



Fondamenti di Internet

IPv6 Basics



Foreword

- In the 1980s, the Internet Engineering Task Force (IETF) released RFC 791 – Internet Protocol, which marks the standardization of IPv4. In the following decades, IPv4 has become one of the most popular protocols. Numerous people have developed various applications based on IPv4 and made various supplements and enhancements to IPv4, enabling the Internet to flourish.
- However, with the expansion of the Internet and the development of new technologies such as 5G and Internet of Things (IoT), IPv4 faces more and more challenges. It is imperative to replace IPv4 with IPv6.
- This course describes the reasons for IPv4-to-IPv6 transition and basic IPv6 knowledge.

- Internet Protocol version 4 (IPv4): a current IP version. An IPv4 address is 32 bits in length and is usually represented by four octets written in dotted decimal notation. Each IPv4 address consists of a network number, an optional subnet number, and a host number. The network and subnet numbers together are used for routing, and the host number is used to address an individual host within a network or subnet.
- Internet Protocol version 6 (IPv6): a set of specifications designed by the IETF. It is an upgraded version of IPv4. IPv6 is also called IP Next Generation (IPng). IPv6 addresses are extended to 128 bits in length.



Objectives

- On completion of this course, you will be able to:
 - Summarize the advantages of IPv6 over IPv4.
 - Describe the basic concepts of IPv6.
 - Describe the formats and functions of IPv6 packet headers.
 - Describe the IPv6 address format and address types.
 - Describe the method and basic procedure for configuring IPv6 addresses.
 - Configure IPv6 addresses and IPv6 static routes.



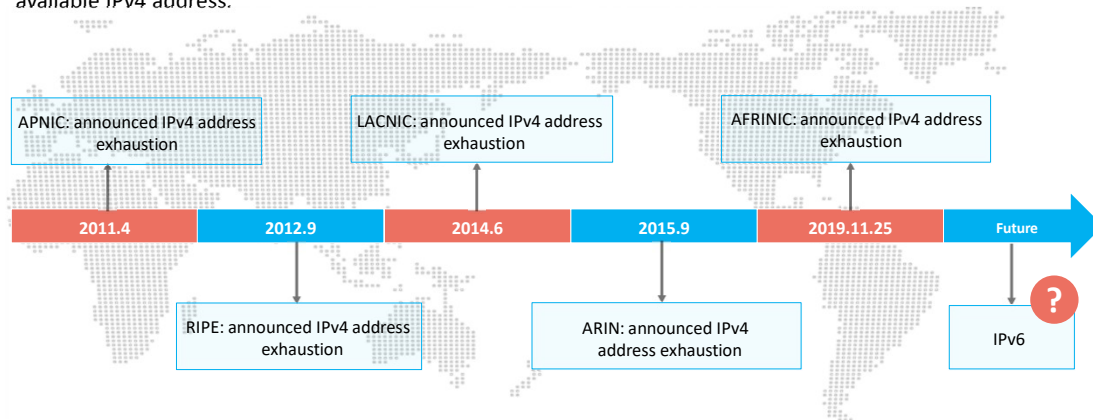
Contents

1. **IPv6 Overview**
2. IPv6 Address Configuration
3. Typical IPv6 Configuration Examples



IPv4 Status

- On February 3, 2011, the Internet Assigned Numbers Authority (IANA) announced even allocation of its last 4.68 million IPv4 addresses to five Regional Internet Registries (RIRs) around the world. The IANA thereafter had no available IPv4 address.



Page 4

- The IANA is responsible for assigning global Internet IP addresses. The IANA assigns some IPv4 addresses to continent-level RIRs, and then each RIR assigns addresses in its regions. The five RIRs are as follows:
 - RIPE: Reseaux IP Europeans, which serves Europe, Middle East, and Central Asia.
 - LACNIC: Latin American and Caribbean Internet Address Registry, which serves the Central America, South America, and the Caribbean.
 - ARIN: American Registry for Internet Numbers, which serves North America and some Caribbean regions.
 - AFRINIC: Africa Network Information Center, which serves Africa.
 - APNIC: Asia Pacific Network Information Centre, which serves Asia and the Pacific.
- IPv4 has proven to be a very successful protocol. It has survived the development of the Internet from a small number of computers to hundreds of millions of computers. But the protocol was designed decades ago based on the size of the networks at that time. With the expansion of the Internet and the launch of new applications, IPv4 has shown more and more limitations.
- The rapid expansion of the Internet scale was unforeseen at that time. Especially over the past decade, the Internet has experienced explosive growth and has been accessed by numerous households. It has become a necessity in people's daily life. Against the Internet's rapid development, IP address depletion becomes a pressing issue.
- In the 1990s, the IETF launched technologies such as Network Address Translation (NAT) and Classless Inter-Domain Routing (CIDR) to delay IPv4 address exhaustion. However, these transition solutions can only slow down the speed of address exhaustion, but cannot fundamentally solve the problem.



Why IPv6?

IPv4

Exhausted public IP addresses
Improper packet header design
Large routing table, leading to inefficient
table query
Dependency on ARP causes broadcast
storms
...

vs.

IPv6

Nearly infinite address space
Hierarchical address allocation
Plug-and-play
Simplified packet header
IPv6 security features
Integrity of E2E communication
Support for mobility
Enhanced QoS features
...



IPv6 Advantages

Nearly infinite address space

The 128-bit address length provides numerous addresses, meeting the requirements of emerging services such as the IoT and facilitating service evolution and expansion.

Hierarchical address structure

IPv6 addresses are allocated more properly than IPv4 addresses, facilitating route aggregation (reducing the size of IPv6 routing tables) and fast route query.

Plug-and-play

IPv6 supports stateless address autoconfiguration (SLAAC), simplifying terminal access.

Simplified packet header

The simplified packet header improves forwarding efficiency. New applications can be supported using extension headers, which facilitate the forwarding processing of network devices and reduce investment costs.

Security features

IPsec, source address authentication, and other security features ensure E2E security, preventing NAT from damaging the integrity of E2E communication.

Mobility

Greatly improves real-time communication and performance of mobile networks.

Enhanced QoS features

A Flow Label field is additionally defined and can be used to allocate a specific resource for a special service and data flow.

- **Nearly infinite address space:** This is the most obvious advantage over IPv4. An IPv6 address consists of 128 bits. The address space of IPv6 is about 8×10^{28} times that of IPv4. It is claimed that IPv6 can allocate a network address to each grain of sand in the world. This makes it possible for a large number of terminals to be online at the same time and unified addressing management, providing strong support for the interconnection of everything.
- **Hierarchical address structure:** IPv6 addresses are divided into different address segments based on application scenarios thanks to the nearly infinite address space. In addition, the continuity of unicast IPv6 address segments is strictly required to prevent "holes" in IPv6 address ranges, which facilitates IPv6 route aggregation to reduce the size of IPv6 address tables.
- **Plug-and-play:** Any host or terminal must have a specific IP address to obtain network resources and transmit data. Traditionally, IP addresses are assigned manually or automatically using DHCP. In addition to the preceding two methods, IPv6 supports SLAAC.
- **E2E network integrity:** NAT used on IPv4 networks damages the integrity of E2E connections. After IPv6 is used, NAT devices are no longer required, and online behavior management and network monitoring become simple. In addition, applications do not need complex NAT adaptation code.
- **Enhanced security:** IPsec was initially designed for IPv6. Therefore, IPv6-based protocol packets (such as routing protocol packets and neighbor discovery packets) can be encrypted in E2E mode, despite the fact that this function is not widely used currently. The security capability of IPv6 data plane packets is similar to that of IPv4+IPsec.
- **High scalability:** IPv6 extension headers are not a part of the main data packet. However, if necessary, the extension headers can be inserted between the basic IPv6 header and the valid payload to assist IPv6 in encryption, mobility, optimal path selection, and QoS, improving packet forwarding efficiency.
- **Improved mobility:** When a user moves from one network segment to another on a traditional network, a typical triangle route is generated. On an IPv6 network, the communication traffic of such

mobile devices can be directly routed without the need of the original triangle route. This feature reduces traffic forwarding costs and improves network performance and reliability.

- Enhanced QoS: IPv6 reserves all QoS attributes of IPv4 and additionally defines a 20-byte Flow Label field for applications or terminals. This field can be used to allocate specific resources to special services and data flows. Currently, this mechanism has not been fully developed and applied yet.



Basic IPv6 Header

- An IPv6 header consists of a mandatory basic IPv6 header and optional extension headers.
- The basic header provides basic information for packet forwarding and is parsed by all devices on a forwarding path.

IPv4 packet header (20–60 bytes)

Version	IHL	ToS	Total Length	
Identification			Flags	Fragment Offset
TTL		Protocol	Head Checksum	
Source Address				
Destination Address				
Options				Padding

Basic IPv6 header (40 bytes)

Version	Traffic Class	Flow Label	
Payload Length		Next Header	Hop Limit
Source Address			
Destination Address			

Deleted

Reserved

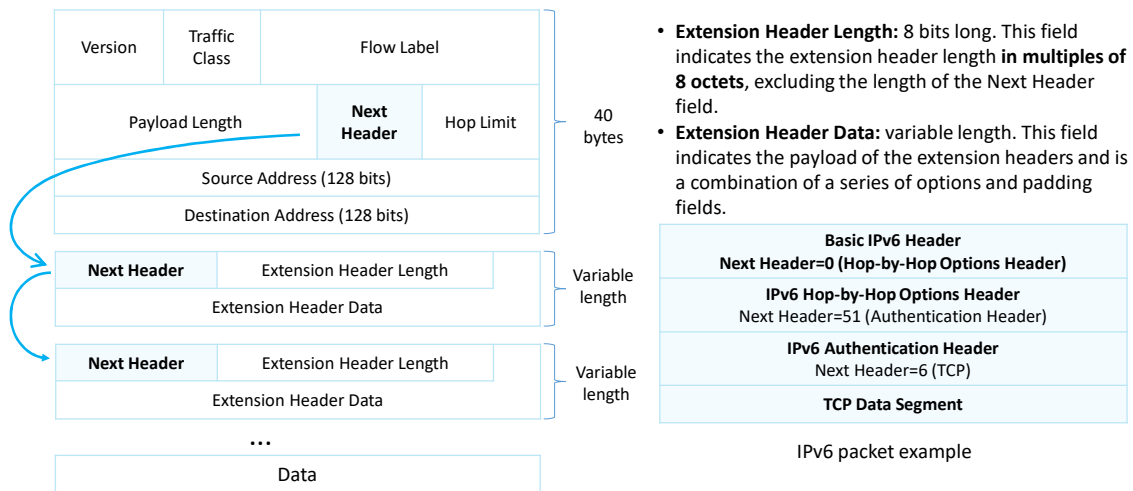
Name/Location
changed

New

- The fields in a basic IPv6 header are described as follows:
 - Version: 4 bits long. In IPv6, the value is 6.
 - Traffic Class: 8 bits long. This field indicates the class or priority of an IPv6 packet. It is similar to the TOS field in an IPv4 packet and is mainly used in QoS control.
 - Flow Label: 20 bits long. This field was added in IPv6 to differentiate real-time traffic. A flow label and a source IP address together can identify a unique data flow. Intermediate network devices can effectively differentiate data flows based on this field.
 - Payload Length: 16 bits long. This field indicates the length of the part (namely, extension headers and upper-layer PDU) in an IPv6 packet following the IPv6 basic header.
 - Next Header: 8 bits long. This field defines the type of the first extension header (if any) following a basic IPv6 header or the protocol type in an upper-layer PDU (similar to the Protocol field in IPv4).
 - Hop Limit: 8 bits long. This field is similar to the Time to Live field in an IPv4 packet. It defines the maximum number of hops that an IP packet can pass through. The value is decreased by 1 each time an IP packet passes through a node. The packet is discarded if Hop Limit is decreased to zero.
 - Source Address: 128 bits long. This field indicates the address of the packet sender.
 - Destination Address: 128 bits long. This field indicates the address of the packet receiver.



IPv6 Extension Header



- An IPv4 packet header carries the optional Options field, which can represent security, timestamp, or record route options. The Options field extends the IPv4 packet header from 20 bytes to 60 bytes. The Options field needs to be processed by all the intermediate devices, consuming a large number of resources. For this reason, this field is seldom used in practice.
- IPv6 removes the Options field from the basic header and puts it in the extension headers, which are placed between a basic IPv6 header and upper-layer PDU. An IPv6 packet may carry zero, one, or more extension headers. A sender adds one or more extension headers to a packet only when the sender requests the destination device or other devices to perform special handling. The length of IPv6 extension headers is not limited to 40 bytes so that new options can be added later. This feature together with the option processing modes enables the IPv6 options to be leveraged. To improve extension header processing efficiency and transport protocol performance, the extension header length, however, is always an integer multiple of 8 bytes.
- When multiple extension headers are used, the Next Header field of the preceding header indicates the type of the current extension header. In this way, a chained packet header list is formed.
- When more than one extension header is used in the same IPv6 packet, those headers must appear in the following order:
 1. Hop-by-Hop Options header: carries optional information that must be examined by every node along a packet's delivery path.
 2. Destination Options header: carries optional information that needs to be examined only by a packet's destination node.
 3. Routing header: used by an IPv6 source to list one or more intermediate nodes to be "visited" on the way to a packet's destination.

4. Fragment header: used by an IPv6 source to send a packet longer than the path MTU to its destination.
5. Authentication header (AH): used by IPsec to provide authentication, data integrity, and replay protection.
6. Encapsulating Security Payload (ESP) header: used by IPsec to provide authentication, data integrity, replay protection, and confidentiality of IPv6 packets.

Extension headers

- Processed only by node identified in IPv6 Destination Address field => much lower overhead than IPv4 options
 - exception: Hop-by-Hop Options header, which carries information that must be examined and processed by every node along a packet's delivery path, including the source and destination nodes (value zero in the Next Header field).

Currently defined Headers should appear in the following order:

- IPv6 header
- Hop-by-Hop Options header
- Destination Options header (for options to be processed by the first destination that appears in the IPv6 Destination Address field plus subsequent destinations listed in the Routing header)
- Routing header
- Fragment header
- Authentication header
- Encapsulating Security Payload header
- Destination Options header (for options to be processed only by the final destination of the packet.)
- upper-layer header

Page 9

- Hop-by-Hop: Carries options that may be examined by intermediate nodes along the forwarding path. Since this header may be read by all routers along the path, it is useful for transmitting management information or debugging commands to routers.
 - The following Hop-by-Hop Options are defined: Jumbo Payload [[RFC2675](#)] Path MTU Record Option [[I-D.ietf-6man-mtu-option](#)] RPL Option [[I-D.ietf-roll-useofrplinfo](#)] ***Quick-Start [[RFC4782](#)] CALIPSO [[RFC5570](#)] ***SMF_DPD [[RFC6621](#)] ***ILNP Nonce [[RFC6744](#)] ***MPL Option [[RFC7731](#)]
- Destination: Specifies packet delivery parameters for intermediate destinations or final destination.
- Routing: Specify a source route; a list of intermediate destinations for the packet to travel to on its path to the final destination(similar to option function in IPv4).
- Fragmentation: Source node uses this option if the packets sent to a particular destination are too large to fit in the maximum size allowed by the links along the path; this is known as Maximum Transmitted Unit (MTU).
- Authentication: Provides data authentication and integrity assurance for IPv6 packets. It also provides protection against replay but not confidentiality.
- Encapsulating Security Payload: Provides data confidentiality, data authentication, and data integrity services to the encrypted payload; not including the header.
- With one exception, extension headers are not examined or processed by any node along a packet's delivery path, until the packet reaches the node (or each of the set of nodes, in the case of multicast) identified in the Destination Address field of the IPv6 header. There, normal demultiplexing on the Next Header field of the

IPv6 header invokes the module to process the first extension header, or the upper-layer header if no extension header is present. The contents and semantics of each extension header determine whether or not to proceed to the next header. Therefore, extension headers must be processed strictly in the order they appear in the packet; a receiver must not, for example, scan through a packet looking for a particular kind of extension header and process that header prior to processing all preceding ones.



IPv6 Addressing

Version	Address size	Total Number of Addresses
IPv4	32 bit	4,294,967,296
IPv6	128 bit	340,282,366,920,938,463,463,374,607,431,768,211,456

- Exhaustion of the limited IPv4 address space.
- IPv6 addressing implemented to resolve address shortages.

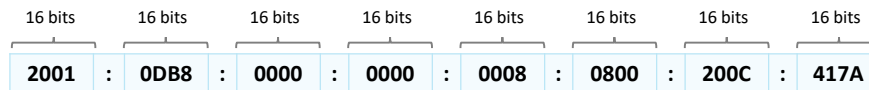
Page 10

- As a set of specifications defined by the Internet Engineering Task Force (IETF), Internet Protocol version 6 (IPv6) is the next-generation network layer protocol standard and the successor to Internet Protocol version 4 (IPv4). The most obvious difference between IPv6 and IPv4 is that IP addresses are lengthened from 32 bits to 128 bits. In doing so, IPv6 is capable of supporting an increased number of levels of addressing hierarchy, a much greater number of addressable nodes, and simpler auto-configuration of addresses.
- The existing IPv4 address range was implemented at a time when such networks as ARPANET and the National Science Foundation Network (NSFNET) represented the mainstream backbone network, and at a time when IPv4 was considered more than ample to support the range of hosts that would connect to these forming networks. The unforeseen evolution of the Internet from these ancestral networks resulted in the rapid consumption of the IPv4 address space of 4.3 billion addresses (of which many are reserved), for which counter measures in the form of NAT and CIDR were put in place to alleviate the expansion, and give time for a more permanent solution to be formed. Additionally, early IPv4 network address allocation was highly discontinuous making it difficult for addressing to be clustered into effective address groups and ease the burden on global IP routing tables used in autonomous system based routing protocols such as BGP.
- Eventually however the IP network is expected to shed its IPv4 addressing to make way for IPv6 and provide the addressing capacity of over 340 undecillion unique addresses, considered more than necessary for continued IP network growth. Along with this, address allocation by the Internet Assigned Numbers Authority (IANA) ensures that address allocation for IPv6 is contiguous for efficient future management of IP routing tables.



