Technology Brief

IPv6 White Paper.

Table of Contents

1 Ipv6 Overview

1.1 Background

IP version 6 (IPv6) is a new IP protocol designed to replace IP version 4 (IPv4).

IPv4 is the Internet protocol that is predominantly deployed and extensively used throughout the world. It has been more than 20 years since IPv4 was initially defined (by RFC791) in 1981. IPv4 has proven to be robust, easily implemented, and interoperable, and has stood the test of scaling an internetwork to a global utility the size of the Internet today.

However, the initial design did not anticipate the following conditions along with the rocketing development of the Internet:

- The IPv4 address space is insufficient

An IPv4 address consists of 32 bits. Theoretically, the 32-bit address space supports 4.3 billion addresses. However, the number of addresses actually available is less than 4.3 billion due to address allocation. Moreover, the allocation of IPv4 addresses is uneven: The America takes up about half of the global address space, Europe has an insufficient address space and the Asia Pacific region is severely short of the address space (some countries have less than 256 addresses). The IPv4 address space shortage becomes more and more serious along with the development of the Internet.

- The backbone router maintains an excessive large number of route entries

Due to improper planning at the initial development stage of IPv4, many IPv4 address blocks are discontinuously allocated and thus routes cannot effectively converge. To solve this problem, CIDR and IPv4 address reclamation and reallocation are employed to effectively suppress the lineal growth of the global IPv4 BGP routing table. However, currently the global IPv4 BGP routing table size keeps increasing. The table has already more than 170 thousand route entries. There are almost 100 thousand route entries even after route convergence via CIDR. The ever-growing routing table consumes quite much memory and affects the device cost and forwarding efficiency to some extent. Therefore, all device manufacturers are making efforts to upgrade their routers so as to improve the route addressing and forwarding performance.

- Auto-configuration and re-addressing are not easy

Because IPv4 addresses use 32 bits only and the address allocation is not even, IP address reallocation is often needed during network capacity expansion or redeployment. Therefore, auto-configuration and re-addressing are needed to reduce the maintenance workload.

- The IPv4 cannot solve the security issues that are more and more prevalent

The security issue becomes more and more important along with the development of the Internet. Security design was not thoroughly considered during the formulation of the IPv4 protocol. For this reason, the inherent framework does not support end-to-end security. Therefore, the security issue is a factor that drives the emergence of the new IP protocol.

Many solutions emerged to solve the IPv4 address shortage problem. CIDR and NAT are two typical ones.

- CIDR

CIDR is short for Classless Inter-Domain Routing. The IPv4 was initially designed with a hierarchical address structure. The addresses fall into three classes: Class A

(mask length = 8), Class B (mask length = 16) and Class C (mask length = 24). The address utilization rate is low. CIDR supports the address mask of any length. It enables ISPs to allocate the address space on demand and thus improves the address space utilization.

The emergence of CIDR greatly alleviated the address shortage. Nevertheless, the requirements for IP addresses increase along with the emergence of network devices and hosts. CIDR still cannot solve the limited IPv4 address space (32 bits).

- NAT

NAT is another solution for solving the IPv4 address shortage problem. The basic operating mechanism is as follows: Private addresses are used inside the network, and the translation between private addresses and external public network addresses is performed on the NAT device so as to reduce the use of public network addresses.

NAT is also a solution widely applied to solve the address shortage problem, but it has the following shortcomings:

- NAT breaks the end-to-end connection model of IP

With IPv4, only the endpoints handle the connection and the underlying layers do not handle any connection. For this reason, the entire network model is clear and concise. However, when NAT is used, the NAT device need care about the state of every connection and thus the network complexity is increased.

- NAT involves single point failure

Because NAT must handle the translation of addresses and ports, NAT requires the network to keep the states of the connections. In case of failure of the NAT device or the links near the NAT device, the need to keep the state of the connections in NAT makes fast rerouting difficult. For this reason, the reliability of the network is reduced.

- NAT does not support non-NAT-friendly applications

With applications that are not "NAT-friendly," more than just port and address translation is necessary. All the data related to addresses, port numbers or security in these applications must undergo NAT translation so as to enable these applications to normally run. Therefore, Every new deployment of a non-NAT-friendly application will require the upgrading of the NAT device.

- NAT does not support end-to-end security

In NAT, the IP header need be modified and sometimes even application-related data need be modified. The integrity of the IP header is protected by some cryptographic functions. This header cannot be changed between the origin of the packet, which protects the integrity of the header and the final destination, where the integrity of the received packet is checked. Any translation of parts of the headers along the path will break the integrity check. Therefore, the network cannot support end-to-end security when NAT is deployed.

- Network capacity expansion or redeployment is difficult

Different networks may use the same private address space such as 192.168.0.1/24. Therefore, address space conflicts occur when these networks are merged or connected. In such a case, re-addressing or secondary NAT is needed to solve the conflict, but this increases the complexity of network management.

- NAT cannot always solve the address shortage problem

NAT adopts the method of mapping between internal private addresses and external addresses or ports to solve the address shortage problem, but the ratio of internal private addresses to external addresses (ports) mapping must be large to make NAT effective. However, when there are many servers inside, the same protocol cannot be multiplexed on the same port using the NAT external address. For example, two

internal servers using the same port (such as HTTP) cannot use the same external address without changing the port number.

As can be seen from above, the major force that drives the emergence of IPv6 is that the IPv4 address space will be exhausted soon. In addition, IPv6 provides some new features and improvement measures:

- The design is concise and transparent, which improves the implementation efficiency and reduces the complexity.
- IPv6 provides support for the emerging mobile services.
- IPv6 re-ushers in end-to-end security and QoS.

1.2 Advantages of IPv6

- 128-bit address structure, which guarantees a sufficient address space

The availability of an almost unlimited number of IP addresses is the most compelling benefit of implementing IPv6 networks. Compared to IPv4, IPv6 increases the number of address bits by a factor of 4, from 32 bits to 128 bits. The 128 bits provide approximately 4.3 billion × 4.3 billion ×4.3 billion × 4.3 billion addressable nodes, which can satisfy any predictable address space requirement. Theoretically speaking, IPv4 can provide at most 4.3 billion addresses whereas IPv6 can provide at most 4.3 billion × 4.3 billion ×4.3 billion × 4.3 billion addresses.

Figure 1 IPv6 address format

- Hierarchical network architecture, which improves the routing efficiency

An IPv6 address consists of 128 bits, which can provide an address space and network prefix far greater than that of IPv4. Therefore, a network can be hierarchically deployed with IPv6. The same organization can use only one prefix in the network. For ISPs, a greater address space can be obtained. Therefore, ISPs can converge all customers into one prefix and distribute the prefix. With hierarchical convergence, the global routing table contains few address entries and thus the forwarding efficiency is higher. Moreover, the same customer can use different prefixes when using multiple ISPs for access, because the address space is huge. In this way, the convergence of the global routing table is not affected.

- Simplified IPv6 header, which enables higher efficiency and easy expansion

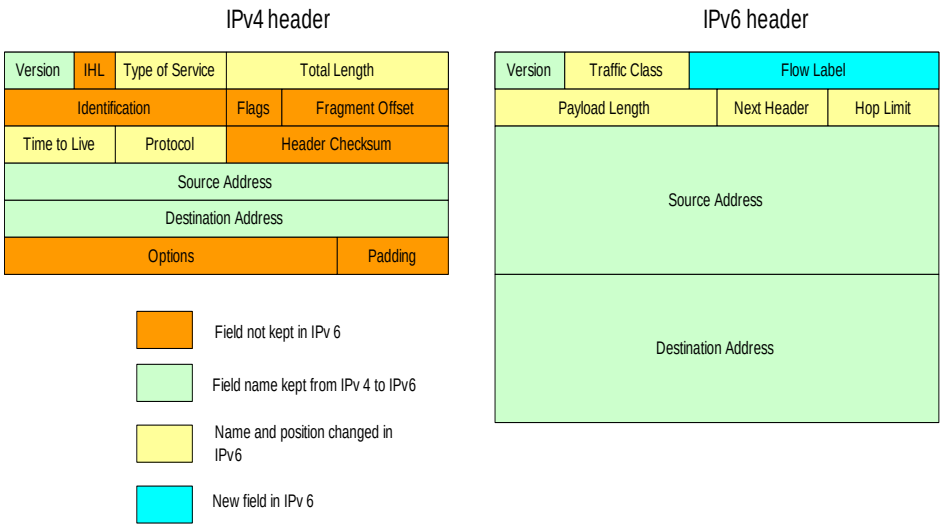

Figure 2 Comparison of IPv4 and IPv6 headers

Fields that are not kept in IPv6:

IHL field

The IHL field in the IPv4 header identifies the length of the IPv4 header. Because the Options field exists in the IPv4 header, the IHL field is mandatory to determine the length of the IPv4 header. However, the IHL field has 4 bits only (the minimum value is 5 in the unit of 4 octets), so the expandability of the options in the header is limited. The IPv6 header is composed of the basic header and the extension headers. The length of the basic header is fixed as 40 octets, so the IHL field is eliminated in the IPv6 header. The Identification field in the IPv4 header is assigned a value by the sender to identify the same group of fragments so as to help fragment reassembly. IPv6 packet fragmentation is implemented through the extension headers. Therefore, the Identification field is no longer needed in the basic header of an IPv6 packet.

Flags field

The Flags field in the IPv4 header identifies whether the packet is a fragment and whether it is the last fragment. IPv6 packet fragmentation is implemented through the extension headers. Therefore, the Flags field is no longer needed in the basic header of an IPv6 packet.

Fragment Offset field

The Fragment Offset field in the IPv4 header identifies the position of the fragment in the original packet before the packet is fragmented. IPv6 packet fragmentation is implemented through the extension headers. Therefore, the Fragment Offset field is no longer needed in the basic header of an IPv6 packet.

Header Checksum field

The Header Checksum field in the IPv4 header is used to check for errors in the IPv4 header. Generally, the link layer in the current networks is highly reliable with a check mechanism and the transmission layer has its own header checksum mechanism. Therefore, the Header Checksum field is excessive to some extent. Moreover, the computation of the Header Checksum field involves TTL and every intermediate router need re-compute the TTL, so the forwarding efficiency is affected. Therefore, the Header Checksum field is eliminated in the IPv6 header (but checksum computation is mandatory in the UDP header).

Options field

The Options field in the IPv4 header is used to support the options. Its length is variable, but cannot exceed the length of the IPv4 header. The expandability of the Options field is limited. In the IPv6 packet, the extension headers implement this function and thus the Options field is no longer needed.

Padding field

In the IPv4 header, the Padding field is used to ensure that the header ends with the 32-bit border to facilitate hardware to access the packet. In the IPv6 packet, the length of the basic header is fixed and thus the Padding field is no longer needed.

New fields in IPv6 :

Flow Label field

The Flow Label field is added in the IPv6 header. The source node can use this field to identify a specific data flow. The flow label is allocated by the source node. A flow can be uniquely identified through the flow label, source address and destination address. There are two great benefits when the flow label instead of the traditional quintuple (source address, destination address, source port, destination port and transmission layer protocol number) is applied:

1) The flow label can associate with any flow. When different types of flows (which can be non-quintuple flows) need be identified, the flow label need not be changed.
2) The Flow Label field in the basic header of the IPv6 packet is visible to the intermediate routers when IPSec is applied. Therefore, the intermediate routers can still perform QoS processing on a specific flow based on the triplet (flow label, source address and destination address) when IPSec is applied in IPv6.

