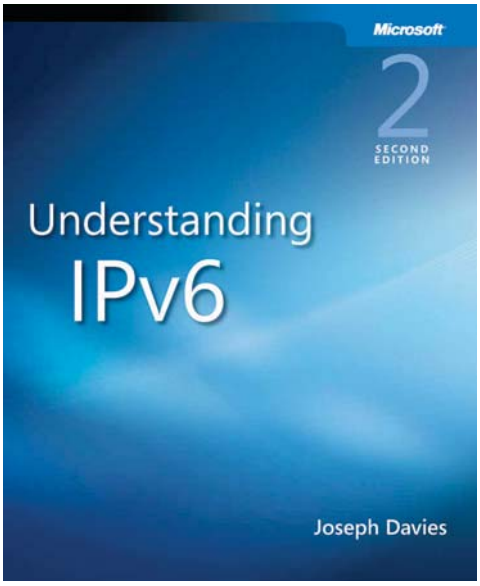

Understanding IPv6, Second Edition

Joseph Davies

PREVIEW CONTENT This excerpt contains uncorrected manuscript from an upcoming Microsoft Press title, for early preview, and is subject to change prior to release. This excerpt is from *Understanding IPv6, Second Edition* from Microsoft Press (ISBN 978-0-7356-2446-7, copyright 2008 Microsoft Corporation, all rights reserved), and is provided without any express, statutory, or implied warranties

To learn more about this book, visit Microsoft Learning at
http://www.microsoft.com/MSPress/books/11607.aspx

Microsoft Press

978-0-7356-2446-7

Table of Contents

5 ICMPv6

6 Neighbor Discovery

Appendix E: Setting Up an IPv6 Test Lab

- IPv6 Test Lab Setup
  - DNS1
  - CLIENT1
  - ROUTER1
  - ROUTER2
  - CLIENT2
- IPv6 Test Lab Tasks
  - Performing
  - Link-Local Pings
  - Enabling Native IPv6 Connectivity on Subnet 1
  - Configuring ISATAP
  - Configuring Native IPv6 Connectivity for All Subnets
  - Using Name Resolution
  - Configuring an IPv6-only Routing Infrastructure

Appendix F: Mobile IPv6

- Overview
  - Mobile IPv6 Components
  - Mobile IPv6 Transport Layer Transparency
- Mobile IPv6 Messages and Options
  - Mobility Header and Messages
  - Type 2 Routing Header
  - Home Address Option for the Destination Options Header
  - ICMPv6 Messages for Mobile IPv6
  - Modifications to Neighbor Discovery Messages and Options
- Mobile IPv6 Data Structures
  - Binding Cache
  - Binding Update List
  - Home Agents List
- Correspondent Registration
  - Return Routability Procedure
  - Detecting Correspondent Nodes that are not Mobile IPv6-Capable
- Mobile IPv6 Message Exchanges
  - Data Between a Mobile Node and a Correspondent Node
  - Binding Maintenance
  - Home Agent Discovery
  - Mobile Prefix Discovery
- Mobile IPv6 Processes
  - Attaching to the Home Link
  - Moving From the Home Link to a Foreign Link
  - Moving to a New Foreign Link
  - Returning Home

Appendix G: IPv6 Reference Tables

Chapter 3

IPv6 Addressing

At the end of this chapter, you should be able to do the following:

- Describe the IPv6 address space, and state why the address length of 128 bits was chosen.
- Describe IPv6 address syntax, including zero suppression and compression and prefixes.
- Enumerate and describe the function of the different types of unicast IPv6 addresses.
- Describe the format of multicast IPv6 addresses.
- Describe the function of anycast IPv6 addresses.
- Describe how IPv6 interface identifiers are determined.
- Describe how to perform bit-level subnetting on the subnet identifier portion of a unicast IPv6 address prefix.
- List and compare the different addressing concepts between IPv4 addresses and IPv6 addresses.

The IPv6 Address Space

The most obvious distinguishing feature of Internet Protocol version 6 (IPv6) is its use of much larger addresses. The size of an address in IPv6 is 128 bits, a bit-string that is four times longer than the 32-bit IPv4 address. A 32-bit address space allows for 2^{32}, or 4,294,967,296, possible addresses. A 128-bit address space allows for 2^{128}, or 340,282,366,920,938,463,463,374,607,431,768,211,456 (3.4×10^{38} or 340 undecillion), possible addresses.

In the late 1970s, when the IPv4 address space was designed, it was unimaginable that it could ever be exhausted. However, the administrative procedures that defined address allocation did not anticipate the recent explosion of hosts on the Internet. The IPv4 address space was thus consumed to the point that, by 1992, it was clear a replacement would be necessary.

With IPv6, it is even more difficult to conceive that the IPv6 address space will ever be consumed. To help put this number in perspective, a 128-bit address space provides 6.65×10^{23} addresses for each square meter of the Earth's surface.

It is important to remember that the decision to make the IPv6 address 128 bits in length was not so that every square meter of the Earth could have 6.65×10^{23} addresses. Rather, the relatively large size of the IPv6 address is designed to be divided into hierarchical unicast routing domains that reflect the topology of the modern-day Internet. The use of 128 bits allows for multiple levels of hierarchy and flexibility in designing hierarchical unicast addressing and routing that is currently lacking on the IPv4-based Internet.

It is easy to get lost in the vastness of the IPv6 address space. As we will discover, the unthinkably large 128-bit IPv6 address that is assigned to an interface on a typical IPv6 host is composed of a 64-bit subnet prefix and a 64-bit interface identifier (a 50-50 split between subnet space and interface space). The 64 bits of subnet prefix leave enough addressing room to satisfy the addressing requirements of three levels of Internet service providers (ISPs) between your organization and the backbone of the Internet and the addressing needs of your organization. The 64 bits of interface identifier accommodate the mapping of current and future link-layer media access control (MAC) addresses.

IPv6 Address Syntax

IPv4 addresses are represented in dotted-decimal format. The 32-bit IPv4 address is divided along 8-bit boundaries. Each set of 8 bits is converted to its decimal equivalent and separated by periods. For IPv6, the 128-bit address is divided along 16-bit boundaries, and each 16-bit block is converted to a 4-digit hexadecimal number and separated by colons. The resulting representation is called *colon hexadecimal*.

The following is an IPv6 address in binary form:

```
0010000000000001000011011011100000000000000000000010111100111011

0000001010101010000000001111111111111111000101000100111000101110 10
```

The 128-bit address is divided along 16-bit boundaries:

```
0010000000000001    0000110110111000    0000000000000000    0010111100111011

0000001010101010    0000000011111111    1111111000101000    1001110001011010
```

Each 16-bit block is converted to hexadecimal and delimited with colons. The result is the following:

```
2001:0DB8:0000:2F3B:02AA:00FF:FE28:9C5A
```

IPv6 address representation is further simplified by suppressing the leading zeros within each 16-bit block. However, each block must have at least a single digit. With leading zero suppression, the result is the following:

```
2001:DB8:0:2F3B:2AA:FF:FE28:9C5A
```

Number System Choice for IPv6

IPv6 uses hexadecimal (the $Base_{16}$ numbering system), rather than decimal (the $Base_{10}$ numbering system), because it is easier to convert between hexadecimal and binary than it is to convert between decimal and binary. Each hexadecimal digit represents four binary digits.

With IPv4, decimal is used to make the IPv4 addresses more palatable for humans and a 32-bit address becomes 4 decimal numbers separated by the period (.) character. With IPv6, dotted-decimal representation would result in 16 decimal numbers separated by the period (.) character. IPv6 addresses are so large that there is no attempt to make them palatable to most humans. Configuration of typical end systems is automated, and end users will almost always use names rather than IPv6 addresses. Therefore, the addresses are expressed in a way to make them more palatable to computers and IPv6 network administrators who understand the semantics and relationship of hexadecimal and binary numbers.

Table 3-1 lists the conversion between binary, hexadecimal, and decimal numbers.

Table 3-1 Converting Between Binary, Hexadecimal, and Decimal Numbers

Binary	Hexadecimal	Decimal
0000	0	0
0001	1	1
0010	2	2
0011	3	3
0100	4	4
0101	5	5
0110	6	6
0111	7	7
1000	8	8
1001	9	9
1010	A	10
1011	B	11
1100	C	12
1101	D	13
1110	E	14
1111	F	15

Compressing Zeros

Some types of IPv6 addresses contain long sequences of zeros. To further simplify the representation of IPv6 addresses, a single contiguous sequence of 16-bit blocks set to 0 in the colon hexadecimal format can be compressed to ::, known as a *double colon*. For example, the link-local address of FE80:0:0:0:2AA:FF:FE9A:4CA2 can be compressed to FE80::2AA:FF:FE9A:4CA2. The multicast address FF02:0:0:0:0:0:0:2 can be compressed to FF02::2.

Note You cannot use zero compression to include part of a 16-bit block. For example, you cannot express FF02:30:0:0:0:0:0:5 as FF02:3::5, but FF02:30::5 is correct.

How Many Blocks or Bits in ::?

To determine how many 0 blocks are represented by the ::, you can count the number of blocks in the compressed address and subtract this number from 8. To determine how many 0 bits are represented by the ::, multiply the number of blocks the :: represents by 16. For example, in the address FF02::2, there are two blocks (the "FF02" block and the "2" block.) The number of blocks expressed by the :: is 6 (8 <;$MI> 2). The number of bits expressed by the :: is 96 (96 = 6 × 16). Zero compression can be used only once in a given address. Otherwise, you could not determine the number of 0 blocks or bits represented by each instance of ::.

IPv6 Prefixes

The prefix is the part of the address where the bits have fixed values or are the bits that define a route or subnet. Prefixes for IPv6 subnets and summarized routes are expressed in the same way as Classless Inter-Domain Routing (CIDR) notation for IPv4. An IPv6 prefix is written in *address/prefix-length* notation.