IPv6 Address

- The length of an IPv6 address is 128 bits. Colons are generally used to divide the IPv6 address into eight segments. Each segment contains 16 bits and is expressed in hexadecimal notation.



The letters in an IPv6 address are case insensitive. For example, **A** is equivalent to **a**.

- Similar to an IPv4 address, an IPv6 address is expressed in the format of IPv6 address/mask length.
 - Example: 2001:0DB8:2345:CD30:1230:4567:89AB:CDEF/64

IPv6 address: 2001:0DB8:2345:CD30:1230:4567:89AB:CDEF

Subnet number: 2001:0DB8:2345:CD30::/64



IPv6 Address Abbreviation Specifications

- For convenience, IPv6 can be abbreviated according to the following rules.

Abbreviation Specifications

2001 : 0DB8 : 0000 : 0000 : 0008 : 0800 : 200C : 417A

The leading 0s in each 16-bit segment can be omitted. However, if all bits in a 16-bit segment are 0s, at least one 0 must be reserved. The trailing 0s cannot be omitted.



2001 : DB8 : 0 : 0 : 8 : 800 : 200C : 417A

If one or more consecutive 16-bit segments contain only 0s, a double colon (::) can be used to represent them, but only one :: is allowed in an entire IPv6 address.



2001 : DB8 : :: 8 : 800 : 200C : 417A

If an abbreviated IPv6 address contains two double colons (::), the IPv6 address cannot be restored to the original one.

Abbreviation Examples

Before 0000:0000:0000:0000:0000:0000:0000:0001 After
::1

Before 2001:0DB8:0000:0000:FB00:1400:5000:45FF After
2001:DB8::FB00:1400:5000:45FF

Before 2001:0DB8:0000:0000:0000:2A2A:0000:0001 After
2001:DB8::2A2A:0:1

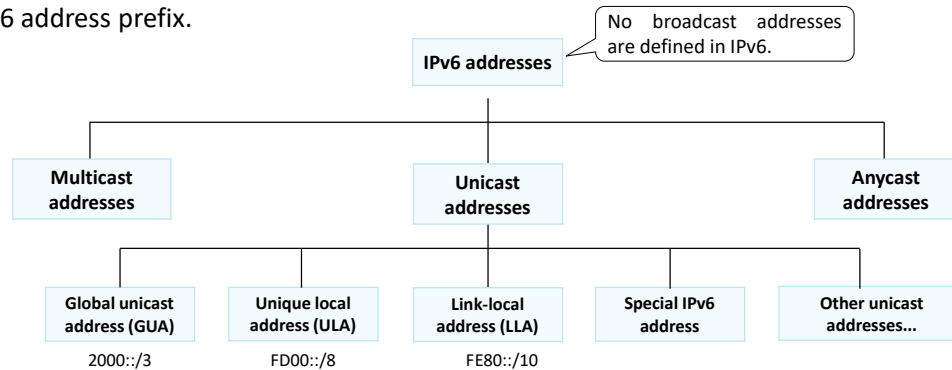
Before 2001:0DB8:0000:1234:FB00:0000:5000:45FF After
2001:DB8::1234:FB00:0:5000:45FF

or 2001:DB8:0:1234:FB00::5000:45FF



IPv6 Address Classification

- IPv6 addresses are classified into unicast, multicast, and anycast addresses according to the IPv6 address prefix.



Page 13

- Unicast address: identifies an interface. A packet destined for a unicast address is sent to the interface having that unicast address. In IPv6, an interface may have multiple IPv6 addresses. In addition to GUAs, ULAs, and LLAs, IPv6 has the following special unicast addresses:
 - Unspecified address: 0:0:0:0:0:0:0:0/128, or ::/128. The address is used as the source address of some packets, for example, Neighbor Solicitation (NS) messages sent during DAD or request packets sent by a client during DHCPv6 initialization.
 - Loopback address: 0:0:0:0:0:0:0:1/128, or ::1/128, which is used for local loopback (same function as 127.0.0.1 in IPv4). The data packets sent to ::1 are actually sent to the local end and can be used for loopback tests of local protocol stacks.
- Multicast address: identifies multiple interfaces. A packet destined for a multicast address is sent to all the interfaces joining in the corresponding multicast group. Only the interfaces that join a multicast group listen to the packets destined for the corresponding multicast address.
- Anycast address: identifies a group of network interfaces (usually on different nodes). A packet sent to an anycast address is routed to the nearest interface having that address, according to the router's routing table.
- IPv6 does not define any broadcast address. On an IPv6 network, all broadcast application scenarios are served by IPv6 multicast.

See <https://www.iana.org/assignments/ipv6-unicast-address-assignments/ipv6-unicast-address-assignments.xhtml>



IPv6 Unicast Address Format

IPv6 Unicast
Address

IPv6 Multicast
Address


IPv6 Anycast
Address

- An IPv6 unicast address is composed of two parts:
 - Network prefix: consists of n bits and is parallel to the network ID of an IPv4 address.
 - Interface ID: consists of $(128 - n)$ bits and is parallel to the host ID of an IPv4 address.
- Common IPv6 unicast addresses, such as GUAs and LLAs, require that the network prefix and interface ID be 64 bits.



- Global unicast addresses that start with binary value 000 can use a non-64-bit network prefix. Such addresses are not covered in this course.

IPv6 Unicast Address
IPv6 Multicast Address
IPv6 Anycast Address



Interface ID of an IPv6 Unicast Address

- 3 methods to generate an interface ID:
 - Manual configuration
 - Automatic generation by the system
 - Using the Extended Unique Identifier (EUI)-64 standard
- EUI-64 is most commonly used. It converts the MAC address of an interface into an IPv6 interface ID.

MAC address (hexadecimal) 3C-52-82-49-7E-9D

MAC address (binary) 00111100-10010010-10000010 - 01001001-01111110-10011101

1 Bit 7 inversion
2 Insert FFFE

EUI-64 ID (binary) 00111110-10010010-10000010-11111111-11111110-01001001-01111110-10011101

EUI-64 ID (hexadecimal) 3E-52-82-FF-FE-49-7E-9D

Page 15

- An interface ID is 64 bits long and is used to identify an interface on a link. The interface ID must be unique on each link. The interface ID is used for many purposes. Most commonly, an interface ID is attached to a link-local address prefix to form the link-local address of the interface. It can also be attached to an IPv6 global unicast address prefix in SLAAC to form the global unicast address of the interface.
- IEEE EUI-64 standard
 - Converting MAC addresses into IPv6 interface IDs reduces the configuration workload. Especially, you only need an IPv6 network prefix in SLAAC to form an IPv6 address.
 - The defect of this method is that IPv6 addresses can be deducted by attackers based on MAC addresses.



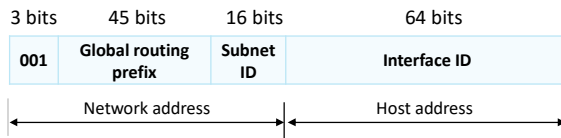
Common IPv6 Unicast Address - GUA

IPv6 Unicast
Address

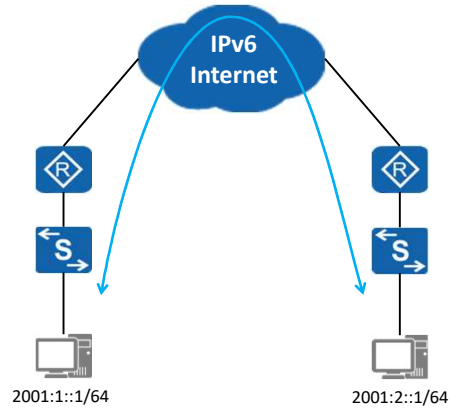
IPv6 Multicast
Address

IPv6 Anycast
Address

- A GUA is also called an aggregatable GUA. This type of address is globally unique and is used by hosts that need to access the Internet. It is equivalent to a public IPv4 address.



- The network address and interface ID of a GUA are each generally 64 bits long.
- Global routing prefix: is assigned by a provider to an organization and is generally at least 45 bits.
- Subnet ID: An organization can divide subnets based on network requirements.
- Interface ID: identifies a device's interface.



- You can apply for a GUA from a carrier or the local IPv6 address management organization.



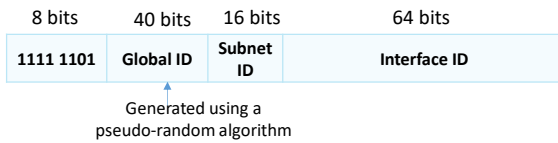
Common IPv6 Unicast Address - ULA

IPv6 Unicast
Address

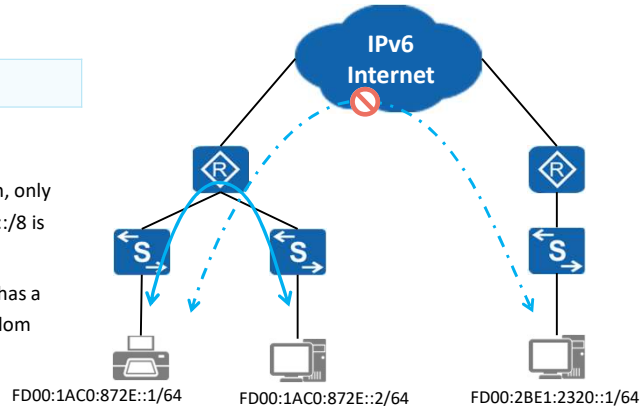
IPv6 Multicast
Address

IPv6 Anycast
Address

- A ULA is a private IPv6 address that can be used only on an intranet. This type of address cannot be routed on an IPv6 public network and therefore cannot be used to directly access a public network.



- ULAs use the FC00::/7 address segment, among which, only the FD00::/8 address segment is currently used. FC00::/8 is reserved for future expansion.
- Although a ULA is valid only in a limited range, it also has a globally unique prefix (generated using a pseudo-random algorithm, low conflict probability).



- A unique local address (ULA) is an Internet Protocol version 6 (IPv6) address in the address range fc00::/7.[1] Its purpose in IPv6 is analogous to IPv4 private network addressing. Unique local addresses may be used freely, without centralized registration, inside a single site or organization or spanning a limited number of sites or organizations. They are routable only within the scope of such private networks, but not in the global IPv6 Internet.

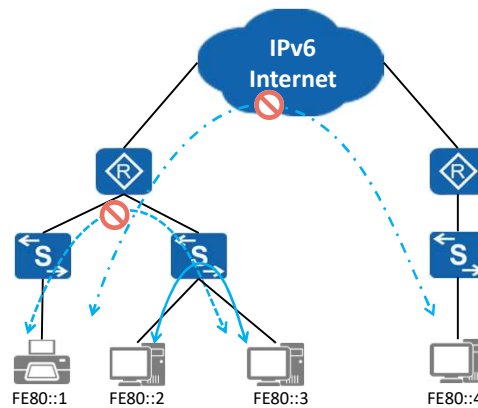


Common IPv6 Unicast Address - LLA

- An LLA is another type of IPv6 address with limited application scope. The valid range of the LLA is the local link, with the prefix of FE80::/10.

10 bit	54 bit	64 bit
1111 1110 10	0	Interface ID
Fixed at 0		

- An LLA is used for communication on a single link, such as during IPv6 SLAAC and IPv6 neighbor discovery.
- Data packets with the source or destination IPv6 address being an LLA are not forwarded out of the originating link. In other words, the valid scope of an LLA is the local link.
- Each IPv6 interface must have an LLA. Huawei devices support automatic generation and manual configuration of LLAs.



IPv6 Unicast Address
IPv6 Multicast Address
IPv6 Anycast Address

IPv6 Multicast Address

- An IPv6 multicast address identifies multiple interfaces and is generally used in one-to-many communication scenarios.
- An IPv6 multicast address can be used only as the destination address of IPv6 packets.

8 bits	4 bits	4 bits	80 bits	32 bits
11111111	Flags	Scope	Reserved (must be 0)	Group ID

- Flags:** indicates a permanent or transient multicast group.
- Scope:** indicates the multicast group scope.
- Group ID:** indicates a multicast group ID.

Address Range	Description
FF02::1	All Nodes Addresses (Link Local)
FF02::2	All Routers Addresses (Link Local)

Page 19

- Types and scope of IPv6 multicast groups:

- Flags:

+--+--+--+

flags is a set of four flags: | 0 | R | P | T |

+--+--+--+

Multicast address flags

bit	flag	Meaning when 0	Meaning when 1
8	<i>reserved</i>	<i>reserved</i>	<i>reserved</i>
9	R (Rendezvous)	Rendezvous point not embedded	Rendezvous point embedded

The embedded RP feature that works for IPv6 multicast is a cool trick that **embeds the IPv6 address of the RP within the IPv6 multicast group address**.

By doing this, multicast-enabled routers can extract the RP address just by looking at the multicast group address and using it for a shared tree.

10	P (Prefix)	Without prefix information	Address based on network prefix
----	------------	----------------------------	---------------------------------

The P flag indicates whether the multicast address is **based on a unicast address** or not.

11	T (Transient)	Well-known multicast address	Dynamically assigned multicast address
----	---------------	------------------------------	--

The [four-bit scope field](#) (sc) is used to indicate where the address is valid and unique.

▫ Scope:

- 0: reserved
- 1: interface-local scope, which spans only a single interface on a node and is useful only for loopback transmission of multicast
- 2: link-local scope (for example, FF02::1)
- 3: realm local: Realm-local scope is defined as larger than link-local, automatically determined by network topology and must not be larger than the following scopes
- 4: admin-local: Admin-local scope is the smallest scope that must be administratively configured, i.e., not automatically derived from physical connectivity or other, non-multicast-related configuration.
- 5: site-local scope: Site-local scope is intended to span a single site belonging to an organisation.
- 8: organization-local scope: Organization-local scope is intended to span all sites belonging to a single organization.
- E: global scope
- F: reserved

IPv6 Unicast Address
IPv6 Multicast Address
IPv6 Anycast Address

Solicited-Node Multicast Address

- If a node has an IPv6 unicast or anycast address, a solicited-node multicast address is generated for the address, and the node joins the corresponding multicast group. This address is used for **neighbor discovery** and **duplicate address detection (DAD)**. A solicited-node multicast address is **valid only on the local link**.

IPv6 unicast or anycast address	64 bits						64 bits					
	IPv6 Address Prefix						Interface ID					
	24 bits copied											
Corresponding solicited-node multicast address	FF02	0000	0000	0000	0000	0001	FF					
	104 bits (fixed prefix)							24 bits				

Page 20

- An application scenario example of a solicited-node multicast group address is as follows: In IPv6, ARP and broadcast addresses are canceled. When a device needs to request the MAC address corresponding to an IPv6 address, the device still needs to send a request packet, which is a multicast packet. The destination IPv6 address of the packet is the solicited-node multicast address corresponding to the target IPv6 unicast address. Because only the target node listens to the solicited-node multicast address, the multicast packet is received only by the target node, without affecting the network performance of other non-target nodes.



IPv6 over Ethernet (and other links)

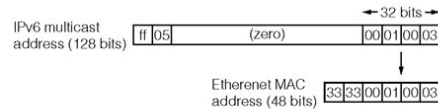
- IPv6 makes heavy use of Ethernet for multicast transmissions.
- A multicast Ethernet packet is one that's potentially addressed to more than one host. The set of hosts it's addresses to is called a multicast group, and is identified by a 48-bit address, just like an individual host is. A multicast address is distinguished by having **the least-significant bit of its first byte set**, so *00-C0-4F-68-12-CB* is a unicast address, while *33-33-FF-68-12-CB* is a multicast address. Note that *FF-FF-FF-FF-FF-FF*, the Ethernet broadcast address, is actually just a special multicast address.
- An Ethernet interface has to listen not only for packets addressed to its unicast address, but also for packets addressed to any of the multicast groups of which it's a member.



IPv6 Ethernet encapsulation (RFC 2464)

- IPv6 packets are encapsulated in Ethernet packets just like IPv4 packets, but with a new Ethertype (*86DD* rather than *0800*)
- To send an IPv6 multicast packet over Ethernet, one simply takes the last 32 bits of the destination IPv6 address, prepends **33-33-** and uses that as the destination Ethernet address. Thus, an IPv6 packet addressed to *FF02::1:FF68:12CB* would be sent to the Ethernet address *33-33-FF-68-12-CB*.
- Any host which is interested in packets for that IPv6 address is expected to be listening for the corresponding Ethernet address.

3333 Coyote Hill Road, Palo Alto,
California, is the address of XEROX PARC



IPv6 to Ethernet multicast address mapping.

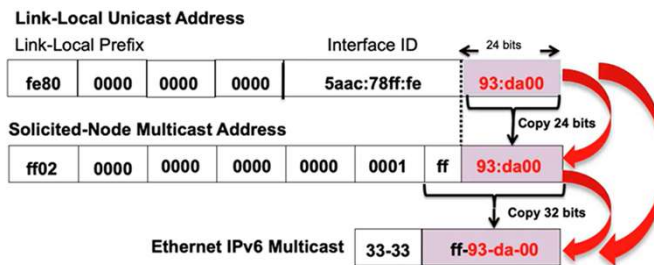
example, the multicast address for DHCPv6

- See https://www-uxsup.csx.cam.ac.uk/courses/moved.ipv6_basics/x84.html



Solicited-Node Multicast Address over Ethernet

- Use of LLA



- Use of GUA





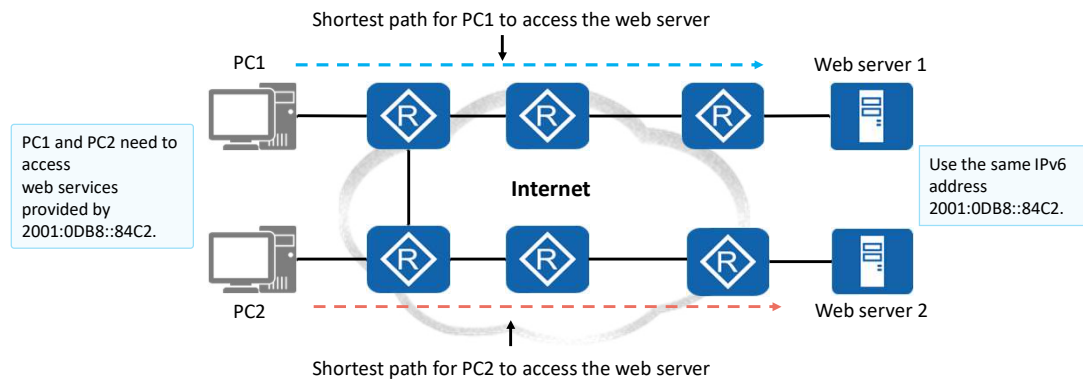
IPv6 Anycast Address

IPv6 Unicast
Address

IPv6 Multicast
Address

IPv6 Anycast
Address

- An anycast address identifies a group of network interfaces, which usually belong to different nodes. An anycast address can be used as the source or destination address of IPv6 packets.



