Microsoft

UNDERSTANDING
IPv6

Joseph Davies

PUBLISHED BY
Microsoft Press
A Division of Microsoft Corporation
One Microsoft Way
Redmond, Washington 98052-6399

Library of Congress Cataloging-in-Publication Data
Davies, Joseph.
Understanding IPv6 / Joseph Davies.
p. cm.
Includes index.
ISBN 0-7356-1245-5
1. TCP/IP (Computer network protocol) 2. Internet. I. Title.

TK5105.585 .D38 2002
004.62--dc21 2002029525

Printed and bound in the United States of America.

1 2 3 4 5 6 7 8 9 QWE 8 7 6 5 4 3

Distributed in Canada by H.B. Fenn and Company Ltd.

A CIP catalogue record for this book is available from the British Library.

Microsoft Press books are available through booksellers and distributors worldwide. For further information about international editions, contact your local Microsoft Corporation office or contact Microsoft Press International directly at fax (425) 936-7329. Visit our Web site at www.microsoft.com/mspress. Send comments to *mspinput@microsoft.com*.

Acquisitions Editors: Juliana Aldous Atkinson, Jeff Koch
Project Editor: Maureen Williams Zimmerman
Technical Editor: Jim Johnson

Body Part No. X08-04478

Table of Contents

Chapters

2 IPv6 Protocol for the Windows .NET Server 2003 Family 17

3 IPv6 Addressing 45

12 IPv6 Mobility 303

Appendixes

Chapter 3

IPv6 Addressing

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

- 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 derived.
- List and compare the different addressing concepts between IPv4 addresses and IPv6 addresses.

THE IPV6 ADDRESS SPACE

The most obvious distinguishing feature of 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 (or 3.4×10^{38}), possible addresses.

In the late 1970s, when the IPv4 address space was designed, it was unimaginable that it could ever be exhausted. However, due to changes in technology and an allocation practice that did not anticipate the recent explosion of hosts on the Internet, the IPv4 address space was consumed to the point that by 1992, it was clear a replacement would be necessary.

With IPv6, it is even harder to conceive that the IPv6 address space will ever be consumed. To help put this number in perspective, a 128-bit address space provides 665,570,793,348,866,943,898,599 (6.65×10^{23}) addresses for every 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 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 addressing and routing that is currently lacking on the IPv4-based Internet.

ADDRESSES PER SQUARE METER OF THE EARTH

The number of 6.65×10^{23} addresses for every square meter of the Earth's surface is derived from the fact that the surface of the Earth is approximately 197,399,019 square miles and there are 2.59×10^6 square meters per square mile. So, the Earth's surface is $197{,}399{,}019 \times 2.59 \times 10^6$, or 511,263,971,197,990 square meters.

Therefore, there are 340,282,366,920,938,463,463,374,607,431,768,211,456 / 511,263,971,197,990, or 665,570,793,348,866,943,898,599 (or 6.65×10^{23}) addresses for each square meter of the Earth's surface.

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 identifier and a 64-bit interface identifier (a 50-50 split between subnet space and interface space). The 64 bits of subnet identifier 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.

Current Allocation

Similar to the way in which the IPv4 address space was divided into unicast addresses (using Internet address classes) and multicast addresses, the IPv6 address space is divided on the basis of the value of high-order bits. The high-order bits and their fixed values are known as a Format Prefix (FP).

Table 3-1 lists the allocation of the IPv6 address space by FPs as defined in RFC 2373.

Table 3-1. CURRENT ALLOCATION OF THE IPV6 ADDRESS SPACE

Allocation	*Format Prefix (FP)*	*Fraction of the Address Space*
Reserved	0000 0000	1/256
Unassigned	0000 0001	1/256
Reserved for Network Service Access Point (NSAP) allocation	0000 001	1/128
Unassigned	0000 010	1/128
Unassigned	0000 011	1/128
Unassigned	0000 1	1/32
Unassigned	0001	1/16
Aggregatable global unicast addresses	001	1/8
Unassigned	010	1/8
Unassigned	011	1/8
Unassigned	100	1/8
Unassigned	101	1/8
Unassigned	110	1/8
Unassigned	1110	1/16
Unassigned	1111 0	1/32
Unassigned	1111 10	1/64
Unassigned	1111 110	1/128
Unassigned	1111 1110 0	1/512
Link-local unicast addresses	1111 1110 10	1/1024
Site-local unicast addresses	1111 1110 11	1/1024
Multicast addresses	1111 1111	1/256

The current set of unicast addresses that can be used with IPv6 nodes consists of aggregatable global unicast addresses, link-local unicast addresses, and site-local unicast addresses. These addresses represent only 12.7 percent of the entire IPv6 address space.

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:

```
0010000111011010000000001101001100000000000000000010111100111011
0000001010101010000000001111111111111110001010001001110001011010
```

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

```
0010000111011010  0000000011010011  0000000000000000  0010111100111011
0000001010101010  0000000011111111  1111111000101000  1001110001011010
```

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

21DA:00D3:0000:2F3B: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:

21DA:D3:0:2F3B:2AA:FF:FE28:9C5A

NUMBER SYSTEM CHOICE FOR IPV6

Hexadecimal (the Base$_{16}$ numbering system), rather than decimal (the Base$_{10}$ numbering system), is used for IPv6 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, with the exception of some types of IPv6 addresses that contain embedded IPv4 addresses. Configuration of typical

continued

Number System Choice for IPv6 *continued*

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 3-2 lists the conversion between binary, hexadecimal, and decimal numbers.

Table 3-2. 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 BITS IN ::?

To determine how many 0 bits are represented by the ::, you can count the number of blocks in the compressed address, subtract this number from 8, and then multiply the result by 16. For example, in the address FF02::2, there are two blocks (the "FF02" block and the "2" block.) The number of bits expressed by the :: is 96 ($96 = (8 - 2) \times 16$). Zero compression can be used only once in a given address. Otherwise, you could not determine the number of 0 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 of a route or subnet identifier. Prefixes for IPv6 subnet identifiers and 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, 21DA:D3::/48 is a route prefix and 21DA:D3:0:2F3B::/64 is a subnet 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 route or address range that is summarizing a portion of the IPv6 address space.

NOTE:
IPv4 implementations commonly use a dotted decimal representation of the network prefix 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 ID from the host ID.

In IPv6, however, there is no notion of a variable length subnet identifier. 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 2373, 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 FEC0::2AC4:2AA:FF:FE9A:82D4 as FEC0::2AC4:2AA:FF:FE9A:82D4/64. Due to the 50-50 split of subnet and interface identifiers, the unicast IPv6 address FEC0::2AC4:2AA:FF:FE9A:82D4 implies that the subnet identifier is FEC0:0:0:2AC4::/64.

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 2373 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. 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 2373 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:

- Aggregatable global unicast addresses
- Link-local addresses
- Site-local addresses
- Special addresses
- Compatibility addresses
- NSAP addresses

Aggregatable Global Unicast Addresses

Aggregatable global unicast addresses, also known as global addresses, are identified by the FP of 001. IPv6 global addresses are equivalent to public IPv4 addresses. They are globally routable and reachable on the IPv6 portion of the Internet.

As the name implies, aggregatable global unicast addresses are designed to be aggregated or summarized to produce 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.

Figure 3-1 shows the structure of an aggregatable global unicast address.

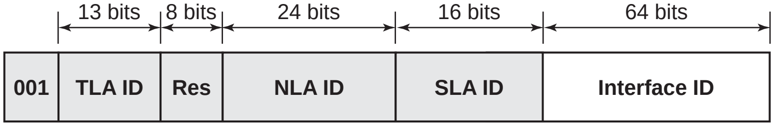

Figure 3-1. *The structure of an aggregatable global unicast address*

The fields in the aggregatable global unicast address are:

TLA ID — Top-Level Aggregation Identifier. The size of this field is 13 bits. The TLA ID identifies the highest level in the routing hierarchy. TLA IDs are administered by the Internet Assigned Numbers Authority (IANA) and allocated to local Internet registries that, in turn, allocate individual TLA IDs to large, long-haul ISPs. A 13-bit field allows up to 8,192 different TLA IDs. Routers in the highest level of the IPv6 Internet routing hierarchy (called default-free routers) do not have a default route—only routes with 16-bit prefixes corresponding to the allocated TLA IDs and additional entries for routes based on the TLA ID assigned to the routing region where the router is located.

Res — Bits that are reserved for future use in expanding the size of either the TLA ID or the NLA ID (defined next). The size of this field is 8 bits.

NLA ID — Next-Level Aggregation Identifier. The size of this field is 24 bits. The NLA ID allows an ISP to create multiple levels of addressing hierarchy within its network to both organize addressing and routing for downstream ISPs and identify organization sites. The structure of the ISP's network is not visible to the default-free routers. The combination of the 001 FP, the TLA ID, the Res field, and the NLA ID form a 48-bit prefix that is assigned to an organization's site that is connecting to the IPv6 portion of the Internet. A site is an organization network or portion of an organization's network that has a defined geographical location (such as an office, an office complex, or a campus).

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

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

Billions 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 FP of 001 and the current definition of the TLA ID (13 bits long) and NLA ID (24 bits long), it is possible to define 2^{37} or 137,438,953,472 possible 48-bit prefixes to assign to sites connected to the Internet. This large number of sites is possible even when we are using only 1/8th 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 network IDs 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 network IDs and then subtracting the network IDs used for the private address space. Even with the adoption of CIDR to make more efficient use of unassigned Class A and Class B network IDs, 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.

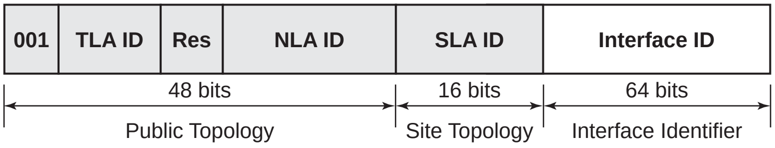

Figure 3-2. *The topological structure of the global address*

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

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

Link-local addresses, identified by the FP of 1111 1110 10, 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. Link-local addresses are equivalent to Automatic Private IP Addressing (APIPA) IPv4 addresses autoconfigured on Microsoft Windows .NET Server 2003 family, Windows XP, Windows 2000, Windows Millennium Edition, and Windows 98 computers using the 169.254.0.0/16 prefix. The scope of a link-local address is the local link.

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

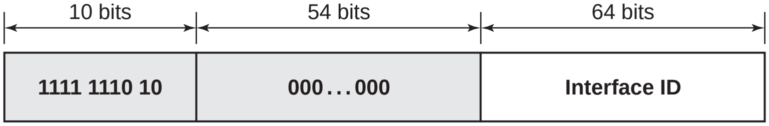

Figure 3-3. *The structure of the link-local address*

A link-local address is required for 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 the FP of 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.

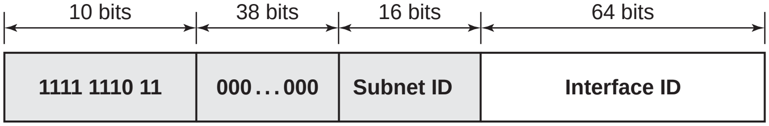

Figure 3-4. *The structure of the site-local address*

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, "Address Autoconfiguration."

The first 48 bits are always fixed for site-local addresses, beginning with FEC0::/48. After the 48 fixed bits is a 16-bit subnet identifier (Subnet ID field) that provides 16 bits with which you can create subnets within your organization. With 16 bits, you can have up to 65,536 subnets in a flat subnet structure, or you can divide the high-order bits of the Subnet ID field to create a hierarchical and aggregatable routing infrastructure. After the Subnet ID field is a 64-bit Interface ID field that identifies a specific interface on a subnet.

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

For example, a specific subnet of your organization can be assigned the global prefix 3FFE:FFFF:4D1C:221A::/64 and the site-local prefix FEC0:0:0:221A::/64 where the subnet is effectively identified by the SLA ID/Subnet ID value of 221A. While 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 used to identify 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-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-only node to an IPv6 node. Windows .NET Server 2003 family and Windows XP IPv6 do not support the use of IPv4-mapped addresses.
- 6over4 address
  An address of the type [*64-bit prefix*]:0:0:WWXX:YYZZ, where WWXX:YYZZ is the colon hexadecimal representation of *w.x.y.z* (a public or private IPv4 address), is used to represent a host for the tunneling mechanism known as 6over4.
- 6to4 address
  An address of the type 2002:WWXX:YYZZ:[*SLA ID*]:[*Interface ID*], where WWXX:YYZZ is the colon hexadecimal representation of *w.x.y.z* (a public IPv4 address), is used to represent a node for the tunneling mechanism known as 6to4.
- 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 used to represent a node for the address assignment mechanism known as Intra-Site Automatic Tunnel Addressing Protocol (ISATAP).

For more information about IPv6 compatibility addresses, see Chapter 11, "Coexistence and Migration."

NSAP Addresses

To provide a way of mapping Open Systems Interconnect (OSI) NSAP addresses to IPv6 addresses, NSAP addresses use the FP of 0000001 and map the last 121 bits of the NSAP address to an IPv6 address. For more information about the four types of NSAP address mappings, see RFC 1888. Figure 3-5 shows the structure of NSAP addresses for IPv6.

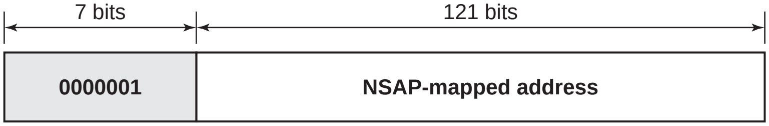

Figure 3-5. *The structure of NSAP addresses for IPv6*

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 FP of 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 header. Beyond the FP, 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.

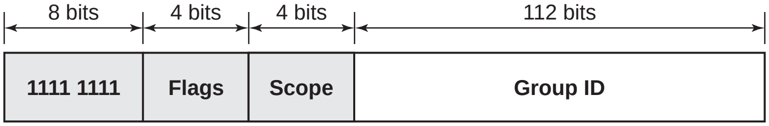

Figure 3-6. *The structure of the IPv6 multicast address*

The fields in the multicast address are:

Flags — Indicates flags set on the multicast address. The size of this field is 4 bits. As of RFC 2373, the only flag defined is the Transient (T) flag, which uses the low-order bit of the Flags field. When set to 0, the T flag indicates that the multicast address is a permanently assigned (well-known) multicast address allocated by IANA. When set to 1, the T flag indicates that the multicast address is a transient (non-permanently-assigned) multicast address.

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 information provided by multicast routing protocols, routers use the multicast scope to determine whether multicast traffic can be forwarded.

Table 3-3 lists the values for the Scope field assigned in RFC 2373.

Table 3-3. DEFINED VALUES FOR THE SCOPE FIELD

Scope Field Value	*Scope*
0	Reserved
1	Node-local scope
2	Link-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 node-local and link-local scopes, the following addresses are defined:

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

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

- FF01::2 (node-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.

Recommended Multicast IPv6 Addresses

With 112 bits in the Group ID field, it is possible to have 2^{112} group IDs. Because of the way in which IPv6 multicast addresses are mapped to Ethernet multicast MAC addresses, RFC 2373 recommends assigning the group ID from the low-order 32 bits of the IPv6 multicast address and setting the remaining original Group ID field bits to 0. By using only the low-order 32 bits, each group ID maps to a unique Ethernet multicast MAC address. Figure 3-7 shows the structure of the recommended IPv6 multicast address.

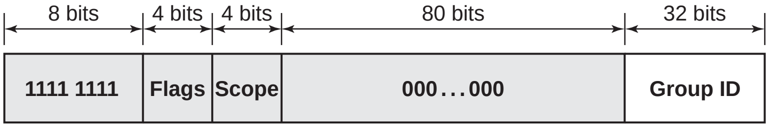

Figure 3-7. *The structure of the recommended IPv6 multicast address*

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 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 of a unicast IPv6 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, due to the relationship between the link-layer MAC address, 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 "IPv6 Interface Identifiers" in this chapter.

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. In order 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, source 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 2373, 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 2373 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 3-8 shows the structure of the Subnet-Router anycast address.

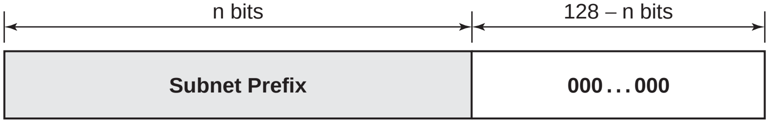

Figure 3-8. *The structure of the Subnet-Router anycast address*

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 site-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 site-local or global address.

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

- The node-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
- 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 site-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 node-local scope all-nodes multicast address (FF01::1)
- The node-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
- 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 network 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.

Subnetting for NLA IDs

If you are an ISP, subnetting the IPv6 address space consists of using subnetting techniques to divide the NLA ID portion of a global address in a manner that allows for route summarization and delegation of the remaining address space for different portions of your network, for downstream providers, or for individual customers. The global address has a 24-bit NLA ID field to be used by the various layers of ISPs between a top-level aggregator (a global ISP identified by the TLA ID) and a customer site.

For a global address allocated to a top-level aggregator, the first 16 bits of the address are fixed and correspond to the FP (set to 001) and the TLA ID (13 bits in length). The TLA ID is followed by the Res portion, which consists of 8 reserved bits set to 0. Therefore, for subnetting of the NLA ID portion of a global address, the first 24 bits are fixed. In a global address, the Res bits are never shown due to the suppression of leading zeros in IPv6 colon hexadecimal notation.

Subnetting the NLA ID portion of a global address requires a two-step procedure:

1. Determine the number of bits to be used for the subnetting.
2. Enumerate the new subnetted network prefixes.

The subnetting technique described here assumes that subnetting is done by dividing the 24-bit address space of the NLA ID using the high-order bits in the NLA ID that do not already have fixed values. While this method promotes hierarchical addressing and routing, it is not required. For example, you can also create a flat addressing space for the NLA ID by numbering the subnets from 0 to 16,777,215.

Step 1: Determining the Number of Subnetting Bits

The number of bits being used for subnetting determines the possible number of new subnetted network prefixes that can be allocated to portions of your network based on geographical, customer segment, or other divisions. In a hierarchical routing infrastructure, you need to determine how many network 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. The last level in the hierarchy is used to assign 48-bit prefixes to customer sites.

For example, a network designer at a large ISP decides to implement a two-level hierarchy reflecting a geographical/customer segment structure and uses 8 bits for the geographical level and 8 bits for the customer segment level.

This means that each customer segment in each geographical location has only 8 bits of subnetting space left (24 – 8 – 8), or only 256 (= 2^8) 48-bit prefixes per customer segment.

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, $f + s + r = 24$. This relationship is shown in Figure 3-9.

