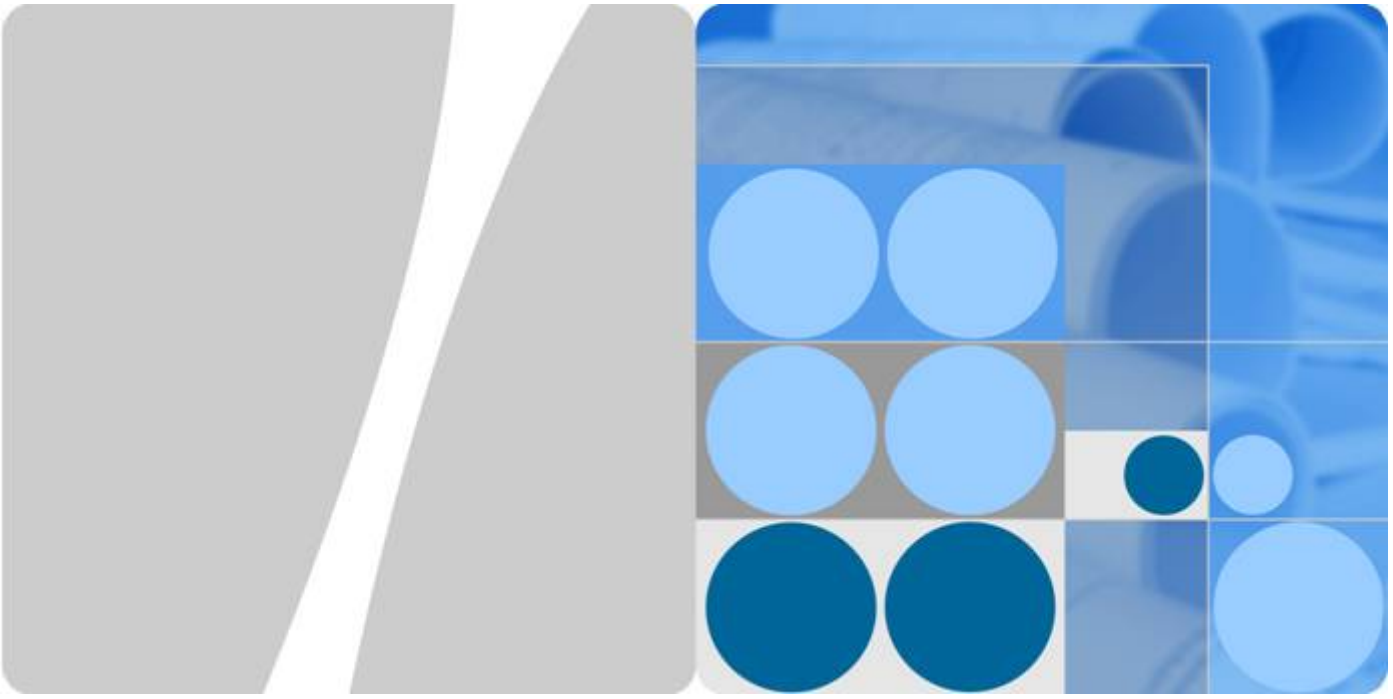


Huawei AR150&200 系列企业路由器

V200R002C00

配置指南-IP 路由

文档版本　02

发布日期　2012-03-30

华为技术有限公司

HUAWEI

商标声明


和其他华为商标均为华为技术有限公司的商标。
本文档提及的其他所有商标或注册商标，由各自的所有人拥有。

注意

您购买的产品、服务或特性等应受华为公司商业合同和条款的约束，本文档中描述的全部或部分产品、服务或特性可能不在您的购买或使用范围之内。除非合同另有约定，华为公司对本文档内容不做任何明示或默示的声明或保证。