Page 24

- The anycast process involves an anycast packet initiator and one or more responders.
 - An initiator of an anycast packet is usually a host requesting a service (for example, a web service).
 - The format of an anycast address is the same as that of a unicast address. A device, however, can send packets to multiple devices with the same anycast address.
- Anycast addresses have the following advantages:
 - Provide service redundancy. For example, a user can obtain the same service (for example, a web service) from multiple servers that use the same anycast address. These servers are all responders of anycast packets. If no anycast address is used and one server fails, the user needs to obtain the address of another server to establish communication again. If an anycast address is used and one server fails, the user can automatically communicate with another server that uses the same address, implementing service redundancy.
 - Provide better services. For example, a company deploys two servers – one in province A and the other in province B – to provide the same web service. Based on the optimal route selection rule, users in province A preferentially access the server deployed in province A when accessing the web service provided by the company. This improves the access speed, reduces the access delay, and greatly improves user experience.



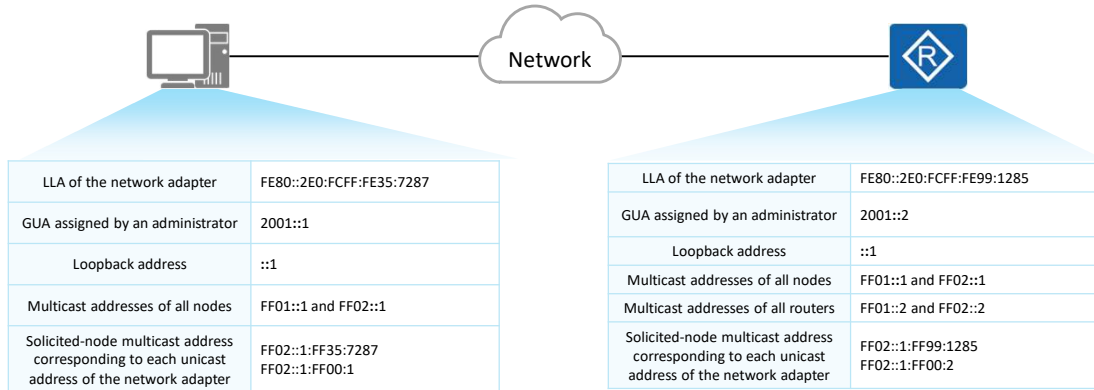
Contents

1. IPv6 Overview
- 2. IPv6 Address Configuration**
3. Typical IPv6 Configuration Examples



IPv6 Addresses of Hosts and Routers

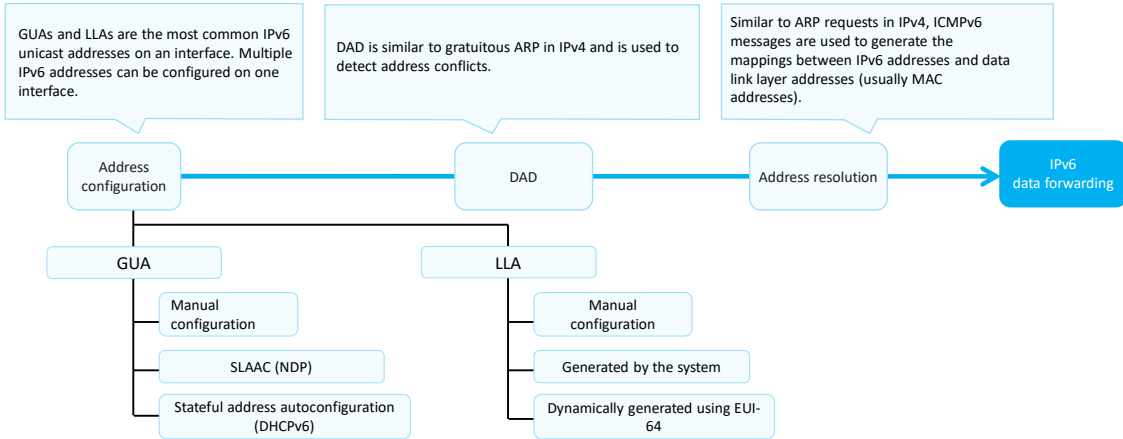
- The unicast IPv6 addresses and multicast addresses of hosts and routers are typically as follows:





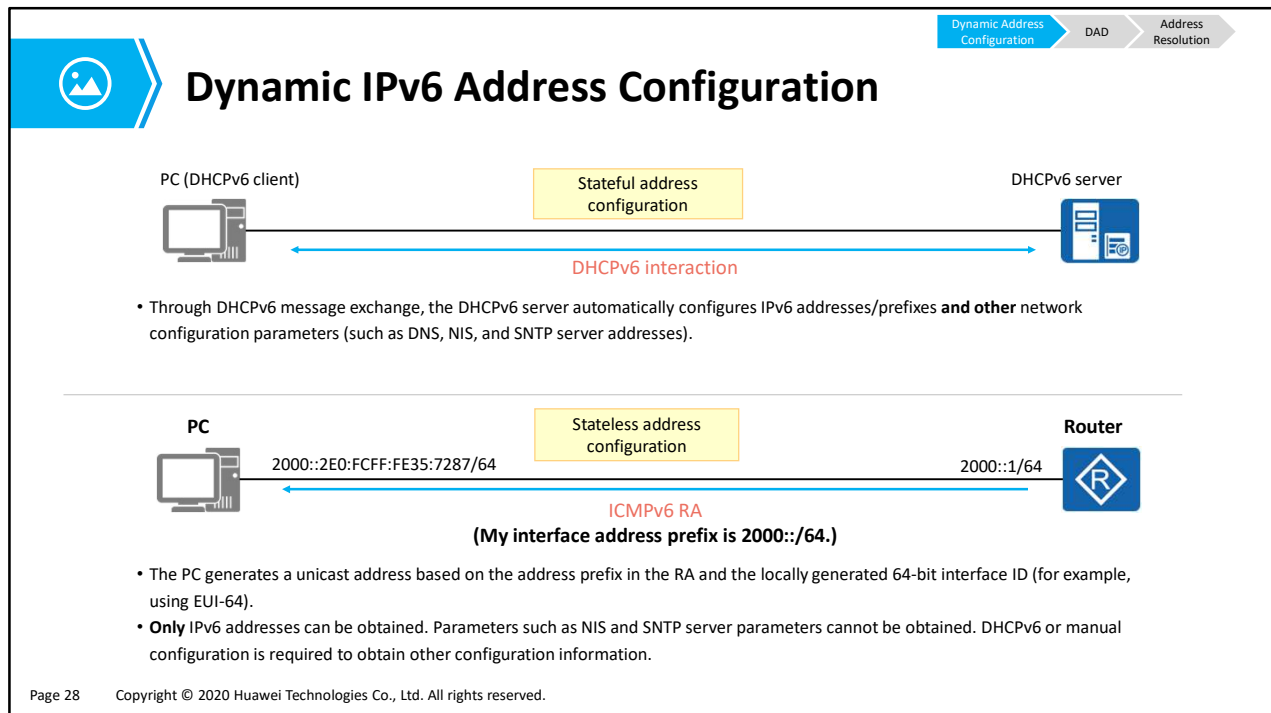
Service Process of IPv6 Unicast Addresses

- Before sending IPv6 packets, an interface undergoes address configuration, DAD, and address resolution. During this process, the ICMPv6-related Neighbor Discovery Protocol (NDP) plays an important role.



Page 27

- Stateless address auto configuration - **SLAAC**

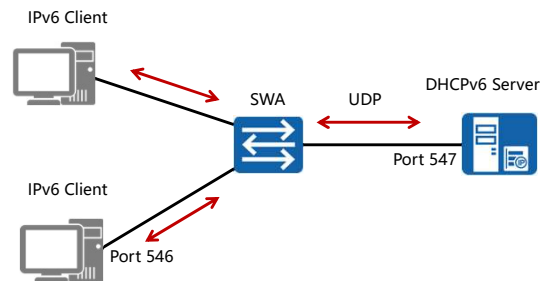


- IPv6 supports stateful and stateless address autoconfiguration. The managed address configuration flag (M flag) and other stateful configuration flag (O flag) in ICMPv6 RA messages are used to control the mode in which terminals automatically obtain addresses.
- For stateful address configuration (DHCPv6), M = 1, O = 1:
 - DHCPv6 is used. An IPv6 client obtains a complete 128-bit IPv6 address, as well as other address parameters, such as DNS and SNTP server address parameter, from a DHCPv6 server.
 - The DHCPv6 server records the allocation of the IPv6 address (this is where stateful comes).
 - This method is complex and requires high performance of the DHCPv6 server.
 - Stateful address configuration is mainly used to assign IP addresses to wired terminals in an enterprise, facilitating address management.
- For SLAAC, M = 0, O = 0:
 - ICMPv6 is used.
 - The router enabled with ICMPv6 RA periodically advertises the IPv6 address prefix of the link connected to a host.
 - Alternatively, the host sends an ICMPv6 RS message, and the router replies with an RA message to notify the link's IPv6 address prefix.
 - The host obtains the IPv6 address prefix from the RA message returned by the router and combines the prefix with the local interface ID to form a unicast IPv6 address.
 - If the host wants to obtain other configuration information, it can use DHCPv6. When DHCPv6 is used, M = 0, and O = 1.

- In SLAAC, routers do not care about the status of hosts or whether hosts are online.
 - SLAAC applies to scenarios where there are a large number of terminals that do not need other parameters except addresses. IoT is such a scenario.
- Domain name system (DNS): a mechanism that maps easy-to-remember domain names to IPv6 addresses that can be identified by network devices
- The Network Information Service, or NIS (originally called Yellow Pages or YP), is a client–server directory service protocol for distributing system configuration data such as user and host names between computers on a computer network. A NIS/YP system maintains and distributes a central directory of user and group information, hostnames, e-mail aliases and other text-based tables of information in a computer network. The NIS environment is composed of *clients* and *servers* logically grouped together in a *domain*. An NIS *server* is a host that provides configuration information to other hosts on the network.
- Simple Network Time Protocol (SNTP) is a less complex implementation of NTP, using the same protocol, but intended for primary servers equipped with a single reference clock, as well as for clients with a single upstream server and no dependent clients.



DHCPv6



- Represents a stateful address auto-configuration protocol.
- UDP based communication between client and server.
- Clients listen for DHCP messages on UDP port 546, whilst servers (and relay agents) listen for DHCP messages on UDP port 547

Page 29

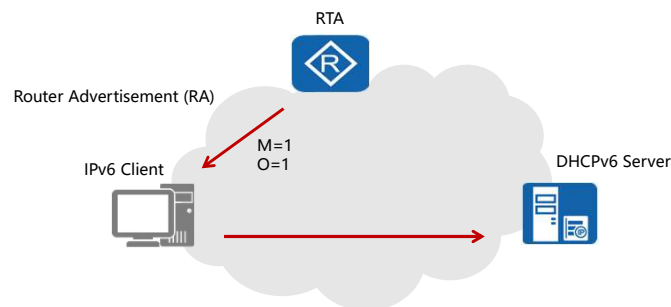
- The Dynamic Host Configuration Protocol for IPv6 (DHCPv6) is a technology that dynamically manages and configures IPv6 addresses in a centralized manner. It is designed to assign IPv6 addresses and other network configuration parameters to hosts. DHCPv6 uses the client/server model. A client requests configurations such as the IPv6 address and DNS server address from the server, the server replies with requested configurations based on policies.
- In stateless address auto-configuration (SLAAC), routers do not record the IPv6 addresses of the hosts, therefore stateless address auto-configuration has poor manageability. In addition, hosts configured with stateless address auto-configuration cannot obtain other configuration parameters such as the DNS server address. ISPs do not provide instructions for automatic allocation of IPv6 prefixes for routers. Users therefore need to manually configure IPv6 addresses for devices during IPv6 network deployment.
- As a stateful protocol for configuring IPv6 addresses automatically, DHCPv6 solves this problem. During stateful address configuration, the DHCPv6 server assigns a complete IPv6 address to a host and provides other configuration. Parameters such as the DNS server address and domain name. A relay agent may be used to forward DHCPv6 packets, however lies outside of the scope of this material. The DHCPv6 server binds the IPv6 address to a client, improving overall network manageability.
- Clients and servers exchange DHCP messages using UDP. The client uses a link-local address, determined through other mechanisms for transmitting and receiving DHCP messages. Clients listen for DHCP messages on UDP port 546, whilst servers (and relay agents) listen for DHCP messages on UDP port 547.
- The DHCPv6 server and DHCPv6 clients could be in different link scopes (that is, the DHCPv6 relay

exists).

- If the DHCPv6 server function is enabled in the interface view, configuration parameters such as IPv6 addresses are assigned only to the clients in one network segment connected to the DHCPv6 relay, because only one IPv6 address pool can be specified on an interface. If configuration parameters such as IPv6 addresses need to be assigned to the DHCPv6 clients in multiple network segments through the DHCPv6 relay, enable the DHCPv6 server function in the system view.
- The configuration method of enabling the DHCPv6 server function in the interface view is affected by the physical interface status. If the interface status is Down, the DHCPv6 server cannot successfully assign network configuration parameters to clients through the DHCPv6 relay. When the DHCPv6 server function is enabled in the system view and there are multiple reachable routes between the DHCPv6 relay and DHCPv6 server, configuration parameters such as IPv6 addresses can be assigned to clients through the DHCPv6 relay as long as one route between the DHCPv6 relay and DHCPv6 server is reachable. This improves reliability of the configuration information obtained by the clients. In addition, no configuration is required on the interface, which reduces the administrator's maintenance workload.



Stateful Addressing



- RA contains managed (M) and other (O) configuration flags.
- Stateful addressing (DHCPv6) used where flags are set to '1'.

Page 30

- Prior to the allocation of addresses, it should be clearly understood that an IPv6 node (client) is required to generate a link-local address and be successfully evaluated by the Duplicate Address Detection (DAD) process. Following this, a link router discovery process is involved, for which the IPv6 client node broadcasts a Router Solicitation (RS) message, and the link router responds with a Router Advertisement (RA) message after receiving the RS message.
- Contained within the RA message are numerous fields containing configuration parameters for the local network. One field in particular referred to as the Autoconfig Flags field, is an 8 bit field that contains two specific bit values to determine the auto-configuration process for the local network. A "managed address configuration flag" (M) is a 1 bit value that is for defining whether stateful address configuration should be used, commonly applied in the presence of a DHCPv6 server on the local network. Where the value is set to 1, stateful addressing should be implemented, meaning the IPv6 client should obtain IPv6 addressing through stateful DHCPv6.
- The other stateful configuration flag (O) represents the second flag bit value in the Autoconfig Flags field, and defines whether other network configuration parameters such as DNS and SNTP (for time management servers) should be determined through stateful DHCPv6. RFC2462 defines that where the M bit is true (a value of 1), the O bit must also be implicitly true, however in practice the M bit and the O bit may be defined interchangeably to support stateless addressing services in DHCPv6, in which an IPv6 address is not assigned but configuration parameters are.
- It should also be noted that the managed address flag and other configuration flag is managed through VRP on the router, and is not set in the RA message by default. In order to set these flags, the commands `ipv6 nd autoconfig managed-address-flag` and `ipv6 nd autoconfig other-flag` should be configured on the gateway responsible for generating RA messages.



Enabling DHCPv6 Communication

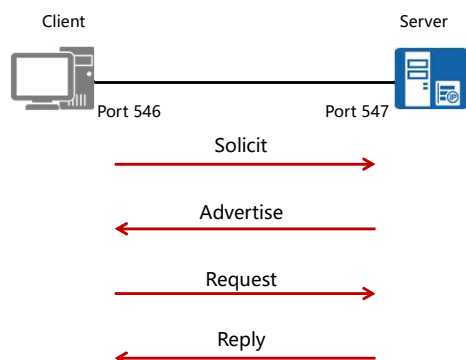


- Link-local addresses are used as source address by clients, and DHCP servers reached via the multicast address `FF02::1:2`.

- Client nodes initiated on a network supporting stateful addressing may be serviced by one or more DHCPv6 servers. The IPv6 client uses a link-local address assigned to the interface for which it is requesting configuration information as the source address in the header of the IPv6 datagram.
- The multicast address `FF02::1:2` is a reserved multicast address that represents "All_DHCP_Relay_Agents_and_Servers", and is used by a client to communicate with neighboring servers. All servers (and relay agents) are members of this multicast group. For any client sending a DHCP message to the All_DHCP_Relay_Agents_and_Servers address, it is expected that the client send the message through the interface for which configuration information is being requested, however exceptions may occur to this rule where two interfaces on the client are associated with the same link, for which it is possible for the alternative interface to be used. In either case the link local address of the forwarding interface must be used as the source address.



Assigning IPv6 Addressing



Example

- *solicit* from **[fe80::aabb:ccff:fedd:eeff]:546** to multicast address **[ff02::1:2]:547**
- *advertise* from **[fe80::0011:22ff:fe33:5566]:547** to **[fe80::aabb:ccff:fedd:eeff]:546**.
- *request* from **[fe80::aabb:ccff:fedd:eeff]:546** to **[ff02::1:2]:547**.
- *reply* from **[fe80::0011:22ff:fe33:5566]:547** to **[fe80::aabb:ccff:fedd:eeff]:54**

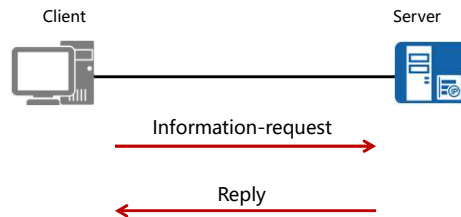
- Discovery of servers and assignment of IPv6 addresses & configuration parameter relies on a set of four messages.