Compared with IPv4, IPv6 removes the fields IHL, Identification, Flags, Fragment Offset, Header Checksum, Options and Padding, and adds the Flow Label field only. Therefore, IPv6 header processing is much simpler than IPv4 header processing and the processing efficiency is higher. In addition, to better support the processing of various options, IPv6 has added the extension header concept. New options can be added without changing the existing structure. Theoretically, the header can be infinitely extended with great flexibility.

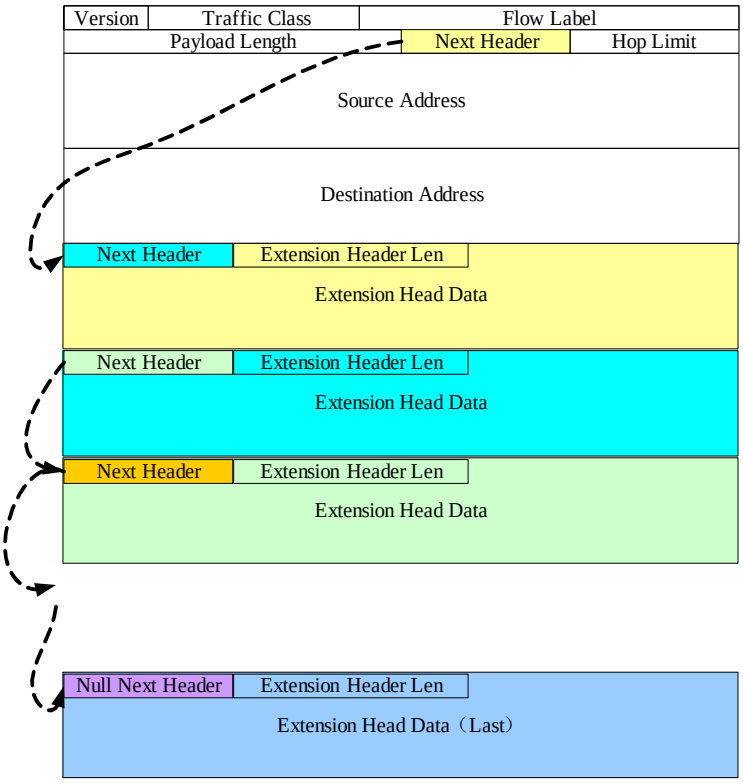

Figure 3 Structure of the IPv6 extension headers

- Supporting auto-configuration and plug-and-play

The IPv6 protocol supports auto address configuration, so as to enable the host to automatically discover the network and obtain an IPv6 address. This greatly improves the manageability of the intranet. Through auto-configuration, users can plug and play user equipment such as mobile phones and wireless terminals without manually configuring or using any dedicated server such as the DHCP server. The routers on the local link transmit the network-related information, such as the prefix of the local link and the default route, in the router advertisement packet. Upon receipt of the packet, the host constructs the host address based on the interface ID of the local interface so as to complete the auto-configuration.

- Supporting end-to-end security

The IPv4 also supports the IP layer security feature (IPSec), but the support is implemented through options. In actual deployment, most nodes do not support this feature. IPSec is a part of the basic definitions in the IPv6 protocol, and all the nodes deployed must be able to support IPSec.

Therefore, it is much easier to support end-to-end security in IPv6. IPv6 supports these security objectives defined for the IP: confidentiality (only the expected receiver can read the data), integrity (the data is not modified during the transmission), and authenticity (the entity sending the data is completely the same as the claimed entity).

- Supporting the mobility feature

The IPv6 protocol stipulates the mandatory support for the mobility feature. Any IPv6 node can use the mobile IP function.

Compared with mobile IPv4, mobile IPv6 can use the neighbor discovery function to directly discover foreign networks and obtain the care-of address, instead of using any foreign agent. In addition, direct communications between a mobile node and the peer node are possible by using the Routing header and the Destination Address

header. This solves the triangle routing and source address filtering issues of mobile IPv4. The mobile communication processing efficiency is thus higher and the processing is transparent to the application layer.

- Better supporting QoS through the new flow label function

The Flow Label field is added in the IPv6 header. The source node can use this field to identify a specific data flow. Both the intermediate routers and the destination node can make special processing based on this Flow Label field, for example, they can process video conference flows and VOIP flows. The IPv6 source node uses the 20-bit Flow Label field in the IPv6 header to identify a flow. When the Flow Label of a packet is 0, this packet does not belong to any flow. A flow is uniquely determined by three fields: Source Address, Destination Address and Flow Label. The Flow Label value set by the source node cannot be changed during the transmission. If an IPv6 node does not support special processing of a flow, it must ignore the Flow Label field when receiving or forwarding the packet. The source node must guarantee that the flow label currently in use is not repeatedly used. A flow label should not be allocated to others within 120 seconds after the flow is terminated. The source node can set the interval to a value more than 120 seconds for various flows. To avoid accidental repeated use of flow labels, the source node must allocate a new flow label according to a certain sequence (such as an ascending order or a pseudo random order) and select the initial value at the system restart. In order to support the processing of specific flows, all the nodes or some of the nodes along the path from the source node to the destination node must record the flow status.

Currently, there is no relevant standard governing the label negotiation between the source node and the other nodes.

2 IPv6 Packet

2.1 IPv6 Basic Header

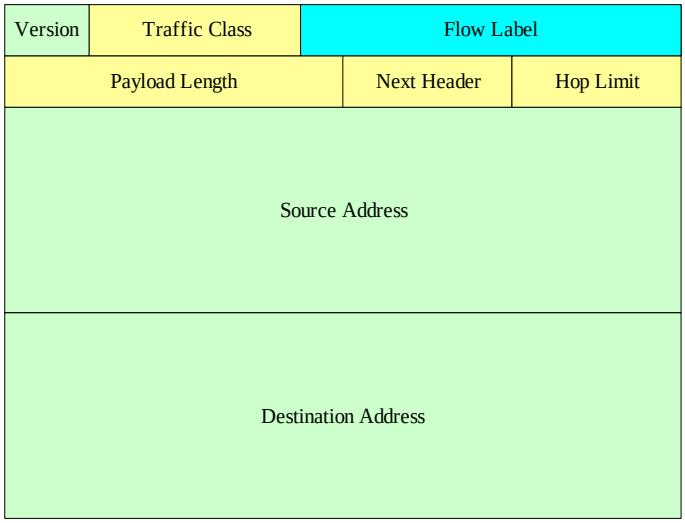

Figure 4 Format of the IPv6 basic header

Version:

This field is 4 bits long. The value 6 indicates that the packet is an IPv6 packet.

Traffic Class:

This field is 8 bits long. It is similar to the TOS field in IPv4.

Flow Label:

This field is 20 bits long. It is a new field in IPv6. The Flow Label field can be used to tag packets of a specific flow to differentiate the packets at the network layer.

The routers in the forwarding path can identify and process a flow based on the flow label. With this label, a router need not check deep into the packet to identify the flow, because this information is available in the IPv6 header. Similarly, the destination node can identify a flow based on the specific flow label. In addition, QoS processing can still be performed based on the flow label even after IPSec is applied, because the flow label is carried in the header.

Payload Length:

This field is 16 bits long. It indicates the IPv6 payload length in octets, that is, the length of the section behind the basic header of the IPv6 packet, including all the extension headers.

Next Header:

This field is 8 bits long. It identifies the type of the header next to the current header (the basic header or an extension header). The type defined in this field is the same as the protocol field value in IPv4. IPv6 defines extension headers that form a chain of headers linked together by the Next Header field, contained in the basic header or each extension header. This mechanism provides more efficiency in the processing of extension headers. The intermediate routers process only the extension headers that need be processed. This improves the forwarding efficiency.

Hop Limit:

This field is 8 bits long. It is similar to the TTL field in IPv4. Every node decrements the value of this field by 1 before forwarding the packet. If the value of this field is already 0, the node simply discards the packet.

Source Address:

This field is 128 bits long. It indicates the source address of the packet.

Destination Address:

This field is 128 bits long. It indicates the destination address of the packet.

2.1.1 IPv6 Extension Headers

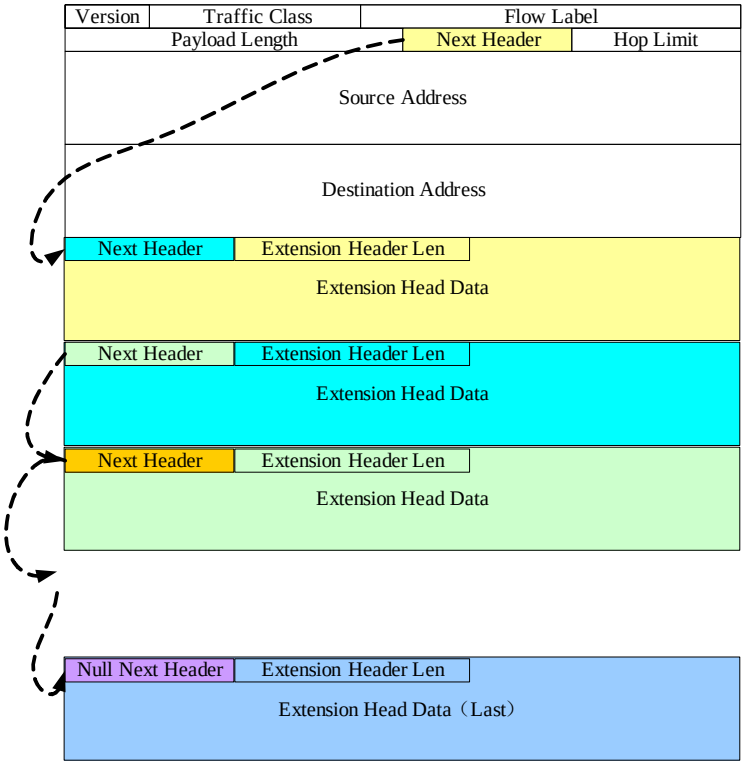

Figure 5 Format of the extension headers in an IPv6 packet

The Options field of an IPv4 packet is substituted by a chain of extension headers in an IPv6 packet. The basic header of an IPv6 packet can be followed by zero or multiple extension headers.

When multiple extension headers are used in the same packet, the order of the headers should be as follows:

- Hop-by-Hop Options header

The value is 0 (defined in the basic header of an IPv6 packet). All the nodes in the forwarding path process this options header. Currently, the Hop-by-Hop Options header is used in the Router Alert (RSVP and MLDv1) and Jumbo frame processing. A Router Alert need be notified to all the nodes in the forwarding path, so the Hop-by-Hop Options header must be used. A Jumbo frame is a packet whose length exceeds 65535 octets. During the transmission of such a packet, all the nodes in the forwarding path must be able to normally process it. Therefore, the Hop-by-Hop Options header must also be used in the processing of a Jumbo frame.

- Destination Options header

The value is 60. This header may appear at two positions only:

- Before the Routing header

In this case, the Destination Options header is processed by the destination node and the nodes specified in the Routing header.

Before the Upper-Layer header (behind any ESP option)

In this case, the Destination Options header can only be processed by the destination node. The Mobile IPv6 uses the Destination Options header. A type of Destination

Options header is added in Mobile IPv6. It is called the “Home Address Option”. The Home Address Option is carried in the Destination Options header. It is used to notify the home address of a mobile node to the receiving node when the mobile node leaves its home. Upon receipt of the packet that carries the Home Address Option, the receiving node interchanges the source address (home address of the mobile node) in the Home Address Option with the source address (care-of address of the mobile node) in the packet, so that the upper-layer protocol always considers that it is communicating with the home address of the mobile node. In this way, the mobile roaming function is implemented.