For example, 2001:DB8:2A0:2F3B::/64 is a subnet prefix and 2001:DB8:3F::/48 is a summarized route prefix. As described earlier in this chapter, the 64-bit prefix is used for individual subnets to which nodes are attached. All subnets have a 64-bit prefix. Any prefix that is less than 64 bits is a summarized route or an address range that is summarizing a portion of the IPv6 address space.

Note IPv4 implementations commonly use a dotted-decimal representation of the prefix length known as the *subnet mask*. A subnet mask is not used for IPv6. Only the prefix length notation is supported.

An IPv6 prefix is relevant only for routes or address ranges, not for individual unicast addresses. In IPv4, it is common to express an IPv4 address with its prefix length. For example, 192.168.29.7/24 (equivalent to 192.168.29.7 with the subnet mask 255.255.255.0) denotes the IPv4 address 192.168.29.7 with a 24-bit subnet mask. Because IPv4 addresses are no longer class-based, you cannot assume the class-based subnet mask based on the value of the leading octet. The prefix length is included so that you can determine which bits identify the subnet and which bits identify the host on the subnet. Because the number of bits used to identify the subnet in IPv4 is variable, the prefix length is needed to separate the subnet prefix from the host ID.

In common IPv6 practice, however, there is no notion of a variable-length subnet prefix. At the individual IPv6 subnet level for currently defined unicast IPv6 addresses, the number of bits used to identify the subnet is always 64 and the number of bits used to identify the host on the subnet is always 64. Therefore, while unicast IPv6 addresses written with their prefix lengths are permitted in RFC 4291, in practice their prefix lengths are always 64 and therefore do not need to be expressed. For example, there is no need to express the IPv6 unicast address 2001:DB8::2AC4:2AA:FF:FE9A:82D4 as 2001:DB8::2AC4:2AA:FF:FE9A:82D4/64. Because of the 50-50 split of subnet prefixes and interface identifiers, the unicast IPv6 address 2001:DB8::2AC4:2AA:FF:FE9A:82D4 implies that the subnet prefix is 2001:DB8:0:0:2AC4::/64.

> **Note** Address prefixes with a prefix length longer than 64 bits can be used for point-to-point links between routers.

Types of IPv6 Addresses

There are three types of IPv6 addresses:

1. **Unicast**

   A unicast address identifies a single interface within the scope of the type of address. The scope of an address is the region of the IPv6 network over which the address is unique. With the appropriate unicast routing topology, packets addressed to a unicast address are delivered to a single interface. To accommodate load-balancing systems, RFC 4291 allows for multiple interfaces to use the same address as long as they appear as a single interface to the IPv6 implementation on the host.

2. **Multicast**

   A multicast address identifies zero or more interfaces on the same or different hosts. With the appropriate multicast routing topology, packets addressed to a multicast address are delivered to all interfaces identified by the address.

3. **Anycast**

   An anycast address identifies multiple interfaces. With the appropriate unicast routing topology, packets addressed to an anycast address are delivered to a single interface—the nearest interface that is identified by the address. The nearest interface is defined as being the closest in terms of routing distance. A multicast address is used for one-to-many communication, with delivery to multiple interfaces. An anycast address is used for one-to-one-of-many communication, with delivery to a single interface.

In all cases, IPv6 addresses identify interfaces, not nodes. A node is identified by any unicast address assigned to any one of its interfaces.

> **Note** RFC 4291 does not define a broadcast address. All types of IPv4 broadcast addressing are performed in IPv6 using multicast addresses. For example, the subnet and limited broadcast addresses from IPv4 are replaced with the link-local scope all-nodes multicast address of FF02::1.

Unicast IPv6 Addresses

The following types of addresses are unicast IPv6 addresses:

- Global unicast addresses
- Link-local addresses
- Site-local addresses

- Unique local addresses
- Special addresses
- Compatibility addresses

Global Unicast Addresses

IPv6 global addresses are equivalent to public IPv4 addresses. They are globally routable and reachable on the IPv6 Internet. Global unicast addresses are designed to be aggregated or summarized for an efficient routing infrastructure. Unlike the current IPv4-based Internet, which is a mixture of both flat and hierarchical routing, the IPv6-based Internet has been designed from its foundation to support efficient, hierarchical addressing and routing. The scope of a global address is the entire IPv6 Internet.

RFC 4291 defines global addresses as all addresses that are not the unspecified, loopback, link-local unicast, or multicast addresses (described later in this chapter). However, Figure 3-1 shows the structure of global unicast addresses defined in RFC 3587 that are currently being used on the IPv6 Internet.

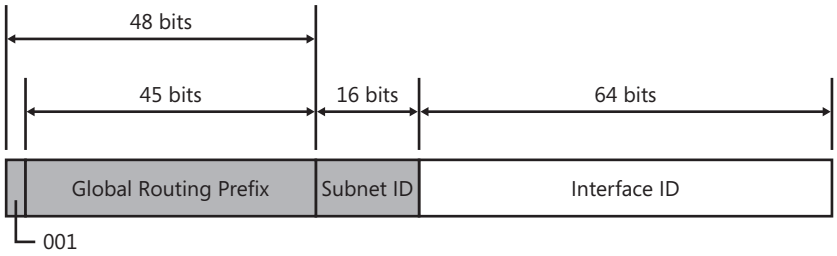

Figure 3-1 The structure of global unicast addresses defined in RFC 3587

The fields in the global unicast address are described in the following list:

Fixed portion set to 001

- The three high-order bits are set to 001.

Global Routing Prefix

- Indicates the global routing prefix for a specific organization's site. The combination of the three fixed bits and the 45-bit Global Routing Prefix is used to create a 48-bit site prefix, which is assigned to an individual site of an organization. A site is an autonomously operating IP-based network that is connected to the IPv6 Internet. Network architects and administrators within the site determine the addressing plan and routing policy for the organization network. Once assigned, routers on the IPv6 Internet forward IPv6 traffic matching the 48-bit prefix to the routers of the organization's site.

Subnet ID

- The Subnet ID is used within an organization's site to identify subnets within its site. The size of this field is 16 bits. The organization's site can use these 16 bits within its site to create 65,536 subnets or multiple levels of addressing hierarchy and an efficient routing infrastructure. With 16 bits of subnetting flexibility, a global unicast prefix assigned to an organization site is equivalent to a public IPv4 Class A address prefix (assuming that the last octet is used for identifying nodes on subnets). The routing structure of the organization's network is not visible to the ISP.

Interface ID

- Indicates the interface on a specific subnet within the site. The size of this field is 64 bits. The interface ID in IPv6 is equivalent to the node ID or host ID in IPv4.

Trillions of Sites

Another way to gauge the practical size of the IPv6 address space is to examine the number of sites that can connect to the IPv6 Internet. With the current allocation practice defined in RFC 3587 of 48-bit global address prefixes, it is possible to define 2^{45} or 35,184,372,088,832 possible 48-bit prefixes to assign to sites connected to the IPv6 Internet. There are more IPv6 sites than possible IPv4 addresses. This large number of sites is possible even when we are using only one-eighth of the entire IPv6 address space.

By comparison, using the Internet address classes originally defined for IPv4, it was possible to assign 2,113,389 address prefixes to organizations connected to the Internet. The number 2,113,389 is derived from adding up all the possible Class A, Class B, and Class C address prefixes and then subtracting the prefixes used for the private address space. Even with the adoption of CIDR to make more efficient use of unassigned Class A and Class B address prefixes, the number of possible sites connected to the Internet is not substantially increased, nor does it approach the number of possible sites that can be connected to the IPv6 Internet.

Topologies Within Global Addresses

The fields within the global address create a three-level topological structure, as shown in Figure 3-2.

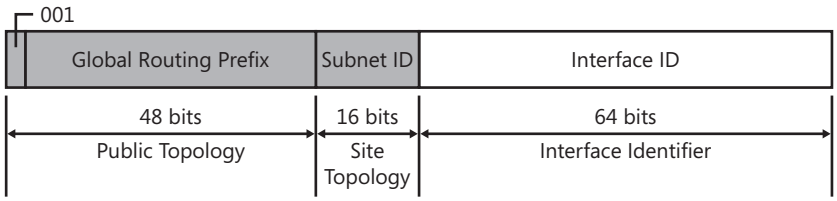

Figure 3-2 The topological structure of the global address

The public topology is the collection of larger and smaller ISPs that provide access to the IPv6 Internet. The site topology is the collection of subnets within an organization's site. The interface identifier specifies a unique interface on a subnet within an organization's site.

Local-Use Unicast Addresses

Local-use unicast addresses do not have a global scope and can be reused. There are two types of local-use unicast addresses:

1. Link-local addresses are used between on-link neighbors and for Neighbor Discovery processes.

2. Site-local addresses are used between nodes communicating with other nodes in the same organization.

Link-Local Addresses

IPv6 link-local addresses, identified by the initial 10 bits being set to 1111 1110 10 and the next 54 bits set to 0, are used by nodes when communicating with neighboring nodes on the same link. For example, on a single-link IPv6 network with no router, link-local addresses are used to communicate between hosts on the link. IPv6 link-local addresses are similar to IPv4 link-local addresses defined in RFC 3927 that use the 169.254.0.0/16 prefix. The use of IPv4 link-local addresses is known as Automatic Private IP Addressing (APIPA) in Windows Vista, Windows Server 2008, Windows Server 2003, and Windows XP. The scope of a link-local address is the local link.

Figure 3-3 shows the structure of the link-local address.

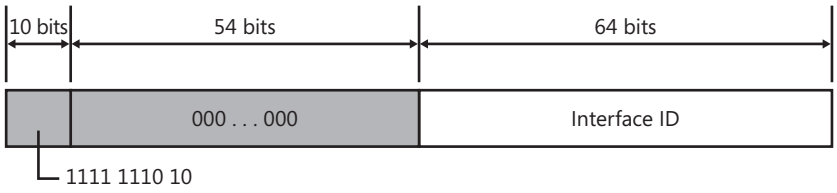

Figure 3-3 The structure of the link-local address

A link-local address is required for some Neighbor Discovery processes and is always automatically configured, even in the absence of all other unicast addresses. For more information about the address autoconfiguration process for link-local addresses, see Chapter 8, "Address Autoconfiguration."