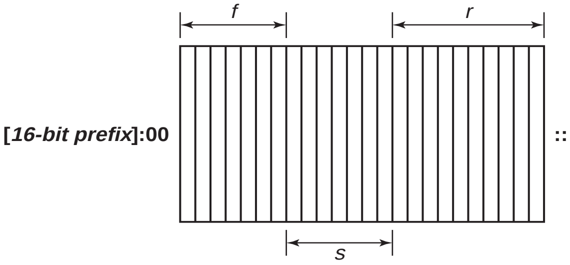

Figure 3-9. *The subnetting of an NLA ID*

Step 2: Enumerating Subnetted Network Prefixes

Based on the number of bits used for subnetting, you must list the new subnetted network prefixes. There are two main approaches:

- Hexadecimal — Enumerate new subnetted network prefixes by using hexadecimal representations of the NLA ID and increment.
- Decimal — Enumerate new subnetted network prefixes by using decimal representations of the NLA ID and increment. The decimal subnetting technique is included here for those who are more comfortable dealing with decimal numbers (Base_{10}).

Either method produces the same result: an enumerated list of subnetted network prefixes.

Creating the enumerated list of subnetted network prefixes by using the hexadecimal method

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

   $$f = m - 24$$

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

$$n = 2^s$$

n is the number of network prefixes that are obtained.

$$i = 2^{24-(f+s)}$$

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

$$l = 24 + f + s$$

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

2. Create a three-column table with n entries. The first column is the network prefix number (starting with 1), the second column is the value of F (the hexadecimal representation of the NLA ID), and the third column is the new subnetted network prefix.
3. In the first table entry, the entry for the NLA ID column is F and the subnetted network prefix is the original network prefix with the new prefix length. To obtain F, combine the last two hexadecimal digits of the second hexadecimal block with the four hexadecimal digits of the third hexadecimal block of the NLA ID being subnetted to form a 6-digit hexadecimal number. Remember to include zeros that may not be present due to leading zero suppression. For example, for the global address prefix 3000:4D:C00::/38, F is 0x4D0C00.
4. In the next table entry, for the NLA ID column, increase the value of F by i. For example, in the second table entry, the NLA ID is $F + i$.
5. For the subnetted network prefix column, convert the NLA ID into two separate 16-bit blocks in colon hexadecimal notation and place them after the 16-bit prefix to express the new subnetted network prefix. For example, for the second table entry, the subnetted network prefix is [*16-bit prefix*]:[$F + i$ (expressed in colon hexadecimal notation)]::/l.
6. Repeat steps 4 and 5 until the table is complete.

For example, to perform a 3-bit subnetting of the global network prefix 3000:4D:C00::/38, we first calculate the values of the number of prefixes, the increment, and the new prefix length. Our starting values are $F = \text{0x4D0C00}$, $s = 3$, and $f = 38 - 24 = 14$. The number of prefixes is 8 ($n = 2^3$). The increment is 0x80 ($i = 2^{24-(14+3)} = 128 = \text{0x80}$). The new prefix length is 41 ($l = 38 + 3$).

Next, we construct a table with 8 entries. The subnetted network prefix for network prefix 1 is 3000:4D:C00::/41. Additional entries in the table are successive increments of *i* in the NLA ID portion of the network prefix, as shown in Table 3-4.

Table 3-4. THE HEXADECIMAL SUBNETTING TECHNIQUE FOR NETWORK PREFIX 3000:4D:C00::/38

Network Prefix Number	*NLA ID (hexadecimal)*	*Subnetted Network Prefix*
1	4D0C00	3000:4D:C00::/41
2	4D0C80	3000:4D:C80::/41
3	4D0D00	3000:4D:D00::/41
4	4D0D80	3000:4D:D80::/41
5	4D0E00	3000:4D:E00::/41
6	4D0E80	3000:4D:E80::/41
7	4D0F00	3000:4D:F00::/41
8	4D0F80	3000:4D:F80::/41

NOTE:
RFC 2373 allows the use of subnetted network prefixes where the bits being used for subnetting are set to all zeros (the all-zeros subnetted network prefix) and all ones (the all-ones subnetted network prefix) for any portion of the IPv6 network prefix being subnetted.

Creating the enumerated list of subnetted network prefixes using the decimal method

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

   $$f = m - 24$$

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

   $$n = 2^s$$

   n is the number of network prefixes that are obtained.

   $$i = 2^{24-(f+s)}$$

   i is the incremental value between each successive NLA ID expressed in decimal form.

$l = 24 + f + s$

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

D = decimal representation of F

2. Create a four-column table with n entries. The first column is the network prefix number (starting with 1), the second column is the decimal representation of the NLA ID portion of the new subnetted network prefix, the third column is the hexadecimal representation of the NLA ID portion of the new subnetted network prefix, and the fourth column is the new subnetted network prefix.
3. In the first table entry, the decimal representation of the NLA ID is D, the hexadecimal representation of the NLA ID is F, and the subnetted network prefix is the original network prefix with the new prefix length.
4. In the next table entry, for the second column, increase the value of the decimal representation of the NLA 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 NLA ID to hexadecimal.
6. For the fourth column, convert the NLA ID into two separate 16-bit blocks in colon hexadecimal notation and place them after the 16-bit prefix to express the new subnetted network prefix. For example, for the second table entry, the subnetted network prefix is [*16-bit prefix*]:[$F + i$ (expressed in colon hexadecimal notation)]::/l.
7. Repeat steps 4 through 6 until the table is complete.

For example, to perform a 3-bit subnetting of the global network prefix 3000:4D:C00::/38, we first calculate the values of the number of prefixes, the increment, and the new prefix length. Our starting values are F = 0x4D0C00, $s = 3$, and $f = 38 - 24 = 14$. The number of prefixes is 8 ($n = 2^3$). The increment is 128 ($i = 2^{24-(14+3)} = 128$). The new prefix length is 41 ($l = 38 + 3$). The decimal representation of the starting NLA ID is 5049344 (D = 0x4D0C00 = 5049344).

Next, we construct a table with 8 entries. The subnetted network prefix for network prefix 1 is 3000:4D:C00::/41. Additional entries in the table are successive increments of i in the NLA ID portion of the network prefix, as shown in Table 3-5.

Table 3-5. **THE DECIMAL SUBNETTING TECHNIQUE FOR NETWORK PREFIX 3000:4D:C00::/38**

Network Prefix Number	*Decimal Representation of NLA ID*	*Hexadecimal Representation of NLA ID*	*Subnetted Network Prefix*
1	5049344	4D0C00	3000:4D:C00::/41
2	5049472	4D0C80	3000:4D:C80::/41
3	5049600	4D0D00	3000:4D:D00::/41
4	5049728	4D0D80	3000:4D:D80::/41
5	5049856	4D0E00	3000:4D:E00::/41
6	5049984	4D0E80	3000:4D:E80::/41
7	5050112	4D0F00	3000:4D:F00::/41
8	5050240	4D0F80	3000:4D:F80::/41

Subnetting for SLA IDs/Subnet IDs

For most network administrators within an organization, subnetting the IPv6 address space consists of using subnetting techniques to divide the SLA ID portion of the global address or the Subnet ID portion of the site-local address in a manner that allows for route summarization and delegation of the remaining address space to different portions of an IPv6 intranet. The global address has a 16-bit SLA ID field to be used by organizations within their sites. The site-local address has a 16-bit Subnet ID field to be used by organizations within a site.

In both cases, the first 48 bits of the address are fixed. For the global address, the first 48 bits are fixed and allocated by an ISP and correspond to the TLA and NLA ID portions of the global address. For the site-local address, the first 48 bits are fixed at FEC0::/48. In the discussion that follows, the term subnet ID refers to either the SLA ID portion of the global address or the Subnet ID portion of a site-local address.

Subnetting the subnet ID portion of a global or site-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 network 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. While 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.

As described in the “Local-Use Unicast Addresses” section of this chapter, you can use the same subnetting scheme and use the same subnet ID for both site-local and global address network prefixes.

Step 1: Determining the Number of Subnetting Bits

The number of bits being used for subnetting determines the possible number of new subnetted network 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 network 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.

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 – 6 – 4), 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.

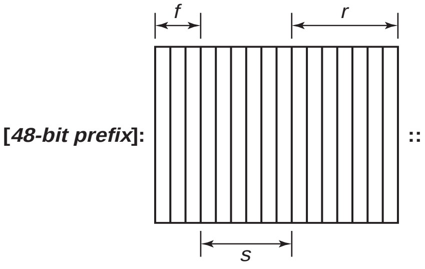

Figure 3-10. *The subnetting of a Subnet ID*

Step 2: Enumerating Subnetted Network Prefixes

Based on the number of bits used for subnetting, you must list the new subnetted network prefixes. There are two main approaches:

- Hexadecimal — Enumerate new subnetted network prefixes by using hexadecimal representations of the subnet ID and increment.
- Decimal — Enumerate new subnetted network prefixes by using decimal representations of the subnet ID and increment.

Either method produces the same result: an enumerated list of subnetted network prefixes.

Creating the enumerated list of subnetted network prefixes using the hexadecimal method

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

   $$f = m - 48$$

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

   $$n = 2^s$$

   n is the number of network prefixes that are obtained.

   $$i = 2^{16-(f+s)}$$

   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 network prefixes.

2. Create a two-column table with n entries. The first column is the network prefix number (starting with 1) and the second column is the new subnetted network prefix.
3. In the first table entry, based on F, the hexadecimal value of the subnet ID being subnetted, the subnetted network prefix is [*48-bit prefix*]:*F*::/*l*.
4. In the next table entry, increase the value within the subnet ID portion of the site-local or global address by i. For example, in the second table entry, the subnetted prefix is [*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 site-local network prefix FEC0:0:0: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 $f = 51 - 48 = 3$. The number of prefixes is 8 ($n = 2^3$). The increment is 0x400 ($i = 2^{16-(3+3)} = 1024 = 0x400$). The new prefix length is 54 ($l = 48 + 3 + 3$).

Next, we construct a table with 8 entries. The entry for the network prefix 1 is FEC0:0:0:C000::/54. Additional entries in the table are successive increments of i in the subnet ID portion of the network prefix, as shown in Table 3-6.

Table 3-6. **THE HEXADECIMAL SUBNETTING TECHNIQUE FOR NETWORK PREFIX FEC0:0:0:C000::/51**

Network Prefix Number	*Subnetted Network Prefix*
1	FEC0:0:0:C000::/54
2	FEC0:0:0:C400::/54
3	FEC0:0:0:C800::/54
4	FEC0:0:0:CC00::/54
5	FEC0:0:0:D000::/54
6	FEC0:0:0:D400::/54
7	FEC0:0:0:D800::/54
8	FEC0:0:0:DC00::/54

Creating the enumerated list of subnetted network prefixes using the decimal method

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

   $$f = m - 48$$

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

   $$n = 2^s$$

   n is the number of network prefixes that are obtained.

   $$i = 2^{16-(f+s)}$$

   i is the incremental value between each successive subnet ID.

   $$l = 48 + f + s$$

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

   D = decimal representation of F

2. Create a three-column table with n entries. The first column is the network prefix number (starting with 1), the second column is the decimal representation of the subnet ID portion of the new network prefix, and the third column is the new subnetted network prefix.
3. In the first table entry, the decimal representation of the subnet ID is D and the subnetted network 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 network 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 site-local network prefix FEC0:0:0: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 $f = 51 - 48 = 3$. The number of prefixes is 8 ($n = 2^3$). The increment is 1024 ($i = 2^{16-(3+3)}$). 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 network prefix 1 is 49152 and FEC0:0:0:C000::/54. Additional entries in the table are successive increments of *i* in the subnet ID portion of the network prefix, as shown in Table 3-7.

Table 3-7. THE DECIMAL SUBNETTING TECHNIQUE FOR NETWORK PREFIX FEC0:0:0:C000::/51

Network Prefix Number	*Decimal Representation of Subnet ID*	*Subnetted Network Prefix*
1	49152	FEC0:0:0:C000::/54
2	50176	FEC0:0:0:C400::/54
3	51200	FEC0:0:0:C800::/54
4	52224	FEC0:0:0:CC00::/54
5	53248	FEC0:0:0:D000::/54
6	54272	FEC0:0:0:D400::/54
7	55296	FEC0:0:0:D800::/54
8	56320	FEC0:0:0:DC00::/54

IPV6 INTERFACE IDENTIFIERS

The last 64 bits of a currently defined IPv6 unicast address are the interface identifier that is unique to the 64-bit prefix of the 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 $2^8 - 2$ 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 most 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 is determined are the following:

- As defined in RFC 2373, all unicast addresses that use the prefixes 001 through 111 must also use a 64-bit interface identifier that is derived from the Extended Unique Identifier (EUI)-64 address. The 64-bit EUI-64 address is defined by the Institute of Electrical and Electronic Engineers (IEEE). EUI-64 addresses are either assigned to a network adapter or derived from IEEE 802 addresses.
- As defined in RFC 3041, it might have a temporarily assigned, randomly generated interface identifier to provide a level of anonymity.
- It is assigned during stateful address autoconfiguration (for example, via Dynamic Host Configuration Protocol version 6 (DHCPv6)). Stateful address autoconfiguration standards and protocols are in progress.
- As defined in RFC 2472, an interface identifier can be based on link-layer addresses or serial numbers, or 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.

EUI-64 Address-based Interface Identifiers

The most common 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, Token Ring, and Fiber Data Distributed Interface (FDDI) 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.

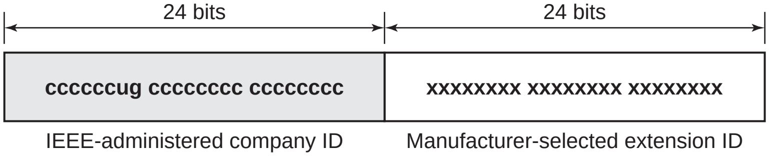
24 bits
24 bits
cccccccug cccccccc cccccccc
xxxxxxxx xxxxxxxx xxxxxxxx
IEEE-administered company ID
Manufacturer-selected extension ID

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

Defined bits within the IEEE 802 address for Ethernet are:

Universal/Local (U/L) — The next-to the low-order bit in the first byte is usedto 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 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 802.x network adapter address, 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.

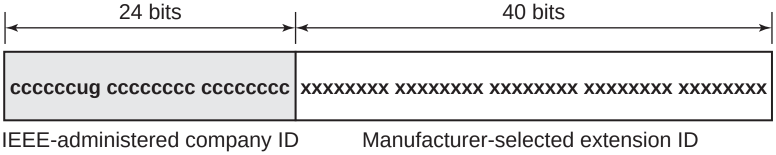
24 bits
40 bits
cccccccug cccccccc cccccccc
xxxxxxxx xxxxxxxx xxxxxxxx xxxxxxxx xxxxxxxx
IEEE-administered company ID
Manufacturer-selected extension ID

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.

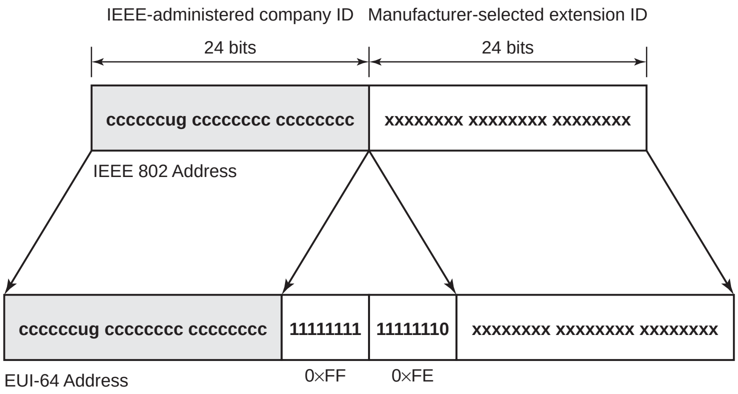
IEEE-administered company ID
Manufacturer-selected extension ID
24 bits
24 bits
ccccccug cccccccc cccccccc
xxxxxxxx xxxxxxxx xxxxxxxx
IEEE 802 Address
ccccccug cccccccc cccccccc
11111111
11111110
xxxxxxxx xxxxxxxx xxxxxxxx
EUI-64 Address
0×FF
0×FE

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 may assign one interface on the link the locally administered EUI-64 address of 02-00-00-00-00-00-00-01 and 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.

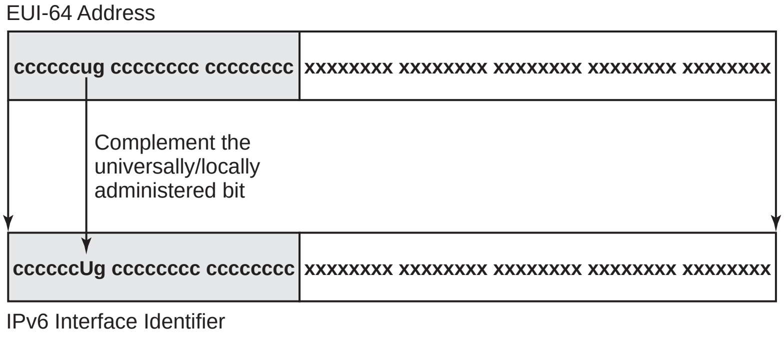
EUI-64 Address
ccccccug cccccccc cccccccc
xxxxxxxx xxxxxxxx xxxxxxxx xxxxxxxx xxxxxxxx
Complement the universally/locally administered bit
ccccccUg cccccccc cccccccc
xxxxxxxx xxxxxxxx xxxxxxxx xxxxxxxx xxxxxxxx
IPv6 Interface Identifier

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.

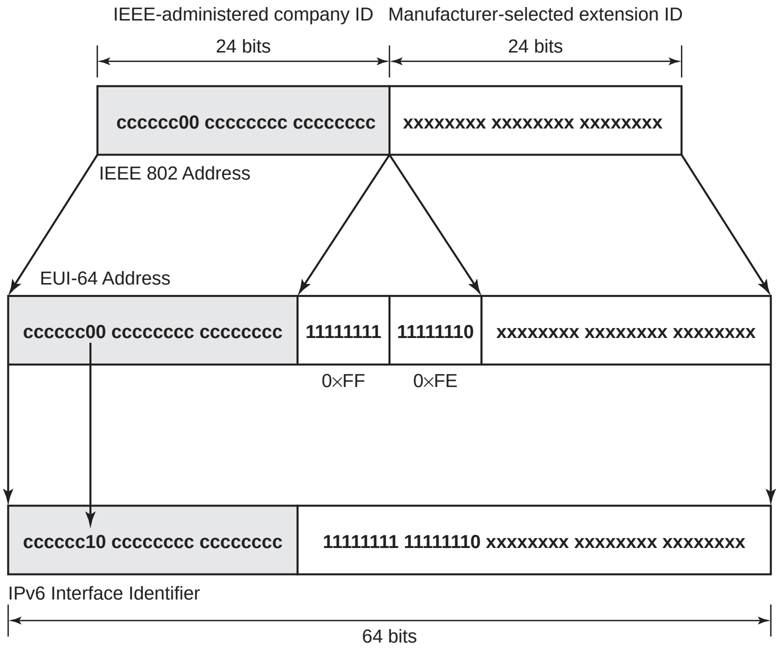
IEEE-administered company ID
Manufacturer-selected extension ID
24 bits
24 bits
cccccc00 cccccccc cccccccc
xxxxxxxx xxxxxxxx xxxxxxxx
IEEE 802 Address
EUI-64 Address
cccccc00 cccccccc cccccccc
11111111
11111110
xxxxxxxx xxxxxxxx xxxxxxxx
0×FF
0×FE
cccccc10 cccccccc cccccccc
11111111 11111110 xxxxxxxx xxxxxxxx xxxxxxxx
IPv6 Interface Identifier
64 bits

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 IP address.