- Routing header

The value is 43. It is used in the Source Route Option and Mobile IPv6.

- Fragment header

The value is 44. This option is used for packet fragmentation when the size of the packet sent by the source node exceeds the Path MTU (the MTU of the transmission path between the source node and the destination node).

- Authentication header (AH header)

The value is 51. It is used for IPSec and provides packet authentication and integrity check. Its definition is the same as that in IPv4.

- Encapsulating Security Payload (ESP) header

The value is 50. It is used for IPSec and provides packet authentication, integrity check and encryption. Its definition is the same as that in IPv4.

- Upper-Layer header

It is a header of the upper-layer protocol such as TCP/UDP/ICMP.

The Destination Options header can appear at most twice (once before the Routing header and the other before the Upper-Layer header). The rest options header can appear at most once.

However, IPv6 nodes can process the case that an options header appears at any position for any number of times (except for the Hop-by-Hop Options header, which can only appear behind the basic header), so as to guarantee the interoperability.

2.1.2 IPv6 ICMP Packet

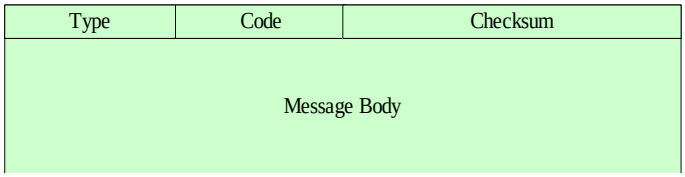

Figure 6 ICMP packet

The function of ICMPv6 is similar to that of ICMPv4. ICMPv6 is used to report the errors that occur when IPv6 nodes process a packet and to complete the functions of other layers such as the diagnosis function (ICMPv6 “ping”). ICMPv6 is a part of IPv6. Every IPv6 node must implement ICMPv6.

ICMPv6 packets are divided into the following two categories:

- Error message
  - ICMP destination unreachable message
  - The message indicating that the packet length exceeds the limited length (this message is used for the path MTU discovery protocol)

- Transmission timeout message (equivalent to the ICMP message triggered when IPv4 TTL is equal to 0)
- Parameter error message
- Information message
- Echo request message
- Echo reply message

2.2 IPv6 Addressing Structure

I. IPv6 address

IPv6 uses 16-bit hexadecimal number fields separated by colons (:) to represent the 128-bit addressing format. The hexadecimal numbers are not case-sensitive. Here is an example of a valid IPv6 address: FEDC:BA98:7654:3210:FEDC:BA98:7654:3210.

When several consecutive intermediate bits are 0, the address can be simplified, that is, a pair of colons (::) represents successive fields of 0. However, the pair of colons is allowed only once in a valid IPv6 address, as shown below:

1080:0:0:0:8:800:200C:417A	Equivalent to 1080::8:800:200C:417A
FF01:0:0:0:0:0:0:101	Equivalent to FF01::101
0:0:0:0:0:0:0:1	Equivalent to ::1
0:0:0:0:0:0:0:0	Equivalent to ::

II. IPv6 address prefix

Just like IPv4, the subnet prefix of IPv6 is associated with the link. Multiple subnet prefixes can be assigned to one link. The IPv6 address prefix is expressed as follows:

ipv6-address/prefix-length

Where:

ipv6-address

128-bit address in hexadecimal system.

prefix-length

Address prefix length in decimal system.

2.3 Types of IPv6 Addresses

Multiple types of IPv6 addresses are defined in RFC2373:

Allocation	Prefix (binary)	Fraction of Address Space
Reserved	0000 0000	1/256
Unassigned	0000 0001	1/256

Reserved for NSAP Allocation	0000 001	1/128
Reserved for IPX Allocation	0000 010	1/128
Unassigned	0000 011	1/128
Unassigned	0000 1	1/32
Unassigned	0001	1/16
Aggregatable Global Unicast Addresses	001	1/8
Unassigned	010	1/8
Unassigned	011	1/8
Unassigned	100	1/8
Unassigned	101	1/8
Unassigned	110	1/8
Unassigned	1110	1/16
Unassigned	1111 0	1/32
Unassigned	1111 10	1/64
Unassigned	1111 110	1/128
Unassigned	1111 1110 0	1/512
Link-Local Unicast Addresses	1111 1110 10	1/1024
Site-Local Unicast Addresses	1111 1110 11	1/1024
Multicast Addresses	1111 1111	1/256

Note:

1) “Unspecified address” (all 0s), loopback address (::1) and the IPv6 addresses with an embedded IPv4 address (See Section 2.3.4) are allocated from the format prefix 0000 0000.
2) Except for the multicast address (format prefix 1111 1111), all the addresses from 001 to 111 in the format prefix must have a 64-bit interface ID in EUI-64 format.

IPv6 address are divided into unicast addresses, anycast addresses and multicast addresses. Compared with IPv4, IPv6 cancels broadcast addresses, replaces broadcast addresses by multicast addresses and adds anycast addresses.

2.3.1 IPv6 Unicast Address

An IPv6 unicast address identifies an interface. Since every interface belongs to a node, the unicast address on any interface of every node can identify this node. The packets destined to a unicast address are received by the interface identified by this unicast address. On an interface, there should be at least one link local unicast

address (LLUA). There can also be other IPv6 addresses of any type or range, such as unicast, anycast or multicast addresses on the interface.

All the IPv6 addresses whose format prefix is not the multicast format prefix (1111 1111) are IPv6 unicast addresses (the format of an anycast address is the same as that of an IPv6 unicast address). Just like IPv5 unicast addresses, IPv6 unicast addresses can converge. Currently, multiple IPv6 unicast address formats are defined. They include aggregatable global unicast addresses, NSAP addresses, IPX layer addresses, site-local addresses, link-local addresses, and the host addresses with IPv4 ability (IPv6 addresses with an embedded IPv4 address). Of them, aggregatable global unicast addresses, site-local addresses and link-local addresses are widely applied.

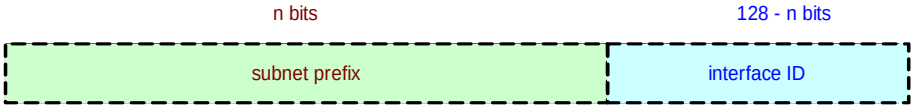

Figure 7 IPv6 unicast address

An IPv6 unicast address consists of the subnet prefix and the interface ID. The subnet prefix is allocated by the Internet Assigned Numbers Authority (IANA), Internet Service Providers (ISPs) and other various organizations.

The interface ID is defined with 64 bits for the time being. It can be generated by the local link ID or the random algorithm so as to guarantee its uniqueness.

2.3.2 IPv6 Anycast Address

The format of an IPv6 anycast address is the same as that of an IPv6 unicast address. An IPv6 anycast address identifies a group of interfaces. In general, these interfaces belong to different nodes. The packets destined to an anycast address are sent to the closest interface among the group of interfaces (the routing protocol judges which interface is the closest).

One of the purposes of IPv6 anycast addresses is to identify a group of routers that belong to the same Internet service provider. These addresses can represent the intermediate forwarding routers in the IPv6 Routing header, so that the packet can be forwarded by a specific group of routers. Another purpose is to identify a group of routers in a specific subnet. The packet need only be received by one of the routers.

Some anycast addresses are already defined:

- Subnet router anycast address

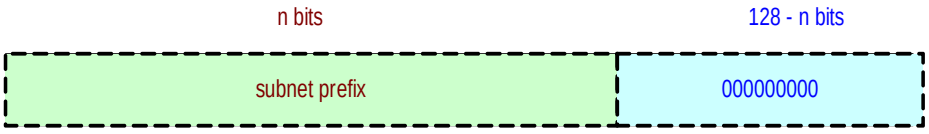

Figure 8 Subnet router anycast address

In a subnet router anycast address, the “subnet prefix” field identifies a specific link. The packets destined to a subnet router anycast address will be sent to a router in the subnet. All routers must support subnet router anycast addresses.

A subnet router anycast address is used for the communication between the node and a router (whichever router) in the remote subnet. For example, it is used when a mobile node need communicate with one of the mobile agents in the home subnet of the mobile node.

2.3.3 IPv6 Multicast Address

I. Format of IPv6 multicast addresses

An IPv6 multicast address identifies a group of interfaces. In general, these interfaces belong to different nodes. One node can belong to zero to multiple multicast groups. The packets destined to a multicast address are received by all the interfaces identified by this multicast address.

Note that the Hop Limit field (equivalent to TTL in IPv4) is not used in IPv6 multicast.

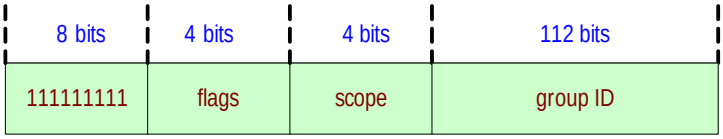

Figure 9 IPv6 multicast address

Where:

11111111

8 bits long. It identifies the address as a multicast address.

flags

4 bits long. The flag field is defined as follows:

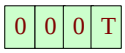

Figure 10 Format of the flag field in an IPv6 multicast address

The three high-order bits are reserved and must be set to 0.

T = 0 indicates a permanently-assigned ("well-known") multicast address, assigned by the global internet numbering authority.

T = 1 indicates a non-permanently-assigned ("transient") multicast address.

scope

4 bits long. It identifies the application scope of this multicast group. It may assume one of the following values:

0 reserved

1 node-local scope //Effective in the local node scope only

2 link-local scope //Effective in the local link scope only

3 (unassigned)

4 (unassigned)

5 site-local scope //Effective in a network only

6 (unassigned)

7 (unassigned)

8 organization-local scope //Effective in the organization only

9 (unassigned)

A (unassigned)

B (unassigned)

C (unassigned)

D (unassigned)

E global scope //Globally effective

F reserved

group ID

It identifies the multicast group. The multicast group may be permanent or transient. The scope is defined by the scope field.

II. IPv6 permanently-assigned multicast address

At present, there are the following permanently-assigned ("well-known") multicast address groups:

- Reserved multicast addresses

FF00::---FF0F:: (total 16 addresses)

- All-nodes multicast address

FF01:0:0:0:0:0:0:1 (node local)

FF02:0:0:0:0:0:0:1 (link local)

- All-routers multicast address

FF01:0:0:0:0:0:0:2 (node local)

FF02:0:0:0:0:0:0:2 (link local)

FF05:0:0:0:0:0:0:2 (site local)

- Solicited-node multicast address

FF02:0:0:0:0:1:FFXX:XXXX

The solicited-node multicast address comes from the unicast or anycast address of the solicited node, that is, the prefix FF02:0:0:0:0:1:FF00::/104 concatenated with the 24 low-order bits of the unicast or anycast address of the solicited node. For example, the solicited-node multicast address corresponding to the IPv6 address 4037::01:800:200E:8C6C is FF02::1:FF0E:8C6C.

This address is used by the IPv6 neighbor discovery protocol and carried in the neighbor solicitation message. Since only the nodes with the same 24 last bits in the unicast address will receive the packets destined to this address, the communication traffic is reduced (as compared with IPv4 ARP).