Link-local addresses always begin with FE80. With the 64-bit interface identifier, the prefix for link-local addresses is always FE80::/64. An IPv6 router never forwards link-local traffic beyond the link.

Site-Local Addresses

Site-local addresses, identified by setting the first 10 bits to 1111 1110 11, are equivalent to the IPv4 private address space (10.0.0.0/8, 172.16.0.0/12, and 192.168.0.0/16). For example, private intranets that do not have a direct, routed connection to the IPv6 Internet can use site-local addresses without conflicting with global addresses. Site-local addresses are not reachable from other sites, and routers must not forward site-local traffic outside the site. Site-local addresses can be used in addition to global addresses. The scope of a site-local address is the site.

Figure 3-4 shows the structure of the site-local address.

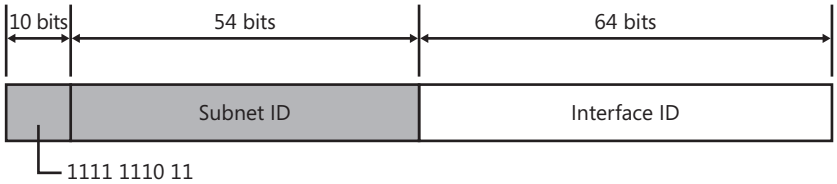

Figure 3-4 The structure of the site-local address

Unlike link-local addresses, site-local addresses are not automatically configured and must be assigned either through stateless or stateful address autoconfiguration. For more information, see Chapter 8.

The first 10 bits are always fixed for site-local addresses, beginning with FEC0::/10. After the 10 fixed bits is a 54-bit Subnet ID field that provides 54 bits with which you can create subnets within your organization. You can have a flat subnet structure, or you can divide the high-order bits of the Subnet ID field to create a hierarchical and summarizable routing infrastructure. After the Subnet ID field is a 64-bit Interface ID field that identifies a specific interface on a subnet.

Site-local addresses have been formally deprecated in RFC 3879 for future IPv6 implementations. However, existing implementations of IPv6 can continue to use site-local addresses.

Zone IDs for Local-Use Addresses

Unlike global addresses, local-use addresses (link-local and site-local addresses) can be reused. Link-local addresses are reused on each link. Site-local addresses can be reused within each site of an organization. Because of this address reuse capability, link-local and site-local addresses are ambiguous. To specify the link on which the destination is located or the site within which the destination is located, an additional identifier is needed. This additional identifier is a zone identifier (ID), also known as a scope ID, which identifies a connected portion of a network that has a specified scope.

The syntax specified in RFC 4007 for identifying the zone associated with a local-use address is *Address***%***zone_ID*, in which *Address* is a local-use unicast IPv6 address and *zone_ID* is an integer value representing the zone. The values of the zone ID are defined relative to the sending host. Therefore, different hosts might determine different zone ID values for the same physical zone. For example, Host A might choose 3 to represent the zone of an attached link and host B might choose 4 to represent the same link.

For Windows-based IPv6 hosts, the zone IDs for link-local and site-local addresses are defined as follows:

- For link-local addresses, the zone ID is typically the interface index of the interface either assigned the address or to be used as the sending interface for a link-local destination. The interface index is an integer starting at 1 that is assigned to IPv6 interfaces, which include a loopback and one or multiple LAN or tunnel interfaces. Multiple interfaces can have the same link-local zone ID if they are attached to the same link. You can view the list of interface indexes from the display of the netsh **interface ipv6 show interface** command. You must include a zone ID with a link-local destination.
- For site-local addresses, the zone ID is the site ID, an integer assigned to the site of an organization. For organizations that do not reuse the site-local address prefix, the site ID is set to 1 by default and does not need to be specified. In Windows, you can view the site ID from the display of the **netsh interface ipv6 show address level=verbose** command.

The following are examples of using Windows tools and the zone ID:

ping fe80::2b0:d0ff:fee9:4143%3

- In this case, *3* is the interface index of the interface attached to the link containing the destination address.

tracert fec0::f282:2b0:d0ff:fee9:4143%2

- In this case, *2* is the site ID of the organization site containing the destination address.

In Windows Vista and Windows Server 2008, the Ipconfig.exe tool displays the zone ID of local-use IPv6 addresses. The following is an excerpt from the display of the **ipconfig** command:

```
Ethernet adapter Local Area Connection:


   Connection-specific DNS Suffix  . : ecoast.example.com
   IPv6 Address. . . . . . . . . . . : 2001:db8:21da:7:713e:a426:d167:37ab
   Temporary IPv6 Address. . . . . . : 2001:db8:21da:7:5099:ba54:9881:2e54
   Link-local IPv6 Address . . . . . : fe80::713e:a426:d167:37ab%6
   IPv4 Address. . . . . . . . . . . : 157.60.14.11
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Default Gateway . . . . . . . . . : fe80::20a:42ff:feb0:5400%6
                                       157.60.14.1
```

For the link-local addresses in the display of the **ipconfig** command, the zone ID indicates the interface index of the interface either assigned the address (for Link-Local IPv6 Address) or the interface through which an address is reachable (for Default Gateway).

Unique Local Addresses

Site-local addresses provide a private addressing alternative to global addresses for intranet traffic. However, because the site-local address prefix can be reused to address multiple sites within an organization, a site-local address prefix can be duplicated. The ambiguity of site-local addresses in an organization adds complexity and difficulty for applications, routers, and network managers. For more information, see section 2 of RFC 3879.

To replace site-local addresses with a new type of address that is private to an organization yet unique across all the sites of the organization, RFC 4193 defines unique local IPv6 unicast addresses. Figure 3-5 shows the structure of the unique local address.

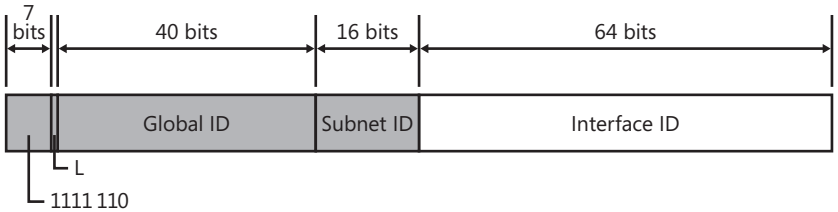

Figure 3-5 The structure of the unique local address

The first 7 bits have the fixed binary value of 1111110. All local addresses have the address prefix FC00::/7. The Local (L) flag is set 1 to indicate that the prefix is locally assigned. The L flag value set to 0 has not yet been defined. Therefore, unique local addresses within an organization with the L flag set to 1 have the address prefix of FD00::/8. The Global ID identifies a specific site within an organization and is set to a randomly derived 40-bit value. By deriving a random value for the Global ID, an organization can have statistically unique 48-bit prefixes assigned to their sites. Additionally, two organizations that use unique local addresses that merge have a low probability of duplicating a 48-bit unique local address prefix, minimizing site renumbering. Unlike the Global Routing Prefix in global addresses, the Global IDs in unique local address prefixes are not designed to be summarized.

Unique local addresses have a global scope, but their reachability is defined by routing topology and filtering policies at Internet boundaries. Organizations will not advertise their unique local address prefixes outside of their organizations or create DNS entries with unique local addresses in the Internet DNS. Organizations can easily create filtering policies at their Internet boundaries to prevent all unique local-addressed traffic from being forwarded. Because they have a global scope, unique local addresses do not need a zone ID.

The global address and unique local address share the same structure beyond the first 48 bits of the address. In both addresses, the 16-bit Subnet ID field identifies a subnet within an organization. Because of this, you can create a subnetted routing infrastructure that is used for both local and global addresses.

For example, a specific subnet of your organization can be assigned both the global prefix 2001:DB8:4D1C:221A::/64 and the local prefix FD0E:2D:BA9:221A::/64, where the subnet is identified for both types of prefixes by the Subnet ID value of *221A*. Although the subnet identifier is the same for both prefixes, routes for both prefixes must still be propagated throughout the routing infrastructure so that addresses based on both prefixes are reachable.

Special IPv6 Addresses

The following are special IPv6 addresses:

- **Unspecified address**

  The unspecified address (0:0:0:0:0:0:0:0 or ::) is used only to indicate the absence of an address. It is equivalent to the IPv4 unspecified address of 0.0.0.0. The unspecified address is typically used as a source address when a unique address has not yet been determined. The unspecified address is never assigned to an interface or used as a destination address.

- **Loopback address**

  The loopback address (0:0:0:0:0:0:0:1 or ::1) is assigned to a loopback interface, enabling a node to send packets to itself. It is equivalent to the IPv4 loopback address of 127.0.0.1. Packets addressed to the loopback address must never be sent on a link or forwarded by an IPv6 router.

Compatibility Addresses

To aid in the migration from IPv4 to IPv6 and the coexistence of both types of hosts, the following addresses are defined:

- **IPv4-compatible address**

  The IPv4-compatible address, 0:0:0:0:0:0:*w.x.y.z* or ::*w.x.y.z* (where *w.x.y.z* is the dotted-decimal representation of a public IPv4 address), is used by IPv6/IPv4 nodes that are communicating with IPv6 over an IPv4 infrastructure that uses public IPv4 addresses, such as the Internet. IPv4-compatible addresses are deprecated in RFC 4291 and are not supported in IPv6 for Windows Vista and Windows Server 2008.

- **IPv4-mapped address**

  The IPv4-mapped address, 0:0:0:0:0:FFFF:*w.x.y.z* or ::FFFF: *w.x.y.z*, is used to represent an IPv4 address as a 128-bit IPv6 address.

