Contents

1 Configuring IPv6 Basics

1.1 Overview

As the Internet develops rapidly and IPv4 address space is becoming exhausted, IPv4 limitations become more and more obvious. At present, many researches and practices on Internet Protocol Next Generation (IPng) have been conducted. The IPng working group of the Internet Engineering Task Force (IETF) has formulated an IPng protocol named IP Version 6 (IPv6).

1.2 Advantages

- Larger address space

Compared with 32-bit IPv4 addresses, the length of an IPv6 address is extended to 128 bits. Therefore, the address space has approximately 2^{128} addresses. IPv6 supports a hierarchical address allocation mode from the Internet core network to intranet subnet.

- Simplified packet header format

Since the design principle of the IPv6 packet header is to minimize the cost of the packet header, some non-key fields and optional fields are removed from the packet header to the extended packet header. Therefore, although the length of an IPv6 address is four times that of an IPv4 address, the IPv6 packet header is only twice the IPv4 packet header in length. Without the checksum field in the IPv6 packet header, IPv6 devices do not need to process fragments during forwarding. Therefore, the packet can be more efficiently forwarded.

- Efficient hierarchical addressing and routing structure

IPv6 uses a convergence mechanism and defines a flexible hierarchical addressing and routing structure. Multiple networks at the same layer are represented with a uniform network prefix on the upstream device, greatly reducing routing entries maintained by the device and routing and storage costs of the device.

- Easy management: plug and play (PnP)

IPv6 provides automatic discovery and auto-configuration functions to simplify management and maintenance of network nodes. For example, neighbor discovery (ND), maximum transmission unit (MTU) discovery, router advertisement (RA), router solicitation (RS), and auto-configuration technologies provide related services for PnP.

IPv6 supports stateful and stateless address configuration modes. In IPv4, Dynamic Host Configuration Protocol (DHCP) realizes auto-configuration of the host IP addresses and related parameters. IPv6 inherits this auto-configuration service from IPv4, which is called stateful auto-configuration. For details, see "Configuring DHCPv6". Besides, IPv6 also offers the stateless auto-configuration service. During stateless auto-configuration, a host automatically obtains the link-local address, address prefix of the local device, and other parameter configurations.

- Security

Internet Protocol Security (IPSec) is an optional extension protocol of IPv4 but a part of IPv6 to ensure security of IPv6. At present, IPv6 provides two mechanisms: authentication header (AH) and encapsulated security payload (ESP). AH ensures data integrity and authenticates IP packet sources to ensure that the packets are from the nodes identified by the source addresses. ESP provides data encryption to realize end-to-end encryption.

- Better QoS support

A new field in the IPv6 packet header defines how to identify and process data streams. The **Flow Label** field in the IPv6 packet header is used to authenticate a data stream, and users can propose requirements on the communication quality by using this field. A device can identify all packets belonging to a specific data stream based on this field and process these packets according to user requirements.

- New protocol for neighboring node interaction

IPv6 Neighbor Discovery Protocol (NDP) uses a series of Internet Control Message Protocol Version 6 (ICMPv6) packets to implement interactive management of adjacent nodes (nodes on the same link). IPv6 uses NDP packets and efficient multicast/unicast ND packets to replace the broadcast-based Address Resolution Protocol (ARP) and Internet Control Message Protocol Version 4 (ICMPv4) router discovery packets.

- Extensibility

IPv6 features strong extensibility, allowing users to add new features to the extension packet headers following the IPv6 packet header. The IPv4 packet header supports at most 40 bytes of options. By contrast, the length of an IPv6 extension packet header is subject to only the maximum number of bytes in the packet.

1.3 Principles

1.3.1 IPv6 Basics

1. IPv6 Address Format

An IPv6 address is represented in the X:X:X:X:X:X:X:X format. A 128-bit IPv6 address is separated into eight groups by colons (:), and the 16 bits in each group are represented by four hexadecimal characters (0-9, A-F). Each "X" represents a group of four hexadecimal values. The following are three valid IPv6 addresses:

2001:ABCD:1234:5678:AAAA:BBBB:1200:2100

800:0:0:0:0:0:0:1

1080:0:0:0:8:800:200C:417A

To simplify the writing of IPv6 addresses, the number "0" in an IPv6 address can be abbreviated as follows:

- Each integer in the address must be represented, except the leading zeros in each integer. For example, 2001:00CD:0034:0078:000A:000B:1200:2100 can be abbreviated as 2001:CD:34:78:A:B:1200:2100.
- If an IPv6 address contains a string of zeros, double colons (::) can be used to represent these zeros. For example, 800:0:0:0:0:0:0:1 can be represented as 800::1. The double colons indicate that this address can be extended to a complete 128-bit address. In this approach, only when the 16-bit integers are all zeros, can they be replaced with double colons. Double colons can exist only once in an IPv6 address.

In an IPv4/IPv6 hybrid environment, in an IPv6 address, the least significant 32 bits can be used to represent an IPv4 address. This IPv6 address can be represented in a hybrid manner, that is, X:X:X:X:X:X:d.d.d.d, "X" is a hexadecimal integer and "d" is an 8-bit decimal integer. For example, 0:0:0:0:0:0:192.168.20.1 is a valid IPv6 address. It can be abbreviated to ::192.168.20.1. One typical application of this hybrid representation is to map an IPv4 address to an IPv6 address, for example, IPv4 address 1.1.1.1 is mapped to an IPv6 address ::FFFF:1.1.1.1.

2. IPv6 Address Structure

An IP address consists of two parts:

- Network prefix: n bits, similar to the network ID in an IPv4 address.
- Interface ID: (128 – n) bits, similar to the host ID in an IPv4 address.

Therefore, the length of the network prefix can be expressed using an additional value according to the classless inter-domain routing (CIDR), and this value is separated from an IPv6 address by a slash (/), for example, 12AB::CD30:0:0:0:0/60, the prefix length used for routing in the address is 60 bits.

3. IPv6 Address Type

RFC 4291 defines three types of IPv6 addresses:

- Unicast address: ID of a single interface. Packets destined to a unicast address are sent to the interface identified by this address.
- Multicast address: ID of an interface group (the interfaces generally belong to different nodes). Packets destined to a multicast address are sent to all interfaces included in this address.
- Anycast address: ID of an interface group. Packets destined to an anycast address are sent to one of the interfaces included in this address (the nearest interface according to the routing protocol).

Note

IPv6 does not define broadcast addresses.

4. Unicast Addresses

Unicast addresses are classified into five types: unspecified address, loopback address, link-local addresses, site-local addresses, and global unicast addresses. Currently site-local addresses have been abolished.

- Unspecified address

  The unspecified address is 0:0:0:0:0:0:0:0, which is usually abbreviated to ::. It has two general purposes:

  - If a host has no unicast address when started, it uses the unspecified address as the source address to send an RS packet to obtain prefix information from the gateway and thereby generate a unicast address.
  - When an IPv6 address is configured for a host, the device detects whether the address conflicts with addresses of other hosts in the same network segment and uses the unspecified address as the source address to send a neighbor solicitation (NS) packet (similar to a gratuitous ARP packet).

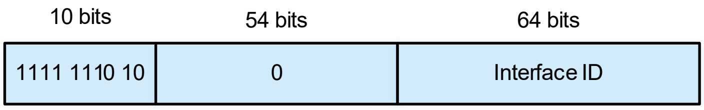
- Loopback address

  The loopback address is 0:0:0:0:0:0:0:1, which is usually abbreviated to ::1. Similar to IPv4 address 127.0.0.1, the loopback address is generally used by a node to send packets to itself.

- Link-local address

  The format of a link-local address is as follows:

Figure 1-1 Link-Local Address



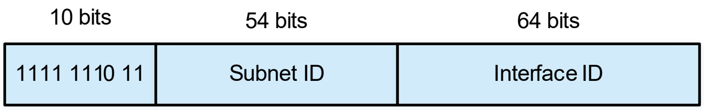
A link-local address is used to number hosts on a single network link. The address identified by the first 10 bits in the prefix is the link-local address. A device will not forward packets in which the source or destination address contains the link-local address. The middle 54 bits of this address is 0, and the last 64 bits represent the interface ID. The address part in the middle allows a single network to connect up to 2^{64} – 1 hosts.

IPv6 link-local addresses can be generated automatically by a device or specified manually.

- Site-local address

  The format of a site-local address is as follows:

Figure 1-2 Site-Local Address

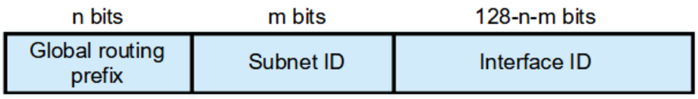


A site-local address can be used to transmit data within a site. A device never forwards packets in which the source or destination address contains a site-local address to the Internet. A site can be assumed as an enterprise's local area network (LAN). Such addresses are similar to IPv4 private addresses such as 192.168.0.0/16. RFC 3879 has abolished site-local addresses. New addresses do not support this prefix and are all regarded as global unicast addresses. Existing addresses can continue to use this prefix.

- Global unicast address

  Except unspecified, loopback, and link-local addresses, all other unicast addresses are global unicast addresses. The format of a global unicast address is as follows:

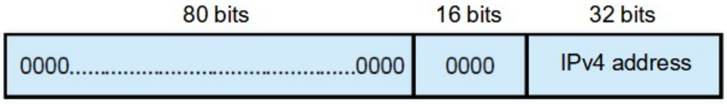
Figure 1-3 Global Unicast Address



Among global unicast addresses, there is a type of IPv4-embedded IPv6 addresses, including IPv4-compatible IPv6 addresses and IPv4-mapped IPv6 addresses. They are used for interconnection between IPv4 nodes and IPv6 nodes.

