目 录

1 IPv6 基础

1.1 IPv6简介

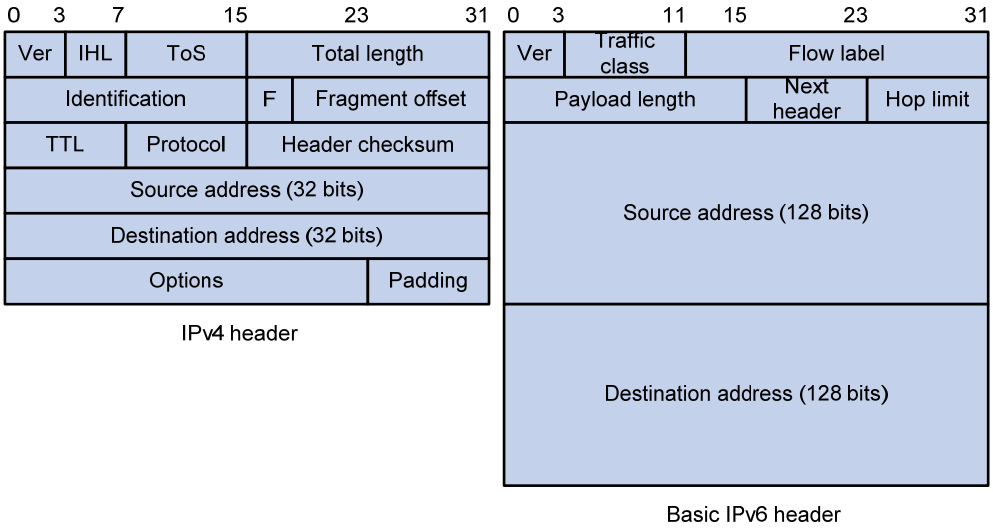
IPv6（Internet Protocol Version 6，互联网协议版本 6）是网络层协议的第二代标准协议，也被称为 IPng（IP Next Generation，下一代互联网协议），它是 IETF（Internet Engineering Task Force，互联网工程任务组）设计的一套规范，是 IPv4 的升级版本。IPv6 和 IPv4 之间最显著的区别为：IP 地址的长度从 32 比特增加到 128 比特。

1.1.1 IPv6 协议特点

1. 简化的报文头格式

通过将 IPv4 报文头中的某些字段裁减或移入到扩展报文头，减小了 IPv6 基本报文头的长度。IPv6 使用固定长度的基本报文头，从而简化了转发设备对 IPv6 报文的处理，提高了转发效率。尽管 IPv6 地址长度是 IPv4 地址长度的四倍，但 IPv6 基本报文头的长度只有 40 字节，为 IPv4 报文头长度（不包括选项字段）的两倍。

图1-1 IPv4 报文头和 IPv6 基本报文头格式比较



2. 充足的地址空间

IPv6 的源地址与目的地址长度都是 128 比特（16 字节）。它可以提供超过 3.4×10^{38} 种可能的地址空间，完全可以满足多层次的地址划分需要，以及公有网络和机构内部私有网络的地址分配。

3. 层次化的地址结构

IPv6 的地址空间采用了层次化的地址结构，有利于路由快速查找，同时可以借助路由聚合，有效减少 IPv6 路由表占用的系统资源。

4. 地址自动配置

为了简化主机配置，IPv6 支持有状态地址配置和无状态地址配置：

- 有状态地址配置是指从服务器（如 DHCPv6 服务器）获取 IPv6 地址及相关信息，详细介绍请参见“三层技术-IP 业务配置指导”中的“DHCPv6”；
- 无状态地址配置是指主机根据自己的链路层地址及路由器发布的前缀信息自动配置 IPv6 地址及相关信息。

同时，主机也可根据自己的链路层地址及默认前缀（FE80::/10）形成链路本地地址，实现与本链路上其他主机的通信。

5. 内置安全性

IPv6 将 IPsec 作为它的标准扩展头，可以提供端到端的安全特性。这一特性也为解决网络安全问题提供了标准，并提高了不同 IPv6 应用之间的互操作性。

6. 支持QoS

IPv6 报文头的流标签（Flow Label）字段实现流量的标识，允许设备对某一流中的报文进行识别并提供特殊处理。

7. 增强的邻居发现机制

IPv6 的邻居发现协议是通过一组 ICMPv6（Internet Control Message Protocol for IPv6，IPv6 互联网控制消息协议）消息实现的，管理着邻居节点间（即同一链路上的节点）信息的交互。它代替了 ARP（Address Resolution Protocol，地址解析协议）、ICMPv4 路由器发现和 ICMPv4 重定向消息，并提供了一系列其他功能。

8. 灵活的扩展报文头

IPv6 取消了 IPv4 报文头中的选项字段，并引入了多种扩展报文头，在提高处理效率的同时还大大增强了 IPv6 的灵活性，为 IP 协议提供了良好的扩展能力。IPv4 报文头中的选项字段最多只有 40 字节，而 IPv6 扩展报文头的大小只受到 IPv6 报文大小的限制。

1.1.2 IPv6 地址介绍

1. IPv6 地址表示方式

IPv6 地址被表示为以冒号（:）分隔的一连串 16 比特的十六进制数。每个 IPv6 地址被分为 8 组，每组的 16 比特用 4 个十六进制数来表示，组和组之间用冒号隔开，比如：2001:0000:130F:0000:0000:09C0:876A:130B。

为了简化 IPv6 地址的表示，对于 IPv6 地址中的“0”可以有下面的处理方式：

- 每组中的前导“0”可以省略，即上述地址可写为 2001:0:130F:0:0:9C0:876A:130B。
- 如果地址中包含一组或连续多组均为 0 的组，则可以用双冒号“::”来代替，即上述地址可写为 2001:0:130F::9C0:876A:130B。


说明

在一个 IPv6 地址中只能使用一次双冒号“::”，否则当设备将“::”转变为 0 以恢复 128 位地址时，将无法确定“::”所代表的 0 的个数。

IPv6 地址由两部分组成：地址前缀与接口标识。其中，地址前缀相当于 IPv4 地址中的网络号码字段部分，接口标识相当于 IPv4 地址中的主机号码部分。

地址前缀的表示方式为：IPv6 地址/前缀长度。其中，前缀长度是一个十进制数，表示 IPv6 地址最左边多少位为地址前缀。

2. IPv6 的地址分类

IPv6 主要有三种类型的地址：单播地址、组播地址和任播地址。

- 单播地址：用来唯一标识一个接口，类似于 IPv4 的单播地址。发送到单播地址的数据报文将被传送给此地址所标识的接口。
- 组播地址：用来标识一组接口（通常这组接口属于不同的节点），类似于 IPv4 的组播地址。发送到组播地址的数据报文被传送给此地址所标识的所有接口。
- 任播地址：用来标识一组接口（通常这组接口属于不同的节点）。发送到任播地址的数据报文被传送给此地址所标识的一组接口中距离源节点最近（根据使用的路由协议进行度量）的一个接口。

IPv6 中没有广播地址，广播地址的功能通过组播地址来实现。

IPv6 地址类型是由地址前面几位（称为格式前缀）来指定的，主要地址类型与格式前缀的对应关系如 表 1-1 所示。

表1-1 地址类型与格式前缀的对应关系

地址类型		格式前缀（二进制）	IPv6 前缀标识
单播地址	未指定地址	00...0 (128 bits)	::/128
	环回地址	00...1 (128 bits)	::1/128
	链路本地地址	1111111010	FE80::/10
	全球单播地址	其他形式	-
组播地址		11111111	FF00::/8
任播地址		从单播地址空间中进行分配，使用单播地址的格式	

3. 单播地址的类型

IPv6 单播地址的类型可有多种，包括全球单播地址、链路本地地址等。

- 全球单播地址等同于 IPv4 公网地址，提供给网络服务提供商。这种类型的地址允许路由前缀的聚合，从而限制了全球路由表项的数量。
- 链路本地地址用于邻居发现协议和无状态自动配置中链路本地上节点之间的通信。使用链路本地地址作为源或目的地址的数据报文不会被转发到其他链路上。
- 环回地址：单播地址 0:0:0:0:0:0:0:1（简化表示为::1）称为环回地址，不能分配给任何物理接口。它的作用与在 IPv4 中的环回地址相同，即节点用来给自己发送 IPv6 报文。
- 未指定地址：地址“::”称为未指定地址，不能分配给任何节点。在节点获得有效的 IPv6 地址之前，可在发送的 IPv6 报文的源地址字段填入该地址，但不能作为 IPv6 报文中的目的地址。

4. 组播地址

表 1-2 所示的组播地址，是预留的特殊用途的组播地址。

表1-2 预留的 IPv6 组播地址列表

地址	应用
FF01::1	表示节点本地范围所有节点的组播地址
FF02::1	表示链路本地范围所有节点的组播地址
FF01::2	表示节点本地范围所有路由器的组播地址
FF02::2	表示链路本地范围所有路由器的组播地址

另外，还有一类组播地址：被请求节点（Solicited-Node）地址。该地址主要用于获取同一链路上邻居节点的链路层地址及实现重复地址检测。每一个单播或任播 IPv6 地址都有一个对应的被请求节点地址。其格式为：