- **6to4 address**

  An address of the type 2002:*WWXX*:*YYZZ*:*Subnet ID*:*Interface ID*, where *WWXX*:*YYZZ* is the colon hexadecimal representation of *w.x.y.z* (a public IPv4 address), is assigned a node for the 6to4 IPv6 transition technology.

- **ISATAP address**

  An address of the type *64-bit prefix*:0:5EFE:*w.x.y.z*, where *w.x.y.z* is a public or private IPv4 address, is assigned to a node for the Intra-Site Automatic Tunnel Addressing Protocol (ISATAP) IPv6 transition technology.

- **Teredo address**

  A global address that uses the prefix 2001::/32 and is assigned to a node for the Teredo IPv6 transition technology. Beyond the first 32 bits, Teredo addresses are used to encode the IPv4 address of a Teredo server, flags, and an obscured version of a Teredo client's external address and UDP port number.

For more information about these addresses, see Chapter 11, "IPv6 Transition Technologies."

Multicast IPv6 Addresses

In IPv6, multicast traffic operates in the same way that it does in IPv4. Arbitrarily located IPv6 nodes can listen for multicast traffic on an arbitrary IPv6 multicast address. IPv6 nodes can listen to multiple multicast addresses at the same time. Nodes can join or leave a multicast group at any time.

IPv6 multicast addresses have the first 8 bits set to 1111 1111. Therefore, an IPv6 multicast address always begins with FF. Multicast addresses cannot be used as source addresses or as intermediate destinations in a Routing extension header. Beyond the first 8 bits, multicast addresses include additional structure to identify flags, their scope, and the multicast group. Figure 3-6 shows the structure of the IPv6 multicast address.

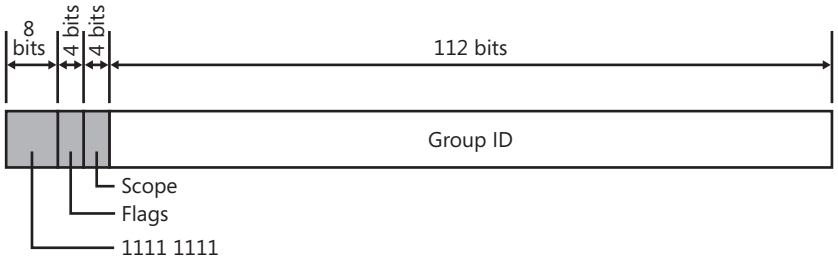

Figure 3-6 The structure of the IPv6 multicast address

The following list describes the fields in the multicast address:

Flags

- Indicates flags set on the multicast address. The size of this field is 4 bits, consisting of three flags in the low-order bits. The first low-order bit is the Transient (T) flag. When set to 0, the T flag indicates that the multicast address is a permanently assigned (well-known) multicast address allocated by the Internet Assigned Numbers Authority (IANA). When set to 1, the T flag indicates that the multicast address is a transient (non-permanently-assigned) multicast address. The second low-order bit is for the Prefix (P) flag, which indicates whether the multicast address is based on a unicast address prefix. RFC 3306 describes the P flag. The third low-order bit is for the Rendezvous Point Address (R) flag, which indicates whether the multicast address contains an embedded rendezvous point address. RFC 3956 describes the R flag.

Scope

- Indicates the scope of the IPv6 network for which the multicast traffic is intended to be delivered. The size of this field is 4 bits. In addition to using information provided by multicast routing protocols, routers use the multicast scope to determine whether multicast traffic can be forwarded.

  Table 3-2 lists the values for the Scope field assigned in RFC 4291. All other values are unassigned.

Table 3-2 Defined Values for the Scope Field

Scope Field Value	Scope
0	Reserved
1	Interface-local scope
2	Link-local scope
3	Reserved
4	Admin-local scope
5	Site-local scope
8	Organization-local scope
E	Global scope
F	Reserved

  For example, traffic with the multicast address of FF02::2 has a link-local scope. An IPv6 router never forwards this traffic beyond the local link.

Group ID

- Identifies the multicast group, and is unique within the scope. The size of this field is 112 bits. Permanently assigned group IDs are independent of the scope. Transient

group IDs are relevant only to a specific scope. Multicast addresses from FF01:: through FF0F:: are reserved, well-known addresses.

To identify all nodes for the interface-local and link-local scopes, the following addresses are defined:

- FF01::1 (interface-local scope all-nodes multicast address)
- FF02::1 (link-local scope all-nodes multicast address)

To identify all routers for the interface-local, link-local, and site-local scopes, the following addresses are defined:

- FF01::2 (interface-local scope all-routers multicast address)
- FF02::2 (link-local scope all-routers multicast address)
- FF05::2 (site-local scope all-routers multicast address)

For the current list of permanently assigned IPv6 multicast addresses, see *http://www.iana.org/assignments/ipv6-multicast-addresses*.

IPv6 multicast addresses replace all forms of IPv4 broadcast addresses. The IPv4 network broadcast (in which all host bits are set to 1 in a classful environment), subnet broadcast (in which all host bits are set to 1 in a non-classful environment), and limited broadcast (255.255.255.255) addresses are replaced by the link-local scope all-nodes multicast address (FF02:01) in IPv6.

Solicited-Node Address

The solicited-node address facilitates the efficient querying of network nodes during link-layer address resolution—the resolving of a link-layer address of a known IPv6 address. In IPv4, the Address Resolution Protocol (ARP) Request frame is sent to the MAC-level broadcast, disturbing all nodes on the network segment, including those that are not running IPv4. IPv6 uses the Neighbor Solicitation message to perform link-layer address resolution. However, instead of using the local-link scope all-nodes multicast address as the Neighbor Solicitation message destination, which would disturb all IPv6 nodes on the local link, the solicited-node multicast address is used. The solicited-node multicast address is constructed from the prefix FF02::1:FF00:0/104 and the last 24 bits (6 hexadecimal digits) of a unicast IPv6 address. Figure 3-7 shows the mapping of a unicast IPv6 address and its corresponding solicited-node multicast address.

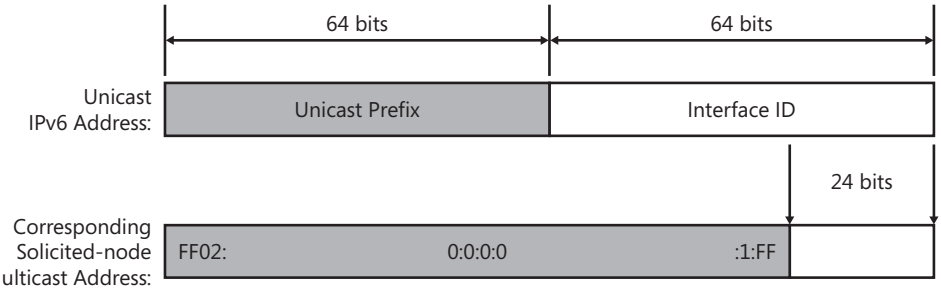

Figure 3-7 The mapping of a unicast address to its solicited-node multicast address

For example, Node A is assigned the link-local address of FE80::2AA:FF:FE28:9C5A and is also listening on the corresponding solicited-node multicast address of FF02::1:FF28:9C5A. (An underline is used to highlight the correspondence of the last six hexadecimal digits.) Node B on the local link must resolve Node A's link-local address FE80::2AA:FF:FE28:9C5A to its corresponding link-layer address. Node B sends a Neighbor Solicitation message to the solicited-node multicast address of FF02::1:FF28:9C5A. Because Node A is listening on this multicast address, it processes the Neighbor Solicitation message and sends a unicast Neighbor Advertisement message in reply.

The result of using the solicited-node multicast address is that link-layer address resolutions, a common occurrence on a link, are not using a mechanism that disturbs all network nodes. By using the solicited-node address, very few nodes are disturbed during address resolution. In practice, because of the relationship between the IPv6 interface ID and the solicited-node address, the solicited-node address acts as a pseudo-unicast address for very efficient address resolution. For more information, see the "IPv6 Interface Identifiers" section in this chapter.

Mapping IPv6 Multicast Addresses to Ethernet Addresses

When sending IPv6 multicast packets on an Ethernet link, the corresponding destination MAC address is 0x33-33-mm-mm-mm-mm, where *mm-mm-mm-mm* is a direct mapping of the last 32 bits (8 hexadecimal digits) of the IPv6 multicast address. Figure 3-8 shows the mapping of an IPv6 multicast address to an Ethernet multicast address.

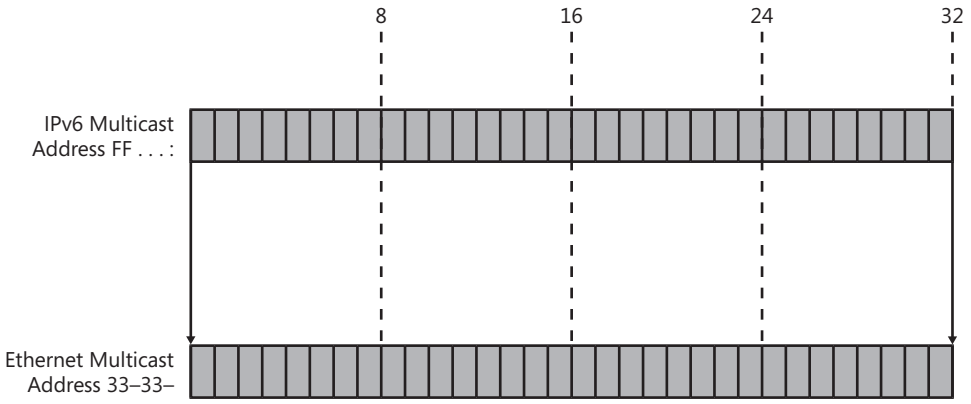

Figure 3-8 The mapping of IPv6 multicast addresses to Ethernet multicast addresses

Ethernet network adapters maintain a table of interesting destination MAC addresses. If an Ethernet frame with an interesting destination MAC address is received, it is passed to upper layers for additional processing. By default, this table contains the MAC-level broadcast address (0xFF-FF-FF-FF-FF-FF) and the unicast MAC address assigned to the adapter. To facilitate efficient delivery of multicast traffic, additional multicast destination addresses can be added or removed from the table. For every multicast address being listened to by the host, there is a corresponding entry in the table of interesting MAC addresses.