For IPv6-based dial-up connections, the user is assigned a 64-bit prefix, at the time of connection, by using router discovery, 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 they are accessing the Internet from home or from work. This makes it easy for Internet merchants and malicious users to track a specific user and their use of the Internet.

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

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, "Address Autoconfiguration."

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 of the IPv6 multicast address. Figure 3-16 shows the mapping of an IPv6 multicast address to an Ethernet multicast address.

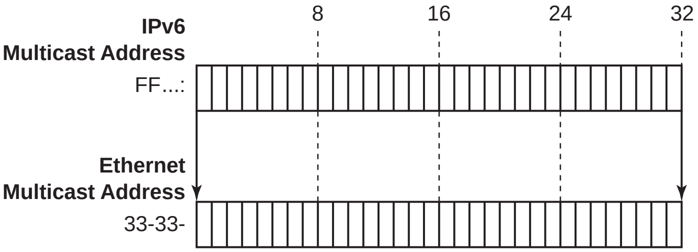

Figure 3-16. *The mapping of IPv6 multicast addresses to Ethernet multicast addresses*

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.
- 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.

IPV4 ADDRESSES AND IPV6 EQUIVALENTS

To summarize the relationships between IPv4 addressing and IPv6 addressing, Table 3-8 lists both IPv4 addresses and addressing concepts and their IPv6 equivalents.

Table 3-8. 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

continued

Table 3-8 *continued*

IPv4 Address	*IPv6 Address*
Public IP addresses	Aggregatable global unicast addresses
Private IP addresses (10.0.0.0/8, 172.16.0.0/12, and 192.168.0.0/16)	Site-local addresses (FEC0::/48)
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. IPv4-compatible addresses are expressed in dotted decimal notation.
Network bits representation: Subnet mask in dotted decimal notation or prefix length	Network bits representation: Prefix length notation only

REFERENCES

RFC 1888 — “OSI NSAPs and IPv6”

RFC 2373 — “IP Version 6 Addressing Architecture”

RFC 2472 — “IP Version 6 over PPP”

RFC 3041 — “Privacy Extensions for Stateless Address Autoconfiguration in IPv6”

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. Define the Format Prefixes (FPs) for commonly used unicast addresses.

3. Express FEC0:0000:0000:0001:02AA:0000:0000:0007A more efficiently.

4. How many bits are expressed by "::" in the addresses 3341::1:2AA:9FF:FE56:24DC and FF02::2?

5. Describe the difference between unicast, multicast, and anycast addresses in terms of a host sending packets to zero or more interfaces.

6. Why are no broadcast addresses defined for IPv6?

7. Define the structure, including field sizes, of the aggregatable global unicast address.

8. Define the scope for each of the different types of typically used unicast addresses.

9. Explain how global and site-local addressing can share the same subnetting infrastructure within an organization.

10. Define the structure, including field sizes, of the multicast address.

11. Why does RFC 2373 recommend using only the last 32 bits of the IPv6 multicast address for the multicast group ID?

12. Explain how the solicited-node multicast address acts as a pseudo-unicast address.

13. How do routers know the nearest location of an anycast group member?

14. Perform a 4-bit subnetting on the site-local prefix FEC0:0:0:3D80::/57.

15. What is the 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?

16. 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?

17. What is the site-local scope multicast address corresponding to the Ethernet multicast MAC address of 33-33-00-0A-4F-11?

18. For each type of address, identify how the address begins in colon hexadecimal notation.

Type of Address	***Begins with…***
Link-local unicast address	FE80
Site-local unicast address	
Global address	
Multicast address	
Link-local scope multicast address	
Site-local scope multicast address	
Solicited-node multicast address	
IPv4-compatible address	
IPv4-mapped address	
6to4 address	

Chapter 12

IPv6 Mobility

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

- Describe how IPv6 mobility provides application transparency for communication to and from mobile nodes when they are away from home.
- List and describe the components of IPv6 mobility.
- List and describe the IPv6 mobility messages and options.
- Describe the information contained in the IPv6 mobility data structures.
- Describe the details of the communication between the mobile node and the correspondent node.
- Describe the details of the communication between the mobile node and the home agent.
- Describe the details of the following IPv6 mobility processes: attaching to the home link, moving from a home link to a foreign link, moving to a different foreign link, and returning home.
- Describe how IPv6 mobility changes the host sending and receiving algorithms.

NOTE:
The discussion of IPv6 mobility in this chapter is based on draft 13 of the Internet draft titled "Mobility Support in IPv6" (draft-ietf-mobileip-ipv6-13.txt in the \RFCs_and_Drafts folder on the companion CD-ROM).

IPV6 MOBILITY OVERVIEW

IPv6 mobility allows an IPv6 node to be mobile—to arbitrarily change its location on the IPv6 Internet—and still maintain existing connections. When an IPv6 node changes its location, it might also change its link. When an IPv6 node changes its link, its IPv6 address must also change in order to maintain reachability. There are mechanisms to allow for the change in addresses when moving to a different link, such as stateful and stateless address autoconfiguration for IPv6. However, these mechanisms ungracefully terminate all the existing connections of the mobile node that are using the address assigned when it was on the previous link.

The key benefit of IPv6 mobility is that, even though the mobile node is changing locations and addresses, the existing connections through which the mobile node is communicating are maintained. Connection maintenance for mobile nodes is not done by modifying connection-oriented protocols such as TCP, but by handling the change of addresses at the Internet layer. Transport-layer protocols are completely unaware that the address of the mobile node has changed. A connection is established with a specific address assigned to the mobile node and remains connected no matter how many times the mobile node changes its location and address.

IPv6 Mobility Components

IPv6 mobility consists of the following components, as shown in Figure 12-1.

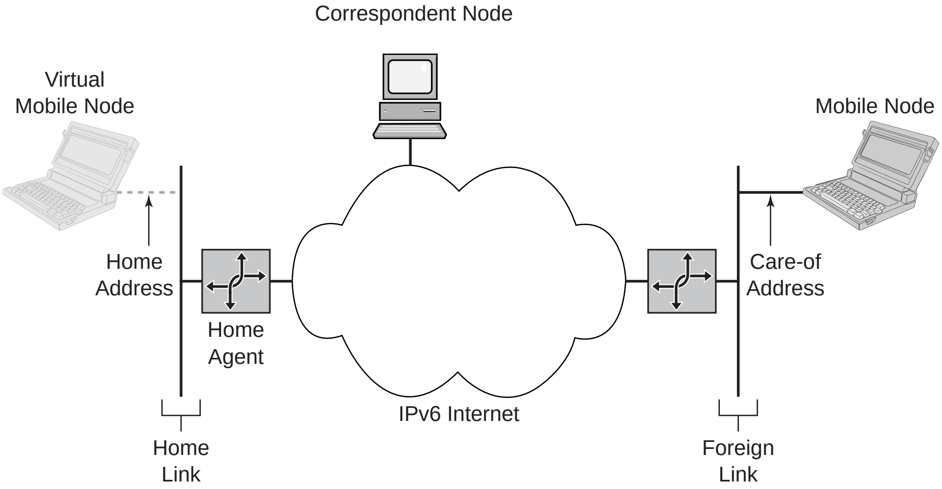

Figure 12-1. *Components of IPv6 mobility*

- Home link
  The home link is the link that is assigned the home subnet prefix. The mobile node uses the home subnet prefix to create a home address.
- Home address
  A home address is an address assigned to the mobile node when it is attached to the home link and through which the mobile node is

always reachable, regardless of its location on the IPv6 Internet. Packets addressed to addresses matching the home subnet prefix are delivered to the home link using normal IPv6 routing processes. If the mobile node is attached to the home link, IPv6 mobility processes are not used and communication occurs normally. If the mobile node is away from home (not attached to the home link), IPv6 mobility processes are used to either deliver or tunnel traffic addressed to the mobile node's home address to its current location on the IPv6 Internet. Because the mobile node is always assigned the home address, it always has a virtual connection to the home link. This relationship is shown in Figure 12-1 as the Virtual Mobile Node.

- Home agent
  The home agent is a router on the home link that maintains an awareness of the mobile nodes of its home link that are away from home and the addresses that they are currently using. If the mobile node is on the home link, the home agent acts as an IPv6 router, forwarding packets addressed to the mobile node. If the mobile node is away from home, the home agent tunnels data sent to the mobile node's home address to the mobile node's current location on the IPv6 Internet.
- Mobile node
  A mobile node is an IPv6 node that can change links, and therefore addresses, and maintain reachability using its home address. A mobile node has awareness of its home address and the global address of its current link address, and indicates its home address/current link address mapping to the home agent and IPv6 nodes with which it is communicating.
- Foreign link
  A foreign link is a link that is not the mobile node's home link. A foreign link is assigned a foreign subnet prefix.
- Care-of address
  A care-of address is an address used by a mobile node while it is attached to a foreign link. The care-of address is a combination of the foreign subnet prefix and an interface ID determined by the mobile node. A mobile node can be assigned multiple care-of addresses; however, only one care-of address is registered as the primary care-of address with the mobile node's home agent. The association of a care-of address with a home address for a mobile node is known as a *binding*. Correspondent nodes and home agents keep information on bindings in a binding cache.

- Correspondent node
  A correspondent node is an IPv6 node that is capable of communicating with a mobile node while it is away from home. A correspondent node can also be a mobile node.

NOTE:
The drawings in this chapter assume that the common IPv6 infrastructure is the IPv6 Internet. However, all of the IPv6 mobility components, messages, and processes also work if the IPv6 nodes are separated by an IPv4 infrastructure (such as the Internet) and are using a coexistence technology (such as 6to4) to achieve IPv6 connectivity.

To achieve Application layer transparency for the home address while the mobile node is assigned a care-of address, the following is used:

- When the mobile node sends data to a correspondent node, the packet is sent from the care-of address and includes the mobile node's home address in a Home Address option in a Destination Options extension header. When the correspondent node receives the packet, it logically replaces the source address of the packet (the care-of address) with the home address stored in the Home Address option. The Home Address option is described later in this chapter.
- When the correspondent node sends data to the mobile node, the packet is sent to the care-of address and includes a Routing extension header containing a single address, the mobile node's home address. When the mobile node receives the packet, it processes the Routing header and logically replaces the destination address of the packet (the care-of address) with the home address from the Routing header.

For IPv6 nodes, there are the following levels of correspondent node support:

- None
  If an IPv6 node has no correspondent node support, then it will be unable to communicate with mobile nodes that are away from home. Packets sent by a mobile node that is away from home always contain a Destination Options header with the Home Address option. The Home Address option uses the Option Type of 201. For this option type, the two high-order bits are set to 11 (binary), which means that if the receiving host does not recognize the option, then

it sends back an ICMPv6 Parameter Problem-Unrecognized IPv6 Option Encountered message to the sender and discards the packet. Therefore, a node that has no correspondent node support does not recognize the Home Address option and never receives the packets sent by a mobile node that is away from home.

- Minimal
  An IPv6 node that has minimal correspondent node support recognizes the Home Address option in the Destination Options header of received packets. However, packets sent by the correspondent node to the mobile node that is away from home are always intercepted by the home agent, who then tunnels the packets to the mobile node. The result is inefficient communication for packets sent by the correspondent node to the mobile node.
- Full
  An IPv6 node that has full correspondent node support recognizes the Home Address and Binding Update options in the Destination Options header, sends the Binding Acknowledgement and Binding Request options as needed, includes a Routing header in packets sent to mobile nodes that are away from home, and maintains a binding cache that maps the home address to the care-of address of mobile nodes that are away from home. The Binding Update, Binding Acknowledgment, and Binding Request options are discussed later in this chapter.

The IPv6 protocol for Windows XP and the Windows .NET Server 2003 family supports full correspondent node functionality, with the exception of sending binding requests. The IPv6 protocol for Windows XP and the Windows .NET Server 2003 family does not support mobile node or home agent functionality.

IPV6 MOBILITY MESSAGES AND OPTIONS

The following options, messages, and modifications to existing messages are needed to facilitate the processes of IPv6 mobility.

Destination Options Header Options

The Destination Options extension header is used to contain the following IPv6 mobility-related options:

- Binding Update
- Binding Acknowledgement

- Binding Request
- Home Address

For information on the structure of these options, see Chapter 4, "The IPv6 Header."

Binding Update Option

The Binding update option is used by a mobile IPv6 node that is away from home to update another node with its new care-of address. The Binding Update option is an option used within the Destination Options extension header for the destination node and can be included in an existing packet sent to the destination or in a packet that contains just the Destination Options header. In this latter case, the Next Header field in the Destination Options header is set to 59, indicating no next header. A binding update is a packet that contains the Binding Update option. A binding update always contains the Home Address option (described later in this chapter).

The Binding Update option is used for the following:

- To update the home agent with a new primary care-of address. This is known as a home registration binding update. The home agent uses the home address in the Home Address option and the source address of the packet to update its Home Address/Primary Care-of Address binding cache entry for the mobile node.
- To update a correspondent node with which the mobile node is actively communicating with a new binding that maps the home address of the mobile node to its care-of address. The correspondent node uses the home address in the Home Address option and the source address of the packet to update its Home Address/Care-of Address binding cache entry for the mobile node.

The mobile node can use the Alternate Care-of Address sub-option in the Binding Update option to specify a care-of address that is different than the source address of the binding update.

Binding Acknowledgement Option

The Binding Acknowledgement option is used to acknowledge the receipt of a binding update whose Acknowledge (A) flag has been set, and to report errors in the binding update. The mobile node sets the A flag when it wants to receive confirmation that the Binding Update message was received. The Binding Acknowledgement option can be included in an existing packet sent to the destination, or in a packet that contains just the Destination Options header. In this latter case, the Next Header field in the Destination Options header is set to 59, indicating no next header. A binding acknowledgement is a packet that contains the Binding Acknowledgement option.

Included in the binding acknowledgement is an indication of how long the node will cache the binding. For home agents, this lifetime indicates how long the home agent will be in service as the home agent for the mobile node. To refresh the binding, either the mobile node sends a new binding update or the correspondent nodes and home agent send a request to update the binding.

The binding acknowledgement also includes an indication of how often the mobile node should send binding updates.

Binding Request Option

A Binding Request option is used to request the current binding from a mobile node. If a mobile node receives a binding request, it responds with a binding update. A correspondent node sends a binding request when the binding cache entry is in active use and the lifetime of the binding cache entry approaches expiration. A home agent sends a binding request when the lifetime of the binding cache entry approaches expiration.

The Binding Request option can be included in an existing packet sent to the destination or in a packet that contains just the Destination Options header. In this latter case, the Next Header field in the Destination Options header is set to 59, indicating no next header. A binding request is a packet that contains the Binding Request option.

Home Address Option

The Home Address option is used to indicate the home address of the mobile node. The Home Address option is included in any packet sent to correspondent nodes and home agents by a mobile node when it is away from home (with the exception of a tunneled Router Solicitation message sent to the home agent). When a mobile node sends a packet, the source address in the IPv6 header is set to the care-of address. If the source address in the IPv6 header is set to the home address, then the router on the foreign link might discard the packet because the source address does not match the prefix of the link on which the mobile node is located. To help minimize Internet attacks in which the source address of attack packets is spoofed with an address that is not assigned to the attacking computer, peripheral routers can implement ingress filtering and silently discard packets that do not have topologically correct source addresses. Ingress in this instance is defined relative to the Internet for packets entering the Internet, rather than packets entering an intranet from the Internet.

By using the care-of address as the source address in the packet (a topologically correct address on the foreign link), and including the Home Address destination option, the packet is forwarded by the router on the foreign link to its destination. When the packet is received at the destination, the correspondent node processes the Destination Options header, and before passing the payload to the upper layer protocol, logically replaces the source address of the packet with the address in the Home Address option. As far as the upper layer protocol is concerned, the packet was sent from the home address.

The Home Address option is also included with the binding update so that the home address for the binding is indicated to the receiving node.

ICMPv6 Messages

The following ICMPv6 messages are used by the mobile node for dynamic home agent address discovery:

- Home Agent Address Discovery Request
- Home Agent Address Discovery Reply

Dynamic home agent address discovery is a process through which the mobile node dynamically discovers the global address of the home agent on the home link. This process is needed only if the mobile node is not already configured with the address of its home agent or if the current home agent becomes unavailable.

Home Agent Address Discovery Request

The ICMPv6 Home Agent Address Discovery Request message is used by a mobile node to begin dynamic home agent address discovery. This message is sent to the Mobile IPv6 Home-Agents anycast address that is described in RFC 2526. The Mobile IPv6 Home-Agents anycast address is composed of the 64-bit home subnet prefix and the interface ID of ::FEFF:FFFF:FFFF:FFFE. All home agents are configured automatically with this anycast address. The home agent that is topologically closest to the mobile node receives the request message.

Figure 12-2 shows the structure of the ICMPv6 Home Agent Address Discovery Request message.

In the Home Agent Address Discovery Request message, the Type field is set to 150 and the Code field is set to 0. Following the Checksum field is a 16-bit Identifier field. The value of the Identifier field is chosen by the sending node and copied to the Identifier field of the Home Agent Address Discovery Reply message to match a reply with its request. Following the Identifier field is an 80-bit Reserved field that is set to 0 by the sender and a 128-bit Home Address field. The Home Address field contains the home address of the mobile node.

The Home Agent Address Discovery Request message is sent with the source address in the IPv6 header set to the mobile node's care-of address. The Home Address destination option is not included.

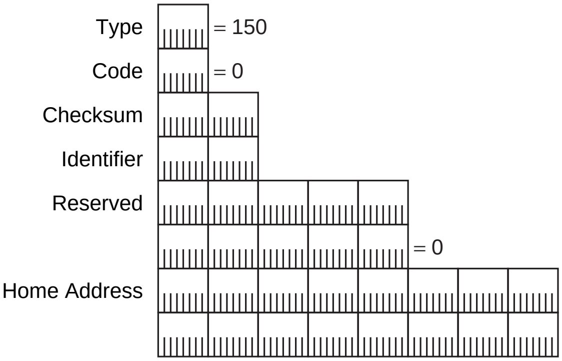

Figure 12-2. *The structure of ICMPv6 Home Agent Address Discovery Request message*

Home Agent Address Discovery Reply

The ICMPv6 Home Agent Address Discovery Reply message is used by a home agent to complete the dynamic home agent address discovery process by informing the mobile node of the addresses of the set of routers attached to the mobile node's home link that are capable of being a home agent.

Figure 12-3 shows the structure of the ICMPv6 Home Agent Address Discovery Reply message.