2.3.4 IPv6 Addresses with an Embedded IPv4 Address

I. IPv4-compatible IPv6 address

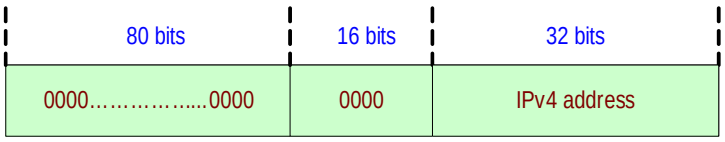

Figure 11 IPv4-compatible IPv6 address

The low-order 32 bits of such an address carry an IPv4 address. The first 96 bits are all 0s. This kind of address is used in an automatic tunneling technique. The packets destined to such an address will be automatically encapsulated in an IPv4 tunnel (the endpoint of the tunnel is the IPv4 address in the IPv6 packet). Because this technique cannot solve the address exhaustion problem, it is gradually eliminated.

II. IPv4-mapped IPv6 address

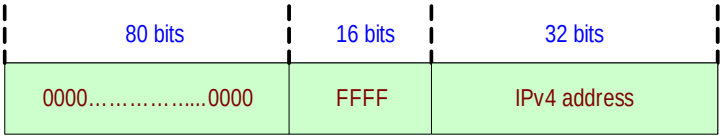

Figure 12 IPv4-mapped IPv6 address

The first 80 bits of such an address are all 0s. The intermediate 16 bits are all 1s. The last 32 bits form an IPv4 address. This kind of address is used to represent an IPv4-only node with an IPv6 address. On an IPv6 node supporting dual stacks, the packet actually sent is an IPv4 packet (the destination address is the IPv4 address in the IPv4-mapped IPv6 address) when the IPv6 application sends a packet destined to such an address.

2.3.5 Special IPv6 Addresses

The following special IPv6 addresses are defined in IPv6:

I. Unspecified IPv6 address

The address format is “0::0”. The unspecified IPv6 address cannot be allocated to any interface. A node with the unspecified IPv6 address means that this node does not have any IPv6 address. For example, if a node does not have any IPv6 address upon startup, it fills the source address in the packet with all 0s to indicate that the node does not have any IP address before it sends the packet. The IPv6 unspecified address should not be used as destination addresses in IPv6 packets or the IPv6 Routing header.

II. IPv6 loopback address

The address format is “::1”. This address is similar to 127.0.0.1 in IPv4. In general, this address is for use by the node itself but cannot be allocated to any physical interface. The IPv6 loopback address cannot be used as the source address. The packets using the IPv6 loopback address as the destination address cannot be sent to the source node and cannot be forwarded by IPv6 routers.

2.3.6 IPv6 Addresses that Must Be Supported by Nodes and Routers

I. IPv6 addresses that a node must support

An IPv6 node requires the following IPv6 addresses for proper operation:

- Link-local addresses for each interface
- Assigned unicast address(es)
- Loopback address
- All-nodes multicast address
- Solicited-node multicast address for each of its assigned unicast or anycast addresses

- Multicast addresses of all other groups to which the host belongs

II. IPv6 addresses that a router must support

An IPv6 router requires the following IPv6 addresses for proper operation:

- All the required node addresses (see subsection 1 in Section 2.3.5)
- Subnet router anycast addresses for the interfaces configured to act as forwarding interfaces
- Anycast addresses configured on all the other routers
- All-routers multicast address
- Multicast addresses of all other groups to which the router belongs

2.4 IPv6 Address Allocation

2.4.1 Global Unicast Address Allocation

The Internet Assigned Numbers Authority (IANA) is responsible for allocating IPv6 addresses. Currently, the IANA allocates 2001::/16 from the entire aggregatable global unicast address space (the format prefix is 001) to registries.

RFC2450 describes the recommended address allocation policy.

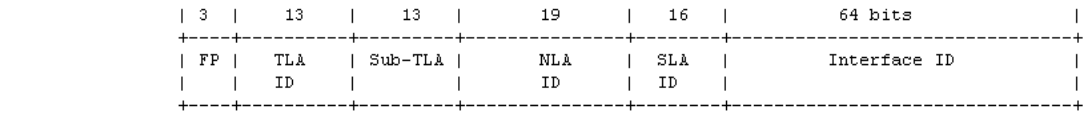

Figure 13 Recommended allocation of aggregatable global unicast addresses

Where:

FP

Format Prefix. It is always 001 for aggregatable global unicast addresses.

TLA ID

Top Level Aggregation Identifier. This field is allocated by the IANA to the specified registries.

Sub-TLA

Sub Top Level Aggregation Identifier. This field is allocated by the registries of the IANA to the organizations that meet certain conditions. These organizations are usually the ISPs with a certain scale.

NLA ID

Next Level Aggregation Identifier. This field is allocated by the registries or the organizations that has a Sub-TLA.

SLA ID

Site Level Aggregation Identifier. This field is usually used inside an organization or enterprise for subnet division.

From IANA, each registry gets a /23 prefix within the 2001::/16 space, as follows:

2001:0200::/23 to 2001:0C00::/23

Allocated to the Asia Pacific Network Information Centre (APNIC).

2001:0400::/23

Allocated to American Registry for Internet Numbers (ARIN) for use in the Americas.

2001:0600::/23 to 2001:0800::/23

Allocated to Reseaux IP Europeens ─ Network Coordination Center (RIPE NCC) for use in Europe and the Middle East.

The registries then allocate an initial /32 prefix to the IPv6 ISPs and the ISPs allocate a /48 prefix (out of the /32) to each customer. The address space of the prefix /48 can be further divided into the subnets of the prefix /64. Therefore, every customer can have at most 65535 subnets.

To conquer the shortcomings of the initial unreasonable IPv4 address allocation scheme, an ISP must satisfy the following conditions so as to receive a /32 prefix address block:

- Exterior routing protocols are deployed.
- The ISP is connected with at least three other ISPs.
- The ISP has at least 40 customers, or the ISP demonstrates a clear intent to provide an IPv6 service within 12 months.

2.4.2 6BONE Network Address Allocation

The 6BONE is a worldwide IPv6 test network. It uses the network prefix 3ffe:0000::/16. Every pseudo TLA (pTLA) receives a /28 prefix. This prefix is inside the 3ffe:0800::/28 range and allows for a maximum of 2048 pTLAs. An end site receives a /48 prefix from its upstream provider and a LAN within a site is assigned a /64 prefix from that site prefix.

The 6BONE network hierarchically allocates addresses. The address space is defined by the IANA. The allocation policy is defined in RFC2921 “6BONE pTLA and pNLA Formats (pTLA)”.

3 Basic Functions of IPv6

The basic functions of IPv6 include IPv6 neighbor discovery protocol (neighbor discovery, router discovery, stateless address auto-configuration and redirection), IPv6 path MTU discovery and IPv6 domain name resolution. Of them, router discovery and stateless address auto-configuration are two new functions of IPv6. The neighbor discovery function is similar to IPv4 ARP but with improvements and enhancements.

3.1 IPv6 Neighbor Discovery Protocol

The IPv6 neighbor discovery process uses IPv6 ICMP (ICMPv6) messages and solicited-node multicast addresses to determine the link-layer address of a neighbor on the same network (local link), verify the reachability of a neighbor, and keep track of neighboring routers. Every IPv6 node is required to join the multicast groups corresponding to its unicast and anycast addresses.

The IPv6 neighbor discovery protocol completes these functions: neighbor discovery, router discovery, IPv6 stateless address auto-configuration and redirection.

3.1.1 Neighbor Discovery

The neighbor discovery function is similar to the ARP function in IPv4.

The neighbor discovery function is implemented by neighbor solicitation and neighbor advertisement messages:

- Neighbor solicitation

Neighbor solicitation messages are sent on the local link when a node wants to determine the link-layer address of another node on the same local link. This function is similar to the ARP in IPv4, except that it uses a multicast address instead of a broadcast address. A node can receive this packet only when the last 24 bits of its IPv6 address are the same as this multicast address. In this way, the possibility of broadcast storm is reduced.

The source node takes the right-most 24 bits of the IPv6 address of the destination node and sends a neighbor solicitation message, which has a value of 135 in the Type field of the ICMP header, to the solicited-node multicast group address on the local link. The destination node will respond with its link-layer address. To send a neighbor solicitation message, the source node must first identify the IPv6 address of the destination node.

Neighbor solicitation message is also used to verify the reachability of a neighbor after the link-layer address of a neighbor is identified.

- Neighbor advertisement

The IPv6 neighbor advertisement message is a response to the IPv6 neighbor solicitation message. After receiving the neighbor solicitation message, the destination node replies by sending a neighbor advertisement message on the local link with a value of 136 in the Type field of the ICMP header. After receiving the neighbor advertisement, the source node and the destination node can communicate.

Neighbor advertisement messages are also sent when there is a change in the link-layer address of a node on a local link.

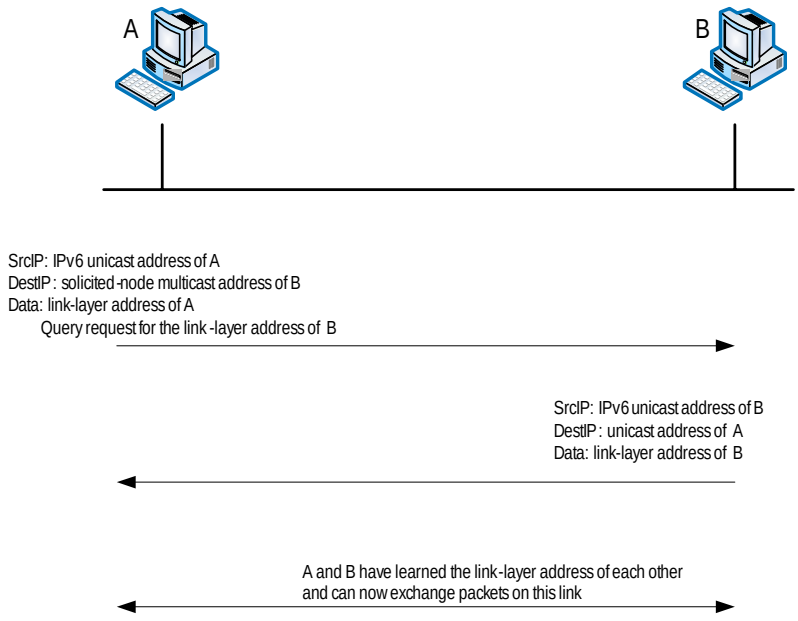

Figure 14 IPv6 neighbor discovery

In the IPv6 neighbor discovery procedure, only one message exchange is needed for two nodes to learn the link-layer address of each other. In contrast, the ARP function in IPv4 involves twice message exchange. Therefore, the IPv6 neighbor discovery efficiency is higher.

In addition, IPv6 neighbor discovery is implemented in the IP layer. Theoretically, IPv6 neighbor discovery can support all kinds of transmission media, which is also a great improvement over IPv4 ARP.

3.1.2 Router Discovery

Router discovery is used to locate neighboring routers and learn the prefix and configuration parameters related to address auto-configuration. The IPv6 router discovery process uses the following messages:

I. Router solicitations

When a host does not have a configured unicast address, for example at system startup, it sends a router solicitation message. A router solicitation is helpful, because it enables the host to automatically configure itself quickly without having to wait for the next scheduled IPv6 router advertisement message.

A router solicitation message has a value of 133 in the Type field of the ICMP packet header.

The source address used in an IPv6 router solicitation message is usually the unspecified IPv6 address (0::0). If the host has a configured unicast address, the unicast address of the interface sending the router solicitation message is used as the source address in the message.

The destination address in the router solicitation messages is the all-routers multicast address (FF02::2) with the link-local scope. When a router advertisement is sent in response to a router solicitation, the destination address used in the router advertisement message is the source address in the router solicitation message.