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

其中，FF02:0:0:0:0:1:FF 为 104 位固定格式；XX:XXXX 为单播或任播 IPv6 地址的后 24 位。

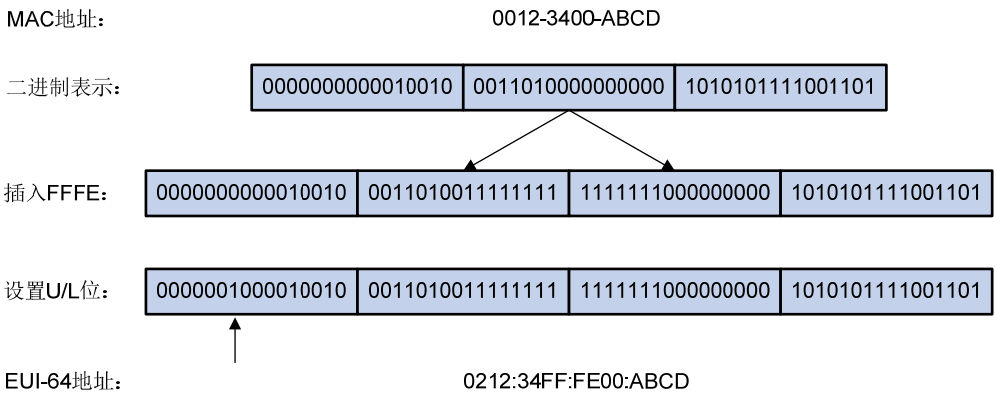
5. IEEE EUI-64 格式的接口标识符

IPv6 单播地址中的接口标识符用来唯一标识链路上的一个接口。目前 IPv6 单播地址基本上都要求接口标识符为 64 位。

不同接口的 IEEE EUI-64 格式的接口标识符的生成方法不同，分别介绍如下：

- 所有 IEEE 802 接口类型（例如，以太网接口、VLAN 接口）：IEEE EUI-64 格式的接口标识符是从接口的链路层地址（MAC 地址）变化而来的。IPv6 地址中的接口标识符是 64 位，而 MAC 地址是 48 位，因此需要在 MAC 地址的中间位置（从高位开始的第 24 位后）插入十六进制数 FFFE（1111111111111110）。为了使接口标识符的作用范围与原 MAC 地址一致，还要将 Universal/Local (U/L)位（从高位开始的第 7 位）进行取反操作。最后得到的这组数就作为 EUI-64 格式的接口标识符。

图1-2 MAC 地址到 EUI-64 格式接口标识符的转换过程



- Tunnel 接口：IEEE EUI-64 格式的接口标识符的低 32 位为 Tunnel 接口的源 IPv4 地址，ISATAP 隧道的接口标识符的高 32 位为 0000:5EFE，其他隧道的接口标识符的高 32 位为全 0。关于各种隧道的介绍，请参见“三层技术-IP 业务配置指导”中的“隧道”。
- 其他接口类型（例如，Serial 接口）：IEEE EUI-64 格式的接口标识符由设备随机生成。

1.1.3 IPv6 邻居发现协议介绍

IPv6 ND（IPv6 Neighbor Discovery，IPv6 邻居发现）协议使用五种类型的 ICMPv6 消息，实现下面一些功能：地址解析、验证邻居是否可达、重复地址检测、路由器发现/前缀发现、地址自动配置和重定向等功能。

邻居发现协议使用的ICMPv6 消息的类型及作用如 表 1-3 所示。

表1-3 邻居发现协议使用的 ICMPv6 消息类型及作用

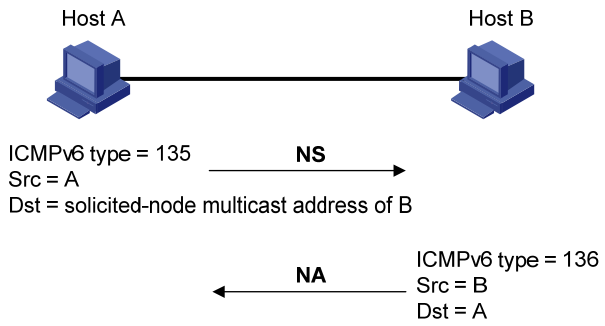
ICMPv6 消息	类型号	作用
邻居请求消息NS（Neighbor Solicitation）	135	获取邻居的链路层地址
		验证邻居是否可达
		进行重复地址检测
邻居通告消息NA（Neighbor Advertisement）	136	对NS消息进行响应
		节点在链路层变化时主动发送NA消息，向邻居节点通告本节点的变化信息
路由器请求消息RS（Router Solicitation）	133	节点启动后，通过RS消息向路由器发出请求，请求前缀和其他配置信息，用于节点的自动配置
路由器通告消息RA（Router Advertisement）	134	对RS消息进行响应
		在没有抑制RA消息发布的条件下，路由器会周期性地发布RA消息，其中包括前缀信息选项和一些标志位的信息
重定向消息（Redirect）	137	当满足一定的条件时，缺省网关通过向源主机发送重定向消息，使主机重新选择正确的下一跳地址进行后续报文的发送

邻居发现协议提供的主要功能如下：

1. 地址解析

获取同一链路上邻居节点的链路层地址（与IPv4 的ARP功能相同），通过邻居请求消息NS和邻居通告消息NA实现。如 图 1-3 所示，节点A要获取节点B的链路层地址。

图1-3 地址解析示意图



(1) 节点 A 以组播方式发送 NS 消息。NS 消息的源地址是节点 A 的接口 IPv6 地址，目的地址是节点 B 的被请求节点组播地址，消息内容中包含了节点 A 的链路层地址和请求的目标地址。

(2) 节点 B 收到 NS 消息后，判断报文的目标地址是否为自己的 IPv6 地址。如果是，则节点 B 可以学习到节点 A 的链路层地址，并以单播方式返回 NA 消息，其中包含了自己的链路层地址。

(3) 节点 A 从收到的 NA 消息中就可获取到节点 B 的链路层地址。

2. 验证邻居是否可达

在获取到邻居节点的链路层地址后，通过邻居请求消息 NS 和邻居通告消息 NA 可以验证邻居节点是否可达。

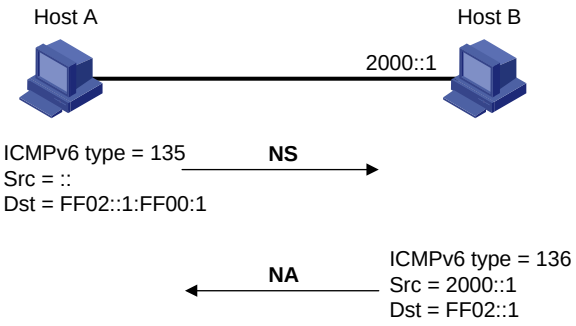
(1) 节点发送 NS 消息，其中目的地址是邻居节点的 IPv6 地址。

(2) 如果收到邻居节点的确认报文，则认为邻居可达；否则，认为邻居不可达。

3. 重复地址检测

当节点获取到一个IPv6 地址后，需要使用重复地址检测功能确定该地址是否已被其他节点使用（与 IPv4 的免费ARP功能相似）。通过NS和NA可以实现重复地址检测，如 图 1-4 所示。

图1-4 重复地址检测示意图



(1) 节点 A 发送 NS 消息，NS 消息的源地址是未指定地址::，目的地址是待检测的 IPv6 地址对应的被请求节点组播地址，消息内容中包含了待检测的 IPv6 地址。

(2) 如果节点 B 已经使用这个 IPv6 地址，则会返回 NA 消息。其中包含了自己的 IPv6 地址。

(3) 节点 A 收到节点 B 发来的 NA 消息，就知道该 IPv6 地址已被使用。反之，则说明该地址未被使用，节点 A 就可使用此 IPv6 地址。

4. 路由器发现/前缀发现及地址无状态自动配置

路由器发现/前缀发现是指节点从收到的 RA 消息中获取邻居路由器及所在网络的前缀，以及其他配置参数。

地址无状态自动配置是指节点根据路由器发现/前缀发现所获取的信息，自动配置 IPv6 地址。

路由器发现/前缀发现通过路由器请求消息 RS 和路由器通告消息 RA 来实现，具体过程如下：

(1) 节点启动时，通过 RS 消息向路由器发出请求，请求前缀和其他配置信息，以便用于节点的配置。

(2) 路由器返回 RA 消息，其中包括前缀信息选项（路由器也会周期性地发布 RA 消息）。

(3) 节点利用路由器返回的 RA 消息中的地址前缀及其他配置参数，自动配置接口的 IPv6 地址及其他信息。

前缀信息选项中不仅包括地址前缀的信息，还包括该地址前缀的首选生命期（preferred lifetime）和有效生命期（valid lifetime）。节点收到周期性发送的 RA 消息后，会根据该消息更新前缀的首选生命期和有效生命期。

有效生命期：表示前缀有效期。在有效生命期内，通过该前缀自动生成的地址可以正常使用；有效生命期过期后，通过该前缀自动生成的地址变为无效，将被删除。

首选生命期：表示首选通过该前缀无状态自动配置地址的时间。首选生命期过期后，节点通过该前缀自动配置的地址将被废止。节点不能使用被废止的地址建立新的连接，但是仍可以接收目的地址为被废止地址的报文。首选生命期必须小于或等于有效生命期。

5. 重定向功能

当主机启动时，它的路由表中可能只有一条到缺省网关的缺省路由。当满足一定的条件时，缺省网关会向源主机发送 ICMPv6 重定向消息，通知主机选择更好的下一跳进行后续报文的发送（与 IPv4 的 ICMP 重定向消息的功能相同）。