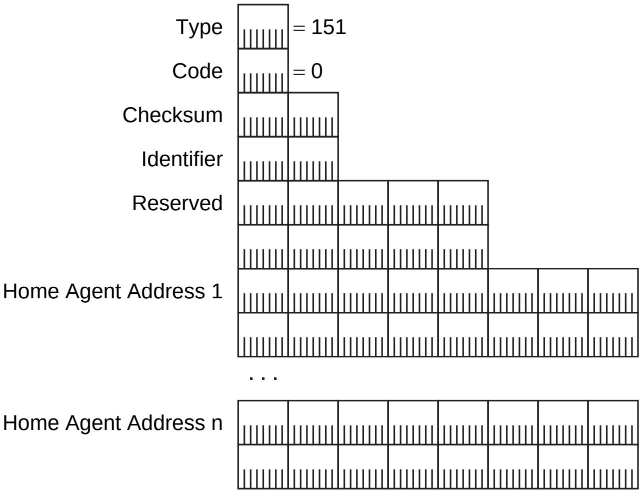

Figure 12-3. *The structure of the ICMPv6 Home Agent Address Discovery Reply message*

In the Home Agent Address Discovery Reply message, the Type field is set to 151 and the Code field is set to 0. Following the Checksum field is a 16-bit Identifier field. The value of the Identifier field is set to the same value as the Identifier field of the received Home Agent Address Discovery Request message.

Following the Identifier field is an 80-bit Reserved field that is set to 0 by the sender, and one or more 128-bit Home Agent Address fields. The Home Agent Address fields contain the global addresses of home agents on the home link in preference order (highest preference first).

The Home Agent Address Discovery Reply message is sent with the source address in the IPv6 header set to the global address of the answering home agent, and the destination address set to the mobile node's care-of address. A Routing extension header is not included.

Modifications to Neighbor Discovery Messages and Options

IPv6 mobility defines the following changes to ND messages and options:

- Modified Router Advertisement message
- Modified Prefix Information option
- New Advertisement Interval option
- New Home Agent Information option

For more information about the structure of these ND messages and options, see Chapter 6, "Neighbor Discovery."

Modifications to the Router Advertisement Message

IPv6 mobility defines an additional flag in the Router Advertisement message to help facilitate home agent discovery by the home agents and mobile nodes on a home link. The new flag, known as the Home Agent (H) flag, indicates whether the advertising router is capable of being a home agent. Each of the home agents on the home link set this flag when they send their router advertisements, and each home agent and mobile node receives each router advertisement. Therefore, each home agent and mobile node can compile the list of possible home agents.

Additionally, IPv6 mobility allows a router advertisement to be sent more frequently than every 3 seconds, as specified in RFC 2461. By sending router advertisements more frequently, IPv6 mobile nodes can use a newly received router advertisement to detect movement to a foreign link more quickly. Recommended values for the pseudo-periodic router advertisement process for routers that might provide connectivity for mobile IPv6 nodes are a minimum of 0.5 seconds and a maximum of 1.5 seconds.

Modified Prefix Information Option

To indicate the global address of the advertising router, IPv6 mobility defines an additional flag and a redefined use of the Prefix field in the Prefix Information option.

As per RFC 2461, which defines Neighbor Discovery, router advertisements are sent from the link-local address. However, the global address for a home agent must be indicated in the router advertisement it sends so that a list of home agents can be compiled by each home agent and mobile node. IPv6 mobility defines the Router Address (R) flag in the Prefix Information option. When set, the R flag indicates to the receiver that the Prefix field contains the global address of the advertising router. In the originally defined Prefix field, the high-order bits corresponding to the value of the Prefix Length field are set to the appropriate values for the advertised prefix and the bits beyond the indicated prefix length are set to 0. With this new definition, the Prefix Length field is advertised in the same way, except the Prefix field contains the entire 128-bit global address of the advertising router.

Advertisement Interval Option

The Advertisement Interval option is sent in Router Advertisement messages to specify how often the router sends unsolicited multicast router advertisements. A mobile node that receives a router advertisement with the Advertisement Interval option can use the advertisement interval to detect whether it has moved to another link.

The Advertisement Interval option contains a 32-bit field that indicates the maximum number of milliseconds between consecutive unsolicited multicast Router Advertisement messages sent by the router using the pseudo-periodic advertising scheme described in section 6.2.4 of RFC 2461.

Home Agent Information Option

The Home Agent Information option is sent in Router Advertisement messages sent by a home agent to specify the home agent's configuration. Included in the Home Agent Information option are the home agent preference (a number indicating a preference level for the advertising router to be a home agent) and the home agent lifetime (how long the home agent is acting as a home agent).

The home agents on a home link use the home agent preference values to order the list of home agents sent to the mobile node during home agent address discovery. The mobile nodes on a home link use the home agent preference values to select the home agent that has the highest preference value.

IPV6 MOBILITY DATA STRUCTURES

The following data structures are needed to facilitate the processes of IPv6 mobility:

- Binding cache
- Binding update list
- Home agents list

Binding Cache

The binding cache is a table maintained by each correspondent node and home agent and contains the current bindings for mobile nodes. Each binding cache entry contains the following information:

- The home address for the mobile node
- The care-of address for the mobile node
- The lifetime of the binding cache entry
  The lifetime is obtained from the Lifetime field of the last binding update that was received for this cache entry.
- A flag indicating whether the binding is a home registration
  This flag is set only for the binding cache entries on home agents.
- A flag indicating whether the mobile node for this binding cache entry should be advertised as a router
  If this flag is set, the home agent will advertise the mobile node as a router (by setting the Router flag) when proxying Neighbor Advertisement messages on behalf of the mobile node. This flag is valid only for home registration entries and set only for the binding cache entries on home agents.
- The value of the Prefix Length field of the last binding update that was received for this cache entry
- The maximum value of the Sequence Number field of the binding updates that have been received for this cache entry
- The time that the last binding request was sent

The actual implementation details for the binding cache are not specified, as long as the external behavior is consistent with the IPv6 mobility draft. For example, you could either maintain a separate binding cache or combine the binding cache with the destination cache. If you have a separate binding cache, you could either check it before you check the destination cache or have a pointer from the destination cache entry to the corresponding binding cache entry.

In any case, the information in the binding cache takes precedence over the information in the neighbor cache. For mobile destinations that are away from home, packets should be sent to the home address by way of the care-of address. If packets are sent directly to the home address while the mobile node is away from home, the home agent must intercept the packets and tunnel them to the mobile node, lowering the efficiency and performance of the communication between the correspondent node and the mobile node.

For the IPv6 protocol for Windows XP and the Windows .NET Server 2003 family, a separate binding cache is maintained. Each binding cache entry stores the home address, its current care-of address, and a pointer to the entry in the destination cache for the care-of address. A destination cache entry for a home address of a mobile node that is away from home has a pointer to an entry in the binding cache. The entry in the binding cache maps the home address to its care-of address and indicates the entry in the destination cache for the care-of address. The care-of address destination cache entry stores the next-hop address and interface for the care-of address. You can view the binding cache entries with the **netsh interface ipv6 show bindingcacheentries** command.

For more information about how a node sends a packet in an IPv6 mobility environment, see the "IPv6 Mobility Host Sending Algorithm" section in this chapter.

Binding Update List

The binding update list is maintained by a mobile node to record the most recent binding updates sent for the home agent and correspondent nodes. A binding update list entry contains:

- The address of the node to which the binding update was sent
- The home address for the binding update
- The care-of address sent in the last binding update
- The value of the Lifetime field in the binding update
- The remaining lifetime of the binding
  The initial value is the value of the Lifetime field in the binding update. When the lifetime expires, the entry is deleted from the binding update list.
- The maximum value of the Sequence Number field sent in previous binding updates
- The time that the last binding update was sent
- An indication of whether a retransmission is needed for binding updates sent with the Acknowledge (A) flag set and when the retransmission is to be sent
- A flag indicating that no future binding updates need to be sent
  This flag is set when the mobile node receives an ICMPv6 Parameter Problem-Unrecognized IPv6 Option Encountered message in response to a binding update.

Home Agents List

The home agents list is maintained by home agents and mobile nodes, and records information about each router from which a router advertisement was received on the home link with the Home Agent (H) flag set. Home agents maintain the home agents list so that they can send the list of home agents to a requesting mobile node away from home during home agent address discovery. Mobile nodes maintain the home agents list so that they can select a home agent.

A home agents list entry contains the following:

- The link-local address of the router on the link, obtained from the source address of the received Router Advertisement message
- The global address or addresses of the home agent, obtained from the Prefix field in the Prefix Information options in the Router Advertisement message with the Router Address (R) flag set
- The remaining lifetime of this entry
  The initial lifetime is obtained from either the Home Agent Lifetime field in the Home Agent Information option or the Router Lifetime field in the Router Advertisement message. When the lifetime expires, the entry is deleted from the home agents list.
- The preference for the home agent, obtained from the Home Agent Preference field in the Home Agent Information option
  If the router advertisement does not contain a Home Agent Information option, the preference is set to 0. Based on the definition of the Home Agent Preference field, 0 is a medium priority level. A mobile node uses the preference value to select the home agent. A home agent uses the preference value to order by preference value the list of home agents returned to a mobile node during home agent address discovery. When the mobile node receives the list of home agents, it chooses the first home agent in the list.

IPV6 MOBILITY COMMUNICATION

Before understanding the various processes used for IPv6 mobility, it is important to understand how packets containing mobility options and Application layer data are sent in a mobility-enabled environment. The following are the types of IPv6 mobility communication:

- Between a mobile node and a correspondent node
- Between a mobile node and a home agent

Communication Between a Mobile Node and a Correspondent Node

Communication between a mobile node and a correspondent node is one of the following:

- From the mobile node to the correspondent node
- From the correspondent node to the mobile node

From the Mobile Node to the Correspondent Node

The mobile node sends the correspondent node the following types of packets:

- Binding updates
- Data

Binding Updates

Binding updates sent from the mobile node to the correspondent node are shown in Figure 12-4.

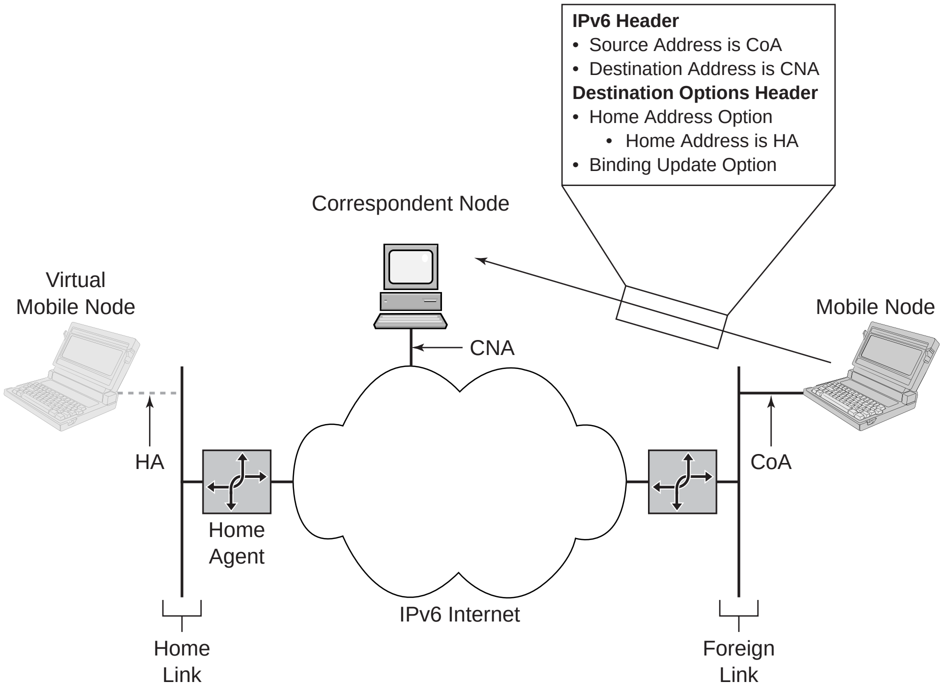

Figure 12-4. *Binding updates sent from the mobile node to the correspondent node*

The packet contains the following:

- IPv6 header
  In the IPv6 header, the source address is the care-of address of the mobile node (indicated by CoA in Figure 12-4) and the destination address is the correspondent node's address (indicated by CNA in Figure 12-4). By using the care-of address rather than the home address, ingress filtering by the foreign link router does not prevent the packet from being forwarded.
- Destination Options header
  The Destination Options extension header contains two options: the Home Address option and the Binding Update option. By including the Home Address option, the home address (indicated by HA in Figure 12-4) for the binding is indicated to the correspondent node.

The binding update can be sent either with data (an upper layer PDU) or in a separate packet. Figure 12-4 shows the binding update sent as a separate packet.

Version 13 of the IPv6 mobility draft (draft-ietf-mobileip-ipv6-13.txt in the \RFCs_and_Drafts folder on the companion CD-ROM) requires the use of an Authentication Header (AH) to provide sender authentication, data integrity, and replay protection for binding updates. Many implementations allow you to disable security for IPv6 mobility messages. If an AH is present, there are two different Destination Options headers: one before the AH and one after the AH. The first Destination Options header contains the Home Address option and the second Destination Options header contains the Binding Update option. This is not shown in Figure 12-4. If an AH is not present, both the Home Address and Binding Acknowledgement options are included in a single Destination Options header. This is shown in Figure 12-4.

If the correspondent node is also mobile, the destination address in the IPv6 header is set to the correspondent node's care-of address and the packet includes a Routing header with the correspondent node's home address. The Routing header is placed before the Destination Options header. If an AH is present, there are two different Destination Options headers: one before the AH and one after the AH. The first Destination Options header contains the Home Address option and the second Destination Options header contains the Binding Update option. In this case, the order of the extension headers is: Routing, Destination Options (w/Home Address option), AH, Destination Options (w/ Binding Update). If an AH is not present, both the Home Address and Binding Acknowledgement options are included in a single Destination Options header that is placed after the Routing header. This is not shown in Figure 12-4.

Data

When the mobile node is away from home, it can choose to either send data from its home address using mobility options, or its care-of address without using mobility options, based on the following:

- For Transport layer connection data (such as TCP sessions) that is long-term and being sent to a correspondent node, the mobile node sends the data from its home address and includes the Home Address option.
- For short-term communication that does not require a logical connection, such as the exchange of DNS Name Query and DNS Name Query Response messages for DNS name resolution, the mobile node can send data from its care-of address and not use a Home Address option. In this case, the mobile node is sending and receiving packets normally from its care-of address.

Packets containing Transport layer connection data sent from the mobile node to the correspondent node are shown in Figure 12-5.

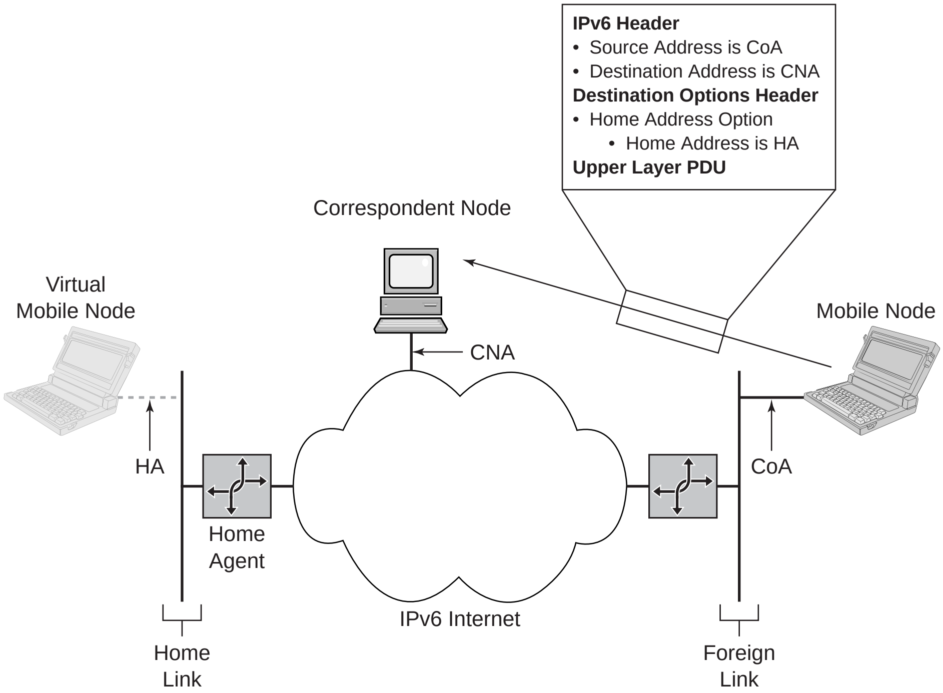

Figure 12-5. *Data sent from the mobile node to the correspondent node*

The packet contains the following:

- IPv6 header
  In the IPv6 header, the source address is the care-of address of the mobile node and the destination address is the correspondent node's address. By using the care-of address rather than the home address, ingress filtering by the foreign link router does not prevent the packet from being forwarded.
- Destination Options header
  In the Destination Options extension header, the Home Address option contains the home address of the mobile node. When the correspondent node processes the Home Address option, it indicates to upper layer protocols that the source address of the packet is the home address, rather than the care-of address.
- Upper layer PDU
  The upper layer PDU contains the Application layer data sent from the mobile node to the correspondent node. From the Application layer perspective, the data was addressed from the home address to the correspondent node address.

If the correspondent node is also mobile, the destination address in the IPv6 header is set to the correspondent node's care-of address and the packet includes a Routing header with the correspondent node's home address. The Routing header is placed before the Destination Options header. This is not shown in Figure 12-5.

From the Correspondent Node to the Mobile Node

The correspondent node sends the mobile node the following types of packets:

- Binding maintenance
- Data

Binding Maintenance

Binding maintenance packets sent from the correspondent node to the mobile node are either binding acknowledgments or binding requests and are shown in Figure 12-6.

The packets contain the following:

- IPv6 header
  In the IPv6 header, the source address is the correspondent node's address and the destination address is the mobile node's care-of address.

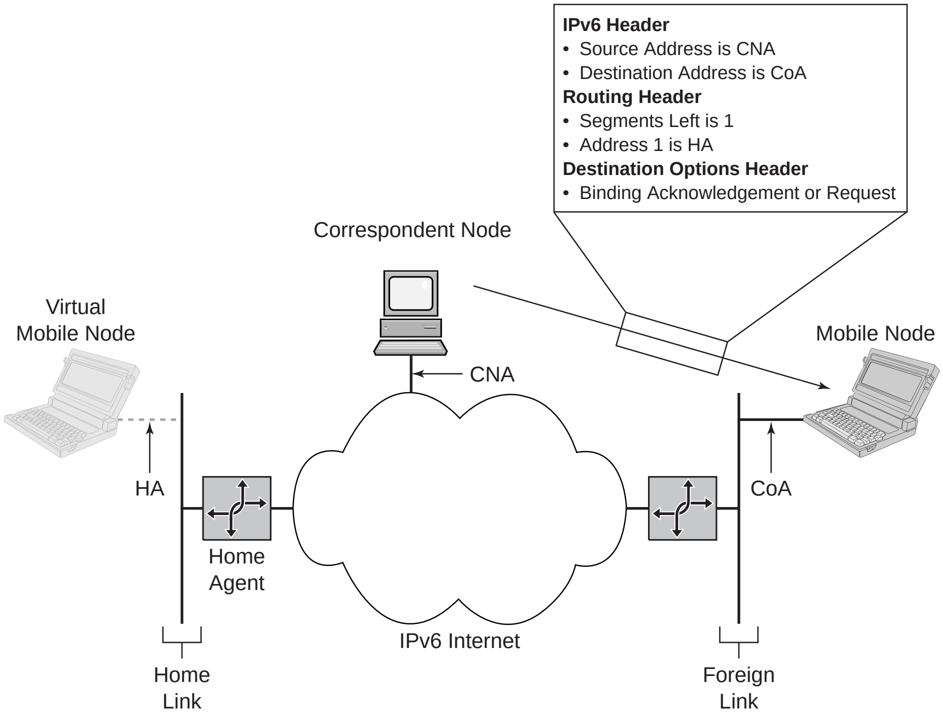

Figure 12-6. *Binding maintenance packets sent from the correspondent node to the mobile node*

- Routing header

  In the Routing extension header, the Routing Type field is set to 0, the Segments Left field is set to 1, and the Address 1 field (the final destination address of the packet) is set to the mobile node's home address. When the mobile node receives the packet, it processes the Routing header and notes that the next destination address (the address in the Address 1 field) is its own home address. The mobile node removes the Routing header and logically replaces the care-of address with the home address as the destination address in the IPv6 header. When the packet is passed to the upper layer protocol, it appears to have been addressed to the mobile node's home address.