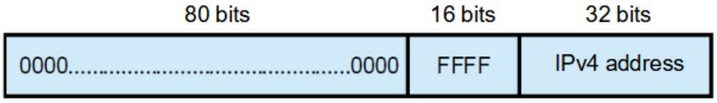
- The format of an IPv4-compatible IPv6 address is as follows:

Figure 1-4 Format of an IPv4-compatible IPv6 Address



- The format of an IPv4-mapped IPv6 address is as follows:

Figure 1-5 Format of an IPv4-mapped IPv6 Address



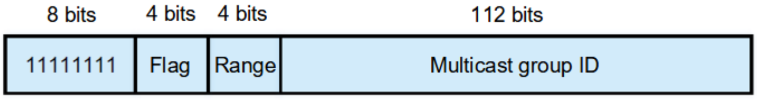
IPv4-compatible IPv6 addresses are mainly used on automatic tunnels. Nodes on automatic tunnels support both IPv4 and IPv6. Using these addresses, IPv4 devices transmit IPv6 packets over tunnels. At present, IPv4-compatible IPv6 addresses have been abolished. IPv4-mapped IPv6 addresses are used by IPv6 nodes to access IPv4-only nodes. For example, if an IPv6 application on an IPv4/IPv6 host requests to resolve the name of an IPv4-only host, the name server dynamically generates an IPv4-mapped IPv6 address and returns it to the IPv6 application.

5. Multicast Addresses

One IPv6 multicast address usually identifies interfaces on a series of different nodes. After a packet is sent to a multicast address, the packet is then forwarded to the interfaces on each node identified by this multicast address.

A multicast address is prefixed with FF00::/8. The format of an IPv6 multicast address is as follows:

Figure 1-1 Multicast Address



- Flag field

  The flag field consists of four bits. Currently only the fourth bit is specified. If this flag bit is 0, the address is a permanent known multicast address assigned by the Internet Assigned Numbers Authority (IANA). If this flag bit is 1, the address is a temporary multicast address used in a certain scenario. The remaining three flag bits are reserved for future use and set to 0.

- Scope field:

  The scope field consists of four bits to indicate the scope type of a multicast address. The following table lists common multicast address scope types and their corresponding values. Unlisted values indicate reserved or unspecified.

Table 1-1 Common Multicast Address Scope Types

Binary Value	Hexadecimal Value	Scope Type
0001	1	Local interface scope
0010	2	Local link scope
0011	3	Local subnet scope
0100	4	Local management scope
0101	5	Local site scope
1000	8	Organization scope
1110	E	Global scope

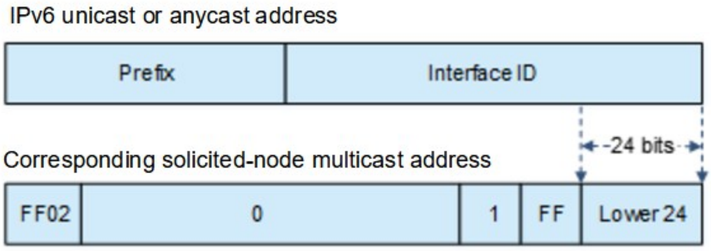
- Multicast group ID field:

  The multicast group ID field consists of 112 bits to identify a multicast group. A multicast group ID can represent different groups based on the flag and scope fields.

Multicast addresses for special purposes:

- FF01::1 represents the multicast address of all nodes in the node-local scope.
- FF02::1 represents the multicast address of all nodes in the link-local scope.
- FF01::2 represents the multicast address of all devices in the node-local scope.
- FF02::2 represents the multicast address of all devices in the link-local scope.
- Solicited-node multicast address. The address is mainly used to obtain the link layer address of a neighbor node on the same link and perform the duplicate address detection. Each unicast or anycast IPv6 address has a corresponding solicited-node address. The address is in the format of FF02:0:0:0:0:1:FFXX:XXXX, FF02:0:0:0:0:1:FF is the fixed 104 bits, and XX:XXXX is the last 24 bits of a unicast or anycast IPv6 address. Solicited-node multicast addresses are usually used in NS packets. The address format is as follows:

Figure 1-2 Format of a Solicited-Node Multicast Address


IPv6 unicast or anycast address
Prefix
Interface ID
-24 bits-
Corresponding solicited-node multicast address
FF02
0
1
FF
Lower 24

6. Anycast Addresses

Similar to a multicast address, an anycast address can also be shared by multiple nodes. The difference is that only one node using the anycast address receives data packets while all nodes using the multicast address receive data packets. Since anycast addresses are allocated from the normal IPv6 unicast address space, they use the same format as unicast addresses. Therefore, each member of an anycast address group must be configured explicitly for easier recognition.

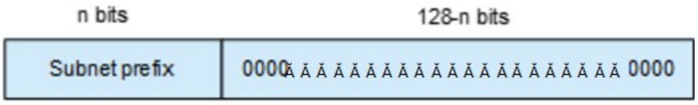
Caution

Anycast addresses can be allocated only to devices not hosts and cannot be used as source addresses of packets.

RFC 2373 defines an anycast address called subnet-router anycast address. The following figure shows the format of a subnet-router anycast address. Such an address consists of the subnet prefix and a series of zeros (interface ID).

The subnet prefix identifies a specified link (subnet). Packets destined to the subnet-router anycast address will be forwarded to a device on this subnet. A subnet-router anycast address is usually used by the application on a node to communicate with a device on a remote subnet.

Figure 1-1 Format of a Subnet Router Anycast Address



Note

The format of an anycast address is shown in Figure 1-1, that is, n bits of subnet prefix plus (128 – n) bits of all zeros. For example, in the anycast address 1000:1::100/120, the length of the subnet prefix is 120 bits, and the host address is 8-bit all zeros. An anycast address cannot be used as the interface address of a switch.

7. IEEE EUI-64 Interface ID

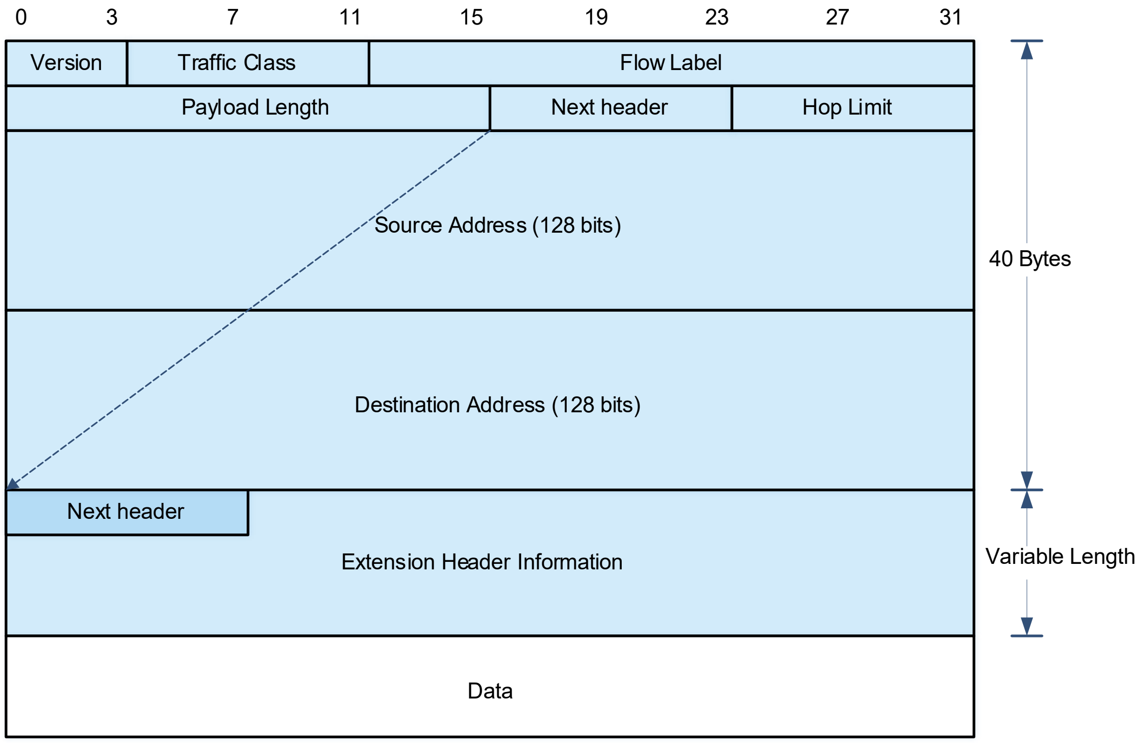
The 64-bit interface ID in an IPv6 address is used to identify a unique interface on the link. This address is a variation of a link-layer address (such as an MAC address) of an interface. This address is generated by inserting a hexadecimal number FFFE after the 24th bit of an MAC address. To make the scope of an interface ID the same as the original MAC address, the U/L bit (the 7th bit starting from the most significant bit) is negated. The resulting set of numbers is used as the interface ID in EUI-64 format.

For the loopback, VBDIF, and tunnel interfaces, the EUI-64 addresses are generated from the MAC addresses of these interfaces, with the last two bytes of the interface index filled in the middle.

8. Format of an IPv6 Packet Header

The format of an IPv6 packet header is shown in the following figure.

Figure 1-1 Format of an IPv6 Packet Header



The IPv6 packet header consists of 40 bytes, in unit of eight bytes. An IPv6 packet header has the following fields:

- Version

  This field consists of 4 bits. In an IPv6 address, this field must be 6.

- Traffic Class

  This field consists of 8 bits. This field indicates the service provided for this packet, and is similar to the type of service (ToS) field in an IPv4 address.

- Flow Label

  This field consists of 20 bits to identify packets belonging to the same service flow. One node can act as the transmission source of multiple service flows. The flow label and source node address uniquely identify one service flow.

- Payload Length

  This field consists of 16 bits, including the packet payload length and the length of IPv6 extended options (if available). That is, it includes the IPv6 packet length except the IPv6 packet header.