Note: A router solicitation is sent at boot time and only three times afterward to avoid flooding of router solicitation messages in the absence of a router on the network.

II. Router advertisements

Router advertisement messages are periodically sent out on each configured interface of an IPv6 router. Router advertisements are also sent out in response to router solicitation messages from IPv6 nodes on the local link. Router advertisements are sent to the all-nodes link-local multicast address (FF02 ::1) or the unicast IPv6 address of a node that sent the router solicitation messages.

Router advertisement has a value of 134 in the Type field of the ICMP packet header and contains the following information in the message:

- Whether nodes could use address auto-configuration
- Flags to indicate the type of auto-configuration (stateless or stateful) that can be completed
- One or more local link prefixes that nodes on the local link could use to automatically configure their IPv6 addresses
- Lifetime information for each local link prefix included in the advertisement
- Whether the router sending the advertisement should be used as a default router and, if so, the amount of time (in seconds) the router should be used as a default router
- Additional information for hosts, such as the hop limit and maximum transmission unit (MTU) a host should use in packets that it originates

The IPv6 nodes on the local link receive the router advertisement messages and use the information to get the information about the default router, prefix list and other configuration parameters updated.

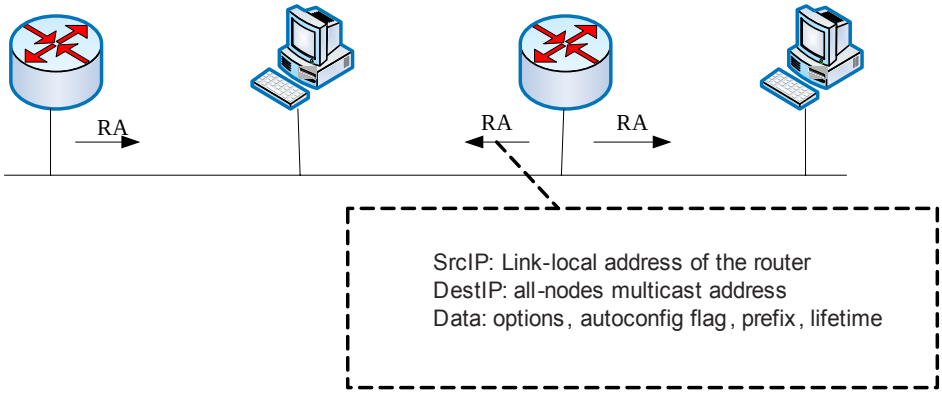

Figure 15 IPv6 router advertisement

3.1.3 IPv6 Stateless Auto-Configuration

Stateless auto-configuration uses the information in the router advertisement messages (and the flag for every prefix) to configure the node. For instance, a router can specify stateful (DHCPv6) configuration or stateless auto-configuration for configuring the IPv6 address of a node.

Upon receipt of a router advertisement message, the node uses the prefix information and local interface ID to automatically configure the IPv6 address. It can also set the default router based on the default router information in the router advertisement message.

Renumbering of IPv6 nodes becomes very easy with the use of stateless address configuration, which reduces the complexity of network redeployment. During renumbering, the router advertisement messages contain both the old prefix and the new prefix. A decrease in the lifetime value of the old prefix alerts the nodes to use the new prefix, while still keeping their current connections intact with the old prefix. During this period of time, nodes have two unicast addresses in use. When the old prefix is no longer usable, the router advertisements will include only the new prefix.

3.1.4 Redirection

As with IPv4, an IPv6 redirect message is sent by a router only to help with the reroute of a packet to a better router. The node receiving the redirect message will then readdress the packet to a better router. Routers send redirect messages only for unicast traffic, only to the originating nodes, and are processed by the nodes.

3.2 IPv6 Path MTU Discovery Protocol

The path MTU discovery protocol is also defined in IPv4, but it is optional. To simplify the packet processing procedure and improve the processing efficiency, IPv6 routers do not handle fragmentation. Fragmentation is done by the originating node or source node of a packet, when necessary. Therefore, the path MTU discovery protocol is mandatory in IPv6. IPv6 uses the path MTU discovery to find the maximum MTU in a path between the source and the destination. The source node starts the path MTU discovery process before actually sending the packets. When the path MTU of every

link along a given data path in an IPv6 network is not large enough to accommodate the size of the packets, the source node fragments the packet and resends it.

The path MTU discovery protocol allows an IPv6 node to dynamically discover and adjust to differences in the MTU size of every link along a given data path. In IPv4, the minimum link MTU size is 68 octets and the recommended minimum is 576 octets. In IPv6, the minimum link MTU is 1280 octets, but the recommended MTU value for IPv6 links is 1500 octets. The maximum packet size supported by the basic IPv6 header is 64000 octets. Larger packets called jumbograms could be handled using a hop-by-hop extension header option.

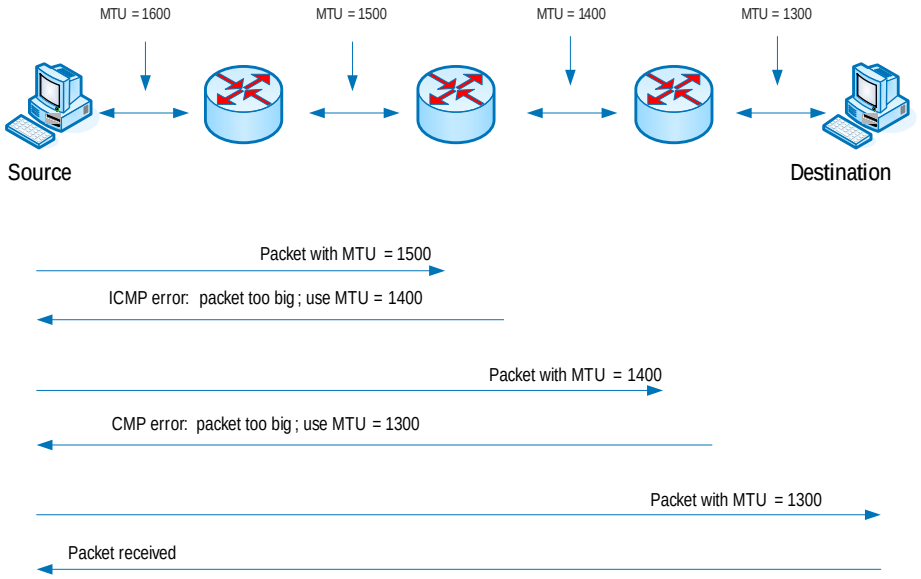

Figure 16 IPv6 path MTU discovery

3.3 IPv6 Domain Name Resolution

In IPv4 DNS, only a 32-bit IPv4 address will be returned for an address query operation. Therefore, the IPv4 DNS does not support IPv6 but must be expanded.

IPv6 introduces new DNS record types for IPv6 addresses that are supported in the DNS name-to-address and address-to-name lookup processes. The record types are as follows:

- AAAA record

Similar to the “quad A” record in IPv4, it maps a host name to an IPv6 address.

- PTR record

Equivalent to a pointer (PTR) record in IPv4, this record maps an IPv6 address to a host name.

The top-level domain for the IPv6 addresses is ip6.arpa.

When a node needs the IPv6 address of another node, it sends an AAAA record request to the DNS. The AAAA record stores a single IPv6 address. A node with more than one address must have more than one record in the DNS database.

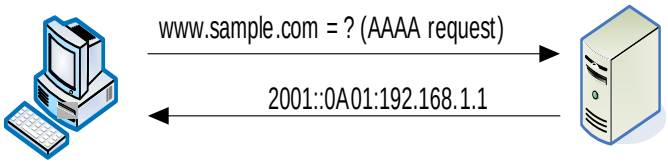

Figure 17 IPv6 DNS resolution

In order to easily modify the corresponding DNS record during IPv6 address aggregation and renumbering, the following two types of records are introduced in IPv6:

- A6 record (RFC2874)

Set to Experimental by the IETF, the A6 record will not be used in production networks. It is equivalent to an AAAA record, but enables the storing of IPv6 addresses in a hierarchical manner to simplify network renumbering.

- DNAME record (RFC2672)
- Binary Labels records (RFC2673)

These new records make renumbering easy for inverse mapping (IP address to host name).

During renumbering, all the nodes must change their IPv6 address prefixes. If DNS is used in the network for renumbering, the address information in the DNS records must also be updated.

4 Technologies for Transition from IPv4 to IPv6

Today, there are plenty of IPv4 networks but more and more IPv6 networks will be deployed. The IPv4 networks and IPv6 networks will coexist for a long time. The transition from IPv4 to IPv6 is usually divided into three phases:

- Initial phase: IPv4 networks dominate. Several isolated IPv6 sites appear and are connected via IPv4 networks.
- Coexistence phase: As more and more IPv6 networks are deployed, the IPv6 application reaches a large scale. Several backbone IPv6 networks appear. IPv6 services also keep increasing. However, various IPv6 networks need be connected together via IPv4 networks, and IPv4 hosts need interoperate with IPv6 hosts. In this phase, the dual-stack, tunneling and network protocol translation technologies need be used.
- IPv6-dominating phase: IPv6 networks and hosts dominate. As IPv6 keeps developing, all the backbone networks are IPv6 networks whereas IPv4 networks become isolated sites. In this phase, the tunneling technology is used for network deployment and IPv4 networks are interconnected through tunnels.

In the transition process, the following technologies may be used:

- Dual-stack technology: The dual-stack nodes communicate with IPv4 nodes using the IPv4 protocol stack and with IPv6 nodes using the IPv6 protocol stack.
- Tunneling technology: Two IPv6 sites are connected and communicate with each other via an IPv4 network. Two IPv4 sites are connected and communicate with each other via an IPv6 network.
- IPv4/IPv6 protocol translation technology: IPv4 networks and IPv6 networks can access each other.

Tunneling is a technology that encapsulates a protocol into another protocol. The devices at the two endpoints of the tunnel (that is, the points where the borders of the two protocols intersect) must support the two protocols. The IPv6 over IPv4 tunnels utilize the existing IPv4 networks to connect two independent IPv6 networks. IPv6 packets are encapsulated in IPv4 packets over the IPv4 networks and are transparently transmitted.

The merit is that there is no need to upgrade all the devices to dual-stack devices but only the devices at the edge of the IPv4/IPv6 network need implement the dual-stack and tunneling functions. The nodes except for the edge nodes need not support the dual protocol stacks. This can greatly improve the utilization of the investment in the existing IPv4 networks. However, the tunneling technology cannot implement the direct communication between an IPv4 host and an IPv6 host.

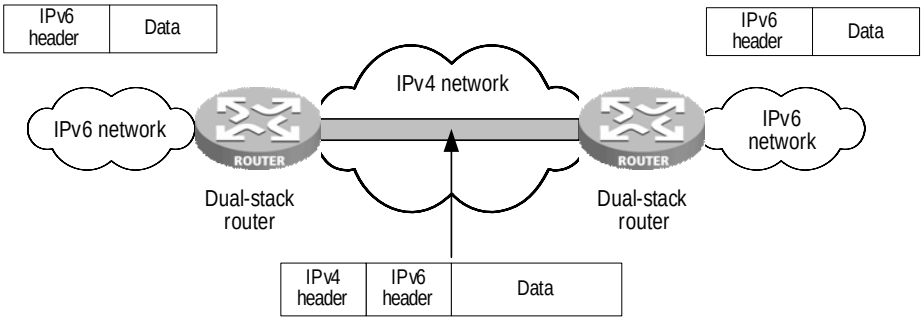

Figure 18 IPv6 over IPv4 tunnel

When the edge device of an IPv6 network receives an IPv6 packet from another IPv6 network, it encapsulates the IPv6 packet in an IPv4 packet. When the IPv4 packet is transmitted via the IPv4 network to the edge device of the destination IPv6 network, the outside IPv4 header is removed through decapsulation and the IPv4 packet is restored into the original IPv6 packet. The IPv6 packet is then forwarded.