For example, an IPv6 host with the Ethernet MAC address of 00-AA-00-3F-2A-1C (link-local address of FE80::2AA:FF:FE3F:2A1C) adds the following multicast MAC addresses to the table of interesting destination MAC addresses on the Ethernet adapter:

- The address of 33-33-00-00-00-01, which corresponds to the link-local scope all-nodes multicast address of FF02::1 (fully expressed as FF02:0000:0000:0000:0000:0000:0000:0001).
- The address of 33-33-FF-3F-2A-1C, which corresponds to the solicited-node address of FF02::1:FF3F:2A1C. Remember that the solicited-node address is the prefix FF02::1:FF00:0/104 and the last 24 bits of the unicast IPv6 address.

Additional multicast addresses on which the host is listening are added and removed from the table as needed.

Anycast IPv6 Addresses

An anycast address is assigned to multiple interfaces. Packets addressed to an anycast address are forwarded by the routing infrastructure to the nearest interface to which the anycast address is assigned. To facilitate delivery, the routing infrastructure must be aware of the interfaces that have anycast addresses assigned to them and their distance in terms of routing metrics. This awareness is accomplished by the propagation of host routes throughout the routing infrastructure of the portion of the network that cannot summarize the anycast address using a route prefix.

For example, for the anycast address 3FFE:2900:D005:6187:2AA:FF:FE89: 6B9A, host routes for this address are propagated within the routing infrastructure of the organization assigned the 48-bit prefix 3FFE:2900:D005::/48. Because a node assigned this anycast address can be placed anywhere on the organization's intranet, host routes for all nodes assigned this anycast address are needed in the routing tables of all routers within the organization. Outside the organization, this anycast address is summarized by the 3FFE:2900:D005::/48 prefix that is assigned to the organization. Therefore, the host routes needed to deliver IPv6 packets to the nearest anycast group member within an organization's intranet are not needed in the routing infrastructure of the IPv6 Internet.

As of RFC 4291, anycast addresses are used only as destination addresses and are assigned only to routers. Anycast addresses are assigned out of the unicast address space, and the scope of an anycast address is the scope of the type of unicast address from which the anycast address is assigned. It is not possible to determine if a given destination unicast address is also an anycast address. The only nodes that have this awareness are the routers that use host routes to forward the anycast traffic to the nearest anycast group member and the anycast group members themselves.

Subnet-Router Anycast Address

The Subnet-Router anycast address is defined in RFC 4291 and is required. It is created from the subnet prefix for a given interface. When the Subnet-Router anycast address is constructed, the bits in the subnet prefix are fixed at their appropriate values and the remaining bits are set to 0. Figure 3-9 shows the structure of the Subnet-Router anycast address.

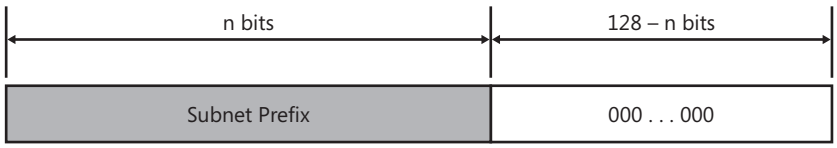

Figure 3-9 The structure of the Subnet-Router anycast address

All router interfaces attached to a subnet are assigned the Subnet-Router anycast address for that subnet. The Subnet-Router anycast address is used to communicate with the nearest router connected to a specified subnet.

IPv6 Addresses for a Host

An IPv4 host with a single network adapter typically has a single IPv4 address assigned to that adapter. An IPv6 host, however, usually has multiple IPv6 addresses assigned to each adapter. The interfaces on a typical IPv6 host are assigned the following unicast addresses:

- A link-local address for each interface
- Additional unicast addresses for each interface (which could be a unique local address and one or multiple global addresses)
- The loopback address (::1) for the loopback interface

Typical IPv6 hosts are always logically multihomed because they always have at least two addresses with which they can receive packets—a link-local address for local link traffic and a routable unique local or global address.

Additionally, each interface on an IPv6 host is listening for traffic on the following multicast addresses:

- The interface-local scope all-nodes multicast address (FF01::1)
- The link-local scope all-nodes multicast address (FF02::1)
- The solicited-node address for each unicast address assigned
- The multicast addresses of joined groups

IPv6 Addresses for a Router

The interfaces on an IPv6 router are assigned the following unicast addresses:

- A link-local address for each interface
- Additional unicast addresses for each interface (which could be a unique local address and one or multiple global addresses)
- The loopback address (::1) for the loopback interface

Additionally, the interfaces of an IPv6 router are assigned the following anycast addresses:

- A Subnet-Router anycast address for each subnet
- Additional anycast addresses (optional)

Additionally, the interfaces of an IPv6 router are listening for traffic on the following multicast addresses:

- The interface-local scope all-nodes multicast address (FF01::1)
- The interface-local scope all-routers multicast address (FF01::2)
- The link-local scope all-nodes multicast address (FF02::1)
- The link-local scope all-routers multicast address (FF02::2)
- The site-local scope all-routers multicast address (FF05::2)
- The solicited-node address for each unicast address assigned
- The multicast addresses of joined groups

Subnetting the IPv6 Address Space

Just as in IPv4, the IPv6 address space can be divided by using high-order bits that do not already have fixed values to create subnetted address prefixes. These are used either to summarize a level in the routing or addressing hierarchy (with a prefix length less than 64), or to define a specific subnet or network segment (with a prefix length of 64). IPv4 subnetting differs from IPv6 subnetting in the definition of the host ID portion of the address. In IPv4, the host ID can be of varying length, depending on the subnetting scheme. For currently defined unicast IPv6 addresses, the host ID is the interface ID portion of the IPv6 unicast address and is always a fixed size of 64 bits.

For most network administrators within an organization, subnetting the IPv6 address space consists of using subnetting techniques to divide the subnet ID portion of a global or unique local address prefix in a manner that allows for route summarization and delegation of the remaining address space to different portions of an IPv6 intranet. For both global and unique local addresses, the first 48 bits of the address are fixed. For the global address, the first 48 bits are fixed and allocated by an ISP. For the unique local address, the first 48 bits are fixed at FD00::/8 and the random 40-bit global ID assigned to a site of an organization.

Subnetting the subnet ID portion of a global or unique local address space requires a two-step procedure:

1. Determine the number of bits to be used for the subnetting.

2. Enumerate the new subnetted address prefixes.

The subnetting technique described here assumes that subnetting is done by dividing the 16-bit address space of the subnet ID using the high-order bits in the subnet ID. Although this method promotes hierarchical addressing and routing, it is not required. For example, in a small organization with a small number of subnets, you can also create a flat addressing space for the subnet ID by numbering the subnets starting at 0.

Step 1: Determining the Number of Subnetting Bits

The number of bits being used for subnetting determines the possible number of new subnetted address prefixes that can be allocated to portions of your network based on geographical or departmental divisions. In a hierarchical routing infrastructure, you need to determine how many address prefixes, and therefore how many bits, you need at each level in the hierarchy. The more bits you choose for the various levels of the hierarchy, the fewer bits you will have available to enumerate individual subnets in the last level of the hierarchy.

Depending on the needs of your organization, your subnetting scheme might be along nibble (hexadecimal digit) or bit boundaries. If you can subnet along nibble boundaries, your subnetting scheme becomes simplified and each hexadecimal digit can represent a level in the subnetting hierarchy. For example, a network administrator decides to implement a three-level hierarchy that uses the first nibble for an organization's campus, the next nibble for a building within a campus, and the last two nibbles for a subnet within a building. An example subnet ID for this scheme is 142A, which indicates campus 1, building 4, and subnet 42 (0x2A).

In some cases, bit-level subnetting is required. For example, a network administrator decides to implement a two-level hierarchy reflecting a geographical/departmental structure and uses 4 bits for the geographical level and 6 bits for the departmental level. This means that each department in each geographical location has only 6 bits of subnetting space left (16<;$MI>4<;$MI>6), or only 64 (= 2^6) subnets per department.

On any given level in the hierarchy, you will have a number of bits that are already fixed by the next level up in the hierarchy (*f*), a number of bits used for subnetting at the current level in the hierarchy (*s*), and a number of bits remaining for the next level down in the hierarchy (*r*). At all times, $f + s + r = 16$. This relationship is shown in Figure 3-10.

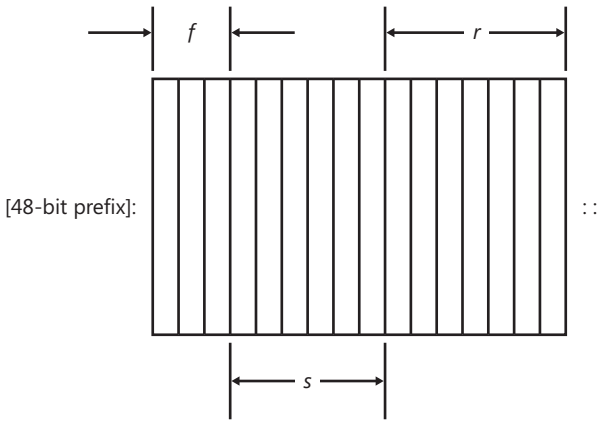

Figure 3-10 The subnetting of a Subnet ID

Step 2: Enumerating Subnetted Address Prefixes

Based on the number of bits used for subnetting, you must determine the new subnetted address prefixes. There are three main approaches:

Binary

- Enumerate new subnetted address prefixes by using binary representations of the subnet ID and converting to hexadecimal for the subnetted address prefixes.