- Destination Options header

  The Destination Options extension header contains either a Binding Acknowledgement option (if a received binding update had the Acknowledge [A] flag set) or a Binding Request option.

The binding acknowledgement or binding request can be sent either with data (an upper layer PDU) or in a separate packet. Figure 12-6 shows the binding acknowledgement or binding request sent as a separate packet.

The IPv6 mobility draft requires the use of an AH to provide data authentication, data integrity, and replay protection for binding acknowledgements. Many implementations allow you to disable security for IPv6 mobility messages. The AH is placed between the Routing header and the Destination Options header and is not shown in Figure 12-6.

If the correspondent node is also mobile, the source address in the IPv6 header is set to the correspondent node's care-of address and the packet includes the Home Address option containing the correspondent node's home address. If an AH is present, there are two different Destination Options headers: one before the AH and one after the AH. The first Destination Options header contains the Home Address option and the second Destination Options header contains the Binding Acknowledgement option. In this case, the order of the extension headers is: Routing, Destination Options (w/Home Address option), AH, Destination Options (w/Binding Acknowledgement). If an AH is not present, both the Home Address and Binding Acknowledgement options are included in a single Destination Options header that is placed after the Routing header. This is not shown in Figure 12-6.

Data with a Binding Cache Entry Present

The form of data packets sent from the correspondent node to mobile nodes depends on whether the correspondent node has a binding cache entry for the mobile node's home address. A packet containing an upper layer PDU sent from the correspondent node to the mobile node when a binding cache entry for the mobile node's care-of address is present is shown in Figure 12-7.

The packet contains the following:

- IPv6 header

  In the IPv6 header, the source address is the correspondent node's address and the destination address is the mobile node's care-of address. By using the care-of address rather than the home address, the packet is delivered to the mobile node's current location on the IPv6 Internet.

- Routing header

  In the Routing extension header, the Routing Type field is set to 0, the Segments Left field is set to 1, and the Address 1 field (the final destination address of the packet) is set to the mobile node's home address. When the mobile node receives the packet, it processes the Routing header and notes that the next destination address (the

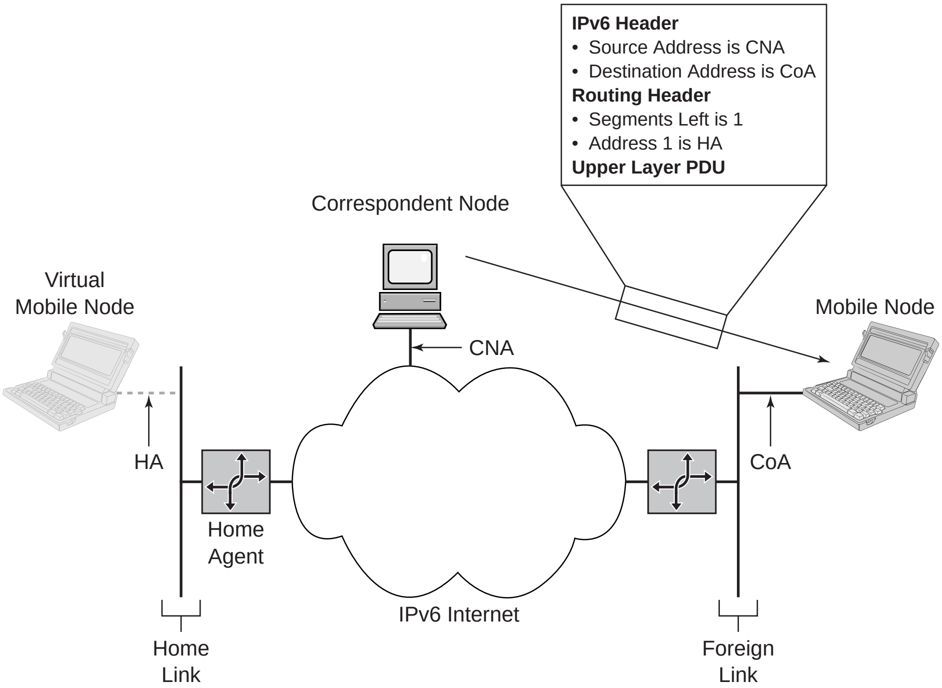

Figure 12-7. *Data sent from the correspondent node when a binding cache entry for the mobile node is present*

address in the Address 1 field) is its own home address. The mobile node removes the Routing header and logically replaces the care-of address with the home address as the destination address in the IPv6 header. When the packet is passed to the upper layer protocol, it appears to have been addressed to the mobile node's home address.

- Upper layer PDU
  The upper layer PDU contains the Application layer data sent from the correspondent node to the mobile node. From the Application layer perspective, the data was addressed from the correspondent node address to the home address.

If the correspondent node is also mobile, the source address in the IPv6 header is set to the correspondent node's care-of address and the packet includes a Destination Options header with the Home Address option containing the correspondent node's home address. The Destination Options header is placed after the Routing header. This is not shown in Figure 12-7.

Data with a Binding Cache Entry Not Present

A packet containing an upper layer PDU sent from the correspondent node to the mobile node when a binding cache entry for the mobile node is not present is shown in Figure 12-8.

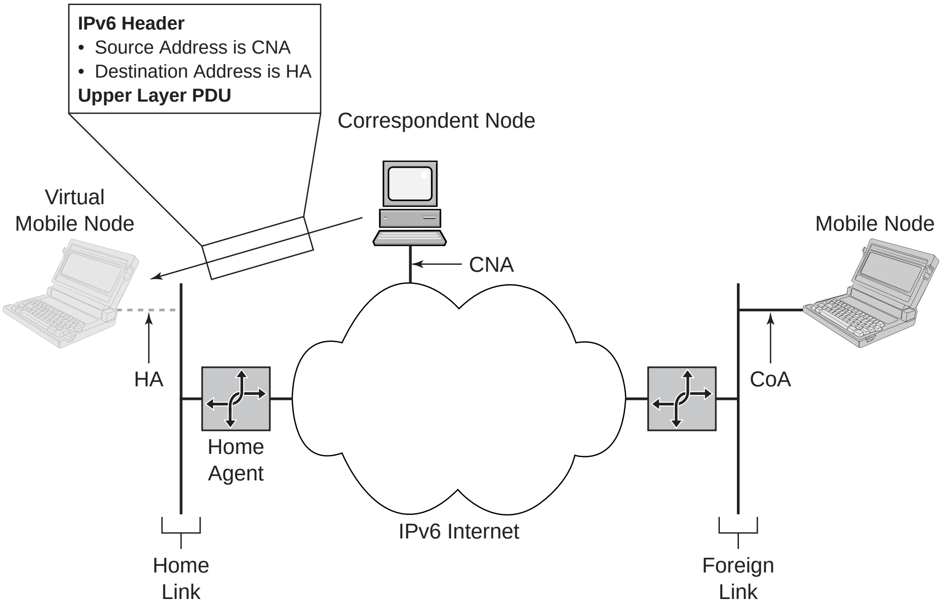

Figure 12-8. *Data sent from the correspondent node when a binding cache entry for the mobile node is not present*

The packet contains the following:

- IPv6 header
  In the IPv6 header, the source address is the correspondent node address and the destination address is the mobile node's home address. Because a binding cache entry does not exist, the correspondent node sends the packet as if the mobile node were physically attached to the home link.
- Upper layer PDU
  The upper layer PDU contains the Application layer data sent from the correspondent node to the mobile node.

While addressed to the mobile node's home address (represented by the Virtual Mobile Node in Figure 12-8), the home agent, which has a binding cache entry for the mobile node, intercepts the packet and forwards it to the mobile node by encapsulating the

IPv6 packet with an IPv6 header. This is known as IPv6-over-IPv6 tunneling. For more information, see the "Communication Between a Mobile Node and Its Home Agent" section in this chapter.

If the correspondent node is also mobile, the source address in the IPv6 header is set to the correspondent node's care-of address and the packet includes a Destination Options header with the Home Address option containing the correspondent node's home address. The Destination Options header is placed after the IPv6 header. This is not shown in Figure 12-8.

Communication Between a Mobile Node and Its Home Agent

Communication between a mobile node and a home agent is one of the following:

- From the mobile node to its home agent
- From the home agent to the mobile node

From the Mobile Node to its Home Agent

The mobile node sends the home agent the following types of packets:

- Binding update
- ICMPv6 Home Agent Address Discovery Request message

Binding Update

Binding updates sent from the mobile node to the home agent are shown in Figure 12-9.

The binding update contains the following:

- IPv6 header

  In the IPv6 header, the source address is the mobile node's care-of address and the destination address is the home agent's address (indicated by HAA in Figure 12-9). By using the care-of address rather than the home address, ingress filtering by the foreign link router does not prevent the packet from being forwarded.
- Destination Options header

  The Destination Options extension header contains two options: the Home Address option and the Binding Update option.

  The Home Address option contains the home address of the mobile node. By including the Home Address option, the home address for the binding is indicated to the home agent.

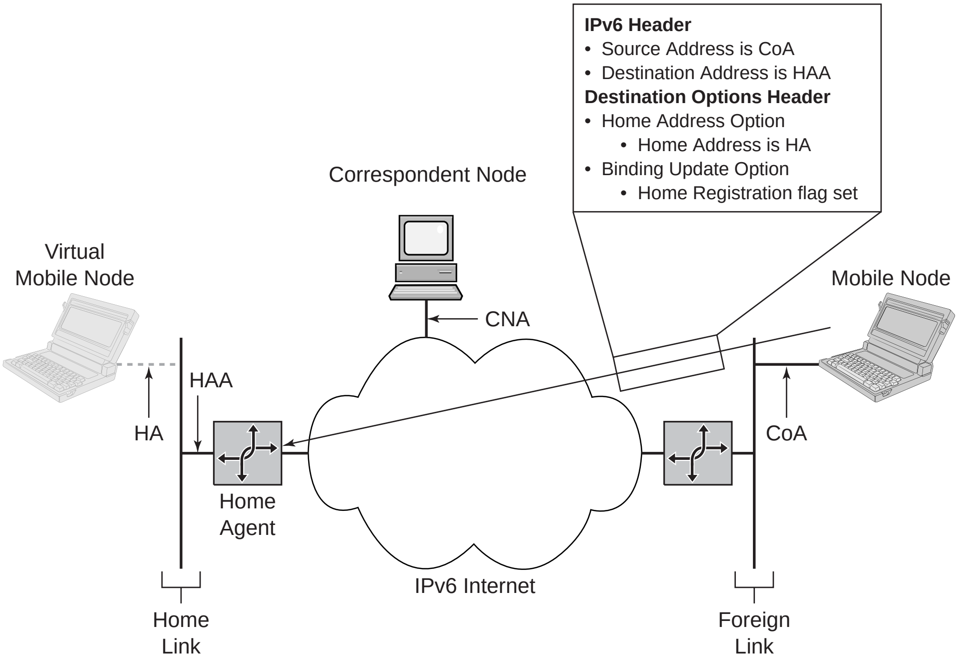

Figure 12-9. *Binding updates sent from the mobile node to the home agent*

In the Binding Update option, the Home Registration (H) flag is set, indicating that the sender is requesting that the receiver be the home agent for the mobile node. The Acknowledge (A) flag is also set to request a binding acknowledgement from the home agent.

If security for binding updates is enabled and an AH is present, there are two different Destination Options headers: one before the AH and one after the AH. The first Destination Options header contains the Home Address option and the second Destination Options header contains the Binding Update option. This is not shown in Figure 12-9.

ICMPv6 Home Agent Address Discovery Request Message

When the mobile node sends an ICMPv6 Home Agent Address Discovery Request message, it has the form shown in Figure 12-10.

- IPv6 header
  In the IPv6 header, the source address is the care-of address of the mobile node and the destination address is the Mobile IPv6 Home-Agents anycast address corresponding to the home link prefix.

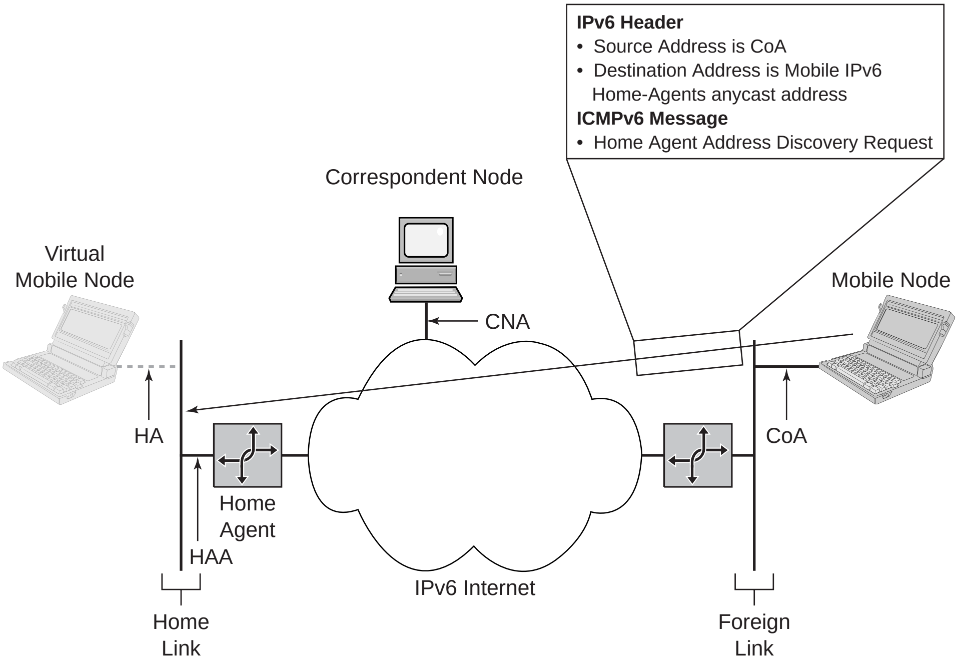

Figure 12-10. *ICMPv6 Home Agent Address Discovery Request message sent from the mobile node*

- ICMPv6 Home Agent Address Discovery Request message
  The ICMPv6 Home Agent Address Discovery Request message is used by the mobile node to query the home link for a list of home agents. For more information, see the “ICMPv6 Messages” section in this chapter.

From the Home Agent to the Mobile Node

Communication from the home agent to the mobile node takes the following forms:

- Binding maintenance
- ICMPv6 Home Agent Address Discovery Reply message
- Tunneled data

Binding Maintenance

Binding maintenance packets sent from the home agent to the mobile node are either binding acknowledgments or binding requests and are shown in Figure 12-11.

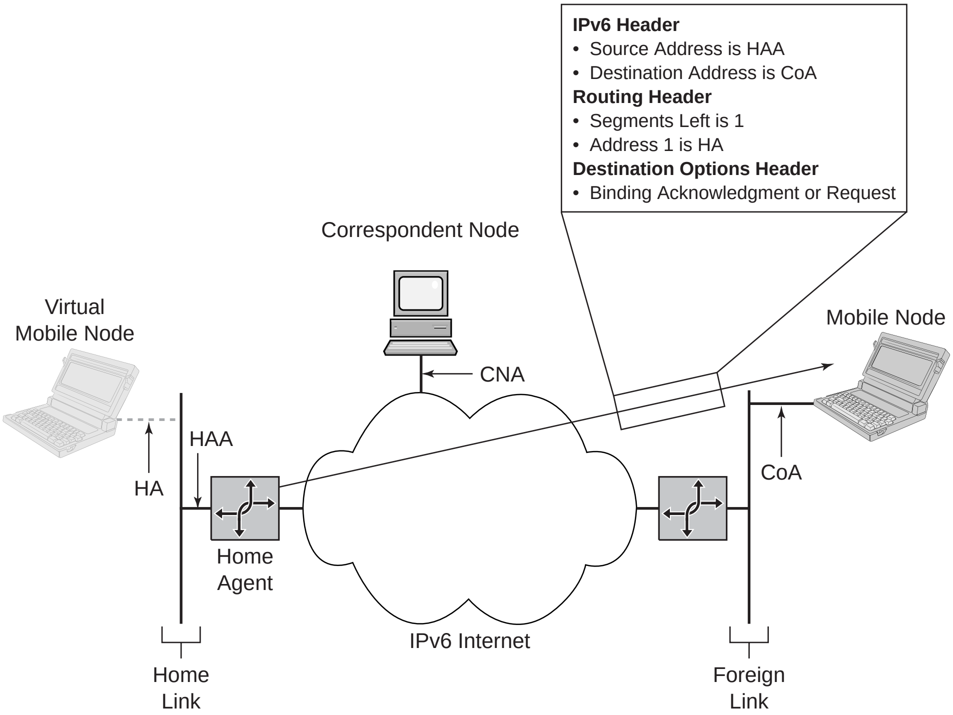

Figure 12-11. *Binding maintenance packets sent from the home agent to the mobile node*

The packet contains the following:

- IPv6 header
  In the IPv6 header, the source address is the home agent's address and the destination address is the mobile node's care-of address.
- Routing header
  In the Routing extension header, the Routing Type field is set to 0, the Segments Left field is set to 1, and the Address 1 field (the final destination address of the packet) is set to the mobile node's home address. When the mobile node receives the packet, it processes the Routing header and notes that the next destination address (the address in the Address 1 field) is its own home address. The mobile node removes the Routing header and logically replaces the care-of address with the home address as the destination in the IPv6 header.
- Destination Options header
  The Destination Options extension header contains either a Binding Acknowledgement option (if a received binding update had the Acknowledge [A] flag set) or a Binding Request option.

If security for binding updates is enabled, an AH is present between the Routing header and the Destination Options header for the binding acknowledgement. This is not shown in Figure 12-11.

ICMPv6 Home Agent Address Discovery Reply Message

When the home agent sends an ICMPv6 Home Agent Address Discovery Reply message, it has the form shown in Figure 12-12.