- Next Header

  This field indicates the protocol type in the header field following the IPv6 packet header. Similar to the protocol field in the IPv4 address header, the **Next Header** field is used to indicate whether the upper layer uses Transmission Control Protocol (TCP) or User Datagram Protocol (UDP). It can also be used to indicate existence of an IPv6 extension header.

- Hop Limit

This field contains eight bits. Every time a device forwards a packet, the hop value is reduced by 1. If the hop value reaches 0, this packet will be discarded. It is similar to the lifetime field in the IPv4 packet header.

- Source Address

  This field consists of 128 bits and indicates the sender address in an IPv6 packet.

- Destination Address

  This field consists of 128 bits and indicates the receiver address in an IPv6 packet.

At present, IPv6 defines the following extension headers.

- Hop-By-Hop Options

  This extension header must follow the IPv6 packet header. It consists of option data to be checked on each node along the path.

- Routing Options

  This extension header indicates the nodes that a packet passes through from the source address to the destination address. It consists of the address list of the passerby nodes. The initial destination address in an IPv6 packet header is the first address among the addresses in the routing header, but not the final destination address of the packet. After the node corresponding to the destination address in an IPv6 packet header receives the packet, it processes the IPv6 packet header and routing header, and sends the packet to the second address, the third address, and so on in the list in the routing header till the packet reaches the final destination address.

- Fragment

  The source node uses this extension header to fragment the packets, of which the length exceeds the path MTU (PMTU).

- Destination Options

  This extension header replaces the option fields of IPv4. At present, the **Destination Options** field can only be filled with integral multiples of 64 bits (eight bytes) if required. This extension header can be used to carry information to be checked by the destination node.

- Upper-layer Header

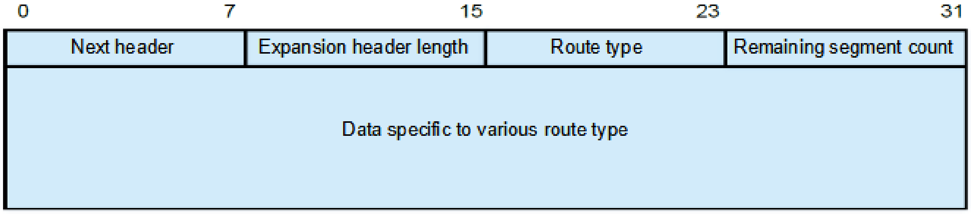
  This extension header indicates the protocol used by the upper layer for data transmission, such as TCP (6) and UDP (17).

Another two extension headers **Authentication** (AH) and **Encapsulating Security Payload** (ESP) will be described in "Configuring IPSec" in "VPN Configuration Guide".

9. IPv6 Source Routing

Similar to the IPv4 loose source routing and loose record routing options, the IPv6 routing header is used to specify the intermediate nodes that a packet passes through. It uses the following format:

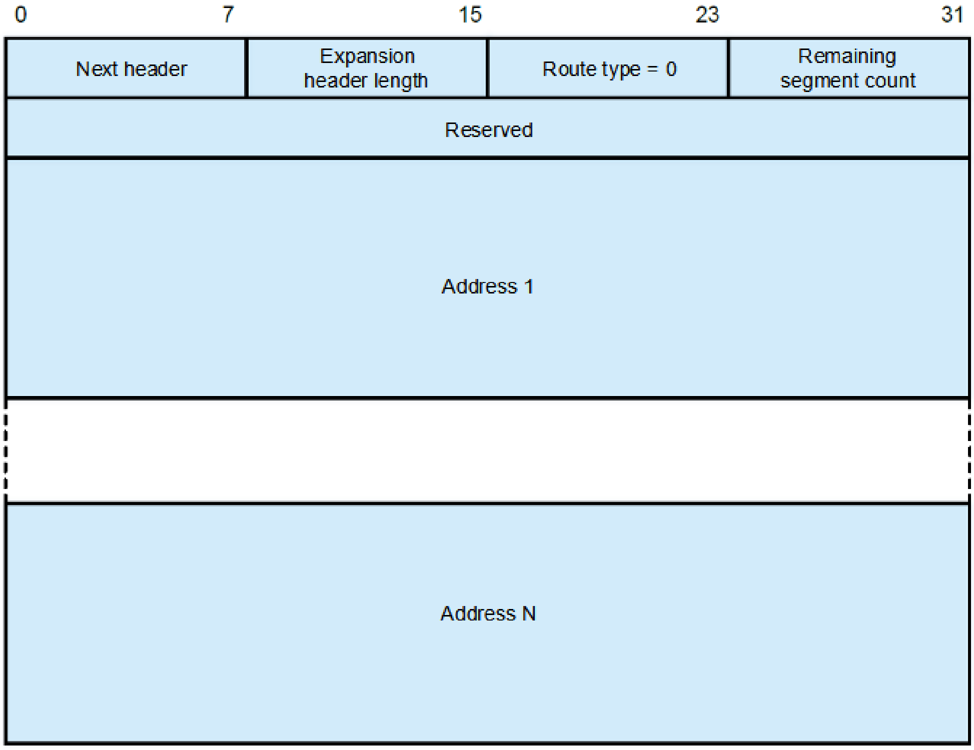
Figure 1-1 IPv6 Routing Header



The **Segments Left** field is used to indicate the number of intermediate nodes specified in the routing header that a packet passes through from the current node to the final destination address.

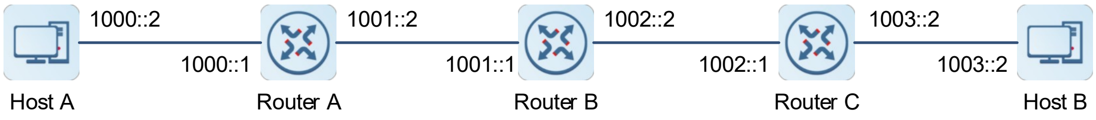
Currently, two routing types are defined: 0 and 2. The Type 2 routing header is used for mobile communication. Type 0 routing header is similar to the loose source routing option of IPv4. The format of the Type 0 routing header is as follows:

Figure 1-2 Type 0 Routing Header



The example describes the application of the Type 0 routing header.

Figure 1-3 Applications of the Type 0 Routing Header



Host A sends host B a packet, which specifies the intermediate nodes router B and router C. The following table lists the changes of fields related to the IPv6 header and routing header during the forwarding process.

Table 1-1 Changes of Fields

Transmission Node	Fields in the IPv6 Header	Fields Related to the Type 0 Routing Header
Host A	Source address = 1000::2 Destination address = 1001::1 (address of router 2)	Segments Left = 2 Address 1 = 1002::1 (address of router 3) Address 2 = 1003::2 (address of host 2)
Router A	No change	
Router B	Source address = 1000::2 Destination address = 1002::1 (address of router 3)	Segments Left = 1 Address 1 = 1001::1 (address of router 2) Address 2 = 1003::2 (address of host 2)
Router C	Source address = 1000::2 Destination address = 1003::2 (address of host 2)	Segments Left = 0 Address 1 = 1001::1 (address of router 2) Address 1 = 1002::2 (address of router 3)
Host B	No change	

The forwarding process is described as follows:

(1) Host A sends a packet in which the destination address is router B's address 1001::1, the Type 0 routing header is filled with router C's address 1002::1 and host B's address 1003::2, and the value of the **Segments Left** field is **2**.

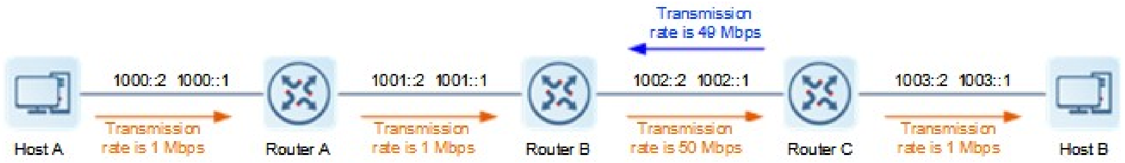
(2) Router A forwards the packet to router B.

(3) Router B interchanges the destination address in the IPv6 header with address 1 in the routing header. That is, the destination address becomes router C's address 1002::1, address 1 in the routing header becomes router B's address 1001::1, and the value of the **Segments Left** field becomes **1**. After modification, router B forwards the packet to router C.

(4) Router C interchanges the destination address in the IPv6 header with address 2 in the routing header. That is, the destination address becomes host B's address 1003::2, address 2 in the routing header becomes router C's address 1002::1, and the value of the **Segments Left** field becomes **0**. After modification, router C forwards the packet to host B.

The Type 0 routing header may be used to initiate denial of service (DoS) attacks. As shown in the following figure, host 1 sends a packet to host 2 at a rate of 1 Mbps and forges a routing header with the **Segments Left** field of 100 to cause multiple round-trips of the packet between router 2 and router 3 (50 times from router 2 to router 3 and 49 times from router 3 to router 2). At the time, the routing header generates the traffic amplification effect: 50 Mbps from router 2 to router 3 and 49 Mbps from router 3 to router 2. Due to this security problem, RFC 5095 abolished the Type 0 routing header.

Figure 1-1 Type 0 Routing Header Used to Initiate DoS Attacks


Transmission rate is 40 Mbps
Host A
1000::2 1000::1
Transmission rate is 1 Mbps
Router A
1001::2 1001::1
Transmission rate is 1 Mbps
Router B
1002::2 1002::1
Transmission rate is 50 Mbps
Router C
1003::2 1003::1
Transmission rate is 1 Mbps
Host B

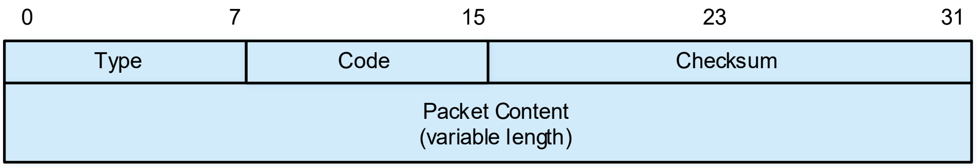
10. IPv6 Hop-limit

An IPv6 data packet passes through routers from the source address to the destination address. If a hop limit is configured, it decreases by one every time the packet passes through a router. When the hop limit decreases to 0, the router discards the packet to prevent this useless packet from being unlimitedly transmitted on the network and wasting network bandwidth. The hop limit is similar to the time to live (TTL) of IPv4.