同时满足下列条件时，设备会发送 ICMPv6 重定向报文：

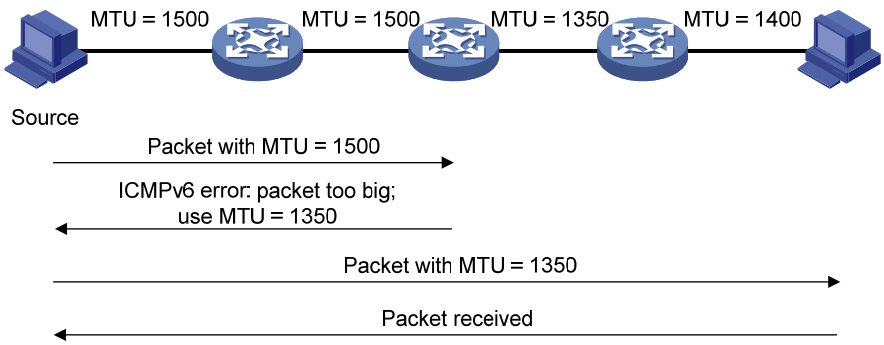
- 接收和转发数据报文的接口是同一接口；
- 被选择的路由本身没有被 ICMPv6 重定向报文创建或修改过；
- 被选择的路由不是设备的缺省路由；
- 被转发的 IPv6 数据报文中不包含路由扩展头。

1.1.4 IPv6 PMTU发现

报文从源端到目的端的传输路径中所经过的链路可能具有不同的 MTU。在 IPv6 中，当报文的长度大于链路的 MTU 时，报文的分片将在源端进行，从而减轻中间转发设备的处理压力，合理利用网络资源。

PMTU（Path MTU，路径MTU）发现机制的目的就是要找到从源端到目的端的路径上最小的MTU。PMTU的工作过程如 图 1-5 所示。

图1-5 PMTU 发现工作过程



(1) 源端主机按照自己的 MTU 对报文进行分片，之后向目的主机发送报文。

(2) 中间转发设备接收到该报文进行转发时，如果发现转发报文的接口支持的 MTU 值小于报文长度，则会丢弃报文，并给源端返回一个 ICMPv6 差错报文，其中包含了转发失败的接口的 MTU。

(3) 源主机收到该差错报文后，将按照报文中所携带的 MTU 重新对报文进行分片并发送。

(4) 如此反复，直到目的端主机收到这个报文，从而确定报文从源端到目的端路径中的最小 MTU。

1.1.5 IPv6 过渡技术介绍

在 IPv6 成为主流协议之前，首先使用 IPv6 协议栈的网络希望能与当前仍被 IPv4 支撑着的互联网进行正常通信，因此必须开发出 IPv4 和 IPv6 互通技术以保证 IPv4 能够平稳过渡到 IPv6。互通技术应该对信息传递做到高效无缝。目前已经出现了多种过渡技术，这些技术各有特点，用于解决不同过渡时期、不同环境的通信问题。

目前解决过渡问题的基本技术主要有 4 种：双协议栈（RFC 2893）、隧道技术（RFC 2893）、NAT-PT（RFC 2766）、6PE。

1. 双协议栈

双协议栈是一种最简单直接的过渡机制。同时支持 IPv4 协议和 IPv6 协议的网络节点称为双协议栈节点。当双协议栈节点配置 IPv4 地址和 IPv6 地址后，就可以在相应接口上转发 IPv4 和 IPv6 报文。当一个上层应用同时支持 IPv4 和 IPv6 协议时，根据协议要求可以选用 TCP 或 UDP 作为传输层的协议，但在选择网络层协议时，它会优先选择 IPv6 协议栈。双协议栈技术适合 IPv4 网络节点之间或者 IPv6 网络节点之间通信，是所有过渡技术的基础。但是，这种技术要求运行双协议栈的节点有一个全球唯一的地址，实际上没有解决 IPv4 地址资源匮乏的问题。

2. 隧道技术

隧道是一种封装技术，它利用一种网络协议来传输另一种网络协议，即利用一种网络传输协议，将其他协议产生的数据报文封装在它自己的报文中，然后在网络中传输。关于隧道技术的详细介绍，请参见“三层技术-IP 业务配置指导”中的“隧道”。

3. NAT-PT

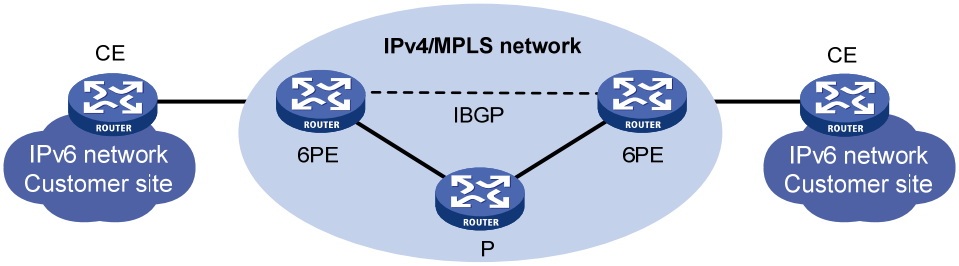
NAT-PT（Network Address Translation-Protocol Translation，附带协议转换的网络地址转换）作用于 IPv4 和 IPv6 网络边缘的设备上，用于实现 IPv6 与 IPv4 报文的转换。NAT-PT 在 IPv4 和 IPv6 网络之间转换 IP 报头的地址，同时根据协议不同对报文做相应的语义翻译，使纯 IPv4 节点和纯 IPv6 节点之间能够透明通信。这种技术适用于仅运行 IPv6 的节点和仅运行 IPv4 的节点之间的通信，具有一定的局限性。关于 NAT-PT 的详细介绍，请参见“三层技术-IP 业务配置指导”中的“NAT-PT”。

4. 6PE

6PE 是一种过渡技术，ISP 可以利用已有的 IPv4 骨干网为分散用户的 IPv6 网络提供接入能力。

6PE 的主要思想是：6PE（IPv6 Provider Edge，IPv6 供应商边缘）路由器将用户的 IPv6 路由信息转换为带有标签的 IPv6 路由信息，并且通过 IBGP（Internal Border Gateway Protocol，内部边界网关协议）会话扩散到 ISP 的 IPv4 骨干网中。6PE 路由器转发 IPv6 报文时，首先会将进入骨干网隧道的数据流打上标签。隧道可以是 GRE 隧道或者 MPLS LSP 等。有关 6PE 的详细介绍及配置请参见“三层技术-IP 路由配置指导”中的“IPv6 BGP”。

图1-6 6PE 组网图



当 ISP 想利用自己原有的 IPv4/MPLS 网络，使其通过 MPLS 具有 IPv6 流量交换能力时，只需要升级 PE 路由器就可以了。所以对于运营商来说，使用 6PE 技术作为 IPv6 过渡机制无疑是一个高效的解决方案，其操作风险也会小得多。

1.1.6 协议规范

与 IPv6 基础相关的协议规范有：

- RFC 1881：IPv6 Address Allocation Management
- RFC 1887：An Architecture for IPv6 Unicast Address Allocation
- RFC 1981：Path MTU Discovery for IP version 6
- RFC 2375：IPv6 Multicast Address Assignments
- RFC 2460：Internet Protocol, Version 6 (IPv6) Specification
- RFC 2464：Transmission of IPv6 Packets over Ethernet Networks
- RFC 2526：Reserved IPv6 Subnet Anycast Addresses
- RFC 3307：Allocation Guidelines for IPv6 Multicast Addresses
- RFC 4191：Default Router Preferences and More-Specific Routes
- RFC 4291：IP Version 6 Addressing Architecture
- RFC 4443：Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification
- RFC 4861：Neighbor Discovery for IP Version 6 (IPv6)
- RFC 4862：IPv6 Stateless Address Autoconfiguration

1.2 IPv6基础配置任务简介

表1-4 IPv6 基础配置任务简介

配置任务		说明	详细配置
配置IPv6基本功能	配置IPv6全球单播地址	三者至少选其一	1.3.1
	配置IPv6链路本地地址		1.3.2
	配置IPv6任播地址		1.3.3
配置IPv6邻居发现协议	配置静态邻居表项	可选	1.4.1

配置任务		说明	详细配置
	配置接口上允许动态学习的邻居的最大个数	可选	1.4.2
	配置STALE状态ND表项的老化时间	可选	1.4.3
	配置链路本地ND表项资源占用最小化	可选	1.4.4
	配置设备的跳数限制	可选	1.4.5
	配置RA消息的相关参数	可选	1.4.6
	配置重复地址检测时发送邻居请求消息的次数	可选	1.4.7
	配置ND Proxy功能	可选	1.4.8
配置PMTU发现	配置接口MTU	可选	1.5.1
	配置指定地址的静态PMTU	可选	1.5.2
	配置PMTU老化时间	可选	1.5.3
配置ICMPv6报文发送	配置指定时间内发送ICMPv6差错报文的最大个数	可选	1.6.1
	配置允许回复组播形式的Echo request报文	可选	1.6.2
	配置ICMPv6目的不可达差错报文发送功能	可选	1.6.3
	配置ICMPv6超时差错报文发送功能	可选	1.6.4
	配置ICMPv6重定向报文发送功能	可选	1.6.5
	配置ICMPv6报文指定源地址功能	可选	1.6.6
IPv6分片报文本地重组功能	配置IPv6分片报文本地重组功能	可选	1.7
IPv6报文扩展头丢弃功能	开启IPv6报文扩展头丢弃功能	可选	1.8