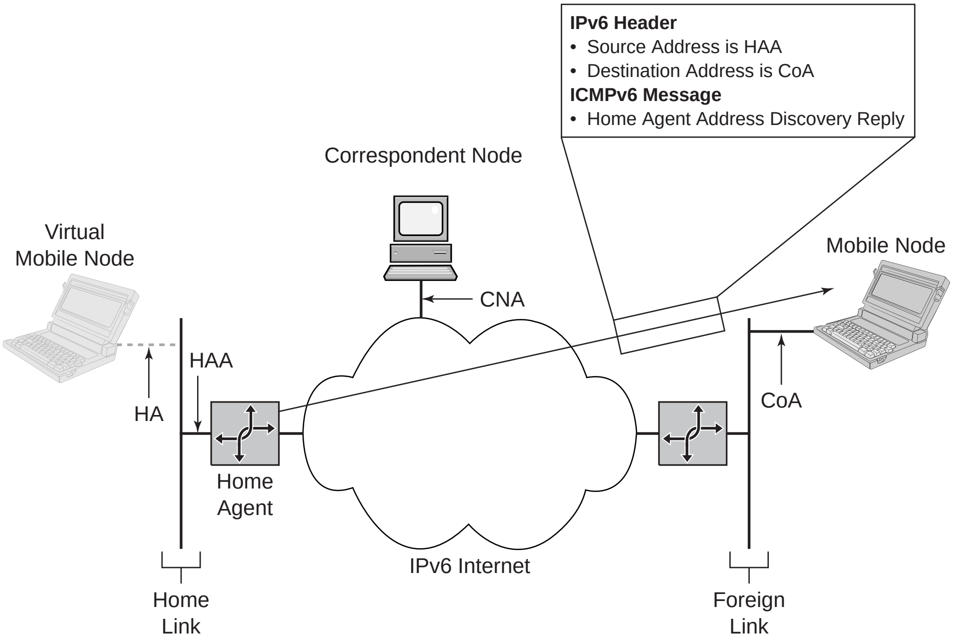

Figure 12-12. *ICMPv6 Home Agent Address Discovery Reply message sent from the home agent*

- IPv6 header
  In the IPv6 header, the source address is the home agent's address and the destination address is the mobile node's care-of address.
- ICMPv6 Home Agent Address Discovery Reply message
  The ICMPv6 Home Agent Address Discovery Reply message contains the list of home agents on the home link in order of preference. For more information, see the "ICMPv6 Messages" section in this chapter.

Tunneled Packet

When the home agent intercepts a packet sent directly to a mobile node's home address when the mobile node is away from home, it forwards the packet to the mobile node by using the form shown in Figure 12-13.

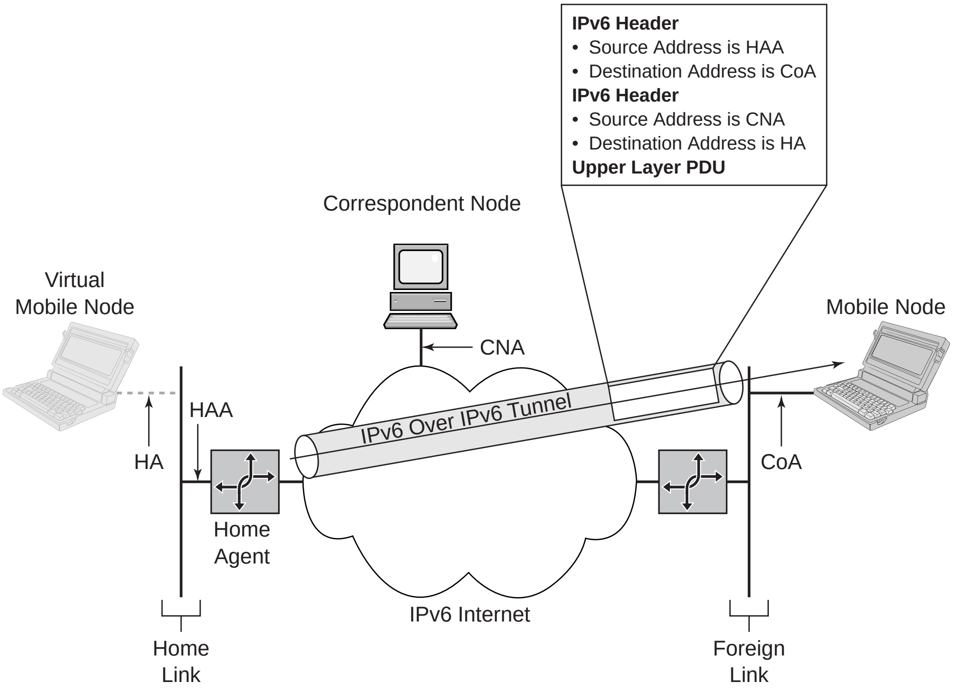

Figure 12-13. *Intercepted packet tunneled to a mobile node by its home agent*

- IPv6 header (outer)
  In the outer IPv6 header, the source address is the home agent's address and the destination address is the mobile node's care-of address.
- IPv6 header (inner)
  In the inner IPv6 header, the source address is the correspondent node's address and the destination address is the mobile node's home address.
- Upper layer PDU
  The upper layer PDU contains the Application layer data sent from the correspondent node to the mobile node at its home address. From the Application layer perspective, the data was addressed from the correspondent node address to the home address.

Notice that this packet is the original packet sent by the correspondent node that did not have a binding cache entry for the mobile node with an additional IPv6 header addressed from the home agent's address to the mobile node's care-of address. The original packet is described in the "Data with a Binding Cache Entry Not Present" section of this chapter.

IPV6 MOBILITY PROCESSES

IPv6 mobility provides a method for a mobile node to determine it is on its home link as well as providing message exchanges for the following processes:

- Moving from the home link to a foreign link
- Moving from a foreign link to another foreign link
- Returning home

Additionally, the sending host and receiving host processes are modified to include special processing for mobility support.

NOTE:
The following discussion assumes that the correspondent node supports full correspondent node functionality and is not a mobile node that is away from home.

Attaching to the Home Link

The method used by a mobile node to determine that it is attached to the home link is not defined in the IPv6 mobility draft. Once an IPv6 mobile node determines that it is connected to its home link, it can store the home subnet prefix, home address, and the global address of its home agent. The following methods for configuring home link parameters are based on implementations in development or existence at the time of the writing of this book:

- Manual configuration
  In the simplest case, the home subnet prefix, home address, and the global address of the home agent are manually configured, typically through a keyboard-based command, and are permanent until manually changed. These implementations do not support the dynamic discovery of home agents or changes in the home subnet prefix.
- Pseudo-automatic configuration
  For pseudo-automatic configuration, when an IPv6 node is attached to a link, the user has the option (typically through a button in the user interface of the operating system) to indicate to the IPv6 protocol that the node is now connected to the home link. Based on this indication, the IPv6 protocol stores the home subnet prefix and home address and listens for additional router advertisements containing the Home Agent (H) flag. The home agent is the router advertising itself with home agent capabilities and has the highest

preference level. Once determined, the IPv6 protocol stores the address of the home agent. These implementations may or may not support the dynamic discovery of home agents or changes in the home subnet prefix.

- Automatic configuration
  With automatic configuration, the IPv6 node is always listening for router advertisements that have the H flag set. Based on additional protocol or operating system parameters, the IPv6 node determines that it is potentially on its home link. Next, it chooses the most preferred home agent and attempts to establish a security relationship with it. If the security relationship with the home agent fails, the IPv6 node concludes it is not on its home link. If the security relationship succeeds, the IPv6 node is on its home link and stores its home subnet prefix, its home address, and the address of its home agent. These implementations may or may not support the dynamic discovery of home agents or changes in the home subnet prefix.

Moving From the Home Link to a Foreign Link

When the mobile node is at home, it autoconfigures its home address through the receipt of a router advertisement, and communication with other nodes occurs normally without the use of IPv6 mobility functionality.

Attaching to the Foreign Link

When the mobile node attaches to the foreign link, it must perform the following functions:

- Receive a new care-of address.
- Discover the home agent on the home link (if needed).
- Register the primary care-of address with the home agent on the home link.

When the mobile node attaches to the foreign link, the following occurs:

1. The mobile node sends a multicast Router Solicitation message on the foreign link. The mobile node might send a router solicitation either because the link layer indicated a media change or because the node received a router advertisement that contained a new prefix. Depending on the IPv6 mobility implementation, the mobile node sends a router solicitation either from its link-local address (assuming that the link-local address of the mobile node is most likely

unique on the foreign link) or from the unspecified address (::) (assuming that the link-local address of the mobile node might not be unique on the foreign link).

2. All routers on the foreign link reply with a Router Advertisement message. Depending on the source address of the Router Solicitation message, the reply is either unicast (because the router solicitation was sent from a link-local address) or multicast (because the router solicitation was sent from the unspecified address). Figure 12-14 shows the router advertisement being unicast to the mobile node.

   From the receipt of the Router Advertisement message(s), the mobile node determines that it has connected to a foreign link because the router advertisements contain new address prefixes. The mobile node forms care-of addresses from the advertised prefixes, verifies their uniqueness by using duplicate address detection, and joins the corresponding solicited node multicast groups (not shown in Figure 12-14).

3. If the mobile node is already configured with the address of its home agent, go to step 5. If not, to discover the home agent on the mobile node's home link, the mobile node sends an ICMPv6 Home Agent Address Discovery Request message to the Mobile IPv6 Home-Agents anycast address formed from the home subnet prefix.

   Depending on the implementation, mobile nodes might not maintain a list of home agents while connected to the home link. To automatically discover the home agents on the home link, it is sufficient for the mobile node to learn its home subnet prefix. When the mobile node leaves its home link and moves to the first foreign link, it sends an ICMPv6 Home Agent Address Discovery Request message to the Mobile IPv6 Home-Agents anycast address formed from the home subnet prefix.

4. A home agent on the home link that is using the Mobile IPv6 Home-Agents anycast address corresponding to the home subnet prefix and is topologically closest to the mobile node receives the Home Agent Address Discovery Request message. Next, it sends back an ICMPv6 Home Agent Address Discovery Reply message containing the entries in the home agent's home agent list in preference order.

   Upon receipt of the ICMPv6 Home Agent Address Discovery Reply message, the mobile node selects the first home agent in the list as its home agent.

5. To register the primary care-of address with the home agent, the mobile node sends the home agent a binding update. In the binding

update, the Home Registration (H) and Acknowledge (A) flags are set.

6. The home agent receives the binding update and updates its binding cache. To intercept packets destined for the mobile node's home address while the mobile node is away from home, the home agent performs proxy Neighbor Discovery for the mobile node by answering neighbor solicitations on behalf of the mobile node. Depending on the implementation, the home agent might send an unsolicited multicast Neighbor Advertisement message as if it were the mobile node immediately or respond only to multicast neighbor solicitations for the mobile node's home address.

   In the first case, to ensure that the nodes on the home link are updated with the new link-layer address of the home agent's interface on the home link, the home agent sends an unsolicited multicast Neighbor Advertisement message to the link-local scope all-nodes multicast address (FF02::1) with the Override (O) flag set. Additionally, the home agent joins the multicast group for the solicited node multicast address corresponding to the mobile node's home address, and registers interest in receiving link-layer multicast frames to the multicast MAC address corresponding to the solicited node multicast address. This is shown in Figure 12-14.

   In the second case, the home agent does not send an unsolicited multicast Neighbor Advertisement message. However, the home agent does join the multicast group for the solicited node multicast address corresponding to the mobile node's home address, and registers interest in receiving link-layer multicast frames to the multicast MAC address corresponding to the solicited node multicast address. If a node on the home link was communicating with the mobile node while it was at home, neighbor unreachability detection would eventually cause the home node to send three unicast neighbor solicitations (while in the PROBE state) and then send a multicast neighbor solicitation. The multicast neighbor solicitation is then answered by the home agent on behalf of the mobile node. This is not shown in Figure 12-14.

7. Because the binding update had the A flag set, the home agent responds with a binding acknowledgement.

This process is shown in Figure 12-14.

Notice that the mobile node does not send a binding update to all the nodes with which the mobile node was communicating when connected to the home link (as there are no entries in the binding update list). Rather, the

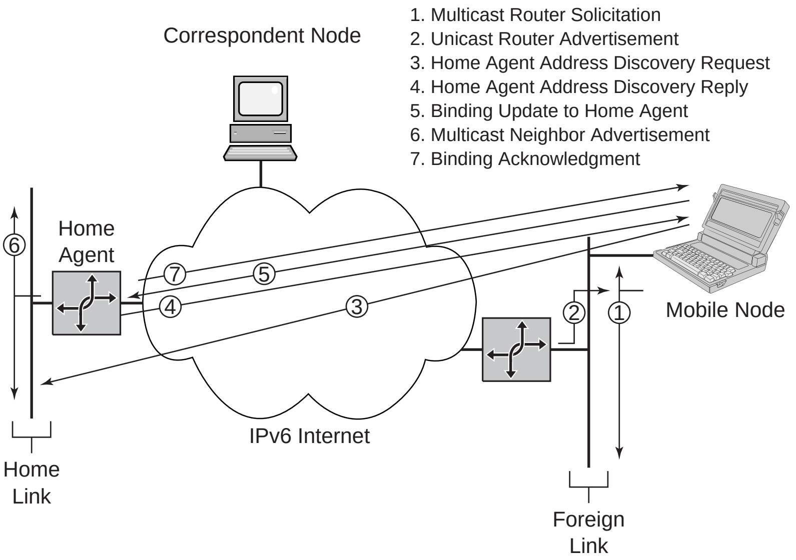
Correspondent Node
1. Multicast Router Solicitation
2. Unicast Router Advertisement
3. Home Agent Address Discovery Request
4. Home Agent Address Discovery Reply
5. Binding Update to Home Agent
6. Multicast Neighbor Advertisement
7. Binding Acknowledgment
Home Agent
⑥
⑦
⑤
④
③
②
①
Mobile Node
IPv6 Internet
Home Link
Foreign Link

Figure 12-14. *Mobile node attaching to the first foreign link*

mobile node relies on the receipt of tunneled traffic from the home agent to send binding updates to correspondent nodes.

Mobile Node Initiates a New TCP Connection with a New Correspondent Node

When a mobile node that is away from home initiates a new TCP connection with a correspondent node, the following occurs:

1. The mobile node sends a TCP SYN (synchronize) segment containing the Destination Options header and the Home Address and Binding Update options to the correspondent node.
2. The correspondent node receives the TCP SYN segment and processes the Home Address and Binding Update options. The correspondent node updates its binding cache and sends the TCP SYN-ACK (synchronize-acknowledgement) segment that includes a Routing header with the mobile node's home address and, if requested, a binding acknowledgment.
3. Upon receipt of the TCP SYN-ACK from the correspondent node, the mobile node sends a TCP ACK (acknowledge) segment that contains the Home Address option to the correspondent node.

This process is shown in Figure 12-15.

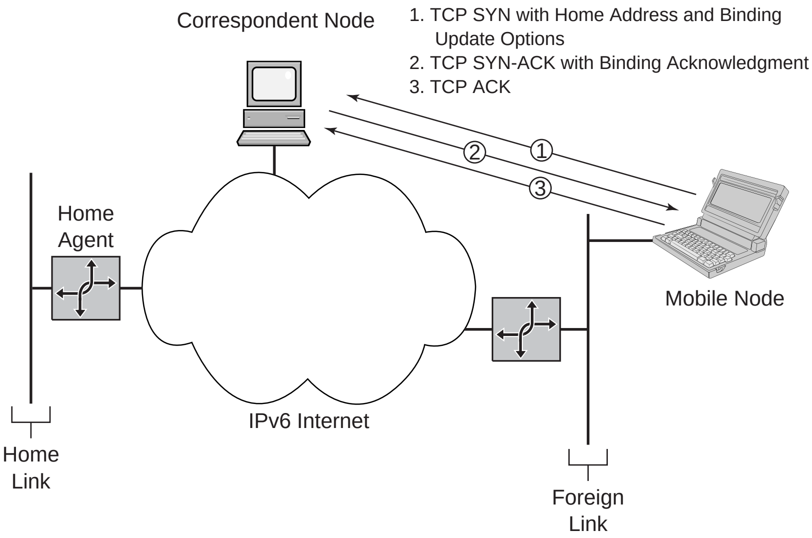
Correspondent Node
1. TCP SYN with Home Address and Binding Update Options
2. TCP SYN-ACK with Binding Acknowledgment
3. TCP ACK
①
②
③
Home Agent
Mobile Node
IPv6 Internet
Home Link
Foreign Link

Figure 12-15. *A mobile node initiating a new TCP connection with a new correspondent node*

If the mobile node is resuming communication using an existing TCP connection, then the same process described here is done for the first three TCP segments exchanged over the resumed TCP connection.

After this process is complete, data between the correspondent node and the mobile node is sent as follows:

- Data from the mobile node is sent from the mobile node's care-of address to the correspondent node's address and includes the Home Address option in the Destination Options header.
- Data from the correspondent node is sent to the mobile node's care-of address and includes a Routing header containing the mobile node's home address.

NOTE:
If the mobile node is multihomed, it is possible for the mobile node to register different care-of addresses with different correspondent nodes. Which care-of address is chosen depends on the source address selection algorithm. The mobile node will choose the care-of address that is matched in scope and topologically closest to the correspondent node.

Mobile Node Initiates Non-TCP-based Communication with a New Correspondent Node

When a mobile node that is away from home either resumes or initiates communication with a correspondent node that does not use a TCP connection (such as ICMPv6 or an Application layer protocol that uses UDP), the following occurs:

1. The mobile node sends the initial message to the correspondent node containing the Destination Options header with the Home Address option.
2. The correspondent node receives the initial message and processes the Home Address option. Because a binding does not yet exist for the mobile node, the correspondent node sends the response message to the home address.
3. Because the home agent has a binding for the mobile node and is acting as an ND proxy for the mobile node, it intercepts the response message sent to the mobile node's home address and tunnels it to the mobile node at the mobile node's care-of address.
4. Upon receipt of the tunneled response message from the home agent, the mobile node queues a binding update to the correspondent node. Whether the binding update is sent as a separate packet or is included as part of upper layer data depends on the implementation. For this example, the binding update is included in the next message sent to the correspondent node.
5. Upon receipt of the next message with the binding update, the correspondent node updates its binding cache and sends back, if requested, a binding acknowledgment that includes a Routing header with the mobile node's home address.

This process is shown in Figure 12-16.

After this process is complete, data between the correspondent node and the mobile node is sent as follows:

- Data from the mobile node is sent from the mobile node's care-of address to the correspondent node's address and includes the Home Address option in the Destination Options header.
- Data from the correspondent node is sent to the mobile node's care-of address and includes a Routing header containing the mobile node's home address.