1.3.2 ICMPv6 Protocol

1. Packet Format

Figure 1-1 ICMPv6 Packet Format



Field descriptions are as follows:

- The **Type** field indicates the type of a packet, 0 to 127 indicates error messages, and 128 to 255 indicates informational messages.
- The **Code** field indicates the subdivision type of this message type.
- The **Checksum** field indicates the checksum of ICMPv6 packets.

2. ICMPv6 Error Messages

ICMPv6 error messages are sent by a destination node or intermediate router to report the errors incurred during IPv6 data packet forwarding and transmission. There are mainly four types of error messages:

- Destination Unreachable (Type=1), indicating that packets cannot be forwarded to the destination node or the upper-layer protocol.
  - Code=0, indicating that there is no route reachable to the destination.
  - Code=1, indicating that the communication with the destination is prohibited according to the administration policy.
  - Code=2, unspecified.
  - Code=3, address unreachable.
  - Code=4, port unreachable.
- Packet Too Big (Type = 2, Code = 0), indicating that the size of a packet exceeds the link MTU of an

interface.

- Time Exceeded (Type=3)
  - Code=0, indicating that the hop limit is exceeded during transmission.
  - Code=1, fragmentation timeout.
- Parameter Problem (Type=4), indicating that an error occurs in the IPv6 header or extension header.
  - Code=0, indicating an incorrect header field.
  - Code=1, indicating that the next header cannot be identified.
  - Code=2, indicating that the IPv6 option cannot be identified.

3. ICMPv6 Informational Messages

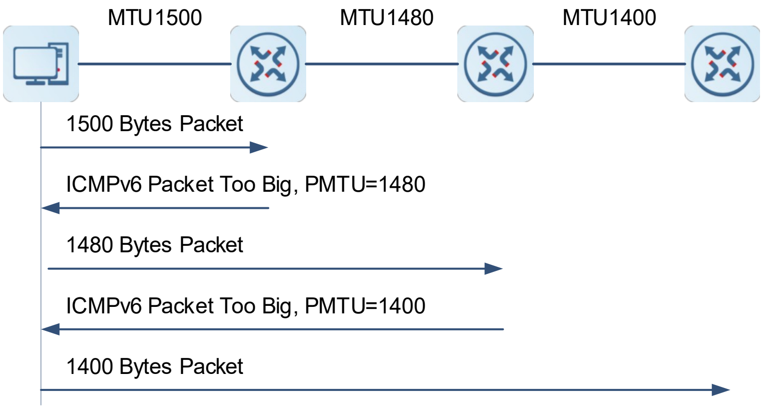
The following describes common informational messages:

- Type=128, echo request, which is sent to the destination node to trigger the destination node to immediately send back an echo reply packet.
- Type=129, echo reply, which is sent in response to an echo request.
- Type=133, router solicitation (RS) packet. After a node is started, it initiates a solicitation to the router via an RS packet to request the prefix and other configurations.
- Type=134, router advertisement (RA) packet, which is sent in response to an RS packet and advertises prefix options and certain flag bits.
- Type=135, neighbor solicitation (NS) packet, which is similar to an ARP request and used to obtain the link layer address of a neighbor, check whether the neighbor is reachable, and perform duplicate address detection.
- Type=136, neighbor advertisement (NA) packet, which is similar to an ARP reply packet.
- Type=137, Redirect packet. By sending a Redirect packet to the source host, the default gateway triggers the host to re-select a better next-hop address for delivering subsequent packets.

1.3.3 IPv6 PMTUD

Similar to IPv4 path MTU discovery (PMTUD), IPv6 PMTUD allows a host to dynamically discover the MTU size in the data path. If the length of a data packet to be sent by a host is greater than the PMTU, the host performs packet fragmentation.

Figure 1-1 IPv6 PMTUD



As shown in the preceding figure, if the length of a packet to be sent by a host is greater than the PMTU, the router discards this packet and sends an ICMPv6 Packet Too Big message containing its PMTU to the host. The host fragments the packet based on the new PMTU. In this manner, the router does not need to perform fragmentation, saving router resources and improving the IPv6 network efficiency.

1.3.4 IPv6 Neighbor Discovery Protocol

Neighbor Discovery Protocol (NDP) is a basic part of IPv6. Its main functions include router discovery, prefix discovery, parameter discovery, address auto-configuration, address resolution (like ARP), next-hop determination, neighbor unreachability detection (NUD), duplicate address detection (DAD), and redirection.

NDP defines five types of ICMP packets: RS, RA, NS, NA, and Redirect.

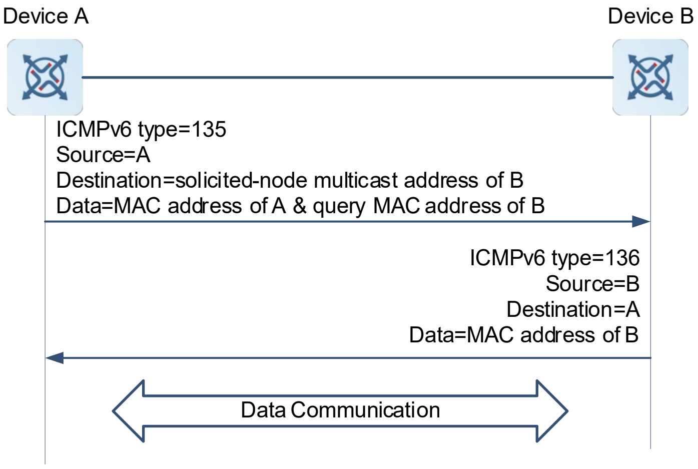
All the above ICMP packets carry one or more options. These options are optional in some cases but are significant in other cases. NDP mainly defines five options: **Source Link-Layer Address** option, Type = 1; **Target Link-Layer Address** option, Type = 2; **Prefix Information** option, Type = 3; **Redirect Header** option, Type = 4; **MTU** option, Type = 5.

Different NDP mechanisms use different ICMPv6 messages.

1. Address Resolution

When a node attempts to communicate with another, the node has to obtain the link-layer address of the peer by sending an NS packet to the peer. In this packet, the destination address is the solicited-node multicast address corresponding to the IPv6 address of the destination node. This packet also contains the link-layer address of the source node. After receiving this NS packet, the destination node replies with an NA packet, in which the destination address is the source address of the NS packet and the content is the link-layer address of the solicited node. After receiving this NA packet, the source node can communicate with the destination node.

Figure 1-1 Address Resolution



The process of address resolution is as follows:

(1) Device A sends an NS packet in multicast mode. The source address of the NS packet is the IPv6 address of the interface of device A, and the destination address is the solicited-node multicast address of device B. The packet contains the link layer address of device A, and solicits the link layer address of device B.

(2) After receiving the NS packet, device B judges whether the destination address of the packet is its own IPv6 address. If yes, device B can learn the link layer address of device A and returns an NA packet containing its link layer address in unicast mode.

(3) Device A can get the link layer address of device B from the received NA packet. The two devices can then begin to communicate with each other.

2. NUD

If the amount of time that a neighbor is considered reachable has elapsed but an IPv6 unicast packet needs to be sent to the neighbor, the device performs NUD. The NS and NA packets can be used to detect whether the neighbor node is reachable.

(1) The node sends an NS packet, in which the destination address is the IPv6 address of the neighbor node.

(2) If the acknowledgment packet is received from the neighbor node, the neighbor is considered reachable. Otherwise, the neighbor is considered unreachable.

While NUD is underway, the device can continue to forward IPv6 packets to the neighbor.

3. DAD

When obtaining an IPv6 address, a node needs to perform DAD to determine whether the IPv6 address is unique on the link. This feature is similar to the gratuitous ARP in IPv4.

The process of DAD is as follows:

(1) A node sends an NS packet. The source IPv6 address of the NS packet is the unspecified address, and the destination address is the solicited-node multicast address corresponding to the IPv6 address to be detected. The packet content is the IPv6 address to be detected.

(2) After receiving the NS packet, other neighbor nodes check whether the IPv6 address exists in their local IPv6 address collections. If yes, the neighbor node that has the IPv6 address responds with an NA packet that carries the IPv6 address to the source node.

(3) When receiving the NA packet from the neighbor, the source node considers the IPv6 address in use. On the contrary, if the source node receives no NA packet in response to the NS packet, the IPv6 address is available.

If a device detects an address conflict, this address is set in the duplicate state and the device cannot receive IPv6 packets destined to this address. Meanwhile, the device also starts a timer for this duplicate address to periodically perform DAD. If no address conflict is detected, this address can be properly used.

4. RA Packets

An RA packet usually contains the following content:

- One or more IPv6 address prefixes (used for on-link determination or stateless address auto-configuration);
- Validity period of the IPv6 address prefix;
- Host auto-configuration method (stateful or stateless);
- Default device information (whether the device acts as the default device; if yes, the duration for acting as the default device is also included);
- Other information provided for host configuration, such as hop limit, MTU and NS retransmission interval.

RA packets can also be used as replies to the RS packets sent by a host. Using RS packets, a host can obtain the auto-configured information immediately after startup rather than wait for the RA packets sent by the device. If no unicast address is configured for a newly started host, the host uses the unspecified address (0:0:0:0:0:0:0:0) as the source address of the RS packet. Otherwise, the host uses the configured unicast address as the source address and the multicast address of all devices in the local link (FF02::2) as the destination address in the RS packet. As a reply to the RS packet, the RA packet uses the source address of the RS packet as the destination address. If the source address is the unspecified address, it uses the multicast address of all nodes in the local link (FF02::1).