Page 32

- Similar to Discover/Offer/Request/Ack of v4
- Obtaining stateful addressing and other parameters from a DHCPv6 server requires a series of messages be sent. A client initially sends a solicit message to locate servers, from which addressing and configuration parameters can be received.
- Following the solicit message, a DHCPv6 server supporting the link will generate an advertise message in response to the solicit message, that indicates to the client, the IPv6 address of the server, providing the required DHCPv6 service. The client is then capable of using this IPv6 address to reference the DHCPv6 server and generate a request message. Where multiple servers respond to the solicit message, the client will need to decide which DHCPv6 server should be used, typically defined by a server preference value defined by the DHCPv6 administrator on the server, and carried in the advertise message. Additionally the server may carry options including a server unicast option which enables the client to use the IPv6 address of the DHCPv6 server to transmit further correspondence with this server as unicast messages.
- The request message is transmitted to the selected DHCP server to request configuration parameters and one or multiple IPv6 addresses to be assigned. Finally the DHCPv6 server responds with a Reply message that contains the confirmed addresses and network configuration parameters.



Stateless Configuration Information



- Information-request used when IPv6 addressing not required.
- Reply used to deliver configuration parameters.

- DHCP may also be employed to support stateless configuration in the event where a host is capable of retrieving IPv6 addressing information through stateless configuration means, and requires only specific configuration parameters from the DHCPv6 server. In such events Information-request messages are generated, and sent by clients to a server to request configuration parameters. The client is able to obtain configuration information such as server addresses and domain information, as a list of available configuration parameters, using only a single message and reply that is exchanged with a DHCP server.
- The Information-Request message is sent to the “All_DHCP_Relay_Agents_and_Servers” multicast address following which servers respond with a Reply message containing the configuration information for the client. Since no dynamic state is being maintained (i.e. in the form of IPv6 address assignment) the allocation of configuration information is understood to be stateless.



DHCP Unique Identifier (DUID) and Identity Association Identifier (IAID)

IAID: 343516489

DUID: 00:01:00:06:51:81:03:c0:f0:de:f1:b8:e1:4d



IAID: 321334513

DUID: 00:01:00:06:50:e2:97:80:f8:1d:4f:a6:21:7f



DUID: 00:03:00:01:00:e0:fc:03:14:f1

FF02::1:2
All DHCP Relay
Agents & Servers

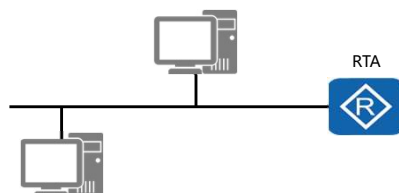
- Unique identifier of clients & servers in the DHCP community.
- Parameters bound to each DUID using Identity Associations (IA).
- DUID length can vary in the range of 12 bytes to 20 bytes, depending on the format, DUID-LL (link-layer), DUID-EN (enterprise number), or DUID-LLT, a combination of the link-layer address and a timestamp.

- **DHCPv4** uses the MAC address and an optional **Client ID** to identify the **client** for purposes of assigning an address. Each time the same client arrives on the network, it gets the same address, if possible.
- **DHCPv6** uses basically the same scheme, but makes the Client ID mandatory and imposes structure on it. The Client ID in DHCPv6 consists of two parts: a DHCP Unique Identifier (DUID) and an Identity Association Identifier (IAID). The DUID identifies the client **system** (rather than just an interface, as in DHCPv4), and the IAID identifies the **interface** on that system.
- A DHCP Unique Identifier (DUID) is a value that is used to distinguish between each client and DHCP server, for which only one DUID is present in each case. Clients may have one or multiple interfaces for which each will be assigned an IPv6 address along with other configuration parameters and is referenced using an Identity Association Identifier (IAID). These are used together with DUID to allow DHCPv6 servers to reference a client and the interface address/configuration parameters that should be assigned.
- In the case of each client, the DUID will be used to identify a specific DHCP server with which a client wishes to communicate. The length of the DUID value can vary from anywhere in the range of **96bits (12 bytes) to 160 bits (20 bytes)**, depending on the format that is used. Three such formats exist, using either the link-layer (**DUID-LL**) address, a combination of the link-layer address and enterprise number (**DUID-EN**), a vendor assigned value at the point of device manufacture, or a combination of the link-layer address and a timestamp value (**DUID-LLT**) generated at the point of DUID creation in seconds from midnight Jan 1st 2000 (GMT), modulo 232.
- The initial 16 bit values (00:01) represent the format used, where "00:01" denotes the DUID-LLT

format, “00:02” the DUID-EN format and “00:03” the DUID-LL format. In the case of the DUID-LL and DUID-LLT formats, the 16 bits immediately after represent the hardware address based on IANA hardware type parameter assignments, with 00:01 representing Ethernet (10Mb) and 00:06 defining IEEE 802 network standards. A time stamp follows in the DUID-LLT format and finally the link layer address value. For DUID-LL only the link layer address follows.



Setting the DHCP DUID



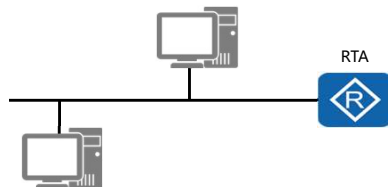
```
[RTA]dhcpv6 duid ll
Warning: The DHCP unique identifier should be globally-unique and
stable. Are you sure to change it? [Y/N]y
```

- Enables assignment of either the DUID-LL or DUID-LLT format.
- The DUID-LL format is assigned by default.

- The DUID format can be assigned through the dhcpv6 duid command, for which either the DUID-LL or DUID-LLT format can be applied. The DUID-LL format is applied by default. For the DUID-LLT, the timestamp value will reference the time from the point at which the dhcpv6 duid llt command is applied. The display dhcpv6 duid command can be used to verify the current format based primarily on the length of the DUID value, as well as the DUID value itself.



IPv6 Address Pool



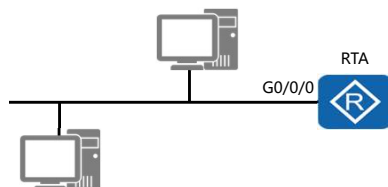
```
[RTA]dhcpv6 pool pool1
[RTA-dhcpv6-pool-pool1]address prefix 3000::/64
[RTA-dhcpv6-pool-pool1]excluded-address 3000::1
[RTA-dhcpv6-pool-pool1]dns-server 3001::1
[RTA-dhcpv6-pool-pool1]dns-domain-name huawei.com
```

- DHCPv6 parameters are assigned for each address pool.

- The implementation of stateful addressing requires that an address pool be defined with a typical address prefix defined for the given range, as well as pool specific parameters. The example demonstrates how a pool is created with the defining of a pool and associated pool name, as well as the address prefix from which the range of host addresses will be allocated.
- Excluded addresses refer to addresses that comprise of the pool range however should not be allocated through DHCP since they are commonly used for other applications such as the interface address of the DHCP server. Additional configuration parameters will also be specified for a given pool with examples such as server addresses and domain names being specified for parameter allocation to DHCPv6 clients.



Enable DHCPv6 Server



```
[RTA]ipv6
[RTA]dhcp enable
[RTA]interface GigabitEthernet 0/0/0
[RTA-GigabitEthernet0/0/0]ipv6 enable
[RTA-GigabitEthernet0/0/0]ipv6 address 3000::1/64
[RTA-GigabitEthernet0/0/0]dhcpv6 server pool1
```

- Address pool is associated with the DHCPv6 server interface.

- A created DHCPv6 pool is required to be associated with an interface through which it will service DHCP clients. An IPv6 address is assigned to the interface of the DHCPv6 server and the interface then associated with the address pool. In this case the excluded address value has been used to represent the interface address in order to ensure no attempts are made by the DHCPv6 server to assign the interface address to a DHCP client.



Displaying DHCPv6 Information

```
<RTA>display dhcpv6 pool
DHCPv6 pool: pool1
  Address prefix: 3000::/64
    Lifetime valid 172800 seconds, preferred 86400 seconds
    2 in use, 0 conflicts
  Excluded-address 3000::1
  Information refresh time: 86400
  DNS server address: 3001::1
  Domain name: huawei.com
  Conflict-address expire-time: 172800
  Active normal clients: 2
```

- Configured pools, pool based parameters, and client activity are referenced under the display dhcp pool command.

- The resulting configuration of the DHCPv6 server can be clarified through the display dhcpv6 pool command, from which point the defined pool(s) can be identified and the address prefix associated with the pool determined. Additional information such as the lifetime can be viewed, for which the default lifetime of a leased address is 86400 seconds, or 1 day and can be reconfigured as necessary using the information-refresh command under the dhcpv6 pool <pool-name> view. Where active clients have leased addresses from the DHCP server, the related statistics can be found here.

ICMPv6

The ICMP typical functions are inherited from v4.

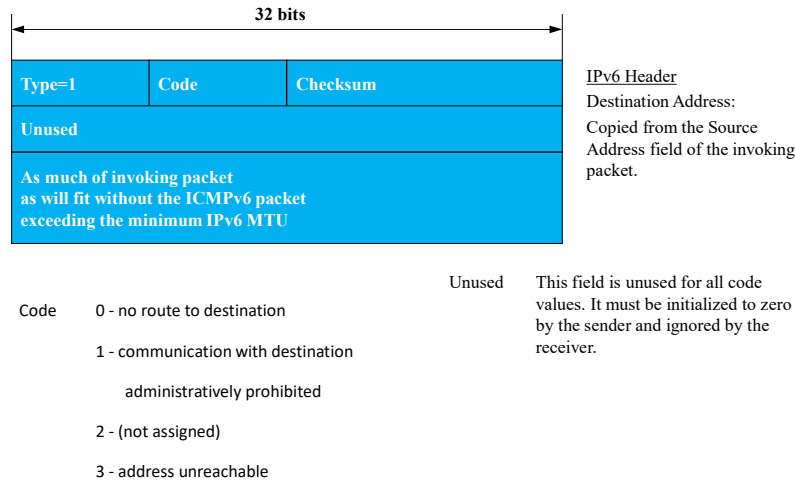
Some important ICMPv6 message types:

Type	Message
1	Destination unreachable
2	Packet too big
3	Time exceeded
4	Parameter problem
128	Echo request
129	Echo reply

Page 39

- ICMPv6 is an integral part of IPv6. Every node that implements IPv6 must fully implement ICMPv6. ICMPv6 is a modified version of ICMP for IPv4.
- ICMP messages are used to report error and informational conditions, as well as diagnostics functions like the Packet Internet Groper (ping) and traceroute.
- ICMP messages are generated as a result of some error condition. For example, if a router is unable to process an IP packet for some reason, it probably generate some type of ICMP message directed back at the packet's source. The source would then be able to take some action to remedy the error condition being reported.
- ICMPv6 is defined in RFC 2463.

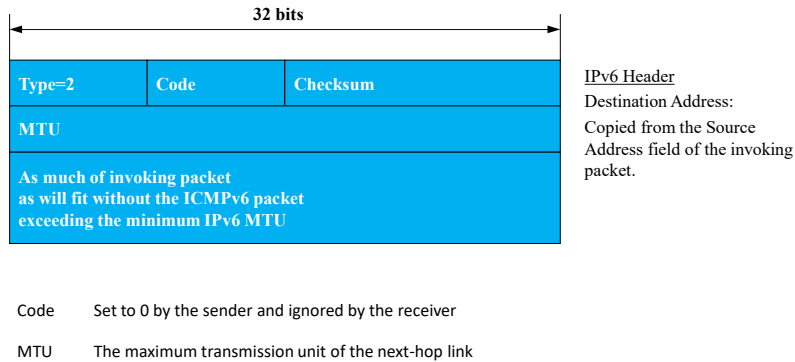
ICMPv6: Destination Unreachable



Page 40

- A Destination Unreachable message should be generated by a router, or by the IPv6 layer in the originating node, in response to a packet that cannot be delivered to its destination address for reasons other than congestion. (An ICMPv6 message MUST NOT be generated if a packet is dropped due to congestion.)
- If the reason for the failure to deliver is lack of a matching entry in the forwarding code's routing table, the Code field is set to 0 (NOTE: this error can occur only in nodes that do not hold a "default route" in their routing tables).
- If the reason for the failure to deliver is administrative prohibition, e.g., a "firewall filter", the Code field is set to 1.
- If there is any other reason for the failure to deliver, e.g., inability to resolve the IPv6 destination address into a corresponding link address, or a link-specific problem of some sort, then the Code field is set to 3.
- A destination node should send a Destination Unreachable message with Code 4 in response to a packet for which the transport protocol (e.g., UDP) has no listener, if that transport protocol has no alternative means to inform the sender.

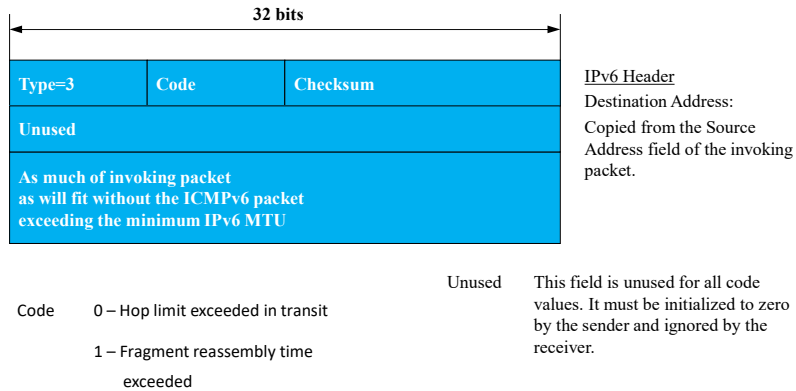
ICMPv6: Packet too big



Page 41

- A Packet Too Big MUST be sent by a router in response to a packet that it cannot forward because the packet is larger than the MTU of the outgoing link. The information in this message is used as part of the Path MTU Discovery process [PMTU].
- Sending a Packet Too Big Message makes an exception to one of the rules of when to send an ICMPv6 error message, in that unlike other messages, it is sent in response to a packet received with an IPv6 multicast destination address, or a link-layer multicast or link-layer broadcast address.

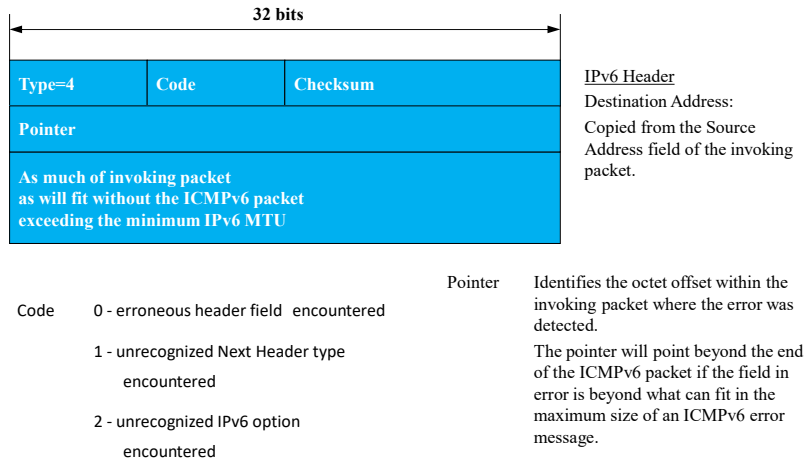
ICMPv6: Time exceeded



Page 42

- If a router receives a packet with a Hop Limit of zero, or a router decrements a packet's Hop Limit to zero, it MUST discard the packet and send an ICMPv6 Time Exceeded message with Code 0 to the source of the packet. This indicates either a routing loop or too small an initial Hop Limit value.
- Default reassembly time is 60s. Can be configured.

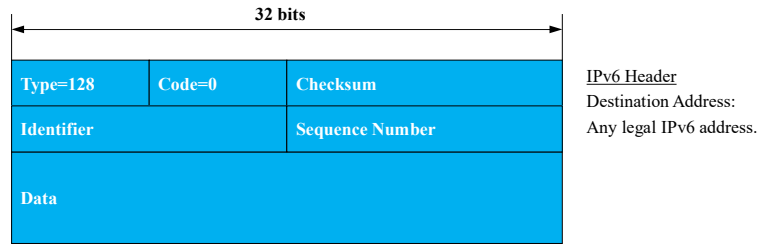
ICMPv6: Parameter problem



Page 43

- If an IPv6 node processing a packet finds a problem with a field in the IPv6 header or extension headers such that it cannot complete processing the packet, it MUST discard the packet and should send an ICMPv6 Parameter Problem message to the packet's source, indicating the type and location of the problem.
- The pointer identifies the octet of the original packet's header where the error was detected. For example, an ICMPv6 message with Type field = 4, Code field = 1, and Pointer field = 40 would indicate that the IPv6 extension header following the IPv6 header of the original packet holds an unrecognized Next Header field value.

ICMPv6: Echo request

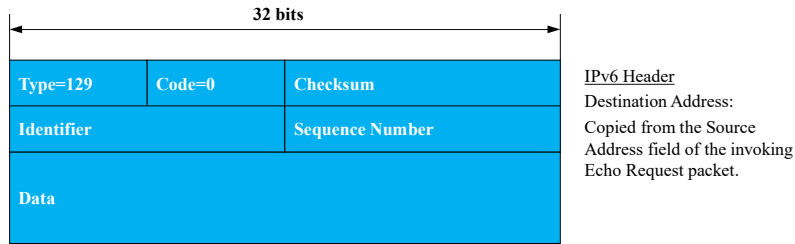


Code	0
Identifier	An identifier to aid in matching Echo Replies to this Echo Request. May be zero.
Sequence Number	A sequence number to aid in matching Echo Replies to this Echo Request. May be zero.
Data	Zero or more octets of arbitrary data.

Page 44

- Every node MUST implement an ICMPv6 Echo responder function that receives Echo Requests and sends corresponding Echo Replies. A node should also implement an application-layer interface for sending Echo Requests and receiving Echo Replies, for diagnostic purposes.

