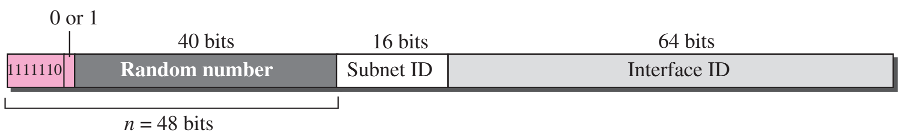
Global Unicast Block

This is the main block used for unicast communication between hosts in the Internet. We will discuss this block later in full detail to show how it will be used in the Internet to provide hierarchical addressing.

Unique Local Unicast Block

We discussed private addresses in Chapter 4 for IPv4 protocol. We discussed that some blocks in the IPv4 address space were reserved for private addressing. IPv6 uses two large blocks for private addressing: one at the site level and one at the link level. We discuss the first in this section and the second in the next. (See Figure 26.11.)

Figure 26.11 *Unique local unicast block*

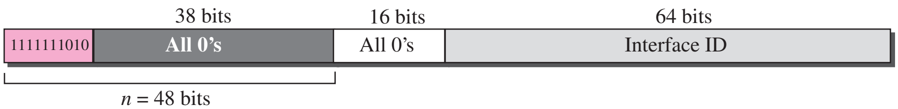


A subblock in a **unique local unicast block** can be privately created and used by a site. The packet carrying this type of address as the destination address is not expected to be routed. This type of address has the block identifier 1111 110, the next bit can be 0 or 1 to define how the address is selected (locally or by an authority). The next 40 bits are selected by the site using a randomly generated number of length 40 bits. This means that the total of 48 bits defines a subblock that looks like a global unicast address. The 40-bit random number makes the probability of duplication of the address extremely small. Note the similarity between the format of these addresses and the global unicast address we discuss later in the chapter.

Link Local Block

The second block designed for private addresses is **link local block**. A subblock in this block can be used as a private address in a network. This type of address has the block identifier 1111111010. The next 54 bits are set to zero. The last 64 bits can be changed to define the interface for each computer (see Figure 26.12). Note the similarity between the format of these addresses and the global unicast address we discuss later in the chapter.

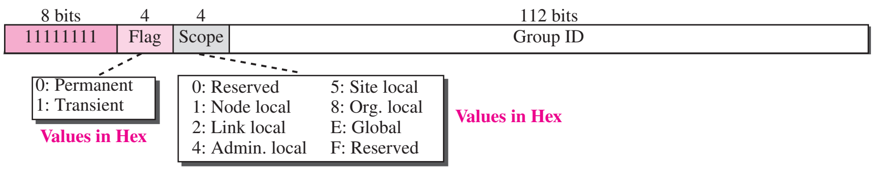
Figure 26.12 *Link local address*



Multicast Block

We discussed multicast addresses of IPv4 in Chapter 4. Multicast addresses are used to define a group of hosts instead of just one. In IPv6 a large block of addresses are assigned for multicasting. All these addresses use the prefix 11111111. The second field is a flag that defines the group address as either permanent or transient. A permanent group address is defined by the Internet authorities and can be accessed at all times. A transient group address, on the other hand, is used only temporarily. Systems engaged in a teleconference, for example, can use a transient group address. The third field defines the scope of the group address. Many different scopes have been defined, as shown in Figure 26.13.

Figure 26.13 *Multicast address*



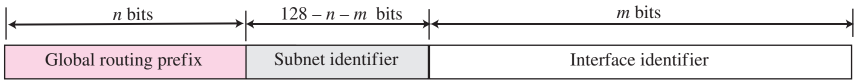
26.3 GLOBAL UNICAST ADDRESSES

This block in the address space that is used for unicast (one-to-one) communication between two hosts in the Internet is called global unicast address block. CIDR notation for the block is 2000::/3, which means that the three leftmost bits are the same for all addresses in this block (001). The size of this block is 2^{125} bits, which is more than enough for the Internet expansion in the many years to come.

Three Levels of Hierarchy

An address in this block is divided into three parts: global routing prefix, subnet identifier, and interface identifier, as shown in Figure 26.14.

Figure 26.14 *Global unicast address*



Recommended length of the different parts are shown in Table 26.2.