In an RA packet, the following parameters can be configured in the IPv6 interface attributes:

- Ra-interval: Indicates the interval for sending the RA packets.
- Ra-lifetime: Indicates the lifetime of a router, that is, whether the device acts as the default router on the local link and the duration for acting as the default router.
- Prefix: Indicates the prefix of an IPv6 address of the local link. It is used for on-link determination or stateless address auto-configuration, including other parameter configurations related to the prefix.
- Ns-interval: Indicates the NS packet retransmission interval.
- Reachable-time: Indicates the period in which the device regards a neighbor reachable after detecting the neighbor reachability event.
- Ra-hoplimit: Indicates the hop count in an RA packet, which is used to set the hop limit for a host to send a unicast packet.
- Ra-mtu: Indicates the value of the MTU field in an RA packet.
- Managed-config-flag: Determines whether a host receiving this RA packet obtains the address through stateful auto-configuration.
- Other-config-flag: Determines whether a host receiving this RA packet uses DHCPv6 to obtain other information except the IPv6 address for auto-configuration.
- RDNSS and DNSSL options: Provide IPv6 terminals with the address of the DNS recursive query server and a search list of DNS domain names.

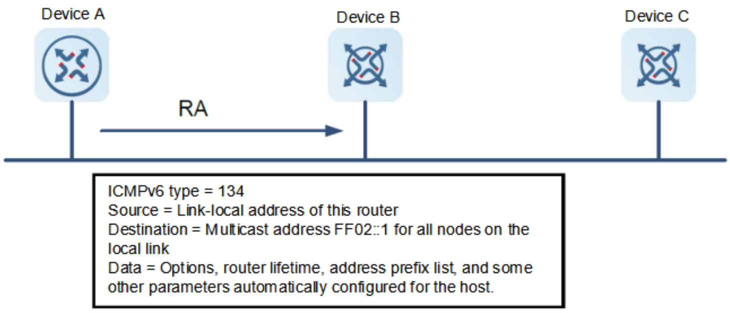
5. Router Discovery/Prefix and Parameter Discovery

RA packets are sent by the device to all link-local nodes on a regular basis. From the RA packets received, the nodes obtain neighbor router information and other configuration parameters. This is called router discovery/prefix and parameter discovery.

The nodes automatically configure the IPv6 address based on the information obtained from router discovery/prefix discovery, which is called stateless auto-configuration.

The RA packet sending process during router discovery/prefix and parameter discovery is shown in the following figure:

Figure 1-1 Router Discovery/Prefix and Parameter Discovery



ICMPv6 type = 134
Source = Link-local address of this router
Destination = Multicast address FF02::1 for all nodes on the local link
Data = Options, router lifetime, address prefix list, and some other parameters automatically configured for the host.

6. Redirection

If a router receiving an IPv6 packet finds a better next hop, it sends an ICMP Redirect packet to inform the host of the better next hop. The host will directly send the IPv6 packet to the better next hop next time. This function is the same as ICMP Redirect packets of IPv4.

1.3.5 Protocols and Standards

- RFC4291: IPv6 Addressing Architecture.
- RFC2460: Internet Protocol, Version 6 (IPv6) Specification
- RFC4443: Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification
- RFC4861: Neighbor Discovery for IP version 6 (IPv6)
- RFC4862: IPv6 Stateless Address Autoconfiguration
- RFC5059: Deprecation of Type 0 Routing Headers in IPv6

1.4 Configuring IPv6 Basic Information

1.4.1 Overview

IPv6 is updated from IPv4 and solves many shortcomings of the IPv4.

The most significant difference between IPv6 and IPv4 is the upgrade of the IP address length from 32 bits to 128 bits. IPv6 features simplified header format, adequate address space, hierarchical address structure, flexible extension headers, and enhanced neighbor discovery mechanism and will be full of vitality in the future market competition.

1.4.2 Configuration Task Summary

The IPv6 basic configuration includes the following tasks:

(1) Configuring an IPv6 Address

(2) (Optional) Configuring the IPv6 MTU for an Interface

(3) (Optional) Configuring IPv6 Source Routing

(4) (Optional) Configuring the IPv6 Hop Limit

(5) (Optional) Configuring the Default Gateway for a Management Interface

(6) Configure the sending of ICMPv6 packets. The following configuration tasks are optional. Select either of them for configuration according to the actual condition.

- Configuring the Specified Source Address for Sending ICMPv6 Packets
- Configuring the Transmission Rate of ICMPv6 Error Messages

1.4.3 Configuring an IPv6 Address

1. Overview

This section describes how to enable an interface to support the IPv6 protocol, and configure an IPv6 address for the interface to achieve IPv6 network communication.

The IPv6 link-local address can be manually configured or automatically obtained.

2. Restrictions and Guidelines

- Anycast addresses cannot be configured as interface IPv6 addresses. Exceptionally, anycast addresses with a subnet prefix of 127 or greater can be configured.
- If an interface is bound to a multiprotocol virtual routing and forwarding (VRF) instance not configured with IPv6 address family, IPv6 cannot be enabled on this interface or an IPv6 address cannot be configured for this interface. You can enable IPv6 or configure an IPv6 address for this interface only after configuring an IPv6 address family for the multiprotocol VRF instance.
- IPv6 can be enabled on an interface by two methods: (1) Run the **ipv6 enable** command in interface configuration mode by referring to Step (4); (2) Configure an IPv6 address for the interface by referring to Step (5). If an IPv6 address is configured on an interface, IPv6 is automatically enabled on this interface. In this case, IPv6 cannot be disabled even when you run the **no ipv6 enable** command.

3. Procedure

(1) Enter the privileged EXEC mode.

enable

(2) Enter the global configuration mode.

configure terminal

(3) Enter the interface configuration mode.

interface *interface-type interface-number*

(4) (Optional) Enable IPv6 on the interface.

ipv6 enable

IPv6 is disabled on an interface by default.

(5) Configure an IPv6 address for the interface. The configuration steps below are mutually exclusive. Please configure only one task.

- Manually configure an IP address for the interface.

  ipv6 address { *ipv6-address*/*prefix-length* | *ipv6-prefix*/*prefix-length* **eui-64** | *prefix-name sub-bits*/*prefix-length* [**eui-64**] }

  No IPv6 address is configured by default.

If an IPv6 interface is created and is in up state, the system automatically generates a link-local address for this interface.

- ○ Use the general prefix mechanism to generate an IPv6 address for the interface.

  ipv6 general-prefix *prefix-name ipv6-prefix*/*prefix-length*

  No IPv6 general prefix is configured by default.

- ○ Enable the IPv6 stateless address auto-configuration on the interface.

  ipv6 address autoconfig

  The IPv6 stateless address auto-configuration is disabled on an interface by default.

1.4.4 Configuring the IPv6 MTU for an Interface

1. Overview

IPv6 PMTUD allows a host to dynamically discover the MTU size on the data path. When sending an IPv6 packet, a host compares the IPv6 MTU of the outbound interface with the PMTU and determines which MTU is smaller. If the packet length is greater than the smaller MTU, the host fragments the packet based on the smaller MTU.

2. Restrictions and Guidelines

The IPv6 MTU of an interface must be less than or equal to the interface MTU.

3. Procedure

(1) Enter the privileged EXEC mode.

enable

(2) Enter the global configuration mode.

configure terminal

(3) Enter the interface configuration mode.

interface *interface-type interface-number*

(4) Configure the IPv6 MTU for the Interface.

ipv6 mtu *bytes*

The IPv6 MTU value is the same as the value configured by running the **mtu** command on an interface by default.

1.4.5 Configuring IPv6 Source Routing

1. Overview

Similar to the IPv4 loose source routing and loose record routing options, the IPv6 routing header is used to specify the intermediate nodes that the packet passes through.

2. Restrictions and Guidelines

Since the Type 0 routing header may cause the device vulnerable DoS attacks, the device does not forward IPv6 packets carrying the routing header by default, but still processes IPv6 packets that are finally destined to itself and carry the Type 0 routing header.

An administrator can run the **ipv6 source-route** command in global configuration mode to enable IPv6 source routing.

3. Procedure

(1) Enter the privileged EXEC mode.

enable

(2) Enter the global configuration mode.

configure terminal

(3) Configures the device to forward IPv6 packets carrying the routing header.

ipv6 source-route

IPv6 packets carrying the routing header are not forwarded by default.

1.4.6 Configuring the IPv6 Hop Limit

1. Overview

This section describes how to configure the hop limit for unicast packets to prevent infinite transmission of the packets on the network.

2. Procedure

(1) Enter the privileged EXEC mode.

enable

(2) Enter the global configuration mode.

configure terminal

(3) Configure the hop limit for unicast packets.

ipv6 hop-limit *hop*

The default hop limit for unicast packets is 64.

1.4.7 Configuring the Default Gateway for a Management Interface

1. Overview

This section describes how to configure the default gateway for a management interface. After configuration, a default route is generated, with the outbound interface of the management interface and the next hop of the configured gateway.

2. Restrictions and Guidelines

This command can be configured on management interfaces only.

3. Procedure

(1) Enter the privileged EXEC mode.

enable

(2) Enter the global configuration mode.

configure terminal

(3) Enter the interface configuration mode.

interface mgmt *interface-number*

(4) Configure the default gateway for the management interface.

ipv6 gataway *ipv6-address*

No IPv6 default gateway is configured for a management interface by default.

1.4.8 Configuring the Specified Source Address for Sending ICMPv6 Packets

1. Overview

In a network with a large number of IPv6 addresses configured, it is complex for receivers to recognize the device, from which an ICMPv6 packet is sent. To simplify the judgment, you can configure a specified source address for ICMPv6 packets.

2. Restrictions and Guidelines

You can configure a specified address, like the address of the loopback interface, as the source address of ICMPv6 packets to simplify judgment.