ICMPv6: Echo reply



Code	0
Identifier	The identifier from the invoking Echo Request message.
Sequence Number	The sequence number from the invoking Echo Request message
Data	The data from the invoking Echo Request message.

Page 45

- Every node MUST implement an ICMPv6 Echo responder function that receives Echo Requests and sends corresponding Echo Replies. A node should also implement an application-layer interface for sending Echo Requests and receiving Echo Replies, for diagnostic purposes.
- The source address of an Echo Reply sent in response to a unicast Echo Request message MUST be the same as the destination address of that Echo Request message.
- An Echo Reply should be sent in response to an Echo Request message sent to an IPv6 multicast address. The source address of the reply MUST be a unicast address belonging to the interface on which the multicast Echo Request message was received.
- The data received in the ICMPv6 Echo Request message MUST be returned entirely and unmodified in the ICMPv6 Echo Reply message.

Neighbor Discovery Protocol (NDP)

Provides functionality for

- Serverless autoconfiguration
- Router discovery
- Prefix discovery
- Address resolution
- Neighbor unreachability detection
- Next-hop determination
- Duplicate address detection

Page 46

- The neighbor discovery protocol addresses many problems related to nodes on a single link. It provides the functionality for serverless autoconfiguration, router discovery, prefix discovery, address resolution, neighbor unreachability detection, link MTU discovery, next-hop determination and duplicate address detection.
- Within IPv4, a combination of many protocols, including DHCP, ICMP router discovery, a routing protocol, and ARP, are required to provide only some of this functionality.
- Neighbor discovery uses ICMPv6 to perform these tasks. Neighbor discovery for IPv6 is described in RFC 2461.
- When a node is initialized, it must know a few things before it begins to communicate:
- It must know its own address (autoconfiguration).
- It must know about any routers on the link (router discovery).
- It must know its own prefix information so that it can figure out how to send packets to nodes located in other prefixes (prefix discovery).
- It needs to know how to obtain the link-level address associated with a known network layer address (address resolution).
- It needs to know how large of a packet it can send (MTU discovery).
- It needs to know how to determine the next hop in the path to a destination (next-hop determination).

Author: G. Amelger, Juniper Networks,
EMEA Central

Copyright © 2001 Juniper Networks, Inc.

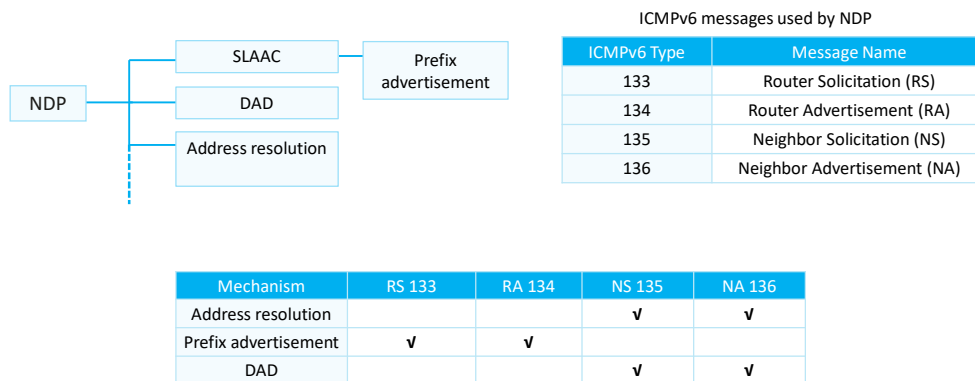
All rights reserved. To make communication run smoother, a node should know some other things as well:

- It should be able to detect when a neighbor is no longer reachable so that it does not sent packets to that neighbor (neighbor unreachability detection).
- It should know about neighbors on its link.
- It should know whether the address it is trying to use is in use already by another node on the link (duplicate address detection).
- It needs to know what other prefixes are assigned to nodes on the same link.
- It should be able to redirect traffic to a better next-hop node, if one exists, for any destination.



NDP

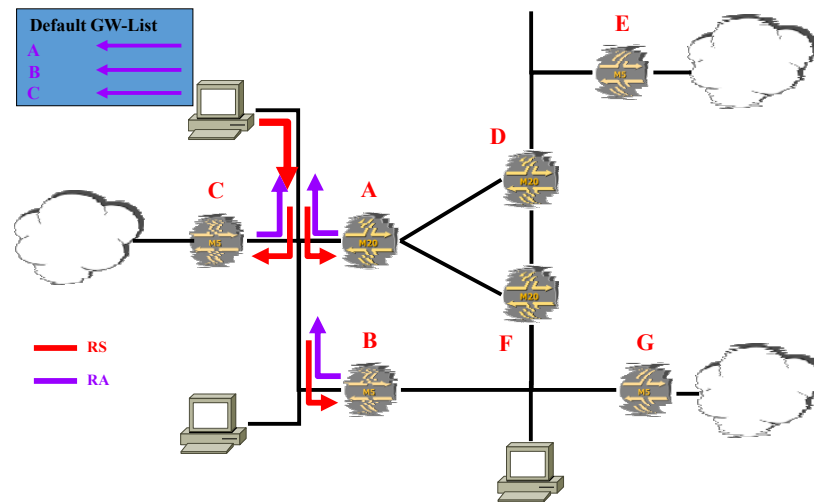
- **Neighbor Discovery Protocol (NDP)** is defined in RFC 2461, which was replaced by RFC 4861.
- NDP uses ICMPv6 messages to implement its functions.



Neighbor Discovery Protocol defines five ICMPv6 packets to provide IPv6 nodes with the information they must and should know before communicating:

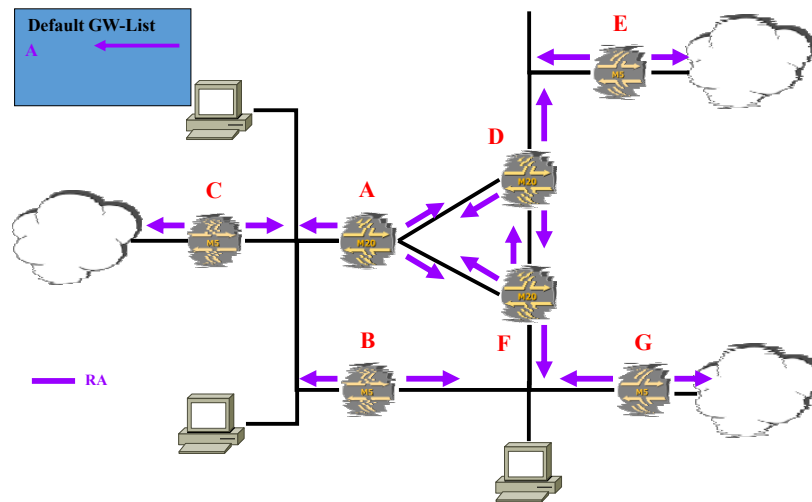
- **Router solicitation** – Multicast by a node when it wants routers to send a Router advertisement immediately instead of waiting for the next scheduled advertisement.
 - **Router advertisement** – Sent periodically or in response to a solicitation. Routers advertise their presence, as well as provide information necessary for a node to configure itself.
 - **Neighbor solicitation** – Enables a node to determine the link layer address of a neighbor or to determine whether the neighbor is still reachable via a cached link layer address. Also enables a node to determine whether a duplicate IP address exists on the link.
 - **Neighbor advertisement** – Sent in response to Neighbor solicitations, or unsolicited if a node's link layer address changes
-
- SLAAC is a highlight of IPv6. It enables IPv6 hosts to be easily connected to IPv6 networks, without the need to manually configure IPv6 addresses and to deploy application servers (such as DHCP servers) to assign addresses to hosts. SLAAC uses ICMPv6 RS and RA messages.
 - Address resolution uses ICMPv6 NS and NA messages.
 - DAD uses ICMPv6 NS and NA messages to ensure that no two identical unicast addresses exist on the network. DAD must be performed on all interfaces before they use unicast addresses.

Neighbor discovery: Router solicitation



Page 48

Neighbor discovery: Router advertisement



Page 49

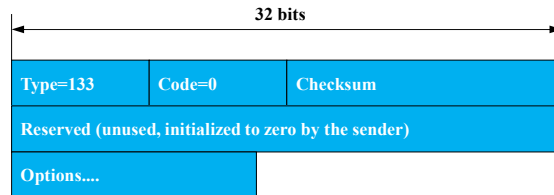
Router solicitation (RS)

- ICMP packet type 133
- Sent by host to speed up learning of link-local routers
- Source address is sending host's address or
`0:0:0:0:0:0:0:0` (if no address is assigned to the sending interface)
- Destination address is typically all-routers multicast address:
`FF02::2`
- May contain sender's link layer address (MUST NOT be included if the Source Address is the unspecified address. Otherwise, it SHOULD be included on link layers that have addresses.)
- Reply is a Router Advertisement (RA)

Page 50

- Hosts send Router Solicitations when they want to receive a Router Advertisement (RA) right away – they do not want to wait for a periodic advertisement. An initializing host sends an RS so that it can quickly learn the information it needs for configuration.

Router solicitation (RS) format



The only valid option is the Source Link-Layer which **SHOULD** be included if known e.g. the EUI-64 value of the interface else no option field should be included.

Page 51

- **IP Fields:**
- Source Address An IP address assigned to the sending interface, or the unspecified address if no address is assigned to the sending interface.
- Destination Address Typically the all-routers multicast address.
- Hop Limit 255
- Authentication Header If a Security Association for the IP Authentication Header exists between the sender and the destination address, then the sender **SHOULD** include this header.
- **ICMP Fields:**
- Type 133
- Code 0
- Checksum The ICMP checksum.
- Reserved This field is unused. It **MUST** be initialized to zero by the sender and **MUST** be ignored by the receiver.
- **Valid Options:**

addresses.

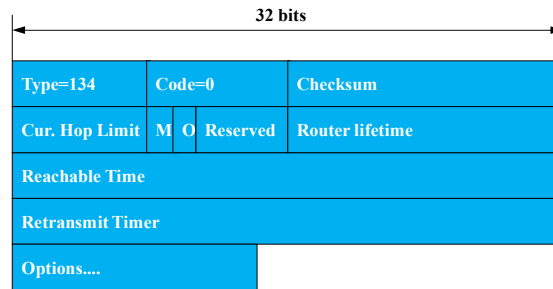
Router advertisement (RA)

- ICMP packet type 134
- Sent by routers periodically or in response to a solicitation to provide information necessary for a node to configure itself
- Source address is link-local address of the sending router
- Destination address is either
 - unicast address of a node that sent an RS, or
 - link-scope all-nodes multicast address: `FF02::1`
- Hop-limit MUST be set to 255
- Possible options contained in RA:
 - Source link layer address of the router
 - MTU
 - Prefix information about on-link prefixes

Page 52

- Routers advertise their presence on a link and provide the information necessary for a node to configure itself. The router advertisement is multicast to the link-scope all-nodes multicast group.
- The hop-limit must be set to 255. This ensures that no off-link device sends router advertisements in an attempt to disrupt traffic flow. If an off-link device does send an RA, the RA traverses a router, which in turn automatically decrements the hop-limit value by one, thus rendering the packet invalid.
- One of the ways the receiving node validates the packet, is by verifying that the hop limit is still 255.

Router advertisement (RA) format



- Cur Hop Limit** 8-bit unsigned integer. The default value that should be placed in the Hop Count field of the IP header for outgoing IP packets. A value of zero means unspecified (by this router).
- M** 1-bit "Managed address configuration" flag. When set, it indicates that addresses are available via DHCPv6.
- O** 1-bit "Other stateful configuration" flag. When set, it indicates that DHCPv6-lite is available for autoconfiguration of other (non-address) information. Examples of such information are DNS- related information or information on other servers within the network.

Page 53

- **IP Fields:**
- Source Address MUST be the link-local address assigned to the interface from which this message is sent.
- Destination Address Typically the Source Address of an invoking Router Solicitation or the all-nodes multicast address.
- Hop Limit 255
- Authentication Header If a Security Association for the IP Authentication Header exists between the sender and the destination address, then the sender SHOULD include this header.
- **ICMP Fields:**
- Type 134
- Code 0
- Checksum The ICMP checksum. See [ICMPv6].
- Cur Hop Limit 8-bit unsigned integer. The default value that should be placed in the Hop Count field of the IP header for outgoing IP packets. A value of zero means unspecified (by this router).
- M 1-bit "Managed address configuration" flag. When set, hosts use the administered (stateful) protocol for address autoconfiguration in addition to any addresses autoconfigured using stateless address autoconfiguration.

- O 1-bit "Other stateful configuration" flag. When set, hosts use the administered (stateful) protocol for autoconfiguration of other (non-address) information.
- **Router Lifetime** 16-bit unsigned integer. The lifetime associated with the default router in units of **seconds**. The field can contain values up to 65535 and receivers should handle any value, while the sending rules in [Section 6](#) limit the lifetime to 9000 seconds. A Lifetime of 0 indicates that the router is not a default router and SHOULD NOT appear on the default Narten, et al. Standards Track [Page 20] [RFC 4861](#) Neighbor Discovery in IPv6 September 2007 router list. The Router Lifetime applies only to the router's usefulness as a default router
- **Reachable Time:** The time, in milliseconds, that a node assumes a neighbor is reachable after having received a reachability confirmation. Used by the Neighbor Unreachability Detection algorithm
- **Retransmission Timer:** The amount of time, in milliseconds, that a host should wait before retransmitting Neighbor Solicitation messages.
- The valid options are the Source Link-Layer, MTU and Prefix Information.

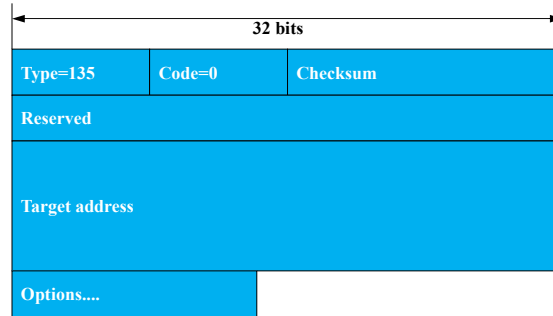
Neighbor solicitation (NS)

- ICMP packet type 135
- Used to provide/obtain link-layer address to/of a neighbor
- Used to verify neighbor reachability
- IPv6 Source-address is link-local address of soliciting node
- IPv6 Destination-address is either
 - solicited-node multicast address associated with target IP address (link layer determination) `FF02:0:0:0:1:FFXX:XXXX`
 - Unicast address of the target (reachability verification)
- Hop-limit MUST be set to 255
- Reply is a Neighbor advertisement (NA)

Page 54

- Nodes send Neighbor Solicitations to request the link-layer address of a target node while also providing their own link-layer address to the target. Neighbor Solicitations are multicast when the node needs to resolve an address and unicast when the node seeks to verify the reachability of a neighbor.

Neighbor solicitation (NS) format



Target Address: The IP address of the target of the solicitation. It **MUST NOT** be a multicast address.

Possible options: Source link-layer address, which is the link-layer address for the sender. **MUST NOT** be included when the **source IP** address is the **unspecified** address. Otherwise, on link layers that have addresses this option **MUST** be included in multicast solicitations and **SHOULD** be included in unicast solicitations.

Page 55

- **IP Fields:**
- Source Address Either an address assigned to the interface from which this message is sent or (if Duplicate Address Detection is in progress) the unspecified address.
- Destination Address Either the solicited-node multicast address corresponding to the target address, or the target address.
- Hop Limit 255
- Authentication Header If a Security Association for the IP Authentication Header exists between the sender and the destination address, then the sender **SHOULD** include this header.
- **ICMP Fields:**
- Type 135
- Code 0
- Checksum The ICMP checksum.
- Reserved This field is unused. It **MUST** be initialized to zero by the sender and **MUST** be ignored by the receiver.
- Target Address The IP address of the target of the solicitation. It **MUST NOT** be a multicast address.
- **Possible options:**
- Source link-layer address The link-layer address for the sender.

MUST NOT be included when the source IP address is the unspecified address. Otherwise, on link layers that have addresses this option MUST be included in multicast solicitations and SHOULD be included in unicast solicitations.

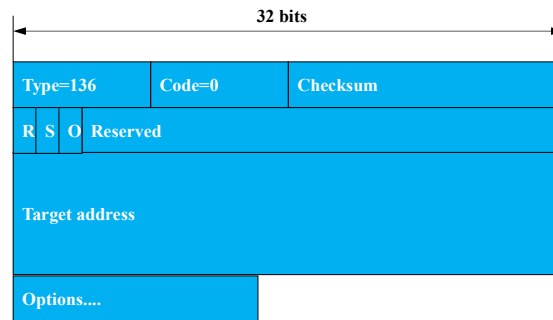
Neighbor advertisement (NA)

- ICMP packet type 136
- Sent in response to NS or unsolicited to immediately propagate new information
- Source address is any valid unicast address assigned to sending node
- Destination address is
 - For solicited advertisements
 - Source address of the solicitation (invoking packet)
 - If address of NS is unspecified: all-nodes multicast address : **FF02::1**
 - For unsolicited advertisements
 - All-nodes multicast: **FF02::1**
- Hop-limit MUST be set to 255
- Possible Option: The link-layer address for the target, i.e., the sender of the advertisement.

Page 56

- A node sends Neighbor Advertisements in response to Neighbor Solicitations and sends unsolicited Neighbor Advertisements in order to (unreliably) propagate new information quickly.