1.3 配置IPv6基本功能

1.3.1 配置IPv6 全球单播地址

IPv6 全球单播地址可以通过下面几种方式配置：

- 采用 EUI-64 格式形成：当配置采用 EUI-64 格式形成 IPv6 地址时，接口的 IPv6 地址的前缀需要手工配置，而接口标识符则由接口自动生成。
- 手工配置：用户手工配置 IPv6 全球单播地址。
- 引用前缀生成 IPv6 地址：引用前缀生成 IPv6 地址时，接口的 IPv6 地址的前缀可以通过手工配置或 DHCPv6 动态获取，同时该前缀还会分配给终端设备。
- 无状态自动配置：根据接收到的 RA 报文中携带的地址前缀信息，自动生成 IPv6 全球单播地址。

每个接口可以有多个全球单播地址。

手工配置的全球单播地址（包括采用 EUI-64 格式形成的全球单播地址）的优先级高于自动生成的全球单播地址。如果在接口已经自动生成全球单播地址的情况下，手工配置前缀相同的全球单播地

址，不会覆盖之前自动生成的全球单播地址。如果删除手工配置的全球单播地址，设备还可以使用自动生成的全球单播地址进行通信。

1. 采用EUI-64 格式形成IPv6 地址

表1-5 采用 EUI-64 格式形成 IPv6 地址

操作	命令	说明
进入系统视图	**system-view**	-
进入接口视图	**interface** *interface-type interface-number*	-
采用EUI-64格式形成IPv6地址	**ipv6 address** { *ipv6-address prefix-length* | *ipv6-address*/*prefix-length* } **eui-64**	缺省情况下，接口上未配置IPv6全球单播地址

2. 手工指定IPv6 地址

表1-6 手工指定 IPv6 地址

操作	命令	说明
进入系统视图	**system-view**	-
进入接口视图	**interface** *interface-type interface-number*	-
手工指定IPv6地址	**ipv6 address** { *ipv6-address prefix-length* | *ipv6-address*/*prefix-length* }	缺省情况下，接口上未配置IPv6全球单播地址

3. 无状态自动配置IPv6 地址

表1-7 无状态自动配置 IPv6 地址

操作	命令	说明
进入系统视图	**system-view**	-
进入接口视图	**interface** *interface-type interface-number*	-
开启无状态地址自动配置功能，使接口通过无状态自动配置方式生成全球单播地址	**ipv6 address auto**	缺省情况下，接口上无状态地址自动配置功能处于关闭状态 在接口上执行**undo ipv6 address auto**命令，将删除该接口上所有自动生成的全球单播地址和链路本地地址

在配置了无状态自动配置 IPv6 地址功能后，接口会根据接收到的 RA 报文中携带的地址前缀信息和接口 ID，自动生成 IPv6 全球单播地址。如果接口是 IEEE 802 类型的接口（例如，以太网接口、VLAN 接口），其接口 ID 是由 MAC 地址根据一定的规则生成，此接口 ID 具有全球唯一性。对于不同的前缀，接口 ID 部分始终不变，攻击者通过接口 ID 可以很方便的识别出通信流量是由哪台设备产生的，并分析其规律，会造成一定的安全隐患。

如果在地址无状态自动配置时，自动生成接口 ID 不断变化的 IPv6 地址，就可以加大攻击的难度，从而保护网络。为此，设备提供了临时地址功能，使得系统可以生成临时地址。配置该功能后，通过地址无状态自动配置，IEEE 802 类型的接口可以同时生成两类地址：

- 公共地址：地址前缀采用 RA 报文携带的前缀，接口 ID 由 MAC 地址产生。接口 ID 始终不变。
- 临时地址：地址前缀采用 RA 报文携带的前缀，接口 ID 由系统根据 MD5 算法计算产生。接口 ID 不断变化。

在配置了优先选择临时地址功能前提下发送报文，系统将优先选择临时地址作为报文的源地址。当临时地址的有效生命期过期后，这个临时地址将被删除，同时，系统会通过 MD5 算法重新生成一个接口 ID 不同的临时地址。所以，该接口发送报文的源地址的接口 ID 总是在不停变化。如果生成的临时地址因为 DAD 冲突不可用，就采用公共地址作为报文的源地址。

临时地址的首选生命期和有效生命期的确定原则如下：

- 首选生命期是如下两个值之中的较小者：“RA 前缀中的首选生命期”和“配置的临时地址首选生命期减去 DESYNC_FACTOR”。DESYNC_FACTOR 是一个 0～600 秒的随机值。
- 有效生命期是如下两个值之中的较小者：“RA 前缀中的有效生命期”和“配置的临时地址有效生命期”。

表1-8 配置系统生成临时地址，并优先选择临时地址作为报文的源地址

操作	命令	说明
进入系统视图	**system-view**	-
配置系统生成临时地址	**ipv6 temporary-address** [*valid-lifetime preferred-lifetime*]	缺省情况下，系统不生成临时地址
优先选择临时地址作为报文的源地址	**ipv6 prefer temporary-address**	缺省情况下，不会用临时地址作为接口发送报文的源地址

设备的接口必须启用地址无状态自动配置功能才能生成临时地址，而且临时地址不会覆盖公共地址，因此会出现一个接口下有多个前缀相同但是接口 ID 不同的地址。

如果公共地址生成失败，例如前缀冲突，则不会生成临时地址。

4. 引用前缀生成接口上的IPv6 地址，并将此前缀分配给终端设备

进行配置前，需要先通过以下方法创建用来引用的 IPv6 前缀：

- 通过 **ipv6 prefix** 命令手工创建静态 IPv6 前缀。
- 配置设备作为 DHCPv6 客户端动态获取 IPv6 前缀，并根据获取到的前缀生成指定编号的 IPv6 前缀。详细介绍请参见“三层技术-IP 业务命令参考/DHCPv6”中的命令 **ipv6 dhcp client pd**。

表1-9 引用前缀生成接口上的 IPv6 地址，并将此前缀分配给终端设备

操作	命令	说明
进入系统视图	**system-view**	-
手工配置静态的IPv6前缀	**ipv6 prefix** *prefix-number ipv6-prefix/prefix-length*	二者必选其一 缺省情况下，不存在IPv6前缀
配置设备作为DHCPv6客户端动态获取IPv6前缀，并生成指定编号的IPv6前缀	配置方法请参见“三层技术-IP业务配置指导”中的“DHCPv6客户端”	
进入接口视图	**interface** *interface-type interface-number*	-
引用前缀生成接口上的IPv6地址，	**ipv6 address** *prefix-number*	缺省情况下，接口上未引用前缀，

操作	命令	说明
并将此前缀分配给终端设备	*sub-prefix/prefix-length*	也不会向终端设备分配该前缀

1.3.2 配置IPv6 链路本地地址

IPv6 的链路本地地址可以通过两种方式获得：

- 自动生成：设备根据链路本地地址前缀（FE80::/10）及接口的链路层地址，自动为接口生成链路本地地址；
- 手工指定：用户手工配置 IPv6 链路本地地址。

每个接口只能有一个链路本地地址，为了避免链路本地地址冲突，推荐使用链路本地地址的自动生成方式。

配置链路本地地址时，手工指定方式的优先级高于自动生成方式。即如果先采用自动生成方式，之后手工指定，则手工指定的地址会覆盖自动生成的地址；如果先手工指定，之后采用自动生成的方式，则自动配置不生效，接口的链路本地地址仍是手工指定的。此时，如果删除手工指定的地址，则自动生成的链路本地地址会生效。

表1-10 配置自动生成链路本地地址

操作	命令	说明
进入系统视图	**system-view**	-
进入接口视图	**interface** *interface-type interface-number*	-
配置自动生成链路本地地址	**ipv6 address auto link-local**	缺省情况下，接口上没有链路本地地址。当接口配置了IPv6全球单播地址后，会自动生成链路本地地址

表1-11 手工指定接口的链路本地地址

操作	命令	说明
进入系统视图	**system-view**	-
进入接口视图	**interface** *interface-type interface-number*	-
手工指定接口的链路本地地址	**ipv6 address** *ipv6-address* **link-local**	缺省情况下，未指定接口的链路本地地址

当接口配置了 IPv6 全球单播地址后，同时会自动生成链路本地地址。且与采用 **ipv6 address auto link-local** 命令生成的链路本地地址相同。此时如果手工指定接口的链路本地地址，则手工指定的有效。如果删除手工指定的链路本地地址，则接口的链路本地地址恢复为系统自动生成的地址。

undo ipv6 address auto link-local 命令只能删除使用 **ipv6 address auto link-local** 命令生成的链路本地地址。即如果此时已经配置了 IPv6 全球单播地址，由于系统会自动生成链路本地地址，则接口仍有链路本地地址；如果此时没有配置 IPv6 全球单播地址，则接口没有链路本地地址。

1.3.3 配置IPv6 任播地址

用户需要手工配置接口的 IPv6 任播地址。

表1-12 配置 IPv6 任播地址

操作	命令	说明
进入系统视图	**system-view**	-
进入接口视图	**interface** *interface-type interface-number*	-
配置IPv6任播地址	**ipv6 address** { *ipv6-address prefix-length* | *ipv6-address/prefix-length* } **anycast**	缺省情况下，接口上未配置任播地址