3. Procedure

(1) Enter the privileged EXEC mode.

enable

(2) Enter the global configuration mode.

configure terminal

(3) Configure a specified source address for sending ICMPv6 packets.

ipv6 icmp source [**vrf** *vrf-name*] *ipv6-address*

No address is configured as the specified source address for sending ICMPv6 packets by default.

1.4.9 Configuring the Transmission Rate of ICMPV6 Error Messages

1. Overview

When receiving an invalid IPv6 packet, a device will discard the packet and send an ICMPv6 error message to the source IPv6 address. When attacked by a large number of invalid IPv6 packets, the device continuously replies with ICMPv6 error messages till resources are exhausted and thereby fail to properly provide services. This is called DoS attacks. To prevent DoS attacks, you can restrict the transmission rate of ICMPv6 error messages.

If the length of an IPv6 packet to be forwarded exceeds the IPv6 MTU of the outbound interface, the router discards this IPv6 packet and sends an ICMPv6 Packet Too Big message to the source IPv6 address. This error message is mainly used for IPv6 PMTUD. When there are too many ICMPv6 error messages, the ICMPv6 Packet Too Big message may be filtered out, causing the failure of the IPv6 PMTUD function. To avoid this problem, you are advised to restrict the transmission rates of the ICMPv6 Packet Too Big message and other ICMPv6 error messages separately.

Note

Although ICMPv6 Redirect packets are not a type of ICMPv6 error messages, the device restricts their rates together with other ICMPv6 error messages except Packet Too Big messages.

2. Restrictions and Guidelines

Since the precision of the timer is 10 milliseconds, you are advised to set the refresh cycle of a token bucket to an integer multiple of 10 milliseconds. If the refresh cycle is set to a value greater than 0 and smaller than 10 milliseconds, the refresh cycle that actually takes effect is 10 milliseconds. For example, if the refresh rate is set to send one packet within the refresh cycle of 5 milliseconds, the refresh rate that actually takes effect is that two packets are sent within the refresh cycle of 10 milliseconds. If the refresh cycle is not an integral multiple of 10 milliseconds, the refresh cycle that actually takes effect is automatically converted into an integral multiple of 10 milliseconds. For example, if the refresh rate is set to send three packets within the refresh cycle of 15 milliseconds, the refresh rate that actually takes effect is that two packets are sent within the refresh cycle of 10 milliseconds.

3. Procedure

(1) Enter the privileged EXEC mode.

enable

(2) Enter the global configuration mode.

configure terminal

(3) Configure the transmission rate of the ICMPv6 Too Big message.

ipv6 icmp error-interval too-big *interval* [*bucket-size*]

Ten messages are transmitted within 100 ms by default.

(4) Configure the transmission rate of other ICMPv6 error messages.

ipv6 icmp error-interval *interval* [*bucket-size*]

Ten messages are transmitted within 100 ms by default.

1.5 Configuring IPv6 NDP

1.5.1 Overview

Neighbor Discovery Protocol (NDP) is a basic part of IPv6. Its main functions include router discovery, prefix discovery, parameter discovery, address auto-configuration, address resolution (like ARP), next-hop determination, neighbor unreachability detection (NUD), duplicate address detection (DAD), and redirection.

1.5.2 Configuration Task Summary

All the configuration tasks below are optional. Perform the configuration tasks as required.

- Configuring a Static Neighbor Entry
- Configuring ND Entry Management
- Configuring Address Resolution
- Configuring NUD
- Configuring DAD
- Configuring Redirection
- Configuring Parameters Related to RA Packets
- Configuring ND Logging

- Configuring the Interval for ND Packet Rate Statistics Collection
- Configuring Local ND Proxy
- Restraining an Interface from Sending NS Packets to Authenticated VLANs
- Configuring the ND Protocol to Allocate Different Prefixes to Different Users
- Configuring Local ND Proxy

1.5.3 Configuring a Static Neighbor Entry

1. Overview

A static neighbor entry is uniquely identified by the IPv6 address of a neighbor node and the ID of an L3 interface connected to this neighbor node.

2. Restrictions and Guidelines

A static neighbor entry can be configured on IPv6-enabled interfaces only. If the neighbor entry to be configured is learned through NDP and stored in the neighbor table, the dynamic neighbor entry will be automatically converted into a static one. An effective static neighbor entry will be always reachable. An invalid static neighbor entry refers to a static neighbor entry with the configured IPv6 address not matching the address configured on the interface (not within any IPv6 network segment of this interface, or in conflict with the address of this interface). In this case, packets will not be forwarded through the MAC address specified in the static neighbor entry.

3. Procedure

(1) Enter the privileged EXEC mode.

enable

(2) Enter the global configuration mode.

configure terminal

(3) Configure a static neighbor entry.

ipv6 neighbor *ipv6-address interface-type interface-number mac-address*

No static neighbor entry is configured by default.

1.5.4 Configuring ND Entry Management

1. Overview

- You can restrict the number of ND entries learned by an interface to prevent malicious neighbor learning attacks. If this number is not restricted, a large number of ND entries will be generated on the device, occupying excessive memory space and affecting forwarding performance.
- You can restrict the number of unresolved ND entries to prevent malicious scanning attacks from causing the generation of a large number of unresolved ND entries and occupying entry resources.
- You can configure the maximum number of ND options to prevent forged ND packets from carrying unlimited ND options and occupying excessive CPU space on the device.
- When an interface is enabled to learn ND entries via DAD NS packets, the interface will create ND entries in stale state when receiving the DAD NS packets.

2. Configuration Tasks

All the configuration tasks below are optional. Perform the configuration tasks as required.

- Configuring the Maximum Number of ND Entries That Can Be Learned
- Configuring the Maximum Number of Unresolved ND Entries
- Configuring the Maximum Number of ND Options That Can Be Processed
- Enabling an Interface to Learn ND Entries via DAD NS Packets

3. Restrictions and Guidelines

By default, a general interface is disabled to learn ND entries via DAD NS packets but interfaces in a super VLAN are allowed to do so.

4. Configuring the Maximum Number of ND Entries That Can Be Learned

(1) Enter the privileged EXEC mode.

enable

(2) Enter the global configuration mode.

configure terminal

(3) Enter the interface configuration mode.

interface *interface-type interface-number*

(4) Configure the maximum number of ND entries that can be learned by an interface.

ipv6 nd cache interface-limit *limit*

The number of neighbor cache entries that can be learned by an interface is not limited by default.

The configured value must be equal to or greater than the number of the neighbor cache entries learned by the current interface. Otherwise, the configuration does not take effect. The configuration is subject to the ND entry capacity supported by the device.

5. Configuring the Maximum Number of Unresolved ND Entries

(1) Enter the privileged EXEC mode.

enable

(2) Enter the global configuration mode.

configure terminal

(3) Enter the interface configuration mode.

interface *interface-type interface-number*

(4) Configure the maximum number of unresolved ND entries.

ipv6 nd unresolved *number*

The default maximum number of unresolved ND entries is 0, indicating no restriction. The number is only subject to the ND entry capacity supported by the device.

6. Configuring the Maximum Number of ND Options That Can Be Processed

(1) Enter the privileged EXEC mode.

enable

(2) Enter the global configuration mode.

configure terminal

(3) Enter the interface configuration mode.

interface *interface-type interface-number*

(4) (Optional) Configure the maximum number of ND options that can be processed.

ipv6 nd max-opt *option*

The device supports 10 ND options by default.

7. Enabling an Interface to Learn ND Entries via DAD NS Packets

(1) Enter the privileged EXEC mode.

enable

(2) Enter the global configuration mode.

configure terminal

(3) Enter the interface configuration mode.

interface *interface-type interface-number*

(4) Enable the interface to learn ND entries via DAD NS packets.

ipv6 nd dad learning enable

The function of learning ND entries via DAD NS packets by an interface is disabled by default.

1.5.5 Configuring Address Resolution

1. Overview

When a node attempts to communicate with another, the node has to obtain the link-layer address of the peer by sending an NS packet to the peer. In this packet, the destination address is the solicited-node multicast address corresponding to the IPv6 address of the destination node. This packet also contains the link-layer address of the source node. After receiving this NS packet, the destination node replies with an NA packet, in which the destination address is the source address of the NS packet and the content is the link-layer address of the solicited node. After receiving the NA packet, the source node can communicate with the destination node.

2. Restrictions and Guidelines

If the device is configured to actively send an NS packet to a specific VLAN in a super VLAN, then it will only send the NS packet to the VLAN specified in the super VLAN.

If an authentication-exempt VLAN is configured but the authentication-exempt VLAN is not in the VLAN list, NS packets will not be actively broadcast to the authentication-exempt VLAN.

3. Procedure

(1) Enter the privileged EXEC mode.

enable

(2) Enter the global configuration mode.

configure terminal

(3) (Optional) Configures a device not to forcibly use the link-local address as the source address when NS packets are sent.

no ipv6 ns-linklocal-src

The link-local address is always used as the source address for sending NS packets by default.

(4) (Optional) Configure the device to actively send NS packets to specific VLANs in the super VLAN.

ipv6 nd resolve vlan { *vlan-list* | **none** }

A device does not actively send NS packets to a specific VLAN in a super VLAN by default.

(5) (Optional) Enter the interface configuration mode.

interface *interface-type interface-number*

(6) Configure the NS packet retransmission interval.

ipv6 nd ns-interval *interval*

The default NS packet retransmission interval on an interface is 0 (indicating unspecified) when the interval is to be filled in the RA packets, and 1,000 ms when the interval is used to control the interval for sending NS packets.

1.5.6 Configuring NUD

1. Overview

If the duration in which a neighbor is considered reachable or the neighbor is in stale state has elapsed but an IPv6 unicast packet needs to be sent to the neighbor, the device performs NUD.