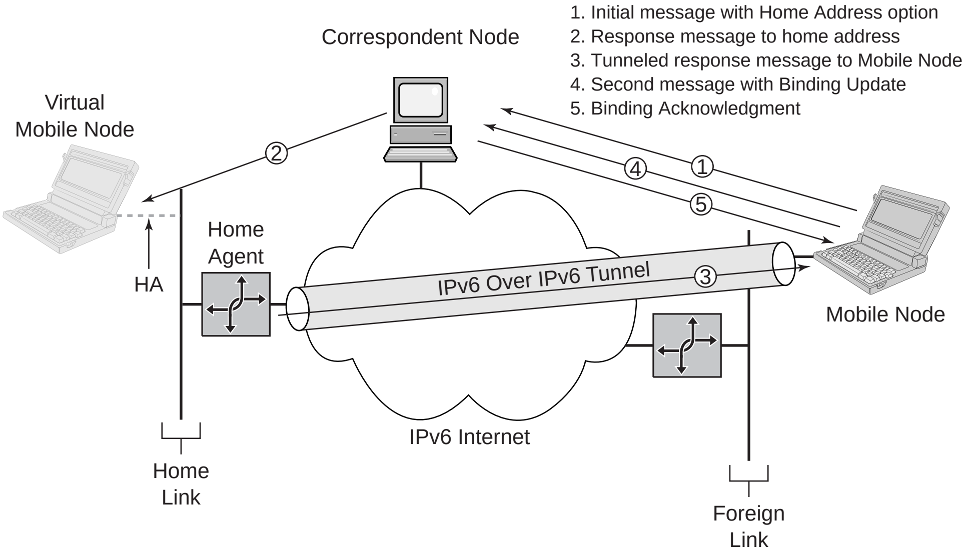
1. Initial message with Home Address option
2. Response message to home address
3. Tunneled response message to Mobile Node
4. Second message with Binding Update
5. Binding Acknowledgment
Correspondent Node
Virtual Mobile Node
①
②
③
④
⑤
Home Agent
HA
IPv6 Over IPv6 Tunnel
Mobile Node
IPv6 Internet
Home Link
Foreign Link

Figure 12-16. *A mobile node initiating non-TCP-based communication with a new correspondent node*

A New Correspondent Node Communicates with a Mobile Node

When a new correspondent node either resumes communication or initiates communication with a mobile node using the mobile node's home address and the mobile node is away from home, the following occurs (example assumes a new TCP connection):

1. The new correspondent node sends a TCP SYN segment to the mobile node's home address. The packet is delivered by the routers of the IPv6 Internet to a router connected to the mobile node's home link.
2. Because the home agent has a binding for the mobile node and is acting as its ND proxy, it intercepts the TCP SYN segment sent to the mobile node's home address and tunnels it to the mobile node's care-of address.
3. Upon receipt of the tunneled TCP SYN segment from the home agent, the mobile node adds an entry for the correspondent node to its binding update list and sends a TCP SYN-ACK with a Destination Options header that contains the Home Address and Binding Update options.
4. Upon receipt of the TCP SYN-ACK segment with the binding update, the correspondent node updates its binding cache and sends back a TCP ACK segment that includes a Routing header with the mobile

node's home address and, if requested, a binding acknowledgement in the Destination Options header.

This process is shown in Figure 12-17.

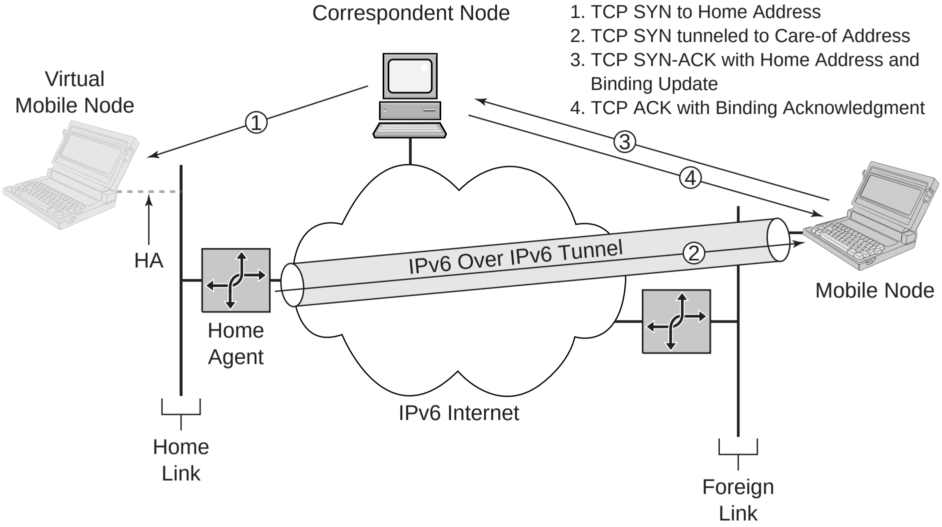
Correspondent Node
1. TCP SYN to Home Address
2. TCP SYN tunneled to Care-of Address
3. TCP SYN-ACK with Home Address and Binding Update
4. TCP ACK with Binding Acknowledgment
Virtual Mobile Node
①
③
④
②
HA
IPv6 Over IPv6 Tunnel
Mobile Node
Home Agent
IPv6 Internet
Home Link
Foreign Link

Figure 12-17. *A new correspondent node communicating with a mobile node*

After this process is complete, data between the correspondent node and the mobile node is sent as follows:

- Data from the mobile node is sent from the care-of address to the correspondent node's address and includes the Home Address option in the Destination Options header.
- Data from the correspondent node is sent to the mobile node's care-of address and includes a Routing header containing the mobile node's home address.

This same process is performed if the correspondent node removes the binding for the mobile node from its binding cache.

A Node on the Home Link Communicates with the Mobile Node

When a node on the home link either resumes or initiates communication with a mobile node using the mobile node's home address and the mobile node is away from home, the following occurs (example assumes a new TCP connection):

1. The node on the home link sends a multicast Neighbor Solicitation message to the solicited node multicast address corresponding to the mobile node's home address.

2. The home agent is acting as an ND proxy for the mobile node. It has registered the solicited node multicast address corresponding to the mobile node's home address as a multicast address to which the home agent is listening. The home agent receives the neighbor solicitation and sends a unicast neighbor advertisement containing the home agent's link-layer address in the Target Link-Layer Address option.
3. The node on the home link sends the TCP SYN segment to the home agent with the mobile node's home address as the destination IPv6 address and the home agent's link-layer address as the destination link-layer address.
4. Because the IPv6 packet is addressed to the home address of the mobile node, the home agent tunnels the TCP SYN segment to the mobile node's care-of address.
5. Upon receipt of the tunneled TCP SYN segment from the home agent, the mobile node adds an entry for the node on the home link to its binding update list and sends a TCP SYN-ACK with a Destination Options header that contains the Home Address and Binding Update options.
6. Upon receipt of the TCP SYN-ACK segment with the binding update, the node on the home link updates its binding cache and sends a TCP ACK segment that includes a Routing header with the mobile node's home address and, if requested, a binding acknowledgement in the Destination Options header.

This process is shown in Figure 12-18.

This same process of intercepting a packet for the mobile node (steps 1 through 3) is used when a packet addressed to the mobile node's home address is delivered to the home link by a router that is not the mobile node's home agent.

Mobile Node Obtains a New Home Address

The home address of the mobile node was initially obtained through the receipt of a router advertisement while the mobile node was connected to the home link, and the stateless address might have a finite lifetime. Because the mobile node is away from home, it does not receive the pseudo-periodic multicast router advertisements sent by the routers on the home link. To refresh a home address that is approaching the end of its valid lifetime or receive a new home address, the following process is used:

1. The mobile node sends an IPv6-tunneled Router Solicitation message to its home agent. The inner IPv6 header is addressed from the mobile node's home address to the home agent's address. The outer

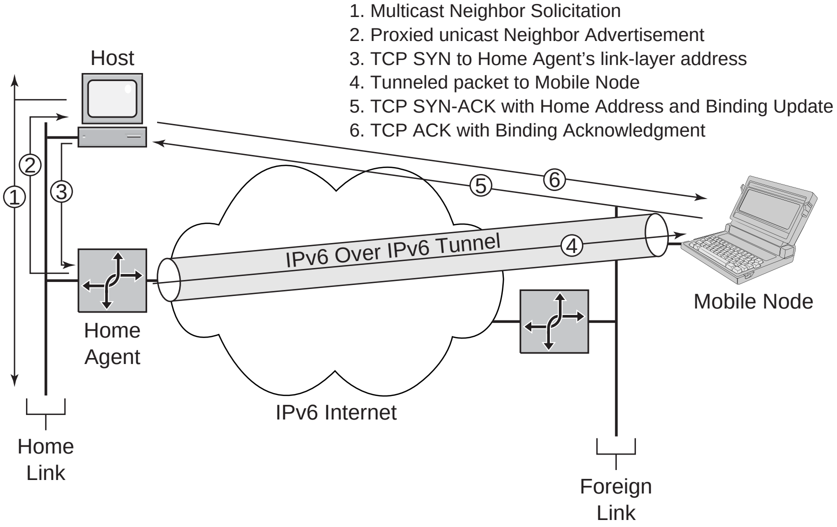
1. Multicast Neighbor Solicitation
2. Proxied unicast Neighbor Advertisement
3. TCP SYN to Home Agent's link-layer address
4. Tunneled packet to Mobile Node
5. TCP SYN-ACK with Home Address and Binding Update
6. TCP ACK with Binding Acknowledgment
Host
①
②
③
④
⑤
⑥
IPv6 Over IPv6 Tunnel
Home Agent
Mobile Node
IPv6 Internet
Home Link
Foreign Link

Figure 12-18. *A node on the home link communicating with the mobile node*

IPv6 header is addressed from the mobile node's care-of address to the home agent's address.

2. Upon receipt of the tunneled router solicitation, the home agent responds with a unicast Router Advertisement message that is sent from the home agent's address to the mobile node's care-of address and includes a Routing header with the mobile node's home address.

Upon receipt of the router advertisement, the mobile node examines the Prefix Information option(s) and does the following:

- If there is no change in the home subnet prefix and therefore no change in the home address, then the mobile node refreshes the valid and preferred lifetimes of the stateless home address.
- If there is a change in the home subnet prefix, then the mobile node autoconfigures a new home address and sends binding updates for all the entries of its binding update list (the home agent and all correspondent nodes).

Moving from a Foreign Link to Another Foreign Link

When the mobile node attaches to a new foreign link after being attached to another foreign link, it must perform the following functions:

- Receive a new care-of address.

- Register the new care-of address with the home agent on the home link.
- Send binding updates to all correspondent nodes.

NOTE:
The IPv6 mobility draft also describes the registration of the new care-of address with a router that has home agent capabilities on the previous foreign link to establish forwarding of packets sent to an outdated care-of address. This is not widely supported and is not described in this chapter.

When the mobile node attaches to the new foreign link, the following occurs:

1. The mobile node sends a multicast Router Solicitation message on the new foreign link. Depending on the IPv6 mobility implementation, the mobile node sends a router solicitation either from its link-local address (assuming that the link-local address of the mobile node is most likely unique on the new foreign link) or from the unspecified address (::) (assuming that the link-local address of the mobile node might not be unique on the new foreign link).
2. All routers on the new foreign link reply with a Router Advertisement message. Depending on the source address of the Router Solicitation message, the reply is either unicast (because the router solicitation was sent from a link-local address) or multicast (because the router solicitation was sent from the unspecified address). Figure 12-19 shows the router advertisement being unicast to the mobile node.

   From the receipt of the Router Advertisement message(s), the mobile node forms care-of address(es), verifies their uniqueness by using duplicate address detection, and joins the corresponding solicited node multicast groups (not shown in Figure 12-19).
3. To register the new primary care-of address with the home agent, the mobile node sends the home agent a binding update. In the binding update, the Home Registration (H) and Acknowledge (A) flags are set.
4. For each correspondent node in the mobile node's binding update list, the mobile node sends a binding update.

   The mobile node does not have to immediately send a binding update. Because the binding update is carried in the Destination

Options header, the mobile node can delay the sending of the binding update to the correspondent node so that it is sent with the next packet. However, a delay in sending the binding update to correspondent nodes can result in data loss when packets sent by the correspondent node are delivered to the previous foreign link.

5. Upon the receipt of the binding update, the home agent updates its binding cache, and responds with a binding acknowledgement.

Upon receipt of the binding update, each correspondent node updates its binding cache and, if requested by the mobile node, sends a binding acknowledgment.

This process is shown in Figure 12-19.

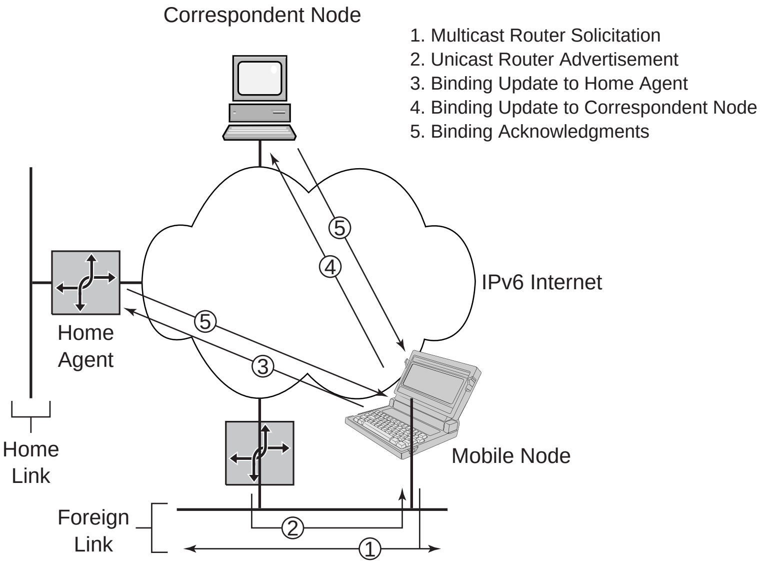
Correspondent Node
1. Multicast Router Solicitation
2. Unicast Router Advertisement
3. Binding Update to Home Agent
4. Binding Update to Correspondent Node
5. Binding Acknowledgments
IPv6 Internet
⑤
④
⑤
③
Home Agent
Home Link
Mobile Node
Foreign Link
②
①

Figure 12-19. *A mobile node attaching to a new foreign link*

If the binding update sent by the mobile node to a correspondent node is dropped from the network, the correspondent node continues to send packets to the mobile node's previous care-of address based on the contents of its now outdated binding cache entry. The packets are forwarded to the previous foreign link and the router on the previous foreign link attempts to deliver them. If the previous foreign link router still considers the mobile node reachable on the previous foreign link, packets are forwarded to the mobile node's link layer address. Because the mobile node is no longer attached to the previous foreign link, the packets are dropped.

The methods for correcting this error condition are the following:

- The mobile node, after not receiving a binding acknowledgment from the correspondent node, retransmits a binding update. The retransmitted binding update is received by the correspondent node and its binding cache is updated with the mobile node's new care-of address.
- The previous foreign link router uses neighbor unreachability detection to determine that the mobile node is no longer attached to the previous foreign link. For a point-to-point link such as a wireless connection, the unreachability of the mobile node is indicated immediately by the lack of a wireless signal from the mobile node. For a broadcast link such as an Ethernet segment, the entry in the previous foreign link router's neighbor cache goes through the REACHABLE, STALE, DELAY, and PROBE states as described in Chapter 6, "Neighbor Discovery." After the neighbor cache entry for the mobile node is removed, attempts to deliver to the mobile node's previous care-of address are unsuccessful and the previous foreign link router will send an ICMPv6 Destination Unreachable-Address Unreachable message to the correspondent node. Upon receiving this message, the correspondent node will remove the entry for the mobile node from its binding cache and communication resumes as described in the "A New Correspondent Node Communicates with a Mobile Node" section of this chapter.
- All binding cache entries have a finite lifetime as determined by the Lifetime field of the last received binding update. After the lifetime expires, the binding cache entry is removed and communication resumes as described in the "A New Correspondent Node Communicates with a Mobile Node" section of this chapter.

Returning Home

When the mobile node attaches to its home link after being away from home, it must perform the following functions:

- Send a binding update to the home agent to delete the binding for the mobile node.
- Inform home link nodes that the correct link-layer address for the home address is now the mobile node's link-layer address.
- Send binding updates to all correspondent nodes to delete the bindings for the mobile node.

These functions are shown in Figure 12-20.

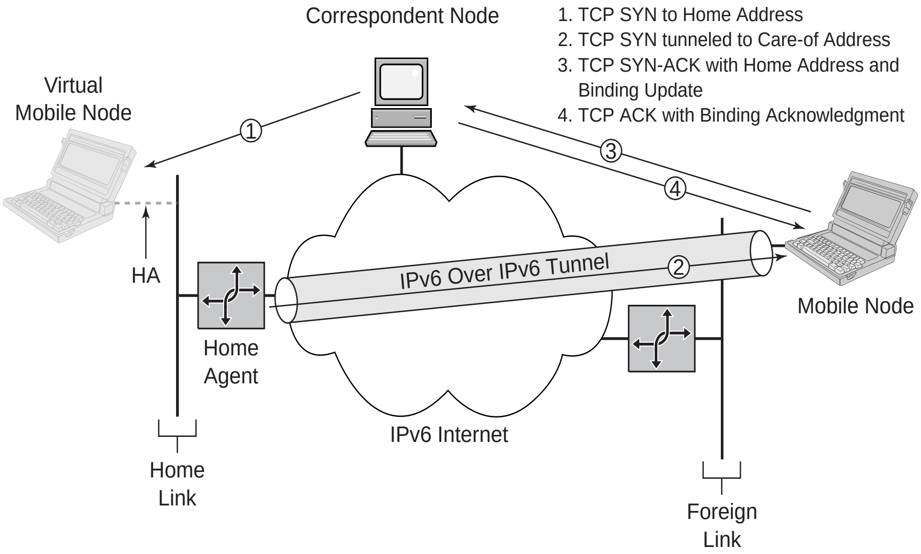
Correspondent Node
1. TCP SYN to Home Address
2. TCP SYN tunneled to Care-of Address
3. TCP SYN-ACK with Home Address and Binding Update
4. TCP ACK with Binding Acknowledgment
Virtual Mobile Node
①
③
④
HA
IPv6 Over IPv6 Tunnel
②
Mobile Node
Home Agent
IPv6 Internet
Home Link
Foreign Link

Figure 12-20. *A mobile node returning home*

When the mobile node returns home (reattaches to its home link), the following occurs:

1. The mobile node sends a multicast Router Solicitation message on the home link. The mobile node might send a router solicitation either because the link layer indicated a media change or because the node received a router advertisement that contained a new prefix. Depending on the IPv6 mobility implementation, the mobile node sends a router solicitation either from its link-local address (assuming that the link-local address of the mobile node is most likely unique on the home link) or from the unspecified address (::) (assuming that the link-local address of the mobile node might not be unique on the home link).

2. All routers on the home link reply with a Router Advertisement message. Depending on the source address of the Router Solicitation message, the reply is either unicast (because the router solicitation was sent from a link-local address) or multicast (because the router solicitation was sent from the unspecified address). Figure 12-20 shows the router advertisement being unicast to the mobile node.

   Because the router advertisement contains an address prefix that matches its home address prefix, the mobile node determines that it is attached to its home link. Depending on the IPv6 mobility implementation, the mobile node may or may not perform duplicate address detection for its home address because the home agent is acting as an ND proxy for the mobile node and defending the use of the mobile node's home address. If the mobile node does perform duplicate address detection, it must ignore the neighbor advertisement reply sent from the home agent.