The following techniques are used for IPv6 over IPv4 tunneling:

- IPv6 Manually Configured Tunnel
- Automatic IPv4-Compatible Tunnel
- Automatic 6to4 Tunnel
- Automatic ISATAP Tunnel
- IPv6 over IPv4 GRE Tunnel
- Tunnel Proxy
- 6over4 Tunnel
- BGP Tunnel
- Teredo Tunnel

The 3Com S7900E supports three tunneling techniques: IPv6 Manually Configured Tunnel, Automatic 6to4 Tunnel and ISATAP Tunnel.

4.1 IPv6 Manually Configured Tunnel

The source address and destination address of an IPv6 manually-configured tunnel are manually specified. An IPv6 manually-configured tunnel provides a point-to-point connection. An IPv6 manually-configured tunnel is located between two edge routers to provide stable connections for two IPv6 networks separated by an IPv4 network, or between an end system and an edge router to enable the end system to access IPv6 networks. The devices used as tunnel endpoints must support both the IPv6 protocol stack and the IPv4 protocol stack. The rest devices need only support a single protocol stack.

IPv6 manually-configured tunnels require configuration of both the source and destination addresses of the tunnel on the devices. If manually-configured tunnels need be established between an edge device and multiple devices, multiple tunnels need be configured on the devices. Therefore, an IPv6 manually-configured tunnel is often established between two edge routers to provide connections for two IPv6 networks.

An IPv6 manually-configured tunnel exists as a virtual interface on the device. After receiving an IPv6 packet from an IPv6 network, the device looks up in the IPv6 forwarding table based on the destination address in the IPv6 packet. If the packet is to be forwarded via the virtual tunnel interface, the device encapsulates the IPv6 packet based on the tunnel source and destination Ipv4 addresses configured on the tunnel interface. The encapsulated packet becomes an IPv4 packet and is delivered to the IPv4 protocol stack for processing. The packet is then forwarded via the IPv4 network to the endpoint of the tunnel.

Upon receipt of a tunnel protocol packet, the endpoint of the tunnel decapsulates the packet and delivers the decapsulated packet to the IPv6 protocol stack for processing.

You cannot configure two IPv6 manual tunnels with the same source and destination on one device.

See RFC 2893 “Transition Mechanisms for IPv6 Hosts and Routers”.

4.2 Automatic 6to4 Tunnel

The 6to4 tunnel is a kind of automatic tunnel. It is established by embedding an IPv4 address in an IPv6 address. Different from the automatic IPv4-compatible tunnel, the automatic 6to4 tunnel supports the Router to Router, Host to Router, Router to Host, and Host to Host tunnel modes. This is because a 6to4 address uses an IPv4 address as the network ID. The following figure shows the address format.

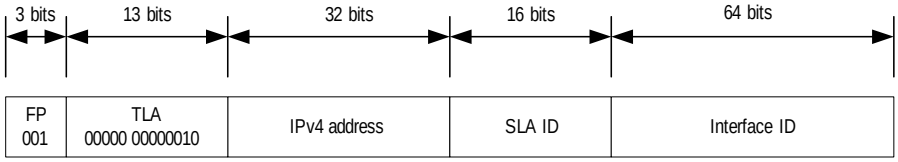

Figure 19 6to4 address

The Format Prefix (FP) is binary 001. The Top Level Aggregator (TLA) is 0002. In other words, a 6to4 address can be expressed by 2002::/16 whereas a 6to4 network can be expressed by 2002:IPv4 address::/48.

An automatic 6to4 tunnel allows isolated IPv6 networks to be connected over an IPv4 network. Automatic 6to4 tunnels are implemented via virtual tunnel interfaces. The IPv4 address of the ingress of a 6to4 tunnel is manually specified, and the destination address of the tunnel is decided by the packet to be forwarded via the tunnel. If the destination address in the IPv6 packet is a 6to4 address, an IPv4 address is extracted from the destination address in the packet and serves as the destination address of the tunnel. If the destination address in the IPv6 packet is not a 6to4 address but the next-hop address is a 6to4 address, an IPv4 address is extracted from the next-hop address and serves as the destination address of the tunnel. In the latter case, it is called the “6to4 relay”.

After an IPv6 packet arrives at the edge router, the router looks up in the forwarding table based on the IPv6 destination address in the packet. If the egress interface is the virtual tunnel interface of an automatic 6to4 tunnel and the destination address in the packet points to a 6to4 address or the next-hop address is a 6to4 address, the

router takes the IPv4 address out of the 6to4 address and uses it as the destination address of the tunnel packet. The source address of the tunnel packet is configured on the tunnel interface.

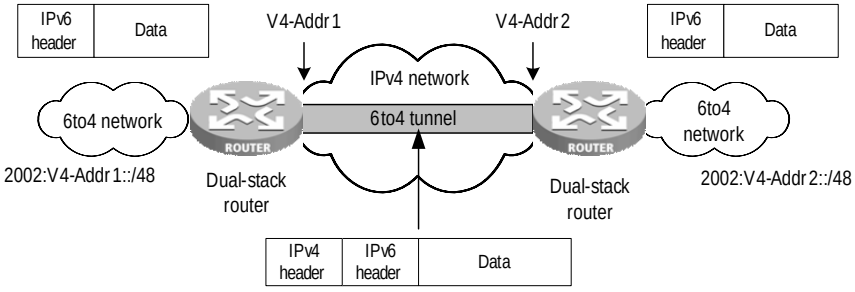

Figure 20 6to4 tunnel networking

One IPv4 address can be used as the source address of one 6to4 tunnel only. If the same IPv4 address is used as the local address of multiple 6to4 networks on an edge router, the 6to4 networks are differentiated by the SLA ID in the 6to4 address but share the same tunnel.

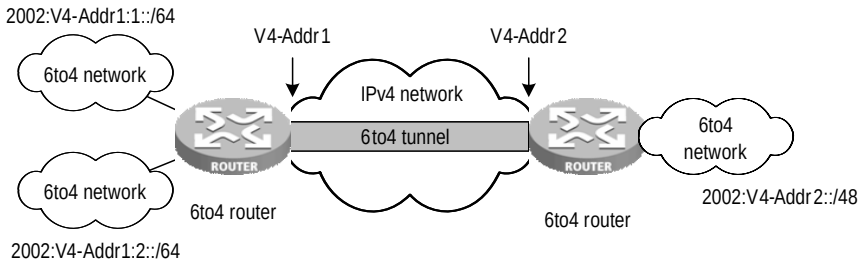

Figure 21 6to4 tunnel networking (2)

See RFC 3056 “Connection of IPv6 Domains via IPv4 Clouds”.

4.3 ISATAP Tunnel

The Intra-Site Automatic Tunnel Addressing Protocol (ISATAP) is another automatic IPv6 tunnel technology. Similar to a 6to4 address, an ISATAP address has also an embedded IPv4 address and the tunnel encapsulation is also performed based on this embedded IPv4 address, except that the formats of the two addresses differ. The 6to4 tunnel technology uses an IPv4 address as the network ID, but ISATAP uses an IPv4 address as the interface ID. The interface ID is constructed in the revised EUI-64 format, as shown in the following figure.

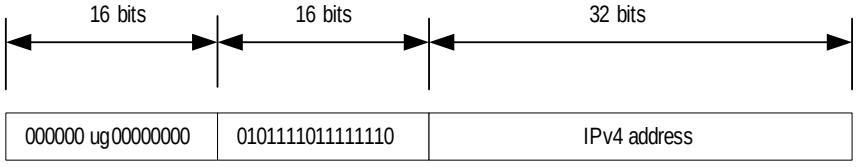

Figure 22 ISATAP interface ID

If the IPv4 address is globally unique, the u bit is set to 1. Otherwise, the u bit is set to 0. The g bit is the IEEE group/individual flag. The interface ID in an ISATAP address seems like 00-00-5E-FE followed by an IPv4 address. 5E-FE is allocated by the IANA.

Since ISATAP acts with the interface ID, ISATAP addresses are divided into many types including global unicast address, site unicast addresses and multicast addresses. The first 64 bits in an ISATAP address are obtained by sending a request to the ISATAP router, which supports address auto-configuration. The ND protocol runs between the two endpoints of an ISATAP tunnel. The ISATAP tunnel regards the IPv4 network as a non-broadcast point-to-multipoint (NBMA) link.

The ISATAP transition mechanism enables the deployment of IPv6 inside the existing IPv4 networks. The technique is simple, features high expandability and can be applied to the transition of local sites. ISATAP supports IPv6 routing within both the site-local and global IPv6 routing domains and automatic IPv6 tunneling. ISATAP also supports automatic tunneling within sites that use non-globally unique IPv4 address assignments combined with network address translation (NAT).

Because ISATAP tunneling typically occurs only within the boundaries of a site, the embedded IPv4 address need not be globally unique.

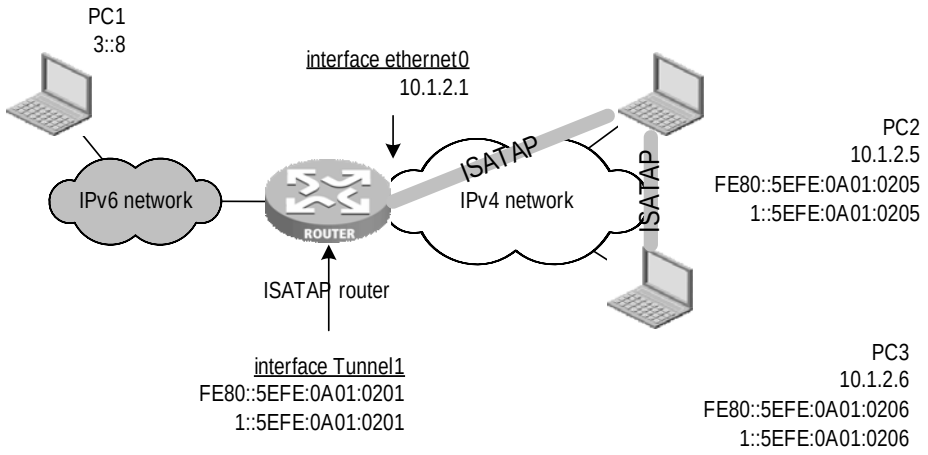

Figure 23 ISATAP tunnel

As shown in the above figure, PC2 and PC3 are two dual-stack hosts inside the IPv4 network and each has a private network IPv4 address. The following operations need be performed on each to enable them to provide the ISATAP function:

- First configure an ISATAP tunnel interface. An ISATAP interface ID will be produced according to the IPv4 address.
- An ISATAP link-local IPv6 address is produced based on the interface ID. Then the host supports the IPv6 connection function.
- The host performs auto-configuration. It obtains the global IPv6 address, site-local address and other addresses.
- When the host communicates with the other IPv6 hosts, the packet is forwarded via the tunnel interface. An IPv4 address is taken out of the next-hop IPv6 address in the packet and serves as the destination address during IPv4 encapsulation. If the destination host is located within the local site, the next hop is then the destination host itself. Otherwise, the next hop is the address of the ISATAP router.