2. Restrictions and Guidelines

A shorter duration in which a neighbor is considered reachable or in stale state indicates that the device detects unreachable neighbors more quickly but more network bandwidth and device resources will be consumed. Therefore, you are not advised to set the duration to a very small value.

3. Procedure

(1) Enter the privileged EXEC mode.

enable

(2) Enter the global configuration mode.

configure terminal

(3) Enter the interface configuration mode.

interface *interface-type interface-number*

(4) Configure the duration in which a neighbor is considered reachable.

pv6 nd reachable-time *time*

By default, the value in an RA packet is 0 (indicating unspecified), and the duration in which a neighbor is considered reachable in neighbor discovery is 30,000 ms.

(5) (Optional) Configure the duration in which a neighbor keeps in stale state.

ipv6 nd stale-time *time*

The default duration of a neighbor in stale state is 1 hour.

1.5.7 Configuring DAD

1. Overview

You need to enable DAD before configuring an IPv6 address for an interface. At this moment, the address is in tentative state. If no address conflict is detected by DAD, this address can be correctly used. If an address conflict is detected and the interface ID of this address is an EUI-64 ID, duplicate link-layer addresses exist on this link. In this case, the system automatically disables this interface to prevent IPv6-related operations on this interface. At the time, you must configure a new address for the interface and disable and then enable the interface to start DAD again.

The software will retry DAD on a conflicting address according to the configured number of NS packets to be sent consecutively and interval. If no address conflict is detected, this address can be normally used.

2. Procedure

(1) Enter the privileged EXEC mode.

enable

(2) Enter the global configuration mode.

configure terminal

(3) Enter the interface configuration mode.

interface *interface-type interface-number*

(4) Configure the number of NS packets to be sent consecutively during DAD.

ipv6 nd dad attempts *attempts*

The default number of NS packets to be sent consecutively during DAD is 1.

(5) Configure the DAD interval.

ipv6 nd dad retry *retry*

The default DAD interval is 1s.

1.5.8 Configuring Redirection

1. Overview

If a router receiving an IPv6 packet finds a better next hop, it sends an ICMP Redirect packet to inform the host of the better next hop. The host will directly send the IPv6 packet to the better next hop next time. ICMPv6 redirection has the same function as ICMP Redirect packets of IPv4.

2. Procedure

(1) Enter the privileged EXEC mode.

enable

(2) Enter the global configuration mode.

configure terminal

(3) Enter the interface configuration mode.

interface *interface-type interface-number*

(4) Enable the IPv6 redirection function on the interface.

ipv6 redirects

The IPv6 redirection function is enabled on an interface by default.

1.5.9 Configuring Parameters Related to RA Packets

1. Overview

An RA packet is a reply to an RS packet and advertises prefix options and certain flag bits.

2. Procedure

(1) Enter the privileged EXEC mode.

enable

(2) Enter the global configuration mode.

configure terminal

(3) Enter the interface configuration mode.

interface *interface-type interface-number*

(4) Enable the sending of RA packets on this interface.

no ipv6 nd suppress-ra

No RA packets are sent by an IPv6 interface by default.

(5) (Optional) Configure the address prefix to be advertised in an RA packet.

ipv6 nd prefix { *ipv6-prefix/prefix-length* | **default** } [[*valid-lifetime* { **infinite** | *preferred-lifetime* }] | [**at** *valid-date preferred-date*] | [**infinite** { **infinite** | *preferred-lifetime* }]] [**no-advertise**] | [[**off-link**] [**no-autoconfig**] | [**pool** *pool-name*] | [**preference** { *high* | *medium* | *low* }] [**proxy**]]

By default, the prefix in an RA packet to be sent by an interface is the prefix configured using the **ipv6 address** command on the interface.

(6) (Optional) Configure the hop limit for RA packets to be sent by the interface.

pv6 nd ra-hoplimit *ra-hoplimit*

The default hop limit of RA packets is 64.

(7) (Optional) Configure the interval for sending RA packets on an interface.

ipv6 nd ra-interval { *interval* | **min-max** *min-interval max-interval* }

The default interval is 200s. The actual interval for sending packets will fluctuate around 200s by ±20%.

(8) (Optional) Configure the MTU for RA packets to be sent by the interface.

ipv6 nd ra-mtu *ra-mtu*

The default MTU of RA packets is the IPv6 MTU value of a network interface.

(9) Configure related parameters of RA packets. All the configuration steps below are optional. Select the configuration steps as required.

- Configure the **Managed address configuration** flag bit in an RA packet.

  ipv6 nd managed-config-flag

  The **Managed address configuration** flag bit in an RA packet is not configured by default.

  The settings of the flag bit determine whether a host receiving this RA packet obtains an address through stateful auto-configuration. If this flag bit is configured, an address will be obtained through stateful address auto-configuration. Otherwise, an address will not be obtained through stateful address auto-configuration.

- Configure the **Other stateful configuration** flag bit in an RA packet.

  ipv6 nd other-config-flag

  The **Other stateful configuration** flag bit in an RA packet is not configured by default.

  When this flag bit is set, the flag bit in an RA packet sent by the device is set to 1. A host receiving this flag bit uses DHCPv6 to obtain other information except the IPv6 address for auto-configuration.

  When **Managed address configuration** is set, **Other stateful configuration** is also set by default.

- Configure the address of the DNS recursive resolution server contained in an RA packet. Run the following commands in turn.

  no ipv6 nd ra dns server suppress

  The RDNSS option is not carried in an RA packet by default.

  ipv6 nd ra dns server *ipv6-address* { *valid-lifetime* | **infinite** } **sequence** *number*

  The address of the DNS recursive resolution server in an RA packet is not configured by default.

- Configure the DNS suffix in an RA packet. Run the following commands in turn.

  no ipv6 nd ra dns search-list suppress

  The DNSSL option is not carried in an RA packet by default.

  ipv6 nd ra dns search-list *ipv6-domain-name* { *valid-lifetime* | **infinite** } **sequence** *number*

  The DNS suffix contained in an RA packet is not configured by default.

- Configure the router lifetime in an RA packet.

  ipv6 nd ra-lifetime *lifetime*

  The default router lifetime in an RA packet is 1,800 seconds.

- Configure the URL address of an RA packet.

  ipv6 nd ra-url [*ra-url*]

  The URL address of an RA packet is not configured by default.

- Configure the URL option type value in an RA packet.

  ipv6 nd ra-url *type*

  The URL option type value in an RA packet is not configured by default.

1.5.10 Configuring ND Logging

1. Overview

After ND logging is configured, system logs about ND packets received and sent by the device will be printed.

2. Procedure

(1) Enter the privileged EXEC mode.

enable

(2) Enter the global configuration mode.

configure terminal

(3) Enable ND logging.

ipv6 nd log enable

ND logging is disabled by default.

(4) Configure ND logging rate.

ipv6 nd log rate *rate*

20 ND logs are printed per minute by default.

1.5.11 Configuring the Interval for ND Packet Rate Statistics Collection

(1) Enter the privileged EXEC mode.

enable

(2) Enter the global configuration mode.

configure terminal

(3) Configure the interval for ND packet rate statistics collection.

ipv6 nd packet rate-statistics interval *interval*

The ND packet rate statistics collection is disabled by default.

1.5.12 Configuring Local ND Proxy

1. Overview

After local ND proxy is enabled on an interface, when receiving NS messages that request addresses of other hosts, the device replies with NA messages containing its MAC address.

If L2 access isolation or inter-subnet isolation (such as sub VLANs) is configured, after local ND proxy is enabled on the gateway, the gateway serves as a proxy to process NS packets from the downlink users and replies with NA packets containing the gateway's MAC address. Thus, the traffic of communication among these users is forwarded by the gateway at L3.

2. Procedure

(1) Enter the privileged EXEC mode.

enable

(2) Enter the global configuration mode.

configure terminal

(3) Enter the interface configuration mode.

interface *interface-type interface-number*

(4) Enable local ND proxy.

local-proxy-nd enable [**force**]

Local ND proxy is disabled on an interface by default.

1.5.13 Configuring the ND Protocol to Allocate Different Prefixes to Different Users

1. Overview

A specific prefix pool can be configured on an interface so that the device assigns different IPv6 prefixes to different users served by the interface.

2. Procedure

(1) Enter the privileged EXEC mode.

enable

(2) Enter the global configuration mode.

configure terminal

(3) Configure a prefix pool for prefix allocation.

ipv6 local pool *pool-name prefix*/*prefix-length assigned-length*

No local prefix pool is configured for DHCPv6 server prefix proxy by default.

(4) Enter the interface configuration mode.

interface *interface-type interface-number*

(5) Configure the name of the prefix pool bound to the interface.

ipv6 nd prefix pool *pool-name*

No prefix pool is bound to an interface by default.

1.5.14 Restraining an Interface from Sending NS Packets to Authenticated VLANs

1. Overview

In gateway authentication mode, all sub VLANs in a super VLAN are authenticated VLANs by default. Users in an authenticated VLAN have to pass authentication to access the network. After authentication, a static ND entry is generated on the device. Therefore, when accessing an authenticated user, the device does not need to send NS packets to the authenticated VLAN. If the device attempts to access users in an authentication-exempt VLAN, it only needs to send ARP requests to the authentication-exempt VLAN.

In gateway authentication mode, the function of restraining from sending NS packets to authenticated VLANs is enabled on the device by default. If the device needs to access non-authenticated users in an authenticated VLAN, disable this function.

2. Restrictions and Guidelines

The configuration is supported only on SVIs and takes effect only in gateway authentication mode.

3. Procedure

(1) Enter the privileged EXEC mode.

enable

(2) Enter the global configuration mode.

configure terminal

(3) Enter the interface configuration mode.

interface *interface-type interface-number*

(4) Restrain the interface from sending NS packets to authentication-exempt VLANs.

ipv6 nd suppress-auth-vlan-ns

NS packets are not sent to authenticated VLANs by default.