1.4 配置IPv6邻居发现协议

1.4.1 配置静态邻居表项

邻居表项保存的是设备在链路范围内的邻居信息，设备邻居表项可以通过邻居请求消息 NS 及邻居通告消息 NA 来动态创建，也可以通过手工配置来静态创建。

设备根据邻居节点的 IPv6 地址和与此邻居节点相连的三层接口号来唯一标识一个静态邻居表项。目前，静态邻居表项有两种配置方式：

- 配置本节点的三层接口相连的邻居节点的 IPv6 地址和链路层地址；
- 配置本节点 VLAN 中的二层端口相连的邻居节点的 IPv6 地址和链路层地址。

表1-13 配置静态邻居表项

操作	命令	说明
进入系统视图	**system-view**	-
配置静态邻居表项	**ipv6 neighbor** *ipv6-address mac-address* { *vlan-id port-type port-number* | **interface** *interface-type interface-number* } [**vpn-instance** *vpn-instance-name*]	缺省情况下，不存在静态邻居表项

对于 VLAN 接口，可以采用上述两种方式来配置静态邻居表项：

- 采用第一种方式配置静态邻居表项后，设备还需要解析该 VLAN 下的二层端口信息。
- 采用第二种方式配置静态邻居表项后，需要保证 *port-type port-number* 指定的二层端口属于 *vlan-id* 指定的 VLAN，且该 VLAN 已经创建了 VLAN 接口。在配置后，设备会将 VLAN 所对应的 VLAN 接口与 IPv6 地址相对应来唯一标识一个静态邻居表项。

当以太网冗余接口的成员接口包含子接口时，不能指定该以太网冗余接口为 IPv6 静态邻居表项所对应的接口。关于以太网冗余接口的详细介绍，请参见“可靠性配置指导”中的“冗余备份”。

1.4.2 配置接口上允许动态学习的邻居的最大个数

设备可以通过 NS 消息和 NA 消息来动态获取邻居节点的链路层地址，并将其加入到邻居表中。为了防止部分接口下的用户占用过多的资源，可以通过设置接口学习动态邻居表项的最大个数来进行

限制。当接口学习到的动态邻居表项的个数达到所设置的最大值时，该接口将不再学习动态邻居表项。

表1-14 配置接口上允许学习的动态邻居表项的最大个数

操作	命令	说明
进入系统视图	**system-view**	-
进入接口视图	**interface** *interface-type interface-number*	-
配置接口上允许学习的动态邻居表项的最大个数	**ipv6 neighbors max-learning-num** *max-number*	缺省情况下，接口上允许学习的动态邻居表项的最大个数为65536

1.4.3 配置STALE状态ND表项的老化时间

为适应网络的变化，ND 表需要不断更新。在 ND 表中，处于 STALE 状态的 ND 表项并非永远有效，而是有一个老化时间。到达老化时间的 STALE 状态 ND 表项将迁移到 DELAY 状态。5 秒钟后 DELAY 状态超时，ND 表项将迁移到 PROBE 状态，并且设备会发送 3 次 NS 报文进行可达性探测。若邻居已经下线，则收不到回应的 NA 报文，此时设备会将该 ND 表项删除。用户可以根据网络实际情况调整老化时间。

表1-15 配置 STALE 状态 ND 表项的老化时间

操作	命令	说明
进入系统视图	**system-view**	-
配置STALE状态ND表项的老化时间	**ipv6 neighbor stale-aging** *aging-time*	缺省情况下，STALE状态ND表项的老化时间为240分钟

1.4.4 配置链路本地ND表项资源占用最小化

本功能可以对链路本地 ND 表项（该 ND 表项的 IPv6 地址为链路本地地址）占用的资源进行优化。缺省情况下，所有 ND 表项均会下发硬件表项。配置本功能后，新学习的、未被引用的链路本地 ND 表项（该 ND 表项的链路本地地址不是某条路由的下一跳）不下发硬件表项，以节省资源。

本功能只对后续新学习的 ND 表项生效，已经存在的 ND 表项不受影响。

表1-16 配置链路本地 ND 表项资源占用最小化

操作	命令	说明
进入系统视图	**system-view**	-
配置链路本地ND表项资源占用最小化	**ipv6 neighbor link-local minimize**	缺省情况下，所有ND表项均会下发硬件表项

1.4.5 配置设备的跳数限制

设备的跳数限制有以下两个作用：

- 决定了设备发送的 IPv6 数据报文的跳数，即 IPv6 数据报文的 Hop Limit 字段的值。
- 如果用户配置了在 RA 消息中发布本设备的跳数限制（配置命令 **undo ipv6 nd ra hop-limit unspecified**），则设备发送的 RA 消息中将携带此处配置的跳数限制值。收到该 RA 消息之后，主机在发送 IPv6 报文时，将使用该跳数值填充 IPv6 报文头中的 Hop Limit 字段。

表1-17 配置设备的跳数限制

操作	命令	说明
进入系统视图	**system-view**	-
配置设备的跳数限制	**ipv6 hop-limit** *value*	缺省情况下，设备的跳数限制为64跳

1.4.6 配置RA消息的相关参数

用户可以根据实际情况，配置接口是否发送RA消息及发送RA消息的时间间隔，同时可以配置RA消息中的相关参数以通告给主机。当主机接收到RA消息后，就可以采用这些参数进行相应操作。可以配置的RA消息中的参数及含义如 表 1-18 所示。

表1-18 RA 消息中的参数及描述

参数	描述
跳数限制（Hop Limit）	在RA消息中发布本设备的跳数限制，收到该RA消息之后，主机在发送IPv6报文时，将使用该跳数值填充IPv6报文头中的Hop Limit字段。
前缀信息（Prefix Information）	在同一链路上的主机收到设备发布的前缀信息后，可以进行无状态自动配置等操作。
MTU	发布链路的MTU，可以用于确保同一链路上的所有节点采用相同的MTU值。
被管理地址配置标志位（M flag）	用于确定主机是否采用有状态自动配置获取IPv6地址。 如果设置该标志位为1，主机将通过有状态自动配置（例如DHCPv6服务器）来获取IPv6地址；否则，将通过无状态自动配置获取IPv6地址，即根据自己的链路层地址及路由器发布的前缀信息生成IPv6地址。
其他信息配置标志位（O flag）	用于确定主机是否采用有状态自动配置获取除IPv6地址外的其他信息。 如果设置其他信息配置标志位为1，主机将通过有状态自动配置（例如DHCPv6服务器）来获取除IPv6地址外的其他信息；否则，将通过无状态自动配置获取其他信息。
路由器生存时间（Router Lifetime）	用于设置发布RA消息的路由器作为主机的默认路由器的时间。主机根据接收到的RA消息中的路由器生存时间参数值，就可以确定是否将发布该RA消息的路由器作为默认路由器。发布RA消息中路由器生存时间为0的路由器不能作为默认路由器。
邻居请求消息重传时间间隔（Retrans Timer）	设备发送NS消息后，如果未在指定的时间间隔内收到响应，则会重新发送NS消息。
保持邻居可达状态的时间（Reachable Time）	当通过邻居可达性检测确认邻居可达后，在所设置的可达时间内，设备认为邻居可达；超过设置的时间后，如果需要向邻居发送报文，会重新确认邻居是否可达。

参数	描述
配置路由优先级 (Router Preference)	用于设置发布RA消息的路由器的路由器优先级，主机根据接收到的RA消息中的路由器优先级，可以选择优先级最高的路由器作为默认网关。在路由器的优先级相同的情况下，遵循“先来先用”的原则，优先选择先接收到的RA消息对应的发送路由器作为默认网关。

表1-19 配置允许发布 RA 消息

操作	命令	说明
进入系统视图	**system-view**	-
进入接口视图	**interface** *interface-type interface-number*	-
取消对RA消息发布的抑制	**undo ipv6 nd ra halt**	缺省情况下，抑制发布RA消息
配置RA消息发布的最大时间间隔和最小时间间隔	**ipv6 nd ra interval** *max-interval-value min-interval-value*	缺省情况下，RA消息发布的最大间隔时间为600秒，最小时间间隔为200秒 RA消息周期性发布时，相邻两次的时间间隔是在最大时间间隔与最小时间间隔之间随机选取一个值作为周期性发布RA消息的时间间隔 配置的最小时间间隔应该小于等于最大时间间隔的0.75倍

表1-20 配置 RA 消息中的相关参数

操作	命令	说明
进入系统视图	**system-view**	-
进入接口视图	**interface** *interface-type interface-number*	-
配置RA消息中的前缀信息	**ipv6 nd ra prefix** { *ipv6-prefix prefix-length* | *ipv6-prefixIprefix-length* } *valid-lifetime preferred-lifetime* [**no-autoconfig** | **off-link**] *	缺省情况下，未配置RA消息中的前缀信息，此时将使用发送RA消息的接口IPv6地址作为RA消息中的前缀信息，其手工配置地址的有效生命期是2592000秒（30天），首选生命期是604800（7天）；其他自动分配地址（如DHCPv6分配地址）的有效生命期和首选生命期与地址本身的生命期相同
配置RA消息中不携带MTU选项	**ipv6 nd ra no-advlinkmtu**	缺省情况下，RA消息中携带MTU选项
配置RA消息中不指定跳数限制	**ipv6 nd ra hop-limit unspecified**	缺省情况下，RA消息中发布本设备的跳数限制。本设备的跳数限制默认为64跳
设置被管理地址配置标志位为1	**ipv6 nd autoconfig managed-address-flag**	缺省情况下，被管理地址标志位为0，即主机通过无状态自动配置获取IPv6地址
设置其他配置标志位为1	**ipv6 nd autoconfig other-flag**	缺省情况下，其他配置标志位为0，即主机通过无状态自动配置获取其他信息
配置RA消息中路由器的生存时间	**ipv6 nd ra router-lifetime** *time*	缺省情况下，RA消息中路由器的生存时间为1800秒