Table 26.2 *Recommended Length of Different Parts in Unicast Addressing*

Block Assignment	*Length*
Global routing prefix (n)	48 bits
Subnet identifier ($128 - n - m$)	16 bits
Interface identifier (m)	64 bits

Global Routing Prefix

The first 48 bits of a global unicast address are called global routing prefix. These 48 bits are used to route the packet through the Internet to the organization site such as ISP that owns the block. Since the first three bits in this part is fixed (001), the rest of the 45 bits can defined up to 2^{45} sites (a private organization or an ISP). The global routers in the Internet route a packet to its destination site based on the value of n.

Subnet Identifier

The next 16 bits defines a subnet in an organization. This means that an organization can have up to $2^{16} = 6553$ subnets, which is more than enough.

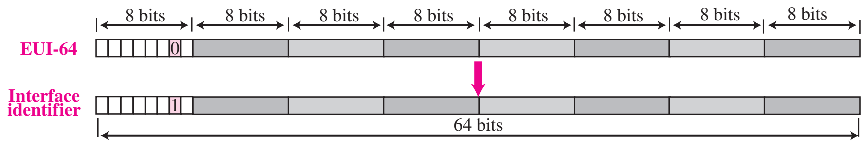
Interface Identifier

The last 64 bits define the interface identifier. The interface identifier is similar to hostid in IPv4 addressing although the term interface identifier is a better choice because, as we discussed in Chapter 5, the host identifier actually defines the interface not the host. If the host is moved from one interface to another, its IP address needs to be changed.

In IPv4 addressing, there is not a specific relation between the hostid (at the IP level) and physical or MAC address (at the data link layer) because the physical address is normally much longer than the hostid. For example, using the Ethernet technology, the physical address is 48 bits while the hostid is less than 32 bits. The IPv6 addressing allows this opportunity. A physical address whose length is less than 64 bits can be embedded as the whole or part of the interface identifier, eliminating the mapping process. Two common physical addressing scheme can be considered for this purpose: the 64-bit extended unique identifier (EUI-64) defined by IEEE and the 48-bit physical address defined by Ethernet.

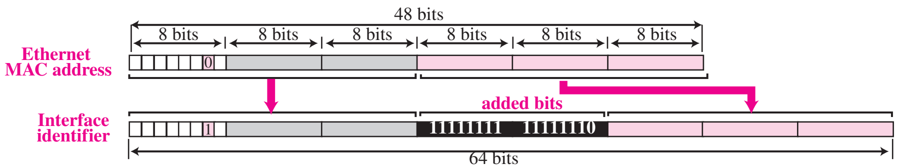
Mapping EUI-64 To map a 64-bit physical address, the global/local bit of this format needs to be changed from 0 to 1 (local to global) to define an interface address, as shown in Figure 26.15.

Figure 26.15 *Mapping for EUI-64*



Mapping Ethernet MAC Address Mapping a 48-bit Ethernet address into a 64-bit interface identifier is more involved. We need to change the local/global bit to 1 and insert an additional 16 bits. The additional 16 bits are defined as 15 ones followed by one zero, or $FFFE_{16}$. Figure 26.16 shows the mapping.

Figure 26.16 *Mapping for Ethernet MAC*



Example 26.11

Find the interface identifier if the physical address in the EUI is $(\textbf{F5-A9-23-EF-07-14-7A-D2})_{16}$ using the format we defined for Ethernet addresses.

Solution

We only need to change the seventh bit of the first octet from 0 to 1 and change the format to colon hex notation. The result is **F7A9:23EF:0714:7AD2**.

Example 26.12

Find the interface identifier if the Ethernet physical address is $(\textbf{F5-A9-23-14-7A-D2})_{16}$ using the format we defined for Ethernet addresses.

Solution

We only need to change the seventh bit of the first octet from 0 to 1, insert two octet $FFFE_{16}$ and change the format to colon hex notation. The result is **`F7A9:23FF:FE14:7AD2`** in colon hex.

Example 26.13

An organization is assigned the block 2000:1456:2474/48. What is the CIDR notation for the blocks in the first and second subnets in this organization.