Neighbor advertisement (NA) format



- R** Router flag. When set, the R-bit indicates that the sender is a router.
- S** Solicited flag. When set, the S-bit indicates that the advertisement was sent in response to a Neighbor Solicitation from the Destination address. The S-bit is used as a reachability confirmation for Neighbor Unreachability Detection. It **MUST NOT** be set in multicast advertisements or in unsolicited unicast advertisements.
- O** Override flag indicates that the information of the message should override an existing Neighbor cache for which no link layer address exists

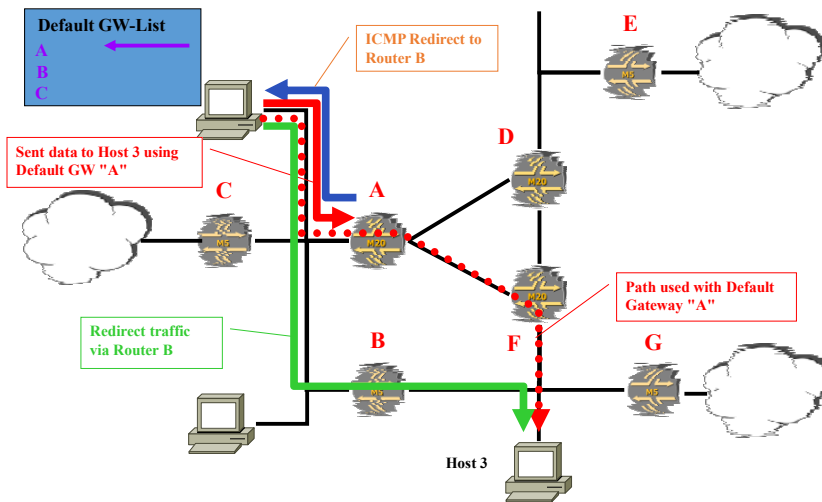
Page 57

- **IP Fields:**
- Source Address An address assigned to the interface from which the advertisement is sent.
- Destination Address For solicited advertisements, the Source Address of an invoking Neighbor Solicitation or, if the solicitation's Source Address is the unspecified address, the all-nodes multicast address. For unsolicited advertisements typically the all-nodes multicast address.
- Hop Limit 255
- Authentication Header If a Security Association for the IP Authentication Header exists between the sender and the destination address, then the sender **SHOULD** include this header.
- **ICMP Fields:**
- Type 136
- Code 0
- Checksum The ICMP checksum.
- R Router flag. When set, the R-bit indicates that the sender is a router. The R-bit is used by Neighbor Unreachability Detection to detect a router that changes to a host.
- S Solicited flag. When set, the S-bit indicates that the advertisement was sent in response to a Neighbor Solicitation from the Destination address. The S-bit is used as a reachability confirmation for

Neighbor Unreachability Detection. It MUST NOT be set in multicast advertisements or in unsolicited unicast advertisements.

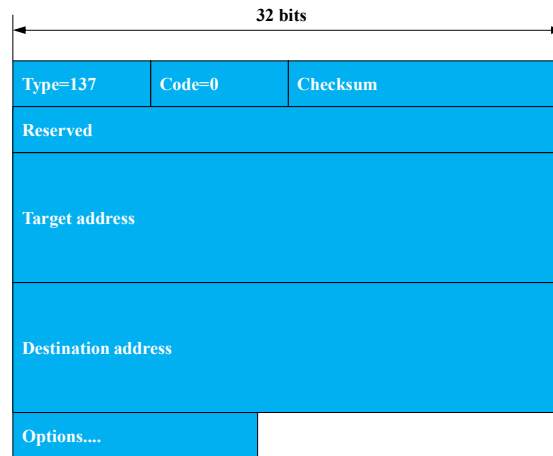
- O Override flag
- Target Address For solicited advertisements, the Target Address field in the Neighbor Solicitation message that prompted this advertisement. For an unsolicited advertisement, the address whose link-layer address has changed. The Target Address MUST NOT be a multicast address
- Options Target link-layer address The link-layer address for the target, i.e., the sender of the advertisement. This option MUST be included on link layers that have addresses when responding to multicast solicitations. When responding to a unicast Neighbor Solicitation this option SHOULD be included. The option MUST be included for multicast solicitations in order to avoid infinite Neighbor Solicitation "recursion" when the peer node does not have a cache entry to return a Neighbor Advertisements message. When responding to unicast solicitations, the option can be omitted since the sender of the solicitation has the correct link-layer address; otherwise, it would not be able to send the unicast solicitation in the first place. However, including the link-layer address in this case adds little overhead and eliminates a potential race condition where the sender deletes the cached link-layer address prior to receiving a response to a previous solicitation.

Redirect



Page 58

Redirect



Page 59

- Routers send Redirect packets to inform a host of a better first-hop node on the path to a destination. Hosts can be redirected to a better first-hop router but can also be informed by a redirect that the destination is in fact a neighbor. The latter is accomplished by setting the ICMP Target Address equal to the ICMP

IP Fields:

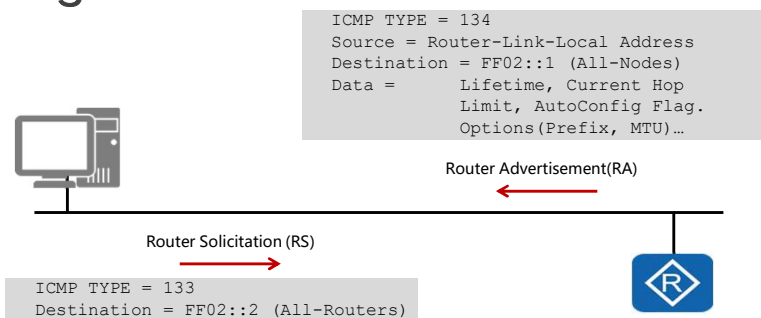
Source Address	MUST be the link-local address assigned to the interface from which this message is sent.
Destination Address	The Source Address of the packet that triggered the redirect.
Hop Limit	255
Authentication Header	If a Security Association for the IP Authentication Header exists between the sender and the destination address, then the sender SHOULD include this header.

Destination Address.

- Target Address: An IP address that is a better first hop to use for the ICMP Destination Address. When the target is the actual endpoint of communication, i.e., the destination is a neighbor, the Target Address field MUST contain the same value as the ICMP Destination Address field. Otherwise, the target is a better first-hop router and the Target Address MUST be the router's link-local address so that hosts can uniquely identify routers.
- Destination Address: The IP address of the destination that is redirected to the target. Possible options: Target link-layer address The link-layer address for the target. It SHOULD be included (if known). Note that on NBMA links, hosts may rely on the presence of the Target Link- Layer Address option in Redirect messages as the means for determining the link-layer addresses of neighbors. In such cases, the option MUST be included in Redirect messages. Redirected Header As much as possible of the IP packet that triggered the sending of the Redirect without making the redirect packet exceed the minimum MTU specified in [\[IPv6\]](#).



IPv6 Stateless Address Auto-configuration – SLAAC



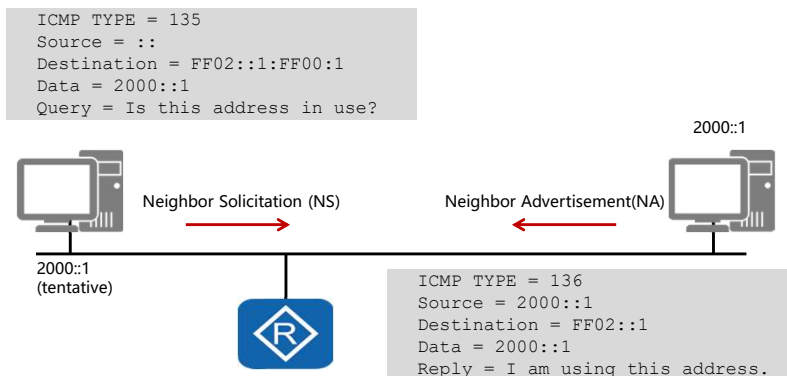
- Hosts are capable of generating IPv6 addresses independently.
- Router Advertisements deliver network parameter information.

Page 60

- Upon physically establishing with an IPv6 network, hosts must establish a unique IPv6 address and associated parameters such as the prefix of the network. Routers send out Router Advertisement messages periodically, and also in response to Router Solicitations (RS) to support router discovery, used to locate a neighboring router and learn the address prefix and configuration parameters for address auto-configuration.
- IPv6 supports stateless address auto-configuration (SLAAC) that allows hosts to obtain IPv6 prefixes and automatically generate interface IDs without requiring an external service such as DHCP. Router Discovery is the basis for IPv6 address auto-configuration and is implemented through two message formats.
- Router Advertisement (RA) messages allow each router to periodically send multicast RA messages that contain network configuration parameters, in order to declare the router's existence to Layer 2 hosts and other routers. An RA message is identified by a value of 134 in the type field of the message. Router Solicitation (RS) messages are generated after a host is connected to the network.
- Routers will periodically send out RA messages however should a host wish to prompt for an RA message, the host will send an RS message. Routers on the network will generate an RA message to all nodes to notify the host of the default router on the segment and related configuration parameters. The RS message generated by a host can be distinguished by the type field which contains a value of 133.



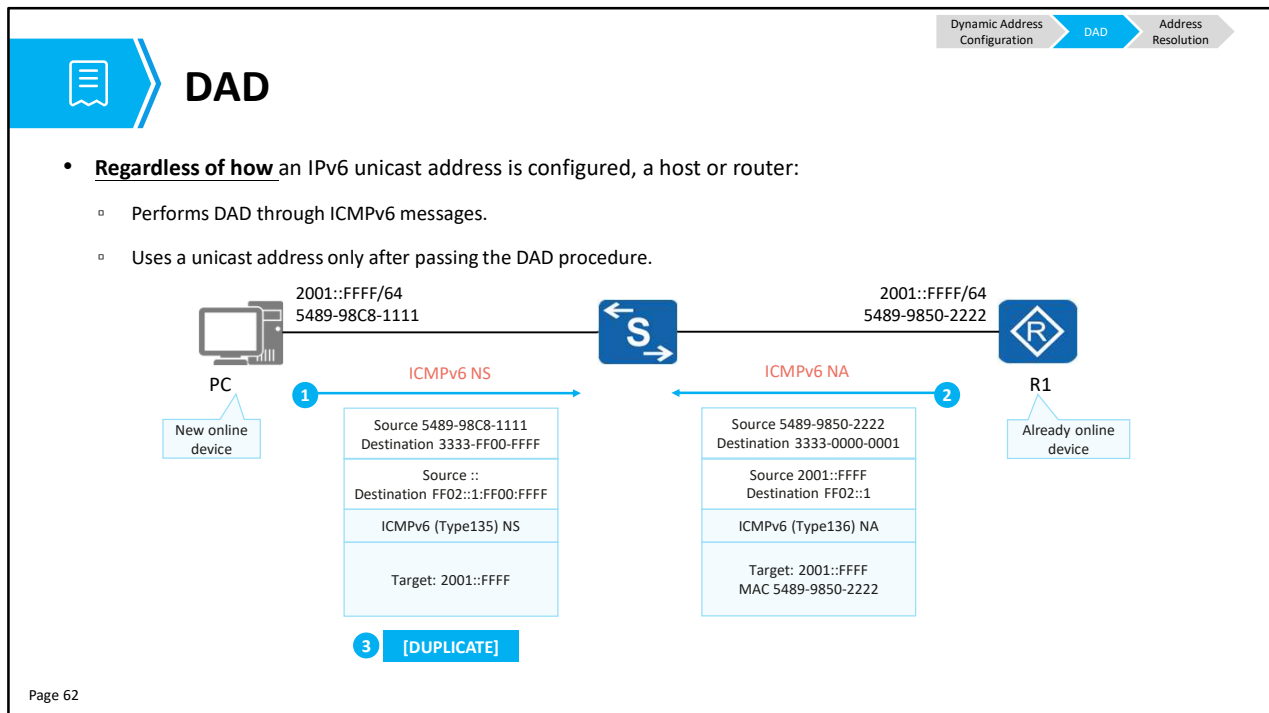
IPv6 Stateless Address Auto-configuration DAD



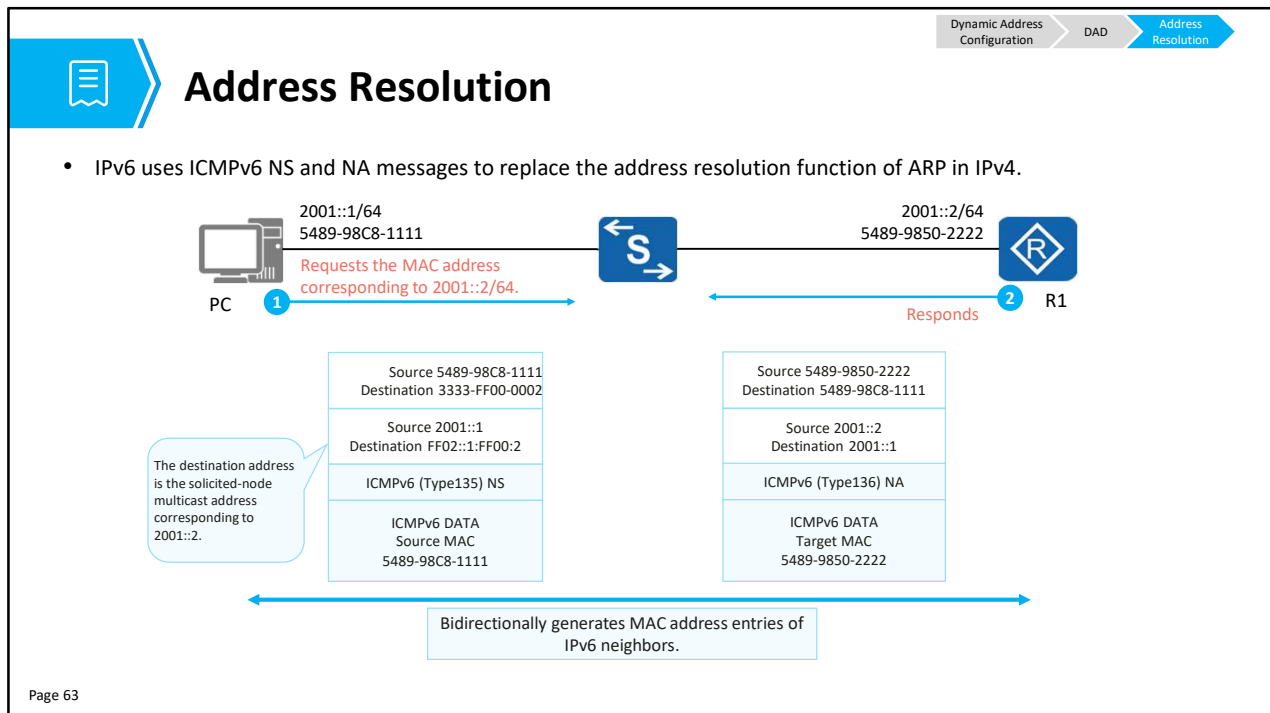
- Duplicate Address Detection (DAD) is used in IPv6 to verify that an address is unique before it is applied to the host interface.

Page 61

- Before an IPv6 unicast address is assigned to an interface, Duplicate Address Detection (DAD) is performed to check whether the address is used by another node. DAD is required if IP addresses are configured automatically. An IPv6 unicast address that is assigned to an interface but has not been verified by DAD is called a tentative address. An interface cannot use the tentative address for unicast communication but will join two multicast groups: ALL-nodes multicast group and Solicited-node multicast group.
- A Solicited-Node multicast address is generated by taking the last 24 bits of a unicast or anycast address and appending the address to the FF02:0:0:0:0:1:FF00::/104 prefix. In the case where the address 2000::1 is used, the address solicited node multicast address FF02::1:FF00:1 would be generated.
- IPv6 DAD is similar to the gratuitous ARP protocol used in IPv4 to detect duplicated IPv4 host addresses upon address assignment or connection of the host to the network. A node sends a neighbor solicitation (NS) message that requests the tentative address be used as the destination address to the Solicited-node multicast group.
- If the node receives a neighbor advertisement (NA) reply message, the tentative address is confirmed as being used by another node and the node will not use this tentative address for communication, following which manual assignment of an address by an administrator would be necessary.



- Assume that R1 is an online device with an IPv6 address 2001::FFFF/64. After the PC goes online, it is configured with the same IPv6 address. Before the IPv6 address is used, the PC performs DAD for the IPv6 address. The process is as follows:
 1. The PC sends an NS message to the link in multicast mode. The source IPv6 address of the NS message is ::, and the destination IPv6 address is the solicited-node multicast address corresponding to 2001::FFFF for DAD, that is, FF02::1:FF00:FFFF. The NS message contains the destination address 2001::FFFF for DAD.
 2. All nodes on the link receive the multicast NS message. The node interfaces that are not configured with 2001::FFFF are not added to the solicited-node multicast group corresponding to 2001::FFFF. Therefore, these node interfaces discard the received NS message. R1's interface is configured with 2001::FFFF and joins the multicast group FF02::1:FF00:FFFF. After receiving the NS message with 2001::FFFF as the destination IP address, R1 parses the message and finds that the destination address of DAD is the same as its local interface address. R1 then immediately returns an NA message. The destination address of the NA message is FF02::1, that is, the multicast address of all nodes. In addition, the destination address 2001::FFFF and the MAC address of the interface are filled in the NA message.
 3. After the PC receives the NA message, it knows that 2001::FFFF is already in use on the link. The PC then marks the address as duplicate. This IP address cannot be used for communication. If no NA message is received, the PC determines that the IPv6 address can be used. The DAD mechanism is similar to gratuitous ARP in IPv4.



- IPv6 address resolution does not use ARP or broadcast. Instead, IPv6 uses the same NS and NA messages as those in DAD to resolve data link layer addresses.
- Assume that a PC needs to parse the MAC address corresponding to 2001::2 of R1. The detailed process is as follows:
 - The PC sends an NS message to 2001::2. The source address of the NS message is 2001::1, and the destination address is the solicited-node multicast address corresponding to 2001::2.
 - After receiving the NS message, R1 records the source IPv6 address and source MAC address of the PC, and replies with a unicast NA message that contains its own IPv6 address and MAC address.
 - After receiving the NA message, the PC obtains the source IPv6 address and source MAC address from the message. In this way, both ends create a neighbor entry about each other.



Contents

1. IPv6 Overview
2. IPv6 Address Configuration
- 3. Typical IPv6 Configuration Examples**



Basic IPv6 Configurations (1)

1. Enable IPv6.

```
[Huawei] ipv6
```

Enable the device to send and receive IPv6 unicast packets, including local IPv6 packets.

```
[Huawei-GigabitEthernet0/0/0] ipv6 enable
```

Enable IPv6 on the interface in the interface view.

2. Configure an LLA for the interface.

```
[Huawei-GigabitEthernet0/0/0] ipv6 address ipv6-address link-local
```

```
[Huawei-GigabitEthernet0/0/0] ipv6 address auto link-local
```

Configure an LLA for the interface manually or automatically in the interface view.