操作	命令	说明
配置邻居请求消息重传时间间隔	**ipv6 nd ns retrans-timer** *value*	缺省情况下，接口发送NS消息的时间间隔为1000毫秒；接口发布的RA消息中Retrans Timer字段的值为0，即不对主机进行指定
配置RA消息中路由器的优先级	**ipv6 nd router-preference { high | low | medium }**	缺省情况下，RA消息中路由器的优先级为**medium**
配置保持邻居可达状态的时间	**ipv6 nd nud reachable-time** *time*	缺省情况下，接口保持邻居可达状态的时间为30000毫秒；接口发布的RA消息中Reachable Timer字段的值为0，即不对主机进行指定

RA 消息发布的最大间隔时间应该小于或等于 RA 消息中路由器的生存时间，以保证在路由器失效之前得到更新的 RA 消息。

在接口上配置的邻居请求消息重传时间间隔及保持邻居可达状态的时间，既可作为 RA 消息中的信息发布给主机，也可作为本接口发送邻居请求消息的时间间隔及保持邻居可达状态的时间。

1.4.7 配置重复地址检测时发送邻居请求消息的次数

接口获得 IPv6 地址后，将发送邻居请求消息进行重复地址检测。如果在指定的时间内（通过 **ipv6 nd ns retrans-timer** 命令配置）没有收到响应，则继续发送邻居请求消息，当发送的次数达到所设置的次数后，仍未收到响应，则认为该地址可用。

表1-21 配置重复地址检测时发送邻居请求消息的次数

操作	命令	说明
进入系统视图	**system-view**	-
进入接口视图	**interface** *interface-type interface-number*	-
配置重复地址检测时发送邻居请求消息的次数	**ipv6 nd dad attempts** *interval*	缺省情况下，重复地址检测时发送邻居请求报文的次数为1，当*interval*值为0时，表示禁止重复地址检测

1.4.8 配置ND Proxy功能

1. ND Proxy功能简介

如果 NS 请求是从一个网络的主机发往同一网段却不在同一物理网络上的另一台主机，那么连接它们的具有代理功能的设备就可以代答该请求，回应 NA 报文，这个过程称作 ND 代理（ND Proxy）。

ND Proxy 功能屏蔽了分离的物理网络这一事实，使用户使用起来，好像在同一个物理网络上。

ND Proxy 功能根据应用场景不同分为普通 ND Proxy 和本地 ND Proxy。

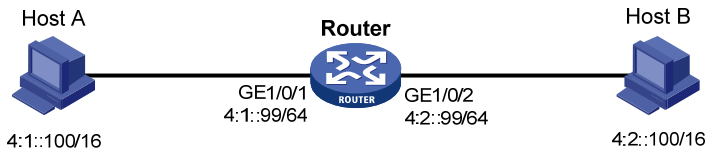
说明

如无特殊说明，本章后续描述中的 ND Proxy 均指普通 ND Proxy。

(1) ND Proxy

ND Proxy的典型应用环境如 图 1-7 所示。设备Router通过两个三层接口GigabitEthernet1/0/1 和GigabitEthernet1/0/2 连接两个网络，两个三层接口的IPv6 地址不在同一个网段，接口地址分别为4:1::99/64、4:2::99/64。但是两个网络内的主机Host A和Host B的地址通过掩码的控制，既与相连设备的接口地址在同一网段，同时二者也处于同一个网段。

图1-7 ND Proxy 的应用环境



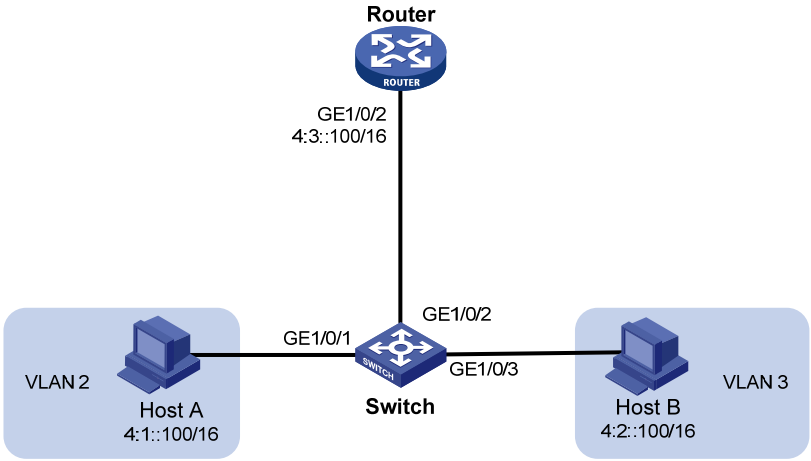
在这种组网情况下，当 Host A 需要与 Host B 通信时，由于目的 IPv6 地址与本机的 IPv6 地址为同一网段，因此 Host A 会直接发出请求 Host B 硬件地址的 NS 请求。但是，此时的两台主机处于不同的广播域中，Host B 无法收到 Host A 的 NS 请求报文，当然也就无法应答。

通过在 Router 上启用 ND Proxy 功能，可以解决此问题。在接口 GigabitEthernet1/0/1 和GigabitEthernet1/0/2 上启用 ND Proxy 后，Router 可以应答 Host A 的 NS 请求。同时，Router 作为 Host B 的代理，把其它主机发送过来的报文转发给 Host B。这样，实现 Host A 与 Host B 之间的通信。

(2) 本地 ND Proxy

本地ND Proxy的应用场景如 图 1-8 所示。Host A属于VLAN 2，Host B属于VLAN 3。但它们分别连接到端口GigabitEthernet1/0/1 和GigabitEthernet1/0/3 上。

图1-8 本地 ND Proxy 的应用环境



在这种组网情况下，当 Host A 需要与 Host B 通信时，由于目的 IPv6 地址与本机的 IPv6 地址为同一网段，因此 Host A 会直接发出请求 Host B 硬件地址的 NS 请求。但是，因为连接两台主机属于不同的 VLAN 中，Host B 无法收到 Host A 的 NS 请求报文。

通过在 Router 上启用本地 ND Proxy 功能，可以解决此问题。在接口 GigabitEthernet1/0/2 上启用本地 ND Proxy 后，Router 会代替 Host B 回应 NA，Host A 发给 Host B 的报文就会通过 Router 进行转发，从而实现 Host A 与 Host B 之间的通信。

本地 ND Proxy 可以在下列四种情况下实现主机之间的三层互通：

- 想要互通的主机分别连接到同一台设备的不同 VLAN 中的端口下；
- 想要互通的主机分别连接到同一个 VLAN 中的同一个隔离组内的不同二层隔离端口下；

2. 配置ND Proxy功能

ND Proxy 和本地 ND Proxy 功能均可在 VLAN 接口视图/三层以太网接口/三层以太网子接口视图下进行配置。

表1-22 配置 ND Proxy 功能

操作	命令	说明
进入系统视图	**system-view**	-
进入接口视图	**interface** *interface-type interface-number*	-
开启ND Proxy功能	**proxy-nd enable**	缺省情况下，ND Proxy功能处于关闭状态

表1-23 配置本地 ND Proxy 功能

操作	命令	说明
进入系统视图	**system-view**	-
进入接口视图	**interface** *interface-type interface-number*	-
开启本地ND Proxy功能	**local-proxy-nd enable**	缺省情况下，本地ND Proxy功能处于关闭状态

1.5 配置PMTU发现

1.5.1 配置接口MTU

由于 IPv6 路由器不支持对报文进行分片，当路由器接口收到一个报文后，如果发现报文长度比转发接口的 MTU 值大，则会将其丢弃；同时将转发接口的 MTU 值通过 ICMPv6 报文的“Packet Too Big”消息发给源端主机，源端主机以该值重新发送 IPv6 报文。为减少报文被丢弃带来的额外流量开销，需要根据实际组网环境设置合适的接口 MTU 值。

表1-24 配置接口 MTU

操作	命令	说明
进入系统视图	**system-view**	-
进入接口视图	**interface** *interface-type interface-number*	-
配置接口MTU	**ipv6 mtu** *size*	缺省情况下，未配置接口上发送IPv6报文的MTU

1.5.2 配置指定地址的静态PMTU

用户可以为指定的目的IPv6 地址配置静态的PMTU值。当设备作为源端从接口发送报文时，将比较该接口的MTU与指定目的IPv6 地址的静态PMTU，如果报文长度大于二者中的最小值，则采用此最小值对报文进行分片发送。发送过程中再通过“1.1.4 IPv6 PMTU发现”中的方法动态确定设备作为源端到目的端主机的PMTU值。

表1-25 配置指定地址的静态 PMTU

操作	命令	说明
进入系统视图	**system-view**	-
配置指定IPv6地址对应的静态PMTU值	**ipv6 pathmtu** [**vpn-instance** *vpn-instance-name*] *ipv6-address value*	缺省情况下，未配置静态PMTU值