Hexadecimal

- Enumerate new subnetted address prefixes by using hexadecimal representations of the subnet ID and a calculated increment between successive subnetted address prefixes.

Decimal

- Enumerate new subnetted address prefixes by using decimal representations of the subnet ID and increment.

Any of these methods produces the same result: an enumerated list of subnetted address prefixes.

Using the Binary Method

In the binary method, the 16-bit subnet ID is expressed as a 16-digit binary number. The bits within the subnet ID that are being used for subnetting are incremented for all their possible values, and for each value, the 16-digit binary number is converted to hexadecimal and combined with the 48-bit site prefix, producing the subnetted address prefixes.

1. Based on *s* (the number of bits chosen for subnetting), *m* (the prefix length of the address prefix being subnetted), and *f* (the number of bits already subnetted), calculate the following:

   $n = 2^s$

   n is the number of address prefixes that are obtained.

   $l = 48 + f + s$

   l is the prefix length of the new subnetted address prefixes.

2. Create a three-column table with *n* entries. The first column is the address prefix number (starting with 1), the second column is the binary representation of the subnet ID portion of the new address prefix, and the third column is the subnetted address prefix (in hexadecimal), which includes the 48-bit site prefix and the subnet ID.

3. In the first table entry, set all the bits being used for subnetting to 0. Convert the resulting 16-digit binary number to hexadecimal, combine it with the 48-bit site prefix, and write the subnetted address prefix. This first subnetted address prefix is just the original address prefix with the new prefix length.

4. In the next table entry, increment the value within the subnet bits. Convert the 16-digit binary number to hexadecimal, combine it with the 48-bit site prefix, and write the resulting subnetted address prefix.

5. Repeat step 4 until the table is complete.

For example, to perform a 3-bit subnetting of the global address prefix 2001:DB8:0:C000::/51, we first calculate the values for the number of prefixes and the new prefix length. Our starting values are s = 3, and f = 51 <;$MI> 48 = 3. The number of prefixes is 8 ($n = 2^3$). The new prefix length is 54 (l = 48 + 3 + 3). The initial value for the subnet ID in binary is 1100 0000 0000 0000 (0xC000 converted to binary).

Next, we construct a table with 8 entries. The entry for the address prefix 1 is 2001:DB8:0:C000::/54. Additional entries are increments of the subnet bits in the subnet ID portion of the address prefix, as shown in Table 3-3.

Table 3-3 The Binary Subnetting Technique for Address Prefix 2001:DB8:0:C000::/51

Address Prefix Number	Binary Representation of Subnet ID	Subnetted Address Prefix
1	1100 0000 0000 0000	2001:DB8:0:C000::/54
2	1100 0100 0000 0000	2001:DB8:0:C400::/54
3	1100 1000 0000 0000	2001:DB8:0:C800::/54
4	1100 1100 0000 0000	2001:DB8:0:CC00::/54
5	1101 0000 0000 0000	2001:DB8:0:D000::/54
6	1101 0100 0000 0000	2001:DB8:0:D400::/54
7	1101 1000 0000 0000	2001:DB8:0:D800::/54
8	1101 1100 0000 0000	2001:DB8:0:DC00::/54

In Table 3-3, the underline in the second column shows the bits that are being used for subnetting.

Using the Hexadecimal Method

Although the binary method allows you to see how the subnetted address prefixes are determined at their most basic level, this method is laborious and does not scale well. For example, imagine performing an 8-bit subnetting using the binary method, producing 256 subnetted prefixes. For an arbitrary subnetting scheme, a more formulaic approach is needed. The following method uses a formula for computing the hexadecimal increment between successive subnetted address prefixes:

1. Based on *s* (the number of bits chosen for subnetting), *m* (the prefix length of the address prefix being subnetted), and *F* (the hexadecimal value of the subnet being subnetted), calculate the following:

   $f = m <;\$MI> 48$

   f is the number of bits within the subnet ID that are already fixed.

   $n = 2^s$

   n is the number of address prefixes that are obtained.

   $i = 2^{16<;\$MI>(f+s)}$

   i is the incremental value between each successive subnet ID expressed in hexadecimal form.

   $l = 48 + f + s$

   l is the prefix length of the new subnetted address prefixes.
2. Create a two-column table with *n* entries. The first column is the address prefix number (starting with 1), and the second column is the new subnetted address prefix.
3. In the first table entry, based on *F*, the hexadecimal value of the subnet ID being subnetted, set the subnetted address prefix to *48-bit prefix*:*F*::/*l*.
4. In the next table entry, increase the value within the subnet ID portion of the site address by *i*. For example, in the second table entry, set the subnetted prefix to *48-bit prefix*:*F* + *i*::/*l*.
5. Repeat step 4 until the table is complete.

For example, to perform a 3-bit subnetting of the global address prefix 2001:DB8:0:C000::/51, we first calculate the values of the number of prefixes, the increment, and the new prefix length. Our starting values are *F* = 0xC000, *s* = 3, and *f* = 51 <;$MI> 48 = 3. The number of prefixes is 8 ($n = 2^3$). The increment is 0x400 ($i = 2^{16<;\$MI>(f+s)} = 1024 = 0x400$). The new prefix length is 54 (*l* = 48 + 3 + 3).

Next, we construct a table with 8 entries. The entry for the address prefix 1 is 2001:DB8:0:C000::/54. Additional entries in the table are successive increments of *i* in the subnet ID portion of the address prefix, as shown in Table 3-4.

Table 3-4 The Hexadecimal Subnetting Technique for Address Prefix 2001:DB8:0:C000::/51

Address Prefix Number	Subnetted Address Prefix
1	2001:DB8:0:C000::/54
2	2001:DB8:0:C400::/54
3	2001:DB8:0:C800::/54
4	2001:DB8:0:CC00::/54
5	2001:DB8:0:D000::/54
6	2001:DB8:0:D400::/54
7	2001:DB8:0:D800::/54
8	2001:DB8:0:DC00::/54

Using the Decimal Method

If you are more comfortable working with decimal numbers, the following formulaic procedure will produce the same results. However, there are additional steps to convert to decimal and then back to hexadecimal for the representation of the subnetted address prefix.

1. Based on *s* (the number of bits chosen for subnetting), *m* (the prefix length of the address prefix being subnetted), and *F* (the hexadecimal value of the subnet ID being subnetted), calculate the following:

   f = m <;$*MI*> 48

   f is the number of bits within the subnet ID that are already fixed.

   $n = 2^s$

   n is the number of address prefixes that are obtained.

   $i = 2^{16<;\$MI>(f-s)}$

   i is the incremental value between each successive subnet ID.

   $l = 48 + f + s$

   l is the prefix length of the new subnetted address prefixes.

   D = decimal representation of *F*

2. Create a three-column table with *n* entries. The first column is the address prefix number (starting with 1), the second column is the decimal representation of the subnet ID portion of the new address prefix, and the third column is the new subnetted address prefix.

3. In the first table entry, the decimal representation of the subnet ID is *D* and the subnetted address prefix is *48-bit prefix*:*F*::/*l*.

4. In the next table entry, for the second column, increase the value of the decimal representation of the subnet ID by *i*. For example, in the second table entry, the decimal representation of the subnet ID is *D* + *i*.

5. For the third column, convert the decimal representation of the subnet ID to hexadecimal and construct the prefix from *48-bit prefix*:*subnet ID*::/*l*. For example, in the second table entry, the subnetted address prefix is *48-bit prefix*:*D* + *i* (converted to hexadecimal)::/*l*.

6. Repeat steps 4 and 5 until the table is complete.

For example, to perform a 3-bit subnetting of the global address prefix 2001:DB8:0:C000::/51, we first calculate the values of the number of prefixes, the increment, the new prefix length, and the decimal representation of the starting subnet ID. Our starting values are *F* = 0xC000, *s* = 3, and *f* = 51 <;$MI> 48 = 3. The number of prefixes is 8 ($n = 2^3$). The increment is 1024 ($i = 2^{16<;\$MI>(f + s)}$). The new prefix length is 54 (*l* = 48 + 3 + 3). The decimal representation of the starting subnet ID is 49152 (*D* = 0xC000 = 49152).

Next, we construct a table with 8 entries. The entry for the address prefix 1 is 49152 and 2001:DB8:0:C000::/54. Additional entries in the table are successive increments of *i* in the subnet ID portion of the address prefix, as shown in Table 3-5.

Table 3-5 The Decimal Subnetting Technique for Address Prefix 2001:DB8:0:C000::/51

Address Prefix Number	Decimal Representation of Subnet ID	Subnetted Address Prefix
1	49152	2001:DB8:0:C000::/54
2	50176	2001:DB8:0:C400::/54
3	51200	2001:DB8:0:C800::/54
4	52224	2001:DB8:0:CC00::/54
5	53248	2001:DB8:0:D000::/54
6	54272	2001:DB8:0:D400::/54
7	55296	2001:DB8:0:D800::/54
8	56320	2001:DB8:0:DC00::/54

IPv6 Interface Identifiers

The last 64 bits of a currently defined IPv6 unicast address are for the interface identifier, which is unique for a 64-bit subnet prefix of a unicast IPv6 address. In IPv4, the host or node ID portion of an IPv4 address is a logical identifier of an interface on an IPv4 subnet. IPv4 host IDs are of variable length, depending on the subnetting scheme and how many interfaces you want to allow on a given subnet. For example, with an 8-bit host ID, there were 2^8 <;$MI> 2 or 254 possible host IDs. (The all-zeros and all-ones combinations are reserved.)

In IPv6, the interface ID is of fixed length. This length was not fixed at 64 bits to allow up to 2^{64} possible hosts on the same subnet. Rather, the IPv6 interface ID is 64 bits long to accommodate the mapping of current 48-bit MAC addresses used by many local area network (LAN) technologies such as Ethernet and the mapping of 64-bit MAC addresses of IEEE 1394 (also known as FireWire) and future LAN technologies.