1.6 Monitoring

Run the **show** command to check the running status of a configured function to verify the configuration effect.

Run the **clear** command to clear information.

⚠ **Caution**

Running the **clear** command may lose vital information and thus interrupt services.

Table 1-1 IPv6 Monitoring

Command	Purpose
clear ipv6 neighbors [**vrf** *vrf-name*] [**oob**] [*interface-type interface-number*]	Clears the dynamically learned neighbor entries.
show ipv6 address [*interface-type interface-number*]	Displays IPv6 addresses.
show ipv6 general-prefix	Displays the prefix information in a general prefix.
show ipv6 nd [**interface** *interface-type interface-number*] **statistics**	Displays the statistics on IPv6 ND packets.
show ipv6 neighbors [**vrf** *vrf-name*] **statistics** [**all**]	Displays the statistics on IPv6 neighbor tables.
show ipv6 packet statistics [**total** | *interface-type interface-number*]	Displays the statistics on IPv6 packets.
show ipv6 neighbors [**vrf** *vrf-name*] [**verbose**] [*interface-type interface-number*] [*ipv6-address*] [**static**] [**oob**]	Displays neighbor information.
show ipv6 raw-socket [*protocol*]	Displays all IPv6 raw sockets
show ipv6 routers [*interface-type interface-number*]	Displays the neighbor router information and RA packets.
show ipv6 sockets	Displays all IPv6 sockets.
show ipv6 udp [**local-port** *port-number*] [**peer-port** *port-number*]	Displays all IPv6 UDP sockets.
show ipv6 udp statistics	Displays the statistics on IPv6 UDP sockets.

Command	Purpose
show ipv6 interface [[*interface*-type interface-number] [**ra-info**] | **brief** [*interface-type interface-number*]]	Displays the information about IPv6 interfaces.

1.7 Configuration Examples

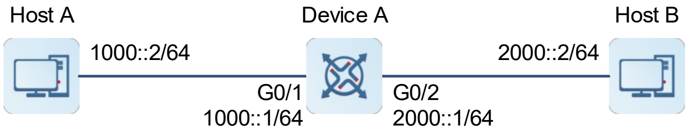
1.7.1 Manually Configuring IPv6 Addresses

1. Requirements

Host A and host B communicate with each other through IPv6 addresses.

2. Topology

Figure 1-1 Topology of IPv6 Addresses



3. Notes

Enable IPv6 on an interface and configure an IPv6 address.

4. Procedure

(1) Configure device A.

Configure an IPv6 address for port GigabitEthernet 0/1 and enable the sending of RA packets on this interface.

```
DeviceA> enable
DeviceA# configure terminal
DeviceA(config)# interface gigabitethernet 0/1
DeviceA(config-if-GigabitEthernet 0/1)# ipv6 enable
DeviceA(config-if-GigabitEthernet 0/1)# ipv6 address 1000::1/64
DeviceA(config-if-GigabitEthernet 0/1)# no ipv6 nd suppress-ra
DeviceA(config-if-GigabitEthernet 0/1)# exit
```

Configure an IPv6 address for port GigabitEthernet 0/2 and enable the sending of RA packets on this interface.

```
DeviceA(config)# interface gigabitethernet 0/2
DeviceA(config-if-GigabitEthernet 0/2)# ipv6 enable
DeviceA(config-if-GigabitEthernet 0/2)# ipv6 address 2000::1/64
DeviceA(config-if-GigabitEthernet 0/1)# no ipv6 nd suppress-ra
DeviceA(config-if-GigabitEthernet 0/2)# exit
```

(2) Configure hosts.

Set the IPv6 address of host A to 1000::2/24.

Set the IPv6 address of host B to 2000::2/24.

5. Verification

Run the **show ipv6 interface** command to check that IPv6 addresses are successfully added to the interfaces.

```
DeviceA# show ipv6 interface

interface GigabitEthernet 0/1 is Up, ifindex: 2, vrf_id 0
  address(es):
    Mac Address: 00:50:56:b0:05:99
    INET6: FE80::250:56FF:FEB0:599 , subnet is FE80::/64
    INET6: 1000::1 , subnet is 1000::/64
  Joined group address(es):
    FF01::1
    FF02::1
    FF02::2
    FF02::1:FF00:0
    FF02::1:FF00:1
    FF02::1:FFB0:599
  MTU is 1500 bytes
  ICMP error messages limited to one every 100 milliseconds
  ICMP redirects are enabled
  ND DAD is enabled, number of DAD attempts: 1
  ND reachable time is 30000 milliseconds
  ND stale time is 3600 seconds
  ND advertised reachable time is 0 milliseconds
  ND retransmit interval is 1000 milliseconds
  ND advertised retransmit interval is 0 milliseconds
  ND router advertisements are sent every 600 seconds<480--720>
  ND router advertisements live for 1800 seconds

interface GigabitEthernet 0/2 is Down, ifindex: 3, vrf_id 0
  address(es):
    Mac Address: 00:50:56:b0:05:9a
    INET6: FE80::250:56FF:FEB0:59A [ TENTATIVE ], subnet is FE80::/64
    INET6: 2000::1 [ TENTATIVE ], subnet is 2000::/64
  Joined group address(es):
  MTU is 1500 bytes
  ICMP error messages limited to one every 100 milliseconds
  ICMP redirects are enabled
  ND DAD is enabled, number of DAD attempts: 1
  ND reachable time is 30000 milliseconds
  ND stale time is 3600 seconds
  ND advertised reachable time is 0 milliseconds
  ND retransmit interval is 1000 milliseconds
  ND advertised retransmit interval is 0 milliseconds
  ND router advertisements are sent every 600 seconds<480--720>
  ND router advertisements live for 1800 seconds
```

On host A, run the **ping** command to test the interconnectivity with host B.

On host B, run the **ping** command to test the interconnectivity with host A

6. Configuration Files

Device A configuration file

```
hostname DeviceA
!
interface gigabitethernet 0/1
ipv6 enable
ipv6 address 1000::1/64
no ipv6 nd suppress-ra
!
interface gigabitethernet 0/2
ipv6 enable
ipv6 address 2000::1/64
no ipv6 nd suppress-ra
!
```

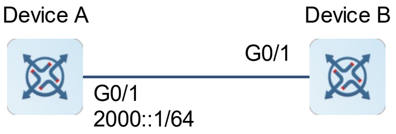
1.7.2 Enabling Stateless IPv6 Address Auto-configuration

1. Requirements

Device A is connected to device B through the GigabitEthernet 0/1 interface which is configured with an IPv6 address. Stateless address auto-configuration needs to be configured on device B to obtain an IPv6 address so that the two devices can communicate with each other through the IPv6 addresses.

2. Topology

Figure 1-1 Topology of Stateless IPv6 Address Auto-configuration



3. Notes

- Configure an IPv6 address on device A.
- Enable stateless address auto-configuration on device B to obtain an IPv6 address.

4. Procedure

(1) Configure device A.

Configure an IPv6 address for port GigabitEthernet 0/1 and enable the sending of RA packets on this interface.

```
DeviceA> enable
DeviceA# configure terminal
```

```
DeviceA(config)# interface gigabitethernet 0/1
DeviceA(config-if-GigabitEthernet 0/1)# ipv6 address 2000::1/64
DeviceA(config-if-GigabitEthernet 0/1)# no ipv6 nd suppress-ra
DeviceA(config-if-GigabitEthernet 0/1)# exit
```

(2) Configure device B.

```
DeviceB> enable
DeviceB# configure terminal
DeviceB(config)# interface gigabitethernet 0/1
DeviceB(config-if-GigabitEthernet 0/1)# ipv6 address autoconfig
```

5. Verification

On device B, run the **show ipv6 interface** command to check that the interface automatically obtains an IPv6 address.

```
DeviceB# show ipv6 interface

interface GigabitEthernet 0/1 is Up, ifindex: 2, vrf_id 0
  address(es):
    Mac Address: 00:50:56:b0:2f:50
    INET6: FE80::250:56FF:FEB0:2F50 , subnet is FE80::/64
    INET6: 2000::250:56FF:FEB0:2F50 [ PRE ], subnet is 2000::/64
      valid lifetime 2591906 sec, preferred lifetime 604706 sec
  Joined group address(es):
    FF01::1
    FF02::1
    FF02::2
    FF02::1:FF00:0
    FF02::1:FFB0:2F50
  MTU is 1500 bytes
  ICMP error messages limited to one every 100 milliseconds
  ICMP redirects are enabled
  ND DAD is enabled, number of DAD attempts: 1
  ND reachable time is 30000 milliseconds
  ND stale time is 3600 seconds
  ND advertised reachable time is 0 milliseconds
  ND retransmit interval is 1000 milliseconds
  ND advertised retransmit interval is 0 milliseconds
  ND router advertisements are sent every 600 seconds<480--720>
  ND router advertisements live for 1800 seconds
  Hosts use stateless autoconfig for addresses.
```

On device A, run the **ping** command to test the interconnectivity with device B.

```
DeviceA# ping ipv6 2000::250:56FF:FEB0:2F50
Sending 5, 100-byte ICMP Echoes to 2000::250:56ff:feb0:2f50, timeout is 2
seconds:
  < press Ctrl+C to break >
!!!!!
```

```
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/13/63 ms.
```

On device B, run the **ping** command to test the interconnectivity with device A.

```
DeviceAB# ping ipv6 2000::1
Sending 5, 100-byte ICMP Echoes to 2000::1, timeout is 2 seconds:
  < press Ctrl+C to break >
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms.
```

6. Configuration Files

- Device A configuration file

```
hostname DeviceA
!
interface gigabitEthernet 0/1
ipv6 enable
ipv6 address 2000::1/64
no ipv6 nd suppress-ra
!
```

- Device B configuration file

```
hostname DeviceB
!
interface gigabitEthernet 0/1
ipv6 address autoconfig
!
```