3. Configure a GUA for the interface.

```
[Huawei-GigabitEthernet0/0/0] ipv6 address { ipv6-address prefix-length | ipv6-address/prefix-length }
```

```
[Huawei-GigabitEthernet0/0/0] ipv6 address auto { global | dhcp }
```

Configure a GUA for the interface manually or automatically (stateful or stateless) in the interface view.



Basic IPv6 Configurations (2)

4. Configure an IPv6 static route.

```
[Huawei] ipv6 route-static dest-ipv6-address prefix-length { interface-type interface-number [ nexthop-ipv6-address ] | nexthop-ipv6-address } [ preference preference ]
```

5. Display IPv6 information on an interface.

```
[Huawei] display ipv6 interface [ interface-type interface-number | brief ]
```

6. Display neighbor entry information.

```
[Huawei] display ipv6 neighbors
```

7. Enable an interface to send RA messages.

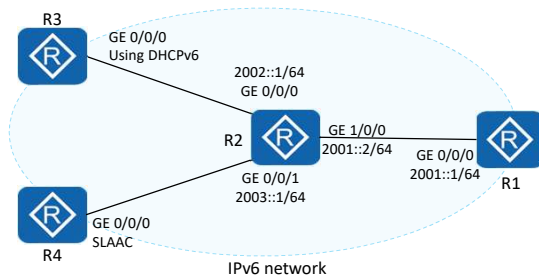
```
[Huawei-GigabitEthernet0/0/0] undo ipv6 nd ra halt
```

By default, a Huawei router's interfaces do not send ICMPv6 RA messages. In this situation, other devices on the links connected to the interfaces cannot perform SLAAC.

To perform SLAAC, you need to manually enable the function of sending RA messages.



Example: Configuring a Small IPv6 Network (1)



- Configuration Requirements

- Connect R1 and R2 through interfaces with static IPv6 addresses.
- Configure R2 as a DHCPv6 server to assign a GUA to GE 0/0/0 of R3.
- Enable R2 to send RA messages, and configure GE 0/0/0 of R4 to automatically perform SLAAC based on the RA messages sent by R2.
- Configure static routes to implement mutual access between the devices.

1. Enable IPv6 globally and on related interfaces of R1, R2, R3, and R4, and enable the interfaces to automatically generate LLAs. The following uses R1 configurations as an example.

```
[R1]ipv6
[R1]interface GigabitEthernet 0/0/0
[R1-GigabitEthernet0/0/0]ipv6 enable
[R1-GigabitEthernet0/0/0]ipv6 address auto link-local
```

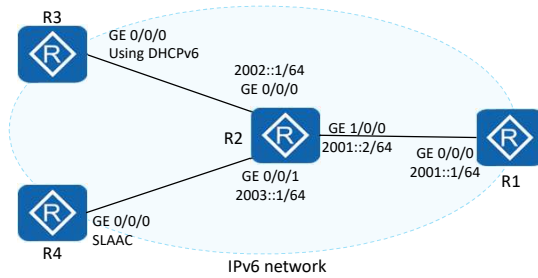
2. Configure static IPv6 GUAs on the related interfaces of R1 and R2.

```
[R1]interface GigabitEthernet 0/0/0
[R1-GigabitEthernet0/0/0]ipv6 address 2001::1 64
```

```
[R2]interface GigabitEthernet 1/0/0
[R2-GigabitEthernet1/0/0]ipv6 address 2001::2 64
[R2-GigabitEthernet1/0/0]interface GigabitEthernet 0/0/0
[R2-GigabitEthernet0/0/0]ipv6 address 2002::1 64
[R2-GigabitEthernet0/0/0]interface GigabitEthernet 0/0/1
[R2-GigabitEthernet0/0/1]ipv6 address 2003::1 64
```



Example: Configuring a Small IPv6 Network (2)



- Configuration Requirements

- Connect R1 and R2 through interfaces with static IPv6 addresses.
- Configure R2 as a DHCPv6 server to assign a GUA to GE 0/0/0 of R3.
- Enable R2 to send RA messages, and configure GE 0/0/0 of R4 to automatically perform SLAAC based on the RA messages sent by R2.
- Configure static routes to implement mutual access between the devices.

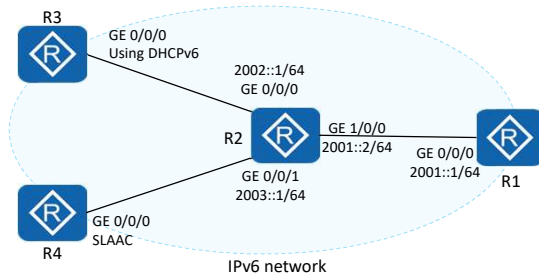
3. Configure R2 as a DHCPv6 server. Configure the related interface of R3 to obtain a GUA using DHCPv6.

```
[R2]dhcp enable
[R2]dhcpv6 pool pool1
[R2-dhcpv6-pool-pool1]address prefix 2002::/64
[R2]interface GigabitEthernet 0/0/0
[R2-GigabitEthernet0/0/0]dhcpv6 server pool1
```

```
[R3]dhcp enable
[R3]interface GigabitEthernet 0/0/0
[R3-GigabitEthernet0/0/0]ipv6 address auto dhcp
```



Example: Configuring a Small IPv6 Network (3)



4. Enable R2 to advertise RA messages. Enable R4 to obtain an address through SLAAC based on the RA messages sent by R2.

```
[R2]interface GigabitEthernet 0/0/1
[R2-GigabitEthernet0/0/1]undo ipv6 nd ra halt
```

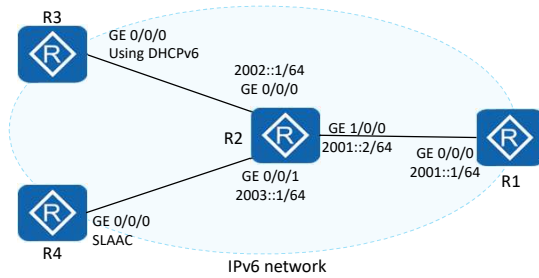
```
[R4]interface GigabitEthernet 0/0/0
[R4-GigabitEthernet0/0/0]ipv6 address auto global
```

• Configuration Requirements

- Connect R1 and R2 through interfaces with static IPv6 addresses.
- Configure R2 as a DHCPv6 server to assign a GUA to GE 0/0/0 of R3.
- Enable R2 to send RA messages, and configure GE 0/0/0 of R4 to automatically perform SLAAC based on the RA messages sent by R2.
- Configure static routes to implement mutual access between the devices.



Example: Configuring a Small IPv6 Network (4)



• Configuration Requirements

- Connect R1 and R2 through interfaces with static IPv6 addresses.
- Configure R2 as a DHCPv6 server to assign a GUA to GE 0/0/0 of R3.
- Enable R2 to send RA messages, and configure GE 0/0/0 of R4 to automatically perform SLAAC based on the RA messages sent by R2.
- Configure static routes to implement mutual access between the devices.

5. Configure static routes on R4.

```
[R4]ipv6 route-static 2001:: 64 2003::1
[R4]ipv6 route-static 2002:: 64 2003::1
```

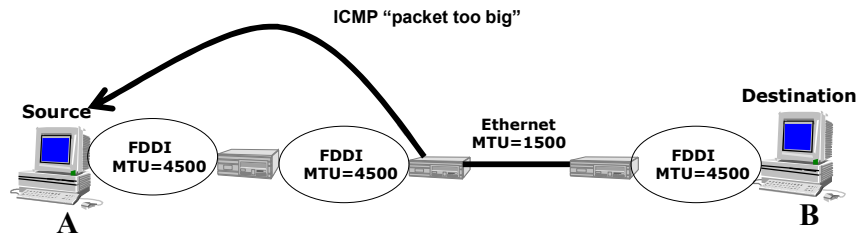
6. Configure an aggregated static route on R1.

```
[R1]ipv6 route-static 2002:: 15 2001::2
```

7. Configure a default route on R3.

```
[R3]ipv6 route-static :: 0 2002::1
```

Path MTU discovery



For packets bigger than 1280 bytes, path MTU discovery is expected:

- start by assuming MTU of the first-hop link
- if a packet reaches a link which couldn't fit, an ICMP "packet too big" is generated and sent back to the source
- then the source will fragmentize the packet into smaller chunks (following this new MTU size) and start this process all over again

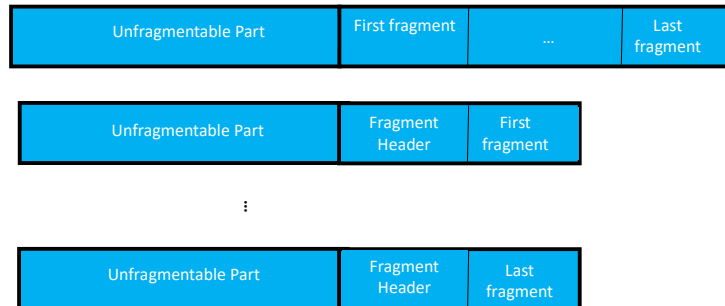
Page 71

- Link MTU is a link's maximum transmission unit, that is the biggest IP packet size that can be transmitted over a specific link. Path MTU is the minimum MTU for all the links in a path between a source and a destination
- IPv6 requires that the link layer support a minimum IPv6 packet size of 1280 bytes (68 bytes in IPv4). Link layers that do not support this must provide a link layer fragmentation and reassembly scheme that is transparent to IPv6. For link layers that can support a configurable MTU size, it is recommended that they be configured with an MTU size of at least 1500 bytes (the Ethernet II encapsulation IPv6 MTU). IPv6 source hosts can fragment payloads of upper layer protocols that are larger than the path MTU by using the Fragment header. However, the use of IPv6 fragmentation is highly discouraged. An IPv6 node must be able to reassemble a fragmented packet that is at least 1500 bytes in size.
- In Path MTU Discovery, the source node sends out a packet with an MTU as large as the local interface can handle. If this MTU is too large for some link along the path, an ICMP "Datagram too big" message will be sent back to the source. This message will contain a packet-too-big indicator and the MTU of the affected link. The source can then chop the packet to smaller chunks(fragment) and retransmit this chunk. This process is repeated until a packet gets all the way to the destination node. The discovered MTU is then used for fragmentation purposes. Although source-based fragmentation is fully supported in IPv6, it is recommended that network applications adjust packet size to accommodate the smallest MTU of the path. This will avoid the overhead associated with fragmentation and reassembly

on source and destination nodes.

Fragmentation revisited

The Fragmentable Part of the original packet is divided into fragments, each, except possibly the last ("rightmost") one, being an integer multiple of 8 octets long. The fragments are transmitted in separate "fragment packets" as illustrated:



The last header of the Unfragmentable Part changed to 44.

Page 72

- The unfragmentable part of a packet consists of the fixed header and some of the extension headers of the original packet (if present): all extension headers up to and including the *Routing* extension header, or else the *Hop-by-Hop* extension header. If neither extension headers are present, the unfragmentable part is just the fixed header.
- The following error conditions may arise when reassembling fragmented packets: If insufficient fragments are received to complete reassembly of a packet within 60 seconds of the reception of the first-arriving fragment of that packet, reassembly of that packet must be abandoned and all the fragments that have been received for that packet must be discarded. If the first fragment (i.e., the one with a Fragment Offset of zero) has been received, an ICMP Time Exceeded -- Fragment Reassembly Time Exceeded message should be sent to the source of that fragment. If the length of a fragment, as derived from the fragment packet's Payload Length field, is not a multiple of 8 octets and the M flag of that fragment is 1, then that fragment must be discarded and an ICMP Parameter Problem, Code 0, message should be sent to the source of the fragment, pointing to the Payload Length field of the fragment packet. If the length and offset of a fragment are such that the Payload Length of the packet reassembled from that fragment would exceed 65,535 octets, then that fragment must be discarded and an ICMP Parameter Problem, Code 0, message should be sent to the source of the fragment, pointing to the Fragment Offset field of the fragment packet.

IP-Sec

- Data encryption
 - IPsec encrypts data to ensure data confidentiality.
- Data integrity authentication
 - IPsec ensures that the data is not tampered with during transmission using data integrity authentication.
- Data origin authentication
 - IPsec authenticates data origins to ensure that data comes from real senders.
- Anti-replay
 - IPsec prevents malicious users from sending obtained packets by enabling the receiver to discard duplicate packets.

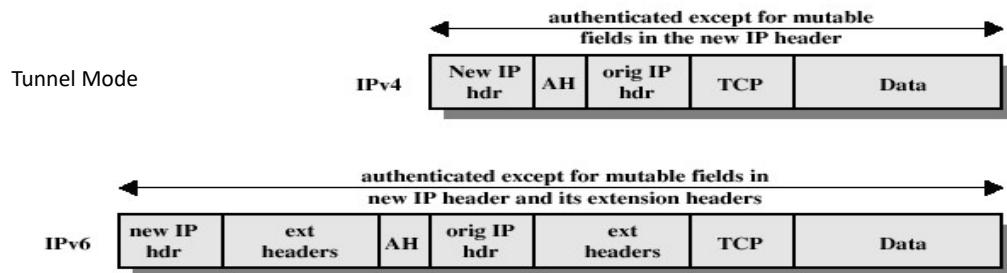
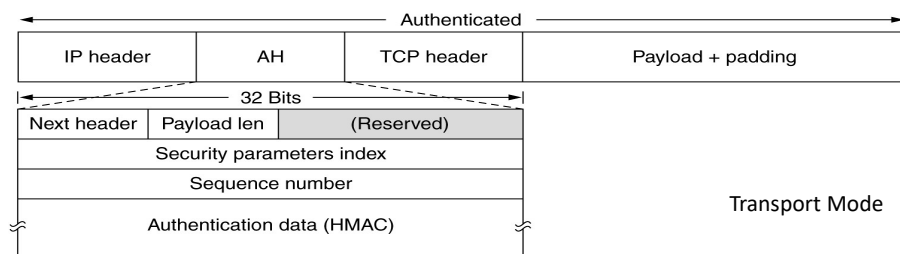
Security Issues: IPSec

Two operation modes: Transport Mode and Tunnel Mode



- **Transport mode:** In transport mode, **only the payload of the IP packet is usually encrypted and/or authenticated**. The routing is intact, since the IP header is neither modified nor encrypted; however, when the authentication header is used, the IP addresses cannot be translated, as this will invalidate the hash value. The transport and application layers are always secured by hash, so they cannot be modified in any way (for example by translating the port numbers). *Transport mode is used for host-to-host communications.*
- **Tunnel mode:** in tunnel mode, **the entire IP packet is encrypted and/or authenticated**. It is then encapsulated into a new IP packet with a new IP header. *Tunnel mode is used to create virtual private networks* for network-to-network communications (e.g. between routers to link sites), host-to-network communications (e.g. remote user access), and host-to-host communications (e.g. private chat).

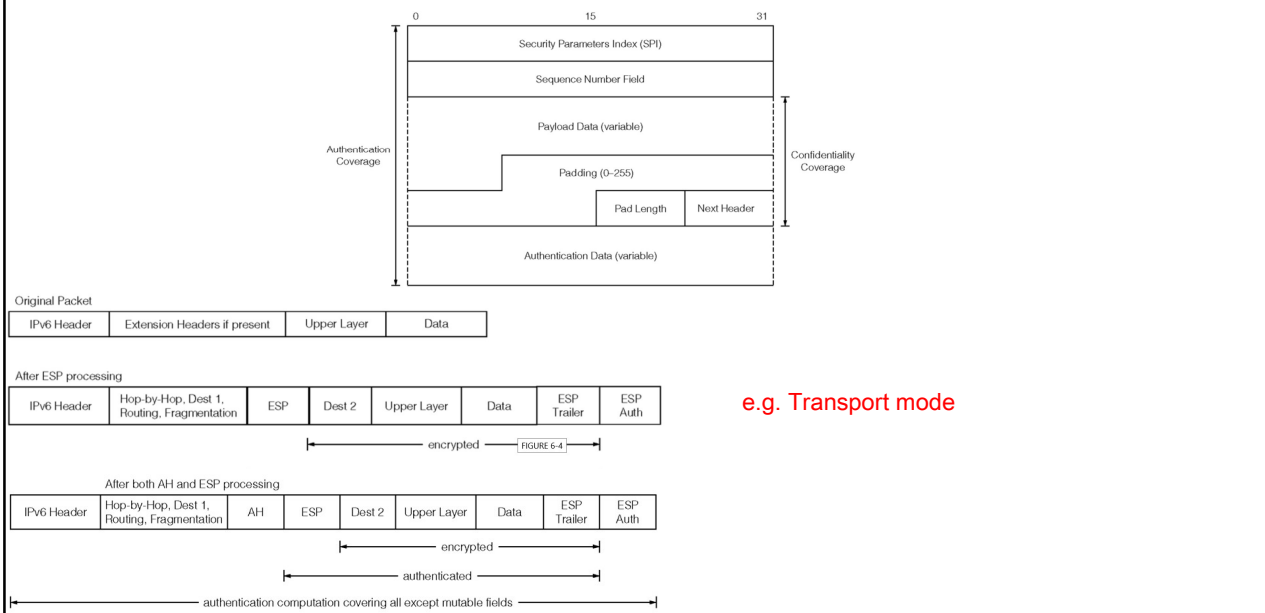
AH Protocol



Page 7

- **Next Header** (8 bits) : it indicates the protected upper-layer protocol. The value is taken from the list of IP protocol numbers.
- **Hdr Ext Len** (8 bits): length of the *Authentication Header* in 4-octet units, minus 2 (a value of 0 means 8 octets, 1 means 12 octets, etcetera). The header length must be a multiple of 8 octets if carried in an IPv6 packet. This restriction does not apply to an *Authentication Header* carried in an IPv4 packet.
- **Reserved** (16 bits): reserved for future use (all zeroes if not used).
- **Security Parameters Index** (32 bits): Arbitrary value which is used (together with the source IP address) to identify the security association of the sending party.
- **Sequence Number** (32 bits): A monotonic strictly increasing sequence number (incremented by 1 for every packet sent), used also to prevent replay attacks.
- **Integrity Check Value** (multiple of 32 bits): variable length check value. It may contain padding to align the field to an 8-octet boundary for IPv6, or a 4-octet boundary for IPv4.