The ways in which an interface identifier for a LAN interface is determined are the following:

- As defined in RFC 4291, it can be derived from the Extended Unique Identifier (EUI)-64 address. The 64-bit EUI-64 address is defined by the Institute of Electrical and Electronics Engineers (IEEE). EUI-64 addresses are either assigned to a network adapter or derived from IEEE 802 addresses. This is the default behavior for IPv6 in Windows XP and Windows Server 2003.
- As defined in RFC 3041, it might have a temporarily assigned, randomly generated interface identifier to provide a level of anonymity. For more information, see the "Temporary Address Interface Identifiers" section in this chapter.
- It is assigned during stateful address autoconfiguration—for example, via Dynamic Host Configuration Protocol for IPv6 (DHCPv6).
- As defined in RFC 2472, an interface identifier can be based on link-layer addresses or serial numbers, or it can be randomly generated when configuring a Point-to-Point Protocol (PPP) interface and an EUI-64 address is not available.
- It is assigned during manual address configuration.
- It is a permanent interface identifier that is randomly generated to mitigate address scans of unicast IPv6 addresses on a subnet. This is the default behavior for LAN interfaces for IPv6 in Windows Vista and Windows Server 2008. You can disable this behavior with the **netsh interface ipv6 set global randomizeidentifiers=disabled** command. When this behavior is disabled, IPv6 for Windows Vista and Windows Server 2008 will use EUI-64-based interface identifiers.

EUI-64 Address-Based Interface Identifiers

One way to derive an IPv6 interface identifier is through the EUI-64 address, a new type of MAC address for network adapters. To gain an understanding of EUI-64 addresses, it is useful to review the current MAC address format known as *IEEE 802 addresses*.

IEEE 802 Addresses

Network adapters for common LAN technologies such as Ethernet and IEEE 802.11 use a 48-bit address called an IEEE 802 address. It consists of a 24-bit company ID (also called the *manufacturer ID*) and a 24-bit extension ID (also called the *board ID*). The combination of the company ID, which is uniquely assigned to each manufacturer of network adapters, and the extension ID, which is uniquely assigned to each network adapter at the time of manufacture, produces a globally unique 48-bit address. This 48-bit address is also called the physical, hardware, or media access control (MAC) address.

Figure 3-11 shows the structure of the 48-bit IEEE 802 address for Ethernet.

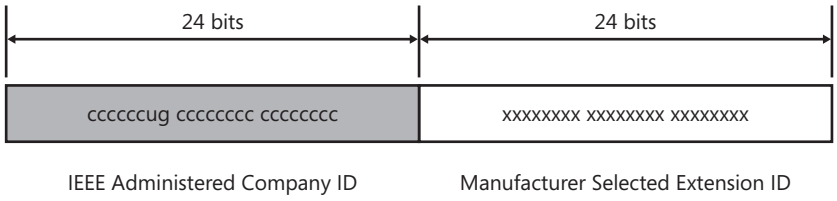

Figure 3-11 The structure of the 48-bit IEEE 802 address for Ethernet

Defined bits within the IEEE 802 address for Ethernet are as follows:

Universal/Local (U/L)

- The seventh bit in the first byte is used to indicate whether the address is universally or locally administered. If the U/L bit is set to 0, the IEEE (through the designation of a unique company ID) has administered the address. If the U/L bit is set to 1, the address is locally administered. In this case, the network administrator has overridden the manufactured address and specified a different address. The U/L bit is designated by the **u** in Figure 3-11.

Individual/Group (I/G)

- The eighth (low-order) bit of the first byte is used to indicate whether the address is an individual address (unicast) or a group address (multicast). When set to 0, the address is a unicast address. When set to 1, the address is a multicast address. The I/G bit is designated by the **g** in Figure 3-11.

For a typical IEEE 802 address assigned to a network adapter, both the U/L and I/G bits are set to 0, corresponding to a universally administered, unicast MAC address.

IEEE EUI-64 Addresses

The IEEE EUI-64 address represents a new standard for network interface addressing. The company ID is still 24-bits long, but the extension ID is 40 bits, creating a much larger address space for a network adapter manufacturer. The EUI-64 address uses the U/L and I/G bits in the same way as the IEEE 802 address.

Figure 3-12 shows the structure of the EUI-64 address.

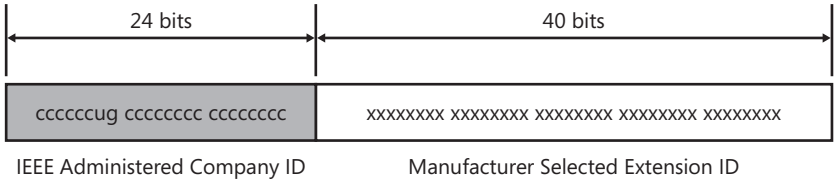

Figure 3-12 The structure of the EUI-64 address

Mapping IEEE 802 Addresses to EUI-64 Addresses

To create an EUI-64 address from an IEEE 802 address, the 16 bits of 11111111 11111110 (0xFFFE) are inserted into the IEEE 802 address between the company ID and the extension ID, as shown in Figure 3-13.

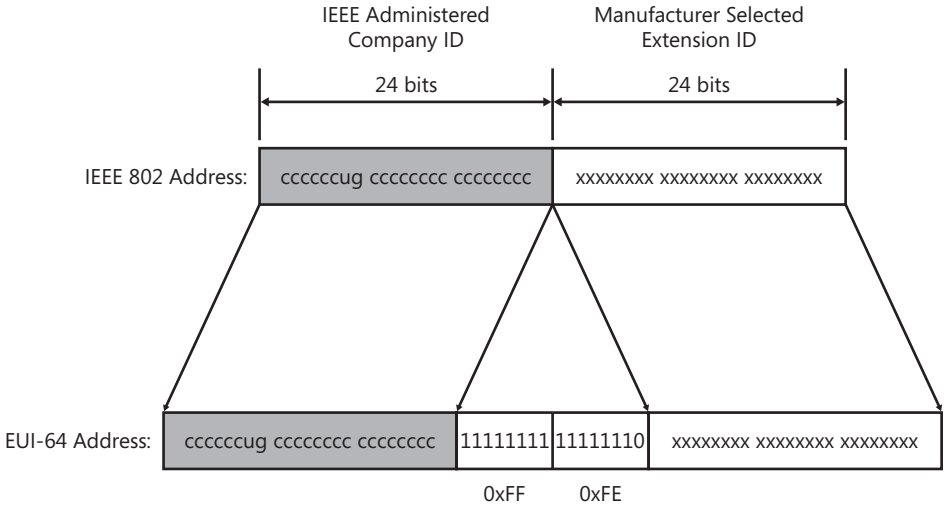

Figure 3-13 The mapping of IEEE 802 addresses to EUI-64 addresses

Obtaining Interface Identifiers for IPv6 Addresses

To obtain the 64-bit interface identifier for IPv6 unicast addresses, the U/L bit in the EUI-64 address is complemented. (If it is a 1 in the EUI-64 address, it is set to 0; and if it is a 0 in the EUI-64 address, it is set to 1.)

The main reason for complementing the U/L bit is to provide greater compressibility of locally administered EUI-64 addresses. It is common practice when assigning locally administered addresses to number them in a simple way. For example, on a point-to-point link, you can assign to one interface on the link the locally administered EUI-64 address of 02-00-00-00-00-00-00-01 and to the other interface the locally administered EUI-64 address of 02-00-00-00-00-00-00-02. If the U/L bit is not complemented, the corresponding link-local addresses for these two interfaces become FE80::200:0:0:1 and FE80::200:0:0:2. By complementing the U/L bit, the corresponding link-local addresses for these two interfaces become FE80::1 and FE80::2.

Figure 3-14 shows the conversion of an EUI-64 address to an IPv6 interface identifier.

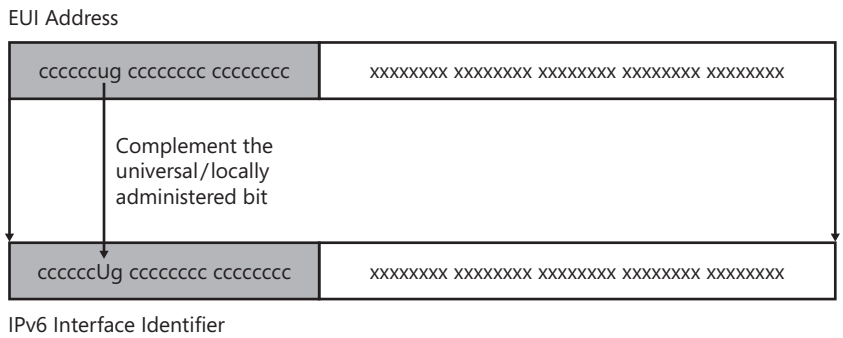

Figure 3-14 The conversion of an EUI-64 address to an IPv6 interface identifier

> **Note** Because the U/L bit is complemented when converting an EUI-64 address to an IPv6 interface identifier, the resulting bit in the IPv6 interface identifier has the opposite interpretation of the IEEE-defined U/L bit. If the seventh bit of the IPv6 interface identifier is set to 0, it is locally administered. If the seventh bit of the IPv6 interface identifier is set to 1, it is universally administered.

Converting IEEE 802 Addresses to IPv6 Interface Identifiers

To obtain an IPv6 interface identifier from an IEEE 802 address, you must first map the IEEE 802 address to an EUI-64 address, and then complement the U/L bit. Figure 3-15 shows this conversion process for a universally administered, unicast IEEE 802 address.