1.5.3 配置PMTU老化时间

通过“1.1.4 IPv6 PMTU发现”中的方法动态确定设备作为源端到目的端主机的PMTU后，设备将使用这个MTU值发送后续报文到目的端主机。当PMTU老化时间超时后，源端主机会通过PMTU机制重新确定发送报文的MTU值。

该配置对静态 PMTU 不起作用。

表1-26 配置 PMTU 老化时间

操作	命令	说明
进入系统视图	**system-view**	-
配置PMTU老化时间	**ipv6 pathmtu age** *age-time*	缺省情况下，PMTU的老化时间是10分钟

1.6 配置ICMPv6报文发送功能

1.6.1 配置发送ICMPv6 差错报文对应的令牌桶容量和令牌刷新周期

如果网络中短时间内发送的 ICMPv6 差错报文过多，将可能导致网络拥塞。为了避免这种情况，用户可以控制在指定时间内发送 ICMPv6 差错报文的最大个数，目前采用令牌桶算法来实现。

用户可以设置令牌桶的容量，即令牌桶中可以同时容纳的令牌数；同时可以设置令牌桶的刷新周期，即每隔多长时间发放一个令牌到令牌桶中，直到令牌桶中的令牌数达到配置的容量。一个令牌表示允许发送一个 ICMPv6 差错报文，每当发送一个 ICMPv6 差错报文，则令牌桶中减少一个令牌。如果连续发送的 ICMPv6 差错报文超过了令牌桶的容量，则后续的 ICMPv6 差错报文将不能被发送出去，直到按照所设置的刷新频率将新的令牌放入令牌桶中。

表1-27 配置发送 ICMPv6 差错报文对应的令牌桶容量和令牌刷新周期

操作	命令	说明
进入系统视图	**system-view**	-

操作	命令	说明
配置发送ICMPv6差错报文对应的令牌桶容量和令牌刷新周期	**ipv6 icmpv6 error-interval** *interval* [*bucketsize*]	缺省情况下，令牌桶容量为10，令牌刷新周期为100毫秒 刷新周期为0时，表示不限制ICMPv6差错报文的发送

1.6.2 配置允许回复组播形式的Echo request报文

缺省情况下，不允许设备回复组播形式的 Echo request 报文。

在某些应用场景下，可能需要使用组播形式的 Echo request 报文来获取信息，此时可以通过下面的命令，配置允许设备回复组播形式的 Echo request 报文。

表1-28 配置允许回复组播形式的 Echo request 报文

操作	命令	说明
进入系统视图	**system-view**	-
配置设备允许回复组播形式的Echo request报文	**ipv6 icmpv6 multicast-echo-reply enable**	缺省情况下，不允许设备回复组播形式的Echo request报文

1.6.3 配置ICMPv6 目的不可达差错报文发送功能

ICMPv6 目的不可达报文发送功能是在设备收到 IPv6 数据报文后，如果发生目的不可达的差错，则将报文丢弃并给源端发送 ICMPv6 目的不可达差错报文。

设备在满足下列任一条件时会发送目的不可达报文：

- 设备在转发报文时，如果在路由表中没有找到对应的转发路由，且路由表中没有缺省路由，则给源端发送“没有到达目的地址的路由”ICMPv6 差错报文；
- 设备在转发报文时，如果是因为管理策略（例如防火墙过滤、ACL 等）导致无法发送报文时，则给源端发送“与目的地址的通信被管理策略禁止”ICMPv6 差错报文；
- 设备在转发报文时，如果报文的目的 IPv6 地址超出源 IPv6 地址的范围（例如，报文的源 IPv6 地址为链路本地地址，报文的目的 IPv6 地址为全球单播地址），会导致报文无法到达目的端，此时要给源端发送“超出源地址范围”ICMPv6 差错报文；
- 设备在转发报文时，如果不能解析目的 IPv6 地址对应的链路层地址，则给源端发送“地址不可达”ICMPv6 差错报文；
- 设备收到目的地址为本地、传输层协议为 UDP 的数据报文时，如果报文的目的端口号与正在使用的进程不匹配，则给源端发送“端口不可达”ICMPv6 差错报文。

由于 ICMPv6 目的不可达报文传递给用户进程的信息为不可达信息，如果有用户恶意攻击，可能会影响终端用户的正常使用。为了避免上述现象发生，可以关闭设备的 ICMPv6 目的不可达报文发送功能，从而减少网络流量、防止遭到恶意攻击。

表1-29 配置 ICMPv6 目的不可达报文发送功能

操作	命令	说明
进入系统视图	**system-view**	-
开启设备的ICMPv6目的不可达报文的发送功能	**ipv6 unreachables enable**	缺省情况下，ICMPv6目的不可达报文发送功能处于关闭状态

1.6.4 配置ICMPv6 超时差错报文发送功能

ICMPv6 超时报文发送功能是在设备收到 IPv6 数据报文后，如果发生超时差错，则将报文丢弃并给源端发送 ICMPv6 超时差错报文。

设备在满足下列任一条件时会发送 ICMPv6 超时报文：

- 设备收到 IPv6 数据报文后，如果报文的目的地不是本地且报文的 Hop limit 字段是 1，则发送“Hop limit 超时”ICMPv6 差错报文；
- 设备收到目的地址为本地的 IPv6 数据报文的第一个分片后，启动定时器，如果所有分片报文到达之前定时器超时，则会发送“重组超时”ICMPv6 差错报文。

如果接收到大量需要发送 ICMPv6 差错报文的恶意攻击报文，设备会因为处理大量该类报文而导致性能降低。

为了避免上述现象发生，可以关闭设备的 ICMPv6 超时报文发送功能，从而减少网络流量、防止遭到恶意攻击。

表1-30 配置 ICMPv6 超时差错报文发送功能

操作	命令	说明
进入系统视图	**system-view**	-
开启设备的ICMPv6超时报文的发送功能	**ipv6 hoplimit-expires enable**	缺省情况下，ICMPv6超时报文发送功能处于开启状态

1.6.5 配置ICMPv6 重定向报文发送功能

当主机启动时，它的路由表中可能只有一条到缺省网关的缺省路由。当满足一定的条件时，缺省网关会向源主机发送 ICMPv6 重定向报文，通知主机重新选择更好的下一跳进行后续报文的发送。

同时满足下列条件时，设备会发送 ICMPv6 重定向报文：

- 接收和转发数据报文的接口是同一接口；
- 被选择的路由本身没有被 ICMPv6 重定向报文创建或修改过；
- 被选择的路由不是设备的缺省路由；
- 被转发的 IPv6 数据报文中不包含路由扩展头。

ICMPv6 重定向报文发送功能可以简化主机的管理，使具有很少选路信息的主机逐渐建立较完善的路由表，从而找到最佳路由。但是由于重定向功能会在主机的路由表中增加主机路由，当增加的主机路由很多时，会降低主机性能。因此缺省情况下设备的 ICMPv6 重定向报文发送功能处于关闭状态。

表1-31 配置 ICMPv6 重定向报文发送功能

操作	命令	说明
进入系统视图	**system-view**	-
开启设备的ICMPv6重定向报文发送功能	**ipv6 redirects enable**	缺省情况下，ICMPv6重定向报文发送功能处于关闭状态

1.6.6 配置ICMPv6 报文指定源地址功能

在网络中 IPv6 地址配置较多的情况下，收到 ICMPv6 报文时，用户很难根据报文的源 IPv6 地址判断报文来自哪台设备。为了简化这一判断过程，可以配置 ICMPv6 报文指定源地址功能。用可配置特定地址（如环回口地址）为 ICMPv6 报文的源地址，可以简化判断。

设备发送 ICMPv6 差错报文（TTL 超时、报文过大、端口不可达和参数错误等）和 ping echo request 报文时，都可以通过上述命令指定报文的源地址。

表1-32 配置 ICMPv6 报文指定源地址功能

操作	命令	说明
进入系统视图	**system-view**	-
开启ICMPv6报文指定源地址功能	**ipv6 icmpv6 source** [**vpn-instance** *vpn-instance-name*] *ipv6-address*	缺省情况下，ICMPv6报文指定源地址功能处于关闭状态

说明

用户发送 ping echo request 报文时，如果 **ping** 命令中已经指定源地址，则使用该源地址，否则使用 **ipv6 icmpv6 source** 配置的源地址。

1.7 配置IPv6分片报文本地重组功能

当分布式设备的某块单板收到目的为本设备的 IPv6 分片报文时，需要把分片报文送到主用主控板进行重组，这样会导致报文重组性能较低的问题。当开启 IPv6 分片报文本地重组功能后，分片报文会在该单板上直接进行报文重组，这样就能提高分片报文的重组性能。

开启 IPv6 分片报文本地重组功能后，如果分片报文是从设备上不同的单板进入的，会导致 IPv6 分片报文本地无法重组成功。

多台设备组成的 IRF 环境下，当某成员设备收到目的为本 IRF 设备的 IPv6 分片报文时，需要把分片报文送到主设备进行重组，这样会导致报文重组性能较低的问题。当开启 IPv6 分片报文本地重组功能后，分片报文会在该成员设备上直接进行报文重组，这样就能提高分片报文的重组性能。

开启 IPv6 分片报文本地重组功能后，如果分片报文是从设备上不同的成员设备进入的，会导致 IPv6 分片报文本地无法重组成功。

表1-33 配置 IPv6 分片报文本地重组功能

操作	命令	说明
进入系统视图	**system-view**	-
开启IPv6分片报文本地重组功能	**ipv6 reassemble local enable**	缺省情况下，IPv6分片报文本地重组功能处于关闭状态