Encapsulating Security Payload (ESP)

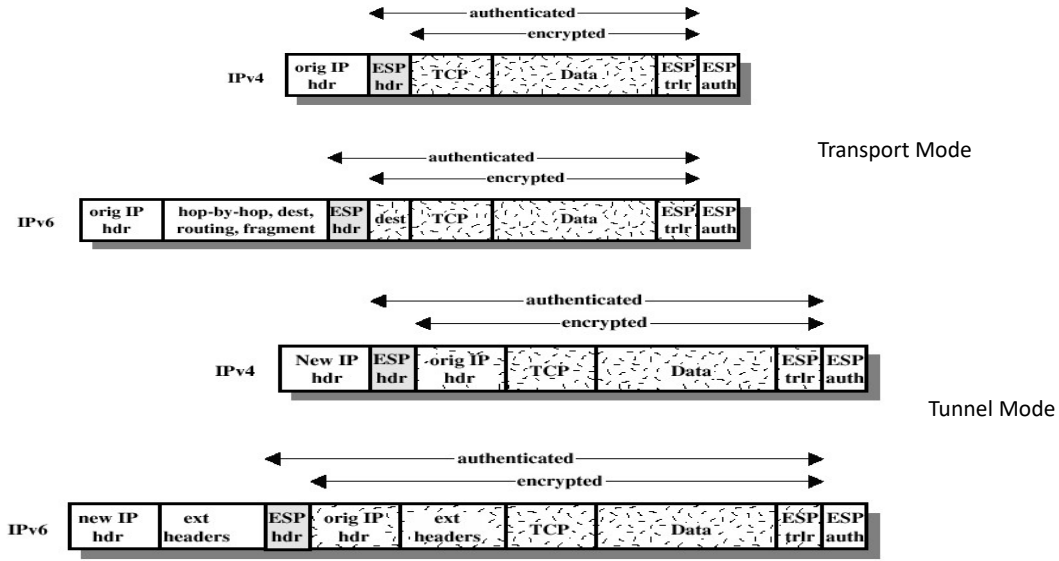


- **Security Parameters Index** (32 bits) : arbitrary value which is used (together with the source IP address) to identify the security association of the sending party.
- **Sequence Number** (32 bits) : a monotonically increasing sequence number (incremented by 1 for every packet sent) used also to protect against replay attacks.
- **Payload data** (variable): the protected contents of the original IP packet, including any data used to protect the contents (e.g. an Initialization Vector for the cryptographic algorithm). The type of content that was protected is indicated by the **Next Header** field.
- **Padding** (0-255 octets): padding for encryption, to extend the payload data to a size that fits the encryption's cypher block size, and to align the next field.
- **Pad Length** (8 bits): size of the padding in octets.
- **Next Header** (8 bits): type of the next header. The value is taken from the list of IP protocol numbers.
- **Authentication Data** (multiple of 32 bits): variable length check value. It may contain padding to align the field to an 8-octet boundary for IPv6, or a 4-octet boundary for IPv4.

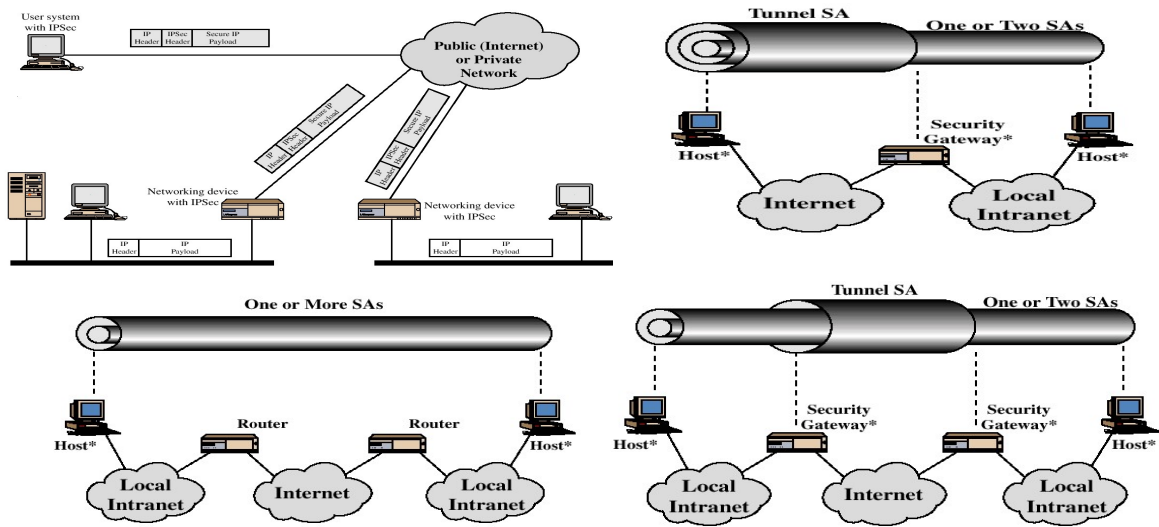
Encryption and Authentication Algorithms

- Encryption:
 - Three-key triple DES
 - RC5
 - IDEA
 - Three-key triple IDEA
 - CAST
 - Blowfish
- Authentication:
 - HMAC-MD5-96
 - HMAC-SHA-1-96

ESP Modes



Combinations of Security Associations



IPsec modes and combinations

	Transport Mode	Tunnel Mode
AH	Authenticates IP payload and selected portions of IP header	Authenticates entire inner IP datagram (header and payload) and selected portions of the outer IP header
ESP	Encrypts IP payload	Encrypts inner IP datagram
ESP with Authentication	Encrypts IP payload and authenticates IP payload but not IP header	Encrypts and authenticates inner IP datagram

Main Extension Headers

Header order	IPv6 default	Fragmented	AH transport	AH tunnel	ESP transport	ESP tunnel
IPv6 Header	IPv6Hdr	IPv6Hdr	IPv6Hdr	NewIPv6	IPv6Hdr	NewIPv6
Hop-by-Hop Opts	TCP	Fragment	EHs	NewEHs	EHs	NewEHs
Destination Opts	Data	TCP	AH	AH	ESP	ESP
Routing		Data	DestOpt	OldIPv6	DestOpt	OldIPv6
Fragment			TCP	OldEHs	TCP	OldEHs
Authentication			Data	TCP	Data	TCP
Encapsulating Security Payload				Data	ESPTail	Data
Destination Opts					ESPAuth	ESPTail
Upper Layer Hdrs						ESPAuth
Data						

If the upper-layer header is another IPv6 header (in the case of IPv6 being tunneled over or encapsulated in IPv6), it may be followed by its own extension headers, which are separately subject to the same ordering recommendations.

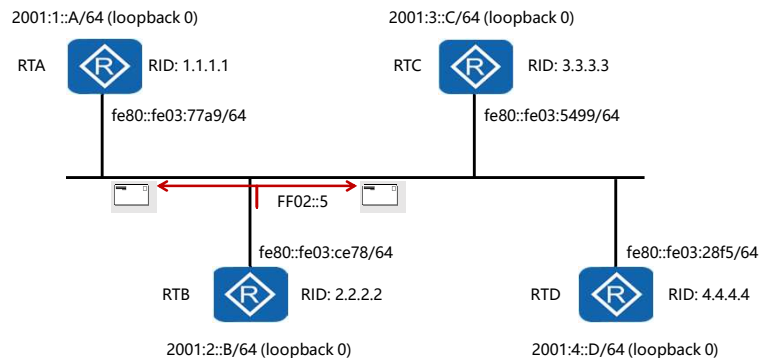
No Next Header

The value 59 in the Next Header field of an IPv6 header or any extension header indicates that there is nothing following that header. i.e. the payload should be empty.

If the Payload Length field of the IPv6 header indicates the presence of octets past the end of a header whose Next Header field contains 59, those octets must be ignored, and passed on unchanged if the packet is forwarded (RFC 2460)



OSPFv3

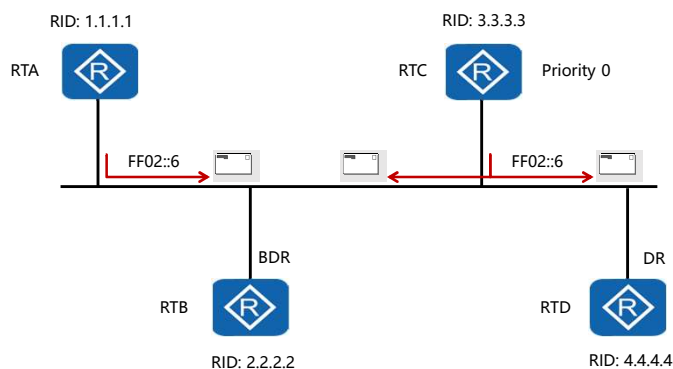


- OSPFv3 updates are sent to the All-SPF-Routers address, that is FF02::5.
- Link-local addressing used by default to define the next-hop.

- OSPFv3 is the version of OSPF that is used over IPv6 networks, and in terms of communication, assumes that each router has been assigned link-local unicast addresses on each of the router's attached physical links. OSPF packets are sent using the associated link-local unicast address of a given physical interface as the source address. A router learns the link-local addresses of all other routers attached to its links and uses these addresses as next-hop information during packet forwarding. This operation is true for all physical links with the exception of virtual links which are outside the scope of this material.
- A reserved "AllSPFRouters" multicast address has been assigned the value FF02::5, reflecting the multicast address 224.0.0.5 used in OSPFv2, for which it should be noted that the two versions are not compatible. All routers running OSPFv3 should be prepared to receive packets sent to this address. Hello packets are always sent to this destination, as well as certain OSPF protocol packets during the flooding procedure.



OSPFv3 Router ID

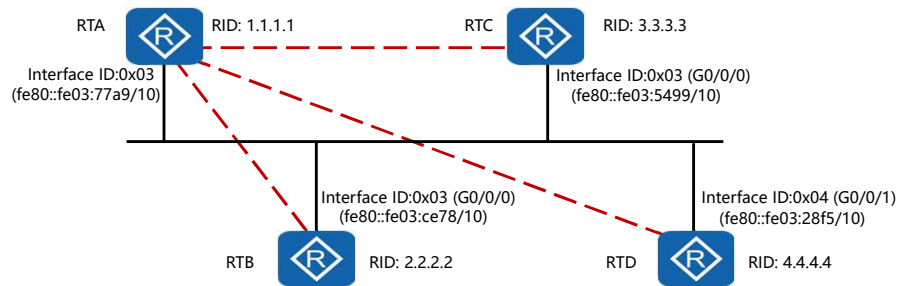


- Not based on any IP address, must be manually defined.
- Router ID continues to be used to support DR and BDR election.

- OSPF Router ID, Area ID and Link-State IDs remain at 32-bits size- they cannot be assigned IPv6 addresses.
- The router ID plays a prominent role in OSPFv3 and is now always used to identify neighboring routers. Previously, they had been identified by an IPv4 address on broadcast, NBMA (Non-Broadcast Multi-Access), and point-to-multipoint links. Each router ID (as well as the area ID) is maintained as a 32 bit dotted decimal value and cannot be configured using an IPv6 address.
- The router ID also continues to actively act as a tie breaker in the event that the priority associated with each OSPFv3 enabled router is equal. In such cases the Designated Router (DR) is identified as the router possessing the highest router ID. Hello messages sent will contain a router ID set to 0.0.0.0 if there is no Designated Router. The same principle applies for the Backup Designated Router, identified by the next highest router ID. A priority of 0 set on a router interface participating in OSPFv3 will deem the router as ineligible to participate in DR/BDR elections. Router Priority is only configured for interfaces associated with broadcast and NBMA networks.
- The multicast address "AllDRouters" has been assigned the value FF02::6, the equivalent of the 224.0.0.6 multicast address used in OSPFv2 for IPv4. The Designated Router and Backup Designated Router are both required to be prepared to receive packets destined to this address.



OSPFv3 Per Link Behavior

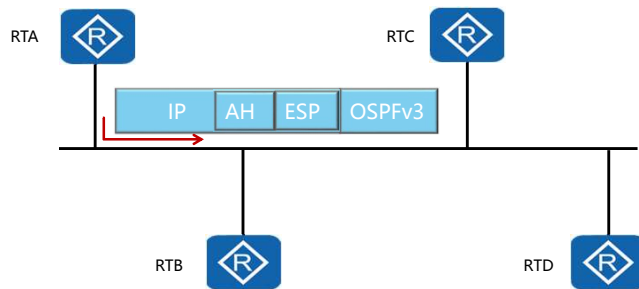


- OSPFv3 operates on the principle of per-link as opposed to the per-network or per-subnet concept used in IPv4.

- IPv6 uses the term "link" instead of "subnet" or "network" to define a medium used to communicate between nodes at the link layer. Multiple IP subnets can be assigned to a single link, and two nodes can communicate with each other even if they do not share a common IP subnet.
- IPv6 refers to the concept of links to mean a medium between nodes over which communication at the link layer is facilitated. Interfaces therefore connect to links and multiple IPv6 subnets can be associated with a single link, to allow two nodes to communicate directly over a single link, even when they do not share a common IPv6 subnet (IPv6 prefix). OSPFv3 as a result operates on a per-link basis instead of per-IP-subnet as is found within IPv4. The term link therefore is used to replace the terms network and subnet which are understood within OSPFv2. OSPF interfaces are now understood to connect to a link instead of an IP subnet. This change affects the receiving of OSPF protocol packets, the contents of Hello packets, and the contents of network-LSAs.
- The impact of links can be understood from OSPFv3 capability to now support multiple OSPF protocol instances on a single link. Where separate OSPF routing domains that wish to remain separate but operate over one or more physical network segments (links) that are common to the different domains. IPv4 required isolation of the domains through authentication which did not provide a practical solution to this design requirement.
- The per-link operation also means the Hello packet no longer contains any address information, and instead it now includes an Interface ID that the originating router assigns to uniquely identify (among its own interfaces) its interface to the link.



OSPFv3 Authentication

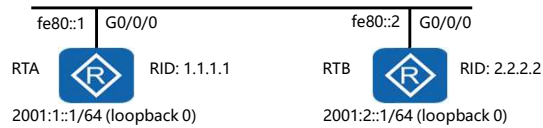


- OSPF authentication removed in OSPFv3, instead relying on the AH & ESP extension headers of IP for security.

- Authentication in OSPFv3 is no longer supported, and as such the authentication type and authentication (value) fields from the OSPF packet header no longer exist. As of IPv6, changes to the IP header have allowed for the utilization of security protocols that were only present in IPv4 as part of the IPsec suite, since initial TCP/IP development in the late 1960's was never designed with security in mind, since security of TCP/IP at the time was not foreseen as a future requirement.
- With the security improvements made to IPv6 and the utilization of IP extension headers in the protocol suite, OSPFv3 is capable of relying on the IP Authentication Header and the IP Encapsulating Security Payload to ensure integrity as well as authentication & confidentiality, during the exchange of OSPFv3 routing information.



Enabling OSPFv3



```
[RTA]ipv6
[RTA]ospfv3
[RTA-ospfv3-1]router-id 1.1.1.1
[RTA-GigabitEthernet0/0/0]ipv6 enable
[RTA-GigabitEthernet0/0/0]ipv6 address fe80::1 link-local
[RTA-GigabitEthernet0/0/0]ospfv3 1 area 0.0.0.0
[RTA-LoopBack0]ipv6 enable
[RTA-LoopBack0]ipv6 address 2001:1::1/64
[RTA-LoopBack0]ospfv3 1 area 0.0.0.0
```

- Implementing OSPFv3 between peers requires, as with RIPng, that the router firstly be capable of supporting IPv6. Additionally the router must also enable the OSPFv3 protocol globally in the system view. Each interface should be assigned an IPv6 address. During the configuration of RIPng it was demonstrated how the interface can be automatically configured to assign a link local address. In this example an alternative and recommended method of link local address assignment is demonstrated. The link local IPv6 address is manually assigned and designated as a link local address using the *ipv6 <link local address> link-local* command. If the address associated with the physical interface is a global IPv6 unicast address, the interface will also automatically generate a link local address.

In order to generate an OSPFv3 route, the example demonstrates assigning global unicast addresses to the logical loopback interface of each router. The physical and logical interfaces both should be associated with the OSPFv3 process and be assigned a process ID (typically the same ID unless they are to operate as separate instances of OSPFv3) and also be assigned to an area, which in this case is confined to area 0.

- As a reminder, OSPFv3 nodes rely on identification by neighboring routers through the use of the router ID, and therefore a unique router ID should be assigned for each router under the OSPFv3 protocol view following the configuration of the *ospfv3* command.



Configuration Validation

```
[RTA]display ospfv3
Routing Process "OSPFv3 (1)" with ID 1.1.1.1
Route Tag: 0
Multi-VPN-Instance is not enabled
SPF Intelligent Timer[milliseconds] Max: 10000, Start: 500, Hold: 2000
LSA Intelligent Timer[milliseconds] Max: 5000, Start: 500, Hold: 1000
LSA Arrival interval 1000 milliseconds
Default ASE parameters: Metric: 1 Tag: 1 Type: 2
Number of AS-External LSA 0. AS-External LSA's Checksum Sum 0x0000
Number of AS-Scoped Unknown LSA 0. AS-Scoped Unknown LSA's Checksum
Sum 0x0000
Number of FULL neighbors 1
Number of Exchange and Loading neighbors 0
.....
```

- Following the configuration of neighboring routers participating in OSPFv3, the display ospfv3 command is used to verify the operation and parameters associated with OSPFv3. Each router will show a running OSPFv3 process and a unique router ID. If the adjacency between neighbors is established, the number of FULL (state) neighbors will show a value greater than zero.



Summary

Comparison	IPv6	IPv4
Address length	128 bits	32 bits
Packet format	A fixed 40-byte basic packet header+variable-length extension headers	A basic header containing the Options field to support extended features
Address type	Unicast, multicast, and anycast	Unicast, multicast, and broadcast
Address configuration	Static, DHCP, and SLAAC	Static and DHCP
DAD	ICMPv6	Gratuitous ARP
Address resolution	ICMPv6	ARP