3. To remove the binding cache entry from the home agent, the mobile node sends the home agent a binding update with the care-of address set to the mobile node's home address and with the Home Registration (H) and Acknowledge (A) flags set.

   If multiple router advertisements are received, the mobile node can determine which router is its home agent from the router advertisement with the Prefix Information option that contains the home agent's global address in the Prefix field.

   The mobile node determines the home agent's link-layer address from the Link-Layer Address field in the Source Link-Layer Address option in the router advertisement sent by the home agent. If the Source Link-Layer Address option is not included, then the mobile node can determine the link-layer address of the home agent using address resolution, because the global address of the home agent is known.

4. For each correspondent node in the mobile node's binding update list, the mobile node sends a binding update to the correspondent node with the care-of address set to the mobile node's home address.

5. Upon receipt of the binding update, the home agent removes the entry for the mobile node from its binding cache, stops defending the use of the mobile node's home address on the home link, and responds with a binding acknowledgement. This is shown in Figure 12-20. Additionally, the home agent removes itself from the multicast group for the solicited node multicast address corresponding to the mobile node's home address and stops listening for link-layer multicast frames addressed to the multicast MAC address corresponding to the solicited node multicast address.

   Upon receipt of the binding update, the correspondent nodes remove the entry for the mobile node in their binding cache and, if requested by the mobile node, send a binding acknowledgment.

6. After receiving the binding acknowledgement from the home agent, the mobile node must inform nodes on the home link that the

link-layer address for the home address has changed to the link-layer address of the mobile node. It sends an unsolicited multicast Neighbor Advertisement message to the link-local scope all-nodes multicast address (FF02::1) with the Override (O) flag set.

The sending of the unsolicited multicast Neighbor Advertisement message is not required. If it is not sent, nodes on the home link that were communicating with the mobile node while it was away from home might still have an entry in their neighbor cache that contains the mobile node's home address and the link-layer address of the home agent. This is a redirect situation, in which a node is sending a packet to a router when the destination is on-link. When the home node sends a packet to the mobile node by using the link-layer address of the home agent, the home agent forwards the packet to the mobile node and sends a Redirect message containing the Target Link-Layer Address option to update the neighbor cache of the home node.

The mobile node also joins the multicast group for the solicited node multicast address corresponding to the mobile node's home address, and registers interest in receiving link-layer multicast frames to the multicast MAC address corresponding to the solicited node multicast address.

IPv6 Mobility Host Sending Algorithm

The IPv6 host sending algorithm is described in Chapter 6, "Neighbor Discovery." However, the discussion in Chapter 6 does not include full IPv6 mobility functionality. An IPv6 mobile node can be both a mobile node and correspondent node at the same time. Therefore, the host sending algorithm for an IPv6 mobility node must take into account the following:

- If the sending host is away from home
  If so, the sending host must set the source address of the IPv6 header to the sending host's care-of address and include the Destination Options header with the Home Address option set to the sending host's home address.

- If the destination node is away from home
  If so, the sending host must set the destination address of the IPv6 header to the destination node's care-of address and include a Routing header with the Address 1 field set to the destination node's home address.

An IPv6 mobility host uses the following algorithm when sending a unicast or anycast packet to an arbitrary destination:

1. Check the destination cache for an entry matching the destination address.
2. If an entry matching the destination address is not found in the destination cache, go to step 7.
3. If an entry matching the destination address is found in the destination cache, check for a pointer to an entry in the binding cache. This pointer will be present if the destination is a mobile node away from home.
4. If there is a pointer to an entry in the binding cache, the sending host sets the destination address in the IPv6 header to the destination node's care-of address and inserts a Routing header that includes the destination node's home address in the Address 1 field. The binding cache entry for the home address contains a pointer to the destination cache entry for the care-of address, from which the sending host obtains the next-hop address and interface for the care-of address.
5. If there is a no pointer to an entry in the binding cache, then the sending host obtains the next-hop address and interface from the destination cache entry.
6. If the sending host is a mobile node away from home, it sets the source address in the IPv6 header to the sending host's care-of address and inserts a Destination Options header that includes the Home Address option containing the sending host's home address. Go to step 10.
7. Check the local IPv6 routing table for the longest matching route that has the lowest metric to the destination address. If there are multiple longest matching routes with the lowest metric, IPv6 chooses a route to use.
8. Based on the chosen route, determine the next-hop interface and address used for forwarding the packet.

   If no route is found, IPv6 assumes that the destination is directly reachable. The next-hop address is set to the destination address and an interface is chosen.
9. Update the destination cache.
10. Check the neighbor cache for an entry matching the next-hop address.

11. If an entry matching the next-hop address is found in the neighbor cache, obtain the link-layer address.
12. If an entry matching the next-hop address is not found in the neighbor cache, use address resolution to obtain the link-layer address for the next-hop address.

    If address resolution is not successful, indicate an error.
13. Send the packet by using the link-layer address of the neighbor cache entry.

Figure 12-21 shows the IPv6 mobility host sending process.

IPv6 Mobility Host Receiving Algorithm

The IPv6 host receiving algorithm is described in Chapter 10, "IPv6 Routing." However, the discussion in Chapter 10 does not include full IPv6 mobility functionality. An IPv6 mobile node can be both a mobile node and correspondent node at the same time. Therefore, the host receiving algorithm for an IPv6 mobility node must take into account the following:

- If the receiving node is away from home

  If so, the receiving node processes the Routing header in the IPv6 packet and logically sets the destination address of the IPv6 header to the value of the Address 1 field in the Routing header.
- If the sending host is away from home

  If so, the receiving node processes the Destination Options header and logically sets the source address of the IPv6 packet to the home address contained in the Home Address option.

Additionally, a receiving IPv6 mobility host must recognize a packet tunneled from its home agent in order to determine when to send a binding update to a new correspondent node.

An IPv6 mobility host uses the following algorithm when receiving a unicast or anycast packet from an arbitrary source:

1. Verify whether the destination address in the IPv6 packet corresponds to an IPv6 address assigned to a local host interface.

   If the destination address is not assigned to a local host interface, silently discard the IPv6 packet.
2. Check to see if there is a Routing header present. If so, process the Routing header and set the destination address of the IPv6 packet to the value of the last address field in the Routing header. For

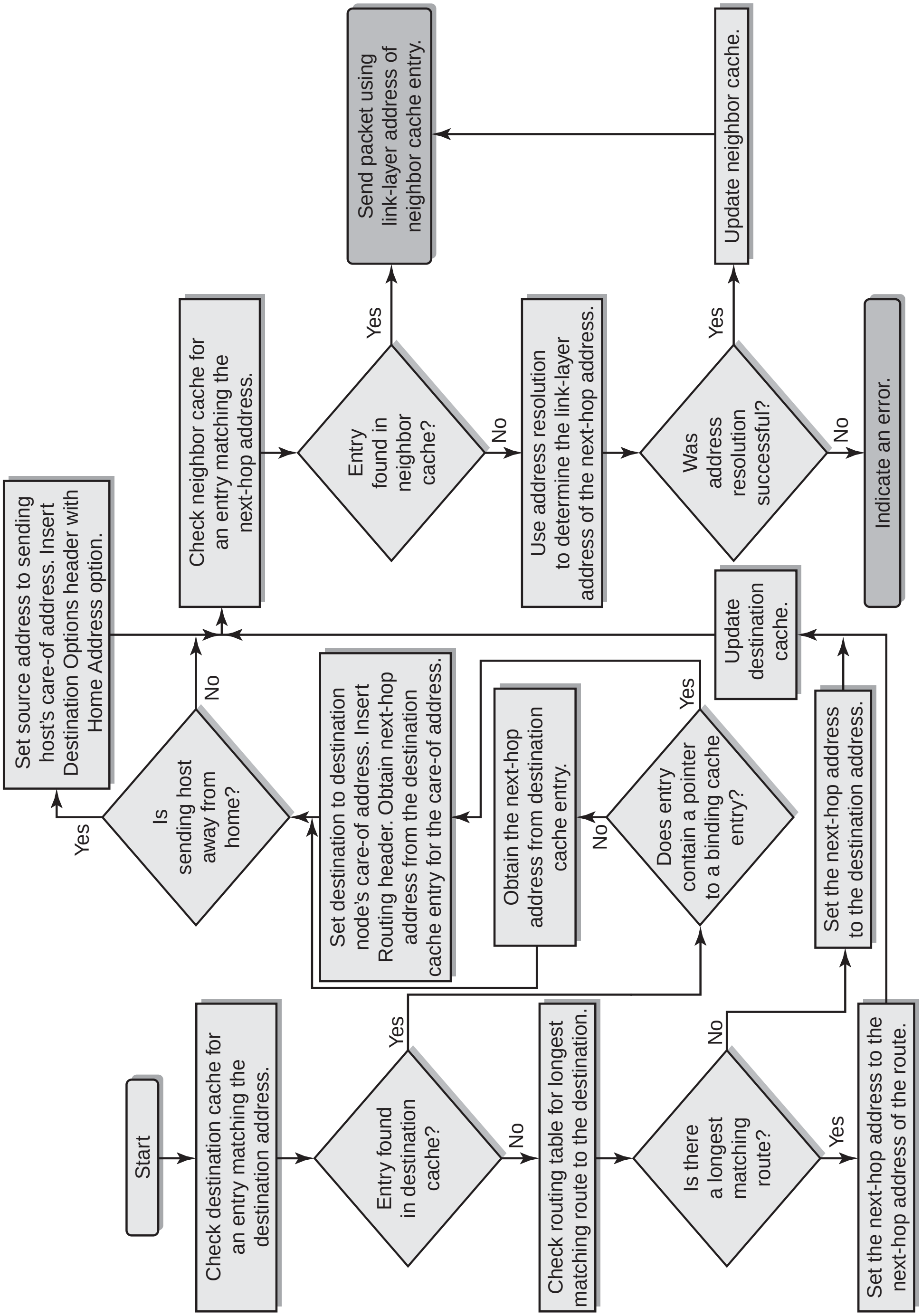
Start
Check destination cache for an entry matching the destination address.
Entry found in destination cache?
Yes
No
Check routing table for longest matching route to the destination.
Is there a longest matching route?
Yes
No
Set the next-hop address to the next-hop address of the route.
Set the next-hop address to the destination address.
Update destination cache.
Does entry contain a pointer to a binding cache entry?
Yes
No
Obtain the next-hop address from destination cache entry.
Set destination to destination node's care-of address. Insert Routing header. Obtain next-hop address from the destination cache entry for the care-of address.
Is sending host away from home?
Yes
No
Set source address to sending host's care-of address. Insert Destination Options header with Home Address option.
Check neighbor cache for an entry matching the next-hop address.
Entry found in neighbor cache?
Yes
No
Send packet using link-layer address of neighbor cache entry.
Use address resolution to determine the link-layer address of the next-hop address.
Was address resolution successful?
Yes
No
Update neighbor cache.
Indicate an error.

Figure 12-21. *The IPv6 mobility host sending process*

packets sent from correspondent nodes, the last address field is the Address 1 field, which contains the mobile node's home address.

Although all IPv6 hosts must support the processing of a Routing header, this was not described in Chapter 10 to keep the discussion of the host receiving algorithm as simple as possible. Because a mobility-based Routing header is placed in the packet when the destination mobile node is away from home, it is explicitly described here.

3. Check to see if the packet was tunneled from the home agent. In the outer IPv6 header, the destination address is set to the receiving node's care-of address, the source address is set to the home agent's address, and the protocol field is set to 41. If so, strip the outer header, set the destination and source addresses of the packet to the addresses in the inner IPv6 header, and queue a binding update to the source address in the inner IPv6 header. The binding update is sent either as a separate packet or is sent with response data to the new correspondent node.

   For data sent to an IPv6 mobile node when it is away from home, an incoming IPv6 packet will either be tunneled from the home agent or sent with a Routing header containing the home address.

   Although all IPv6 hosts must support the processing of IPv6 tunneled packets, this was not described in Chapter 10 to keep the discussion of the host receiving algorithm as simple as possible. Because IPv6 packets are tunneled by a home agent when the destination mobile node is away from home, it is explicitly described here.

4. Check to see if there is a Destination Options header with a Home Address option. If so, logically set the source address of the IPv6 packet to the home address in the Home Address option.

5. Based on the Next Header field, process extension headers (if present) and pass the upper layer PDU to the appropriate upper-layer protocol.

   If the protocol does not exist, send an ICMPv6 Parameter Problem-Unrecognized Next Header Type Encountered message back to the sender and discard the packet.

6. If the upper layer PDU is not a TCP segment or UDP message, pass the upper layer PDU to the appropriate protocol.

7. If the upper layer PDU is a TCP segment or UDP message, check the destination port.

If no application exists for the UDP port number, send an ICMPv6 Destination Unreachable-Port Unreachable message back to the sender and discard the packet. If no application exists for the TCP port number, send a TCP Connection Reset segment back to the sender and discard the packet.

8. If an application exists for the UDP or TCP destination port, process the contents of the TCP segment or UDP message.

Figure 12-22 shows the IPv6 mobility host receiving process.

REFERENCES

RFC 2461 – "Neighbor Discovery for IP Version 6 (IPv6)"

RFC 2526 – "Reserved IPv6 Subnet Anycast Addresses"

Internet draft – "Mobility Support in IPv6" (version 13)

TESTING FOR UNDERSTANDING

To test your understanding of IPv6 mobility, answer the following questions. See Appendix D to check your answers.

1. How does a mobile node determine its home subnet prefix, home address, and the address of its home agent?

2. When does a home agent or correspondent node send a binding request?

3. How does a home agent compile a list of home agents on the home link and then convey that information to the mobile node while it is away from home?

4. How does the mobile node determine when it has attached to a new link?

5. What kinds of packets are sent between the home agent and the mobile node?

6. What kinds of packets are sent between the correspondent node and the mobile node?

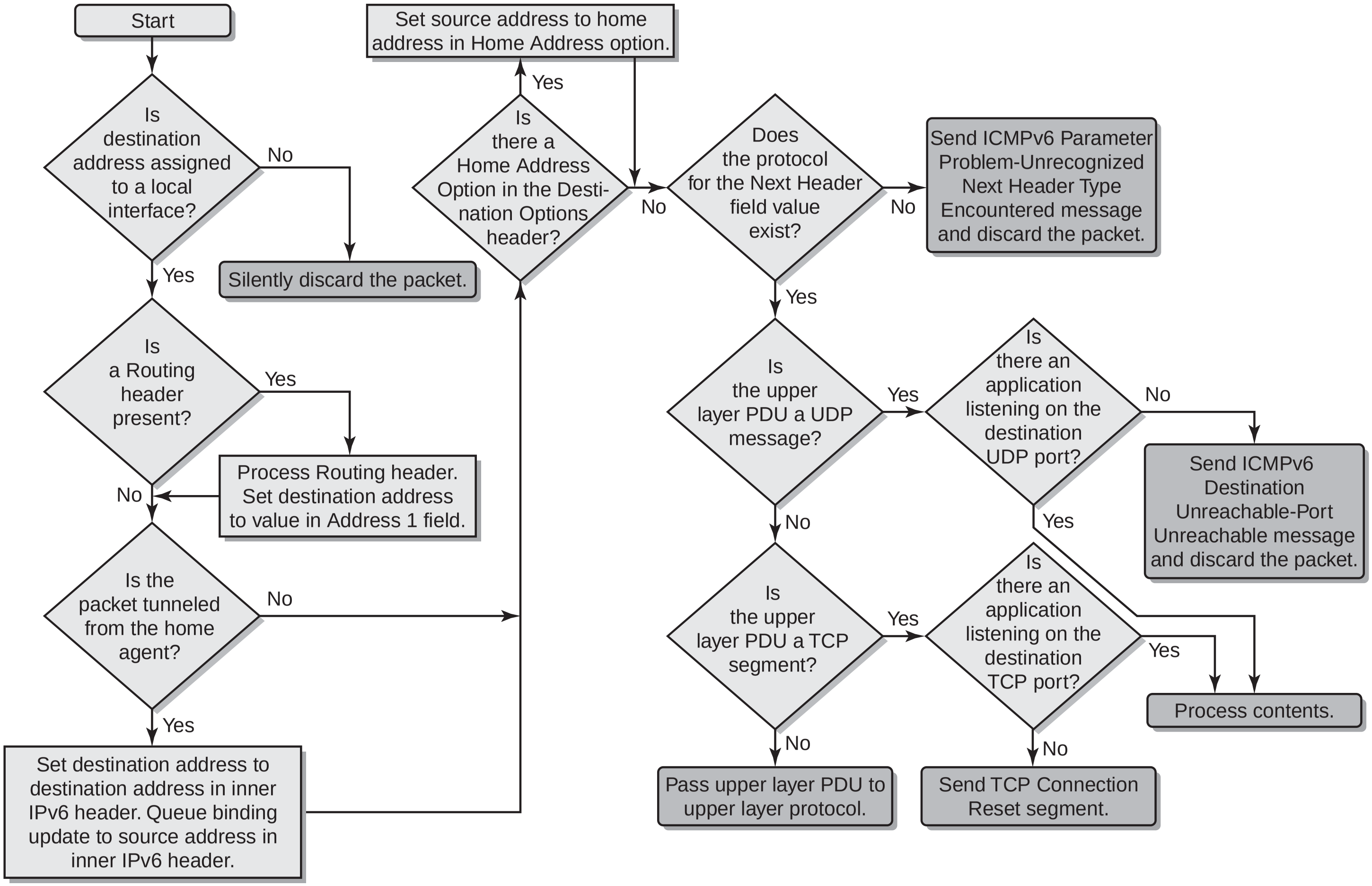
Start
Is destination address assigned to a local interface?
No
Silently discard the packet.
Yes
Is a Routing header present?
Yes
Process Routing header. Set destination address to value in Address 1 field.
No
Is the packet tunneled from the home agent?
No
Yes
Set destination address to destination address in inner IPv6 header. Queue binding update to source address in inner IPv6 header.
Is there a Home Address Option in the Destination Options header?
Yes
Set source address to home address in Home Address option.
No
Does the protocol for the Next Header field value exist?
No
Send ICMPv6 Parameter Problem-Unrecognized Next Header Type Encountered message and discard the packet.
Yes
Is the upper layer PDU a UDP message?
Yes
Is there an application listening on the destination UDP port?
No
Send ICMPv6 Destination Unreachable-Port Unreachable message and discard the packet.
Yes
No
Is the upper layer PDU a TCP segment?
Yes
Is there an application listening on the destination TCP port?
Yes
Process contents.
No
Send TCP Connection Reset segment.
No
Pass upper layer PDU to upper layer protocol.

Figure 12-22. *The IPv6 mobility host receiving process*

7. What kinds of packets are sent between the correspondent node and the home agent?

8. Describe the addressing in the IPv6 header, and the sequence of IPv6 extension headers and their contents, for a packet sent by a mobile node that is away from home to another mobile node that is away from home for which a binding cache entry is present.

9. When does the mobile node send a binding update to the home agent? When does the mobile node send a binding update to the correspondent node?

10. How does a mobile node determine when it has returned home?

11. How does the mobile node avoid duplicate address conflicts when it returns home?