However, if a node is located within the private network behind the NAT device, then its IPv4 address is a private address. To communicate with other sites over the NAT device, UDP/IPv4 encapsulation can apply with the port number being 3544.

See the Intra-Site Automatic Tunnel Addressing Protocol (draft-ietf-ngtrans-isatap-04.txt).

5 IPv6 Unicast Routing Technology

IPv6 supports various unicast routing protocols such as IGP and EGP. The implementation of IPv6 unicast routing protocols is similar to that in IPv4. Some are simply expanded (to ISISv6 and BGP4) and others are totally new versions (RIPng and OSPFv3).

5.1 RIPng

The Routing Information Protocol Next-Generation (RIPng) protocol is an extension of the RIP-2 protocol applied in IPv4 networks. The majority of RIP concepts can be applied to RIPng.

In order to be applied in IPv6 networks, RIPng has the following changes as compared with the original RIP protocol:

- UDP port No.: UDP port 521 is used to send and receive routing information.
- Multicast address: FF02::9 is used as the RIPng router multicast address within the link-local scope.
- Route prefix: A 128-bit IPv6 address is used as the route prefix.
- Next-hop address: 128-bit IPv6 address.

5.2 OSPFv3

OSPF version 3 (OSPFv3) provides support for IPv6 and conforms to RFC2740 (OSPF for IPv6). Compared with OSPFv2, OSPFv3 completely takes into account the network-independent feature and expandability of the protocol to further clarify the relationship between topologies and routes, in addition to providing support for IPv6. Therefore, the protocol logic of OSPF is much simpler and clearer, and the expandability of OSPF is greatly improved.

The major differences between OSPFv3 and OSPFv2 are as follows:

- The types and formats of LSAs are modified, so that OSPFv3 supports the distribution of IPv6 routing information.
- Some protocol procedures are modified, so that OSPFv3 is independent of network protocols and has higher expandability. The major revisions include the use of the Router-ID to identify a neighbor and the use of a link-local address to discover a neighbor. These revisions enable the topology to be independent of network protocols for easy expansion in the future.
- OSPFv3 further clarifies the relationship between topologies and routes. In OSPFv3 LASs, the topology is separated from routing information. Classes 1 and 2 LSAs no longer carry any routing information but carry only the simple topology description information. Moreover, the new Class 8 and Class 9 LSAs are combined with the old Class 3, Class 5 and Class 7 LSAs to distribute route prefix information.
- The protocol adaptability is improved in OSPFv3. By introducing the LSA flooding range concept, OSPFv3 further clarifies the processing of unknown LSAs, so that the protocol can make appropriate processing without identifying the LSA. In this way, the protocol's adaptability to future expansion is greatly improved.

5.3 ISISv6

IS-IS is a dynamic routing protocol released by the International Standardization Organization (ISO) for connectionless network protocol (CLNP). To enable IS-IS to support IPv4, IETF expanded the IS-IS protocol in RFC1195 and named it as the Integrated IS-IS or Dual IS-IS. This new IS-IS protocol can also be applied to TCP/IP and OSI environments. To effectively support IPv6, IETF further expanded IS-IS in draft-ietf-isis-ipv6-05.txt, that is, it added two Type-Length-Values (TLVs) and a new Network Layer Protocol Identifier (NLP ID) supporting IPv6 routing information.

TLV is a variable-length structure in the Link State PDU (LSP).

The two new TLVs are:

- IPv6 Reachability: The type value is 236 (0xEC). It describes the network reachability by defining the information such as routing information prefix and measurements.
- IPv6 Interface Address: The type value is 232 (0xE8). It is equivalent to the “IP Interface Address” TLV in IPv4, except that the 32-bit IPv4 address is changed to the 128-bit IPv6 address.

The NLP ID is a 8-bit field that identifies the specific network layer protocol supported by IS-IS. For IPv6, the NLP ID is 142 (0x8E). If an IS-IS router supports IPv6, it must carry the NLP ID in the Hello packet to tell its neighbors that it supports IPv6.

5.4 BGP4+

The legacy BGP-4 can only manage the routing information of IPv4. It is restricted in the event of inter-AS transmission for the applications that use other network layer protocols such as IPv6.

To provide support for multiple network layer protocols, IETF expanded BGP-4 to BGP4+. At present, the BGP4+ standard is RFC2858 (Multiprotocol Extensions for BGP-4).

To support the IPv6 protocol, BGP-4+ need indicate the IPv6 network layer protocol information in the Network Layer Reachable Information (NLRI) and Next_Hop attributes.

BGP4+ has introduced two NLRI attributes:

- MP_REACH_NLRI: Multiprotocol Reachable NLRI, which is used to distribute the information on reachable routes and the next hop.
- MP_UNREACH_NLRI: Multiprotocol Unreachable NLRI, which is used to cancel unreachable routes.

In BGP4+, the Next_Hop attribute is expressed by an IPv6 address, which can be an IPv6 global unicast address or the next-hop link-local address.

BGP4+ uses the multi-protocol extension attribute of BGP so as to be applied in IPv6 networks. The original message mechanism and routing mechanism of BGP do not change.

5.5 IPv6 Unicast Routing Capacity of the S7900E

The S7900E supports the large-capacity routing mode. In this mode, OSPFv3, ISISv6 and BGP4+ support up to 64K routes whereas RIPng supports up to 5K routes.

6 IPv6 Deployment Strategy

IPv4 networks are dominating today and will still dominate in a rather long period of time. IPv6-related technologies are continuously improved. Therefore, the deployment of IPv6 networks is a progressive process, and IPv6 networks and IPv4 networks will coexist in a long time.

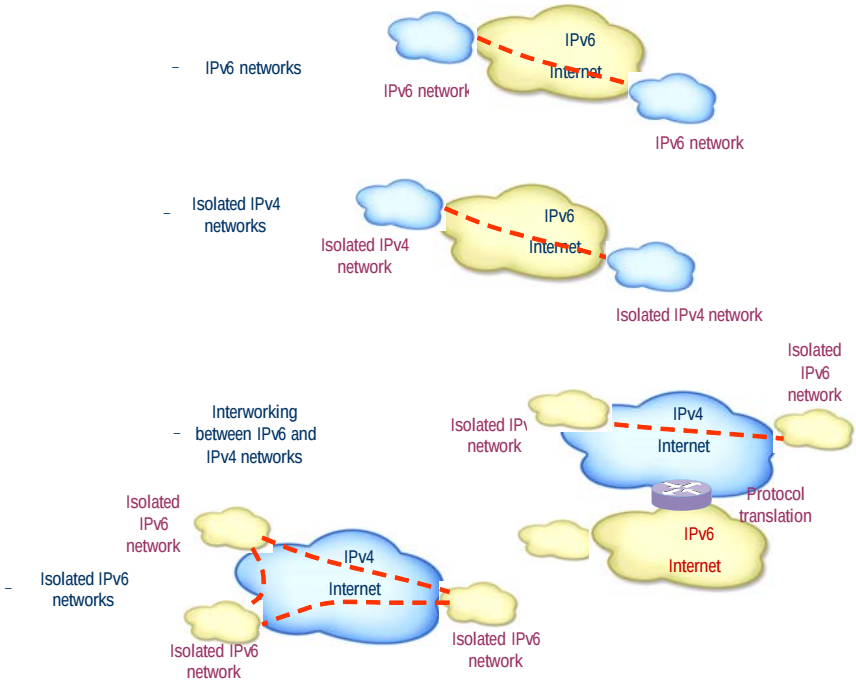

Figure 24 IPv6 network evolution

The evolution of IPv6 networks is a process from isolated IPv6 networks to IPv6 backbone networks. Different deployment strategies can be adopted to suit the specific scenarios and applications:

- Deploying IPv6 over dual protocol stacks (IPv4 and IPv6)

This mode is applicable to small-sized enterprise networks (e.g. small campus networks). The deployment is easy. Both IPv4 and IPv6 applications can be provided. All routers must support dual stacks and the IPv4/IPv6 routing table. However, routers must have both IPv4 and IPv6 routing protocols. Therefore, the upgrade workload is huge for a large-sized network.

- Deploying IPv6 over dedicated data links

This mode is applicable when ATM, Frame Relay or DWDM is deployed in the carrier's WAN or MAN. Only the routers to access the WAN need support dual stacks. The deployment is simple and does not affect the existing IPv4 traffic. However, there is no dedicated hardware supporting IPv6 and thus the forwarding rate cannot be improved through hardware.

- Deploying IPv6 through tunneling

This mode is applied for carriers to deploy IPv6 networks at the initial stage or for enterprises to connect IPv6 domains over an IPv4 network.

Only the IPv6 DNS query function need be provided. In addition, the routers between an IPv4 network and an IPv6 network need support dual stacks. Both the investment and the risk level are low. However, the network topology is complex with the

tunneling technology and management is more difficult. Faults can hardly be located. Moreover, the forwarding efficiency is affected to some extent due to the use of tunnel encapsulation.

- Deploying IPv6 over MPLS networks

This mode is applicable to mobile carriers or the carriers with MPLS networks. IPv6 runs over an MPLS network without the need for software or hardware upgrade of the core network. Only slight changes need be done to the PE or CE equipment (the most typical application is 6PE). However, an MPLS network is needed. Similar to the tunneling mode, the network management overheads are big.

- Deploying IPv6 through the protocol translation mechanism

The most common protocol translation mechanism is NAT-PT. In general, the other protocol translation mechanisms are not implemented on network devices.

NAT-PT enables an IPv4-only node and an IPv6-only node to communicate with each other. NAT-PT requires only the devices supporting NAT-PT and IPv6 DNS. It does not require the devices to be dual-stacked. The end system need not be upgraded. Therefore, the deployment is much simpler. However, NAT-PT involves single point failure and the forwarding efficiency is affected since translation between IPv4 and IPv6 packets is needed.

7 Typical Networking of the S7900E

The 7900E can be used as a core layer switch in a small-sized or medium-sized enterprise, a convergence layer switch in a large-sized enterprise network, or a switch in the distribution room. It provides high-performance and large-capacity switching services, and offers higher bandwidth for access devices.

7.1 Applying the S7900E in the Core Layer

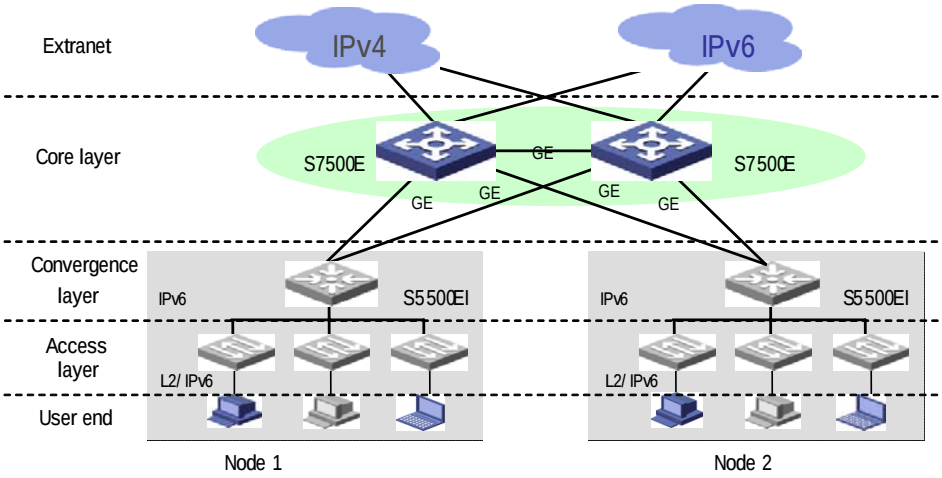

Figure 25 Applying the S7900E in the core layer

In small- or medium-sized enterprises, the S7900E can serve as a core layer switch that provides high-performance, large-capacity switching services. In the large-capacity routing mode, the S7900E can have 64K IPv6 routes to offer powerful L3 switching capability.