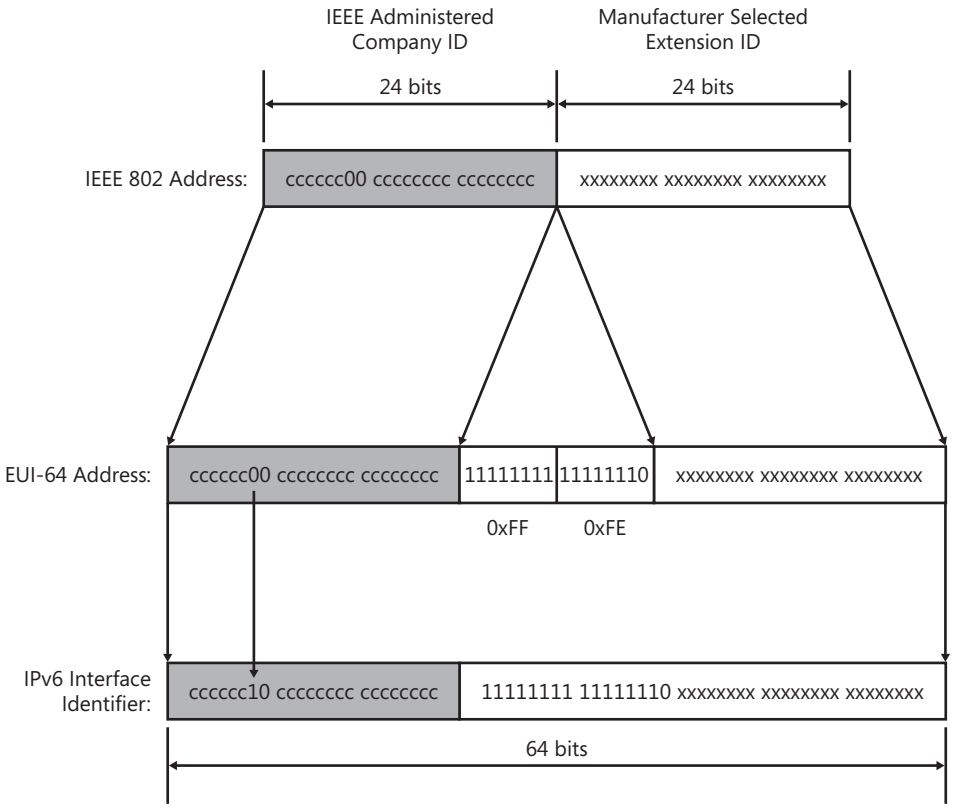

Figure 3-15 The conversion of an IEEE 802 address to an IPv6 interface identifier

IEEE 802 Address Conversion Example

Host A has the Ethernet MAC address of 00-AA-00-3F-2A-1C. First, it is converted to EUI-64 format by inserting FF-FE between the third and fourth bytes, yielding 00-AA-00-FF-FE-3F-2A-1C. Then, the U/L bit, which is the seventh bit in the first byte, is complemented. The first byte in binary form is 00000000. When the seventh bit is complemented, it becomes 00000010 (0x02). The final result is 02-AA-00-FF-FE-3F-2A-1C which, when converted to colon hexadecimal notation, becomes the interface identifier 2AA:FF:FE3F:2A1C. As a result, the link-local address that corresponds to the network adapter with the MAC address of 00-AA-00-3F-2A-1C is FE80::2AA:FF:FE3F:2A1C.

Note When complementing the U/L bit, add 0x2 to the first byte if the EUI-64 address is universally administered, and subtract 0x2 from the first byte if the EUI-64 address is locally administered.

Temporary Address Interface Identifiers

In today's IPv4-based Internet, a typical Internet user dials an ISP and obtains an IPv4 address using PPP and the Internet Protocol Control Protocol (IPCP). Each time the user dials, a different IPv4 address might be obtained. Therefore, it is not easy to track a dial-up user's traffic on the Internet based on the user's IPv4 address.

For IPv6-based dial-up connections, the user is assigned a 64-bit prefix, at the time of connection, by using router discovery, which consists of an exchange of Router Solicitation and Router Advertisement messages. If the interface identifier is always based on the EUI-64 address (as derived from the static IEEE 802 address), it is possible to identify the traffic of a specific node regardless of the prefix assigned at the time of connection. The use of the same 64-bit interface identifier allows identification of a user's traffic whether the user is accessing the Internet from home or from work. This makes it easy for Internet merchants and malicious users to track a specific user and his or her use of the Internet.

To address this concern and provide the same level of anonymity as that provided with IPv4, RFC 3041 describes an alternative derivation of the IPv6 interface identifier that is randomly generated and changes over time.

The initial interface identifier is generated using random number techniques. For IPv6 systems that do not have the ability to store any history information for generating future values of the interface identifier, a new random interface identifier is generated each time the IPv6 protocol is initialized. For IPv6 systems that do have storage capabilities, a history value is stored and when the IPv6 protocol is initialized, a new interface identifier is created through the following process:

1. Retrieve the history value from storage, and append the interface identifier based on the EUI-64 address of the adapter.
2. Compute the Message Digest-5 (MD5) hash over the quantity in step 1. The MD5 hash computation will produce a 128-bit value.
3. Store the low-order 64 bits of the MD5 hash computed in step 2 as the history value for the next computation of the interface identifier.
4. Take the high-order 64 bits of the MD5 hash computed in step 2 and set the seventh bit to zero. The seventh bit corresponds to the U/L bit, which, when set to 0, indicates a locally administered interface identifier. The result is the interface identifier.

The resulting IPv6 address, based on this random interface identifier, is known as a *temporary address*. Temporary addresses are generated for public address prefixes that use stateless address autoconfiguration. Temporary addresses are used for the lower of the following values of the valid and preferred lifetimes:

- The lifetimes included in the Prefix Information option in the received Router Advertisement message.
- Local default values of 1 week for valid lifetime and 1 day for preferred lifetime.

After the temporary address valid lifetime expires, a new interface identifier and temporary address is generated. For more information about router discovery, see Chapter 6, "Neighbor Discovery." For more information about stateless address autoconfiguration and valid and preferred lifetimes, see Chapter 8.

IPv4 Addresses and IPv6 Equivalents

To summarize the relationships between IPv4 addressing and IPv6 addressing, Table 3-6 lists both IPv4 addresses and addressing concepts and their IPv6 equivalents.

Table 3-6 IPv4 Addressing Concepts and Their IPv6 Equivalents

IPv4 Address	IPv6 Address
Internet address classes	Not applicable in IPv6
Multicast addresses (224.0.0.0/4)	IPv6 multicast addresses (FF00::/8)
Broadcast addresses	Not applicable in IPv6
Unspecified address is 0.0.0.0	Unspecified address is ::
Loopback address is 127.0.0.1	Loopback address is ::1
Public IP addresses	Global unicast addresses
Private IP addresses (10.0.0.0/8, 172.16.0.0/12, and 192.168.0.0/16)	Unique local (FD00::/8) or site-local addresses (FEC0::/10) (deprecated)
APIPA addresses (169.254.0.0/16)	Link-local addresses (FE80::/64)
Text representation: Dotted-decimal notation	Text representation: Colon hexadecimal format with suppression of leading zeros and zero compression.
Prefix representation: Subnet mask in dotted-decimal notation or prefix length notation	Prefix representation: Prefix length notation only

References

The following references were cited in this chapter:

- RFC 2472 — "IP Version 6 over PPP"
- RFC 3041 — "Privacy Extensions for Stateless Address Autoconfiguration in IPv6"
- RFC 3306 — "Unicast-Prefix-based IPv6 Multicast Addresses"
- RFC 3587 — "IPv6 Global Unicast Address Format"
- RFC 3879 — "Deprecating Site Local Addresses"
- RFC 3927 — "Dynamic Configuration of IPv4 Link-Local Addresses"

- RFC 3956 — "Embedding the Rendezvous Point (RP) Address in an IPv6 Multicast Address"
- RFC 4007 — "IPv6 Scoped Address Architecture"
- RFC 4193 — "Unique Local IPv6 Unicast Addresses"
- RFC 4291 — "IP Version 6 Addressing Architecture"

You can obtain these RFCs from *http://www.ietf.org/rfc.html*.

Testing for Understanding

To test your understanding of IPv6 addressing, answer the following questions. See Appendix D, "Testing for Understanding Answers," to check your answers.

1. Why is the IPv6 address length 128 bits?
2. Express FEC0:0000:0000:0001:02AA:0000:0000:007A more efficiently.
3. How many blocks and bits are expressed by "::" in the addresses 2001:DB8::2AA:9FF:FE56:24DC and FF02::2?
4. Describe the difference between unicast, multicast, and anycast addresses in terms of a host sending packets to zero or more interfaces.
5. Why are no broadcast addresses defined for IPv6?
6. Define the structure, including field sizes, of the global unicast address.
7. Define the scope for each of the different types of unicast addresses.
8. Explain how global and unique local addressing can share the same subnetting infrastructure within an organization.
9. Define the structure, including field sizes, of the multicast address.
10. Explain how the solicited-node multicast address acts as a pseudo-unicast address.
11. How do routers know the nearest location of an anycast group member?
12. Perform a 4-bit subnetting on the unique local prefix FD1A:39C1:4BC2:3D80::/57.
13. What is the EIU-64-based IPv6 interface identifier for the universally administered, unicast IEEE 802 address of 0C-1C-09-A8-F9-CE? What is the corresponding link-local address? What is the corresponding solicited-node multicast address?

14. What is the IPv6 interface identifier for the locally administered, unicast EUI-64 address of 02-00-00-00-00-00-00-09? What is the corresponding link-local address?

15. For each type of address shown in the following table, identify how the address begins in colon hexadecimal notation.

Type of Address	Begins with…
Link-local unicast address	FE80
Site-local unicast address	
Unique local unicast address	
Global address (as defined by RFC 3587)	
Multicast address	
Link-local scope multicast address	
Site-local scope multicast address	
Solicited-node multicast address	
IPv4-mapped address	
6to4 address	
Teredo address	