1.8 开启IPv6报文扩展头丢弃功能

IPv6 协议引入了多种扩展报文头，开启 IPv6 扩展报文丢弃功能后，如果接收到无法处理的 IPv6 扩展头的报文，设备将直接丢弃。

表1-34 开启 IPv6 报文扩展头丢弃功能

操作	命令	说明
进入系统视图	**system-view**	-
开启IPv6报文扩展头丢弃功能	**ipv6 extension-header drop enable**	缺省情况下，IPv6报文扩展头丢弃功能处于关闭状态

1.9 IPv6基础显示和维护

在完成上述配置后，在任意视图下执行 **display** 命令可以显示 IPv6 配置后的运行情况，用户可以通过查看显示信息验证配置的效果。

在用户视图下，执行 **reset** 命令可以清除相应的统计信息。

表1-35 IPv6 基础显示和维护

操作	命令
显示IPv6 FIB信息	**display ipv6 fib** [**vpn-instance** *vpn-instance-name*] [*ipv6-address* [*prefix-length*]]
显示接口的IPv6信息	**display ipv6 interface** [*interface-type* [*interface-number*]] [**brief**]
显示接口的IPv6前缀信息	**display ipv6 interface** *interface-type interface-number* **prefix**
显示邻居信息（分布式设备－独立运行模式）	**display ipv6 neighbors** { { *ipv6-address* | **all** | **dynamic** | **static** } [**slot** *slot-number* [**cpu** *cpu-number*]] | **interface** *interface-type interface-number* | **vlan** *vlan-id* } [**verbose**]
显示邻居信息（分布式设备－IRF模式）	**display ipv6 neighbors** { { *ipv6-address* | **all** | **dynamic** | **static** } [**chassis** *chassis-number* **slot** *slot-number* [**cpu** *cpu-number*]] | **interface** *interface-type interface-number* | **vlan** *vlan-id* } [**verbose**]
显示邻居表项的个数（分布式设备－独立运行模式）	**display ipv6 neighbors** { { **all** | **dynamic** | **static** } [**slot** *slot-number* [**cpu** *cpu-number*]] | **interface** *interface-type interface-number* | **vlan** *vlan-id* } **count**
显示邻居表项的个数（分布式设备－IRF模式）	**display ipv6 neighbors** { { **all** | **dynamic** | **static** } [**chassis** *chassis-number* **slot** *slot-number* [**cpu** *cpu-number*]] | **interface**

操作	命令
	interface-type interface-number | **vlan** *vlan-id* } **count**
显示指定VPN实例的邻居信息	**display ipv6 neighbors vpn-instance** *vpn-instance-name* [**count**]
显示IPv6的PMTU信息	**display ipv6 pathmtu** [**vpn-instance** *vpn-instance-name*] { *ipv6-address* | { **all** | **dynamic** | **static** } [**count**] }
显示IPv6前缀信息	**display ipv6 prefix** [*prefix-number*]
显示IPv6报文及ICMPv6报文的统计信息（分布式设备－独立运行模式）	**display ipv6 statistics** [**slot** *slot-number* [**cpu** *cpu-number*]]
显示IPv6报文及ICMPv6报文的统计信息（分布式设备－IRF模式）	**display ipv6 statistics** [**chassis** *chassis-number* **slot** *slot-number* [**cpu** *cpu-number*]]
显示IPv6 RawIP连接摘要信息（分布式设备－独立运行模式）	**display ipv6 rawip** [**slot** *slot-number* [**cpu** *cpu-number*]]
显示IPv6 RawIP连接摘要信息（分布式设备－IRF模式）	**display ipv6 rawip** [**chassis** *chassis-number* **slot** *slot-number* [**cpu** *cpu-number*]]
显示IPv6 RawIP连接详细信息（分布式设备－独立运行模式）	**display ipv6 rawip verbose** [**slot** *slot-number* [**cpu** *cpu-number*] [**pcb** *pcb-index*]]
显示IPv6 RawIP连接详细信息（分布式设备－IRF模式）	**display ipv6 rawip verbose** [**chassis** *chassis-number* **slot** *slot-number* [**cpu** *cpu-number*] [**pcb** *pcb-index*]]
显示IPv6 TCP连接摘要信息（分布式设备－独立运行模式）	**display ipv6 tcp** [**slot** *slot-number* [**cpu** *cpu-number*]]
显示IPv6 TCP连接摘要信息（分布式设备－IRF模式）	**display ipv6 tcp** [**chassis** *chassis-number* **slot** *slot-number* [**cpu** *cpu-number*]]
显示IPv6 TCP代理连接的简要信息（分布式设备－独立运行模式）	**display ipv6 tcp-proxy slot** *slot-number* [**cpu** *cpu-number*]
显示IPv6 TCP代理连接的简要信息（分布式设备－IRF模式）	**display ipv6 tcp-proxy chassis** *chassis-number* **slot** *slot-number* [**cpu** *cpu-number*]
显示IPv6 TCP代理非保留端口的使用信息（分布式设备－独立运行模式）	**display ipv6 tcp-proxy port-info slot** *slot-number* [**cpu** *cpu-number*]
显示IPv6 TCP代理非保留端口的使用信息（分布式设备－IRF模式）	**display ipv6 tcp-proxy port-info chassis** *chassis-number* **slot** *slot-number* [**cpu** *cpu-number*]
显示IPv6 TCP连接详细信息（分布式设备－独立运行模式）	**display ipv6 tcp verbose** [**slot** *slot-number* [**cpu** *cpu-number*] [**pcb** *pcb-index*]]
显示IPv6 TCP连接详细信息（分布式设备－IRF模式）	**display ipv6 tcp verbose** [**chassis** *chassis-number* **slot** *slot-number* [**cpu** *cpu-number*] [**pcb** *pcb-index*]]
显示IPv6 UDP连接摘要信息（分布式设备－独立运行模式）	**display ipv6 udp** [**slot** *slot-number* [**cpu** *cpu-number*]]
显示IPv6 UDP连接摘要信息（分布式设备－IRF模式）	**display ipv6 udp** [**chassis** *chassis-number* **slot** *slot-number* [**cpu** *cpu-number*]]
显示IPv6 UDP连接详细信息（分布式设备－独立运行模式）	**display ipv6 udp verbose** [**slot** *slot-number* [**cpu** *cpu-number*] [**pcb** *pcb-index*]]
显示IPv6 UDP连接详细信息（分布式设备	**display ipv6 udp verbose** [**chassis** *chassis-number* **slot**

操作	命令
－IRF模式）	*slot-number* [**cpu** *cpu-number*] [**pcb** *pcb-index*]]
显示IPv6 ICMP流量统计信息（分布式设备－独立运行模式）	**display ipv6 icmp statistics** [**slot** *slot-number* [**cpu** *cpu-number*]]
显示IPv6 ICMP流量统计信息（分布式设备－IRF模式）	**display ipv6 icmp statistics** [**chassis** *chassis-number* **slot** *slot-number* [**cpu** *cpu-number*]]
显示IPv6 TCP连接的流量统计信息（分布式设备－独立运行模式）	**display tcp statistics** [**slot** *slot-number* [**cpu** *cpu-number*]]
显示IPv6 TCP连接的流量统计信息（分布式设备－IRF模式）	**display tcp statistics** [**chassis** *chassis-number* **slot** *slot-number* [**cpu** *cpu-number*]]
显示IPv6 UDP流量统计信息（分布式设备－独立运行模式）	**display udp statistics** [**slot** *slot-number* [**cpu** *cpu-number*]]
显示IPv6 UDP流量统计信息（分布式设备－IRF模式）	**display udp statistics** [**chassis** *chassis-number* **slot** *slot-number* [**cpu** *cpu-number*]]
清除IPv6邻居信息（分布式设备－独立运行模式）	**reset ipv6 neighbors** { **all** | **dynamic** | **interface** *interface-type interface-number* | **slot** *slot-number* | **static** }
清除IPv6邻居信息（分布式设备－IRF模式）	**reset ipv6 neighbors** { **all** | **dynamic** | **interface** *interface-type interface-number* | **chassis** *chassis-number* **slot** *slot-number* | **static** }
清除PMTU值	**reset ipv6 pathmtu** { **all** | **dynamic** | **static** }
清除IPv6报文及ICMPv6报文的统计信息（分布式设备－独立运行模式）	**reset ipv6 statistics** [**slot** *slot-number* [**cpu** *cpu-number*]]
清除IPv6报文及ICMPv6报文的统计信息（分布式设备－IRF模式）	**reset ipv6 statistics** [**chassis** *chassis-number* **slot** *slot-number* [**cpu** *cpu-number*]]
清除IPv6 TCP连接的流量统计信息	**reset tcp statistics**
清除IPv6 UDP流量统计信息	**reset udp statistics**

display tcp statistics、**display udp statistics**、**reset tcp statistics** 和 **reset udp statistics** 命令的详细介绍请参见“三层技术-IP 业务命令参考”中的“IP 性能优化”。

1.10 常见配置错误举例

1. 故障现象

无法 Ping 通对端的 IPv6 地址。

2. 故障排除

- 在任意视图下使用 **display ipv6 interface** 命令检查接口配置的 IPv6 地址是否正确，接口状态是否为 up。
- 在用户视图下使用 **debugging ipv6 packet** 命令打开 IPv6 报文调试开关，根据调试信息进行判断。