7.2 Applying the S7900E in the Convergence Layer

In a large-sized enterprise network, the S7900E can serve as a convergence layer switch that provides high-performance, large-capacity switching services. It supports 10GE uplink interfaces and offers higher bandwidth for access devices.

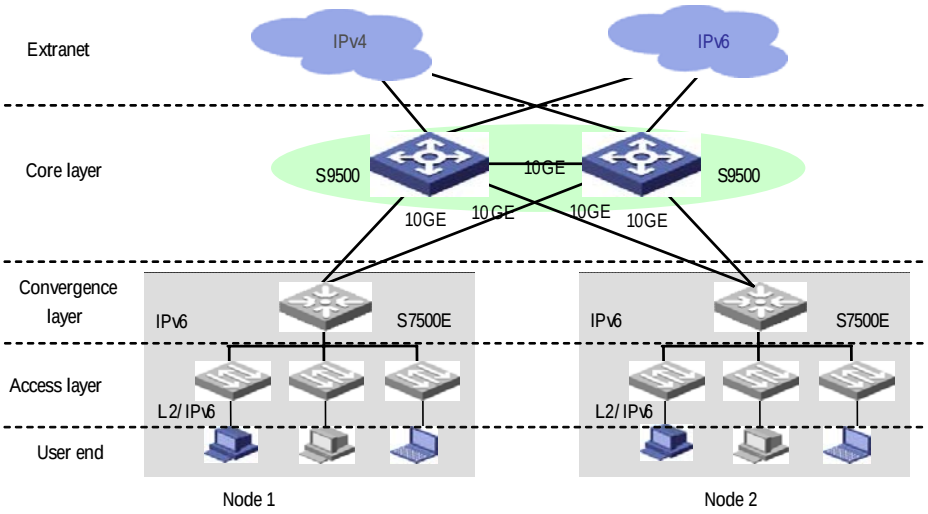

Figure 26 Applying the S7900E in the convergence layer

7.3 Applying the S7900E in Data Center Interconnection

The S7900E can provide high-speed access for servers in a data center. It provides wire-speed L2 and L3 switching performance, offering high-performance and large-capacity switching capability for the data enter. In addition, it provides strong chip buffer capability, and the 10GE uplink capability to make possible further expansion of the egress bandwidth of the data center.

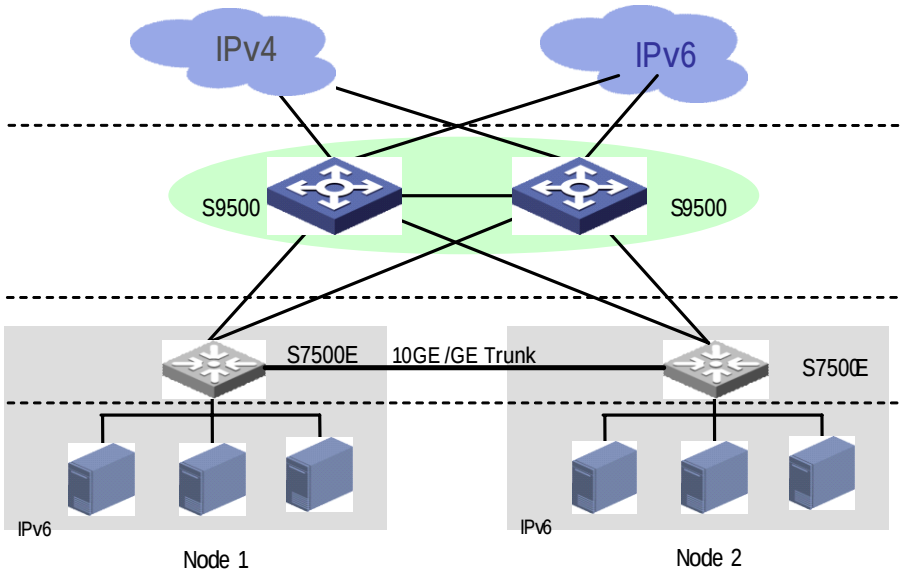

Figure 27 Applying the S7900E in data center networking

7.4 Applying the S7900E in the Transition from IPv4 to IPv6

The S7900E supports IPv6 tunneling techniques including 6to4 and ISATAP. It can be applied in the smooth evolution from IPv4 to IPv6, that is, it connects isolated IPv6 networks via tunneling, so that the networks smoothly evolve to IPv6 networks along with the development of the IPv6 technology and networks.

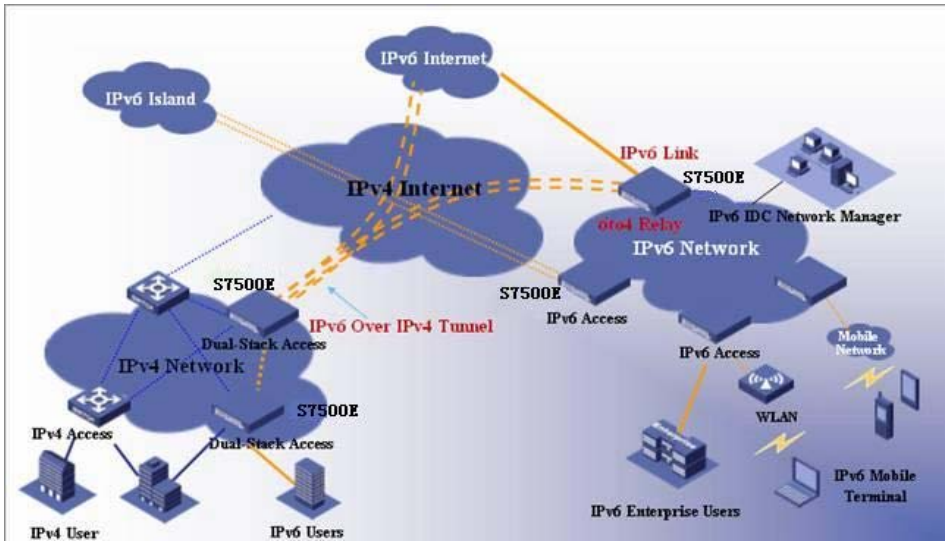

Figure 28 Applying the S7900E in the transition from IPv4 to IPv6

7.5 Abbreviations:

Abbreviation	Full Spelling
ND	Neighbour Discovery Protocol
PMTUD	Path MTU Discovery Protocol
ISATAP	Intra-Site Automatic Tunnel Addressing Protocol
Teredo	Tunneling IPv6 over UDP through NATs
RIPng	Route Information Protocol Next Generation
OSPFv3	Open Short Path First Protocol Version 3
BGP4+	Boarder Gateway Protocol 4+

8 Protocol Standards

- RFC 1886, DNS Extensions to support IP version 6
- RFC 1887, An Architecture for IPv6 Unicast Address Allocation
- RFC 1888, OSI NSAPs and IPv6
- RFC 1981, Path MTU Discovery for IP version 6
- RFC 2553, Basic Socket Interface Extensions for IPv6
- RFC 2374, An IPv6 Provider-Based Unicast Address Format
- RFC 2375, IPv6 Multicast Address Assignments

- RFC 2471, IPv6 Testing Address Allocation
- RFC 2450, Proposed TLA and NLA Assignment Rules
- RFC 2460, Internet Protocol, Version 6 (IPv6) Specification
- RFC 2461, Neighbor Discovery for IP Version 6 (IPv6)
- RFC 2462, IPv6 Stateless Address Autoconfiguration
- RFC 2463, Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6)
- RFC 2464, A Method for the Transmission of IPv6 Packets over Ethernet Networks
- RFC 2465, Management Information Base for IP Version 6: Textual Conventions and General Group
- RFC 2466, Management Information Base for IP Version 6
- RFC 2467, Transmission of IPv6 Packets Over FDDI
- RFC 2472, IP Version 6 over PPP
- Generic Packet Tunneling in IPv6 Specification
- RFC 2529, Transmission of IPv6 over IPv4 Domains without Explicit Tunnels
- RFC 2663, IP Network Address Translator (NAT) Terminology and Considerations
- RFC 2675, TCP and UDP over IPv6 Jumbograms
- RFC 2710, Multicast Listener Discovery (MLD) for IPv6
- RFC 2711, IPv6 Router Alert Option
- RFC 2765, Stateless IP/ICMP Translation Algorithm (SIIT)
- RFC 2766, Network Address Translation - Protocol Translation (NAT-PT)
- RFC 2767, Dual Stack Hosts using the "Bump-In-the-Stack" Technique (BIS)
- RFC 2874, DNS Extensions to Support IPv6 Address Aggregation and Renumbering
- RFC 2893, Transition Mechanisms for IPv6 Hosts and Routers
- RFC 2894, Router Renumbering for IPv6
- RFC 2928, Initial IPv6 Sub-TLA ID Assignments
- RFC 3019, IP Version 6 Management Information Base for the Multicast Listener Discovery Protocol
- RFC 3041, Privacy Extensions for Stateless Address Autoconfiguration in IPv6
- RFC 3053, IPv6 Tunnel Broker
- RFC 3056, Connection of IPv6 Domains via IPv4 Clouds
- RFC 3089, A SOCKS-based IPv6/IPv4 Gateway Mechanism
- RFC 3122, Extensions to IPv6 Neighbor Discovery for Inverse Discovery Specification
- RFC 3142, An IPv6-to-IPv4 Transport Relay Translator
- RFC 3146, Transmission of IPv6 Packets over IEEE 1394 Networks
- RFC 3178, IPv6 multihoming support at site exit routers
- RFC 3306, Unicast-Prefix-based IPv6 Multicast Addresses
- RFC 3314, Recommendations for IPv6 in 3GPP Standards
- RFC 3316, IPv6 for Some Second and Third Generation Cellular Hosts
- RFC 3338, Dual Stack Hosts Using "Bump-in-the-API" (BIA)
- RFC 3484, Default Address Selection for Internet Protocol version 6 (IPv6)
- RFC 3493, Basic Socket Interface Extensions for IPv6
- RFC 3513, IP Version 6 Addressing Architecture
- RFC 3531, A Flexible Method for Managing the Assignment of Bits of an IPv6 Address Block
- RFC 3542, Advanced Sockets API for IPv6
- RFC 3587, An IPv6 Aggregatable Global Unicast Address Format
- RFC 3697, IPv6 Flow Label Specification
- RFC 3701, IPv6 Testing Address Allocation
- RFC 3769, Requirements for IPv6 prefix delegation
- RFC 3879, Deprecating Site Local Addresses
- RFC 3986, Format for Literal IPv6 Addresses in URL's
- RFC 4007, IPv6 Scoped Address Architecture
- RFC 4022, IP Version 6 Management Information Base for the Transmission Control Protocol
- RFC 4087, IP Tunnel MIB

- RFC 4113, IP Version 6 Management Information Base for the User Datagram Protocol
- RFC 4193, Unique Local IPv6 Unicast Addresses
- draft-ietf-ngtrans-isatap, Intra-Site Automatic Tunnel Addressing Protocol (ISATAP)
- draft-huitema-v6ops-teredo, Teredo: Tunneling IPv6 over UDP through NATs
- draft-ietf-pim-sm-v2-new-11
- draft-ietf-pim-dm-new-v2-05
- draft-ietf-magma-snoop-11