Solution

Theoretically, the first and second subnets should use the block with subnet identifier 0001_{16} and 0002_{16}. This means that the blocks are 2000:1456:2474:0000/64 and 2000:1456:2474:0001/64.

Example 26.14

An organization is assigned the block 2000:1456:2474/48. What is the IPv6 address of an interface in the third subnet if the IEEE physical address of the computer is $(\textbf{F5-A9-23-14-7A-D2})_{16}$.

Solution

The interface identifier for this interface is **F7A9:23FF:FE14:7AD2** (see Example 26.12). If we this identifier to the global prefix and the subnet identifier, we get:

```
2000:1456:2474:0003:F7A9:23FF:FE14:7AD2/128
```

26.4 AUTOCONFIGURATION

One of the interesting features of IPv6 addressing is the **autoconfiguration** of hosts. As we discussed in IPv4, the host and routers are originally configured manually by the network manager. However, the Dynamic Host Configuration Protocol, DHCP, can be used to allocate an IPv4 address to a host that joins the network. In IPv6, DHCP protocol can still be used to allocate an IPv6 address to a host, but a host can also configure itself.

When a host in IPv6 joins a network, it can configure itself using the following process:

1. The host first creates a link local address for itself. This is by taking the 10-bit link local prefix (1111 1110 10), adding 54 zeros, and adding the 64-bit interface identifier, which any host knows how to generate it from its interface card. The result is a 128-bit link local address.
2. The host then tests to see if this link local address is unique and not used by other hosts. Since the 64-bit interface identifier is supposed to be unique, the link local address generated is unique with a high probability. However, to be sure, the host sends a *neighbor solicitation message* (see Chapter 28) and waits for *neighbor advertisement message*. If any host in the subnet is using this link local address, the process fails and the host cannot autoconfigure itself; it needs to use other means such as DHCP protocol for this purpose.
3. If the uniqueness of the link local address is passed, the host stores this address as its link-local address (for private communication), but it still needs a global unicast address. The host then sends a *router solicitation message* (see Chapter 28) to a local router. If there is a router running on the network, the host receives a *router advertisement message* that includes the global unicast prefix and the subnet prefix that the host needs to add to its interface identifier to generate its global unicast address. If the router cannot help the host with the configuration, it informs the host in the *router advertisement message* (by setting a flag). The host then needs to use other means for configuration.

Example 26.15

Assume a host with Ethernet address $(\mathbf{F5\text{-}A9\text{-}23\text{-}11\text{-}9B\text{-}E2})_{16}$ has joined the network. What would be its global unicast address if the global unicast prefix of the organization is 3A21:1216:2165 and the subnet identifier is A245:1232.

Solution

The host first creates its interface identifier as **F7A9:23FF:FE11:9BE2** using the Ethernet address read from its card. The host then creates its link-local address as

```
FE80::F7A9:23FF:FE11:9BE2
```

Assuming that this address is unique, the host sends a router solicitation message and receives the router advertisement message that announces the combination of global unicast prefix and the subnet identifier as 3A21:1216:2165:A245:1232. The host then appends its interface identifier to this prefix to find and store its global unicast address as:

```
3A21:1216:2165:A245:1232:F7A9:23FF:FE11:9BE2
```

26.5 RENUMBERING

To allow sites to change the service provider, **renumbering** of the address prefix (*n*) was built into IPv6 addressing. As we discussed before, each site is given a prefix by the service provider to which it is connected. If the site changes the provider, the address prefix needs to be changed. A router to which the site is connected can advertise a new prefix and let the site use the old prefix for a short time before disabling it. In other words, during the transition period, a site has two prefixes. The main problem in using the renumbering mechanism is the support of the DNS, which needs to propagate the new addressing associated with a domain name. A new protocol for DNS, called Next Generation DNS, is under study to provide support for this mechanism.

26.6 FURTHER READING

For more details about subjects discussed in this chapter, we recommend the following books and RFCs. The items enclosed in brackets refer to the reference list at the end of the book.

Books

Several books give thorough coverage of IPv6. We recommend [Com 06] and [Los 04].

RFCs

Several RFCs show updates on IPv6 addressing, including RFC 2375, RFC 2526, RFC 3513, RFC 3587, RFC 3789, and RFC 4291.