由于产品版本升级或其他原因，本文档内容会不定期进行更新。除非另有约定，本文档仅作为使用指导，本文档中的所有陈述、信息和建议不构成任何明示或暗示的担保。

华为技术有限公司

地址： 深圳市龙岗区坂田华为总部办公楼 邮编：518129

网址： http://www.huawei.com

客户服务邮箱： support@huawei.com

客户服务电话： 4008302118

前 言

读者对象

本文档介绍了 AR150/200 的 IP 路由（静态路由、RIP、OSPF、IS-IS、路由策略）的基本概念、在不同应用场景中的配置过程和配置举例。

本文档提供了 IP 路由（静态路由、RIP、OSPF、IS-IS、路由策略）特性的配置方法。

本文档主要适用于以下工程师：

- 数据配置工程师
- 调测工程师
- 网络监控工程师
- 系统维护工程师

符号约定

在本文中可能出现下列标志，它们所代表的含义如下。

符号	说明
危险	以本标志开始的文本表示有高度潜在危险，如果不能避免，会导致人员死亡或严重伤害。
警告	以本标志开始的文本表示有中度或低度潜在危险，如果不能避免，可能导致人员轻微或中等伤害。
注意	以本标志开始的文本表示有潜在风险，如果忽视这些文本，可能导致设备损坏、数据丢失、设备性能降低或不可预知的结果。
窍门	以本标志开始的文本能帮助您解决某个问题或节省您的时间。
说明	以本标志开始的文本是正文的附加信息，是对正文的强调和补充。

命令行格式约定

格式	意义
粗体	命令行关键字（命令中保持不变、必须照输的部分）采用**加粗**字体表示。
斜体	命令行参数（命令中必须由实际值进行替代的部分）采用*斜体*表示。
[ ]	表示用“[ ]”括起来的部分在命令配置时是可选的。
{ x | y | ... }	表示从两个或多个选项中选取一个。
[x | y | ...]	表示从两个或多个选项中选取一个或者不选。
{ x | y | ... } *	表示从两个或多个选项中选取多个，最少选取一个，最多选取所有选项。
[x | y | ...] *	表示从两个或多个选项中选取多个或者不选。
&<1-n>	表示符号&的参数可以重复 1 ～ n 次。
#	由“#”开始的行表示为注释行。

接口编号约定

本手册中出现的接口编号仅作示例，并不代表设备上实际具有此编号的接口，实际使用中请以设备上存在的接口编号为准。

修订记录

修改记录累积了每次文档更新的说明。最新版本的文档包含以前所有文档版本的更新内容。

文档版本 02 (2012-03-30)

相对于版本 01 (2011-12-30)的变化如下：

修改：

- **7.3.2 创建 IS-IS 进程(IPv4)**

文档版本 01 (2011-12-30)

第一次正式发布。

目 录

1 IP 路由基础配置

关于本章

本章介绍数据通信网络的基本要素。

1.1 IP 路由管理
为了实现数据转发，路由器必须有能力建立、刷新路由表，并根据路由表转发数据包。

1.1 IP 路由管理

为了实现数据转发，路由器必须有能力建立、刷新路由表，并根据路由表转发数据包。

1.1.1 路由表的显示

通过查看路由表，有助于了解网络拓扑结构和定位问题。

背景信息

掌握路由表信息的查看是定位路由问题的基本要求，下面列举了通用的路由表信息显示命令。

display 命令可以在所有视图下使用。

操作步骤

- 使用 **display ip routing-table** 命令查看路由表中当前激活路由的摘要信息。
- 使用 **display ip routing-table verbose** 命令查看路由表详细信息。
- 使用 **display ip routing-table** *ip-address* [*mask* | *mask-length*] [**longer-match**] [**verbose**]命令查看指定目的地址的路由。
- 使用 **display ip routing-table** *ip-address1* { *mask1* | *mask-length1* } *ip-address2* { *mask2* | *mask-length2* } [**verbose**]命令查看指定目的地址范围内的路由。
- 使用 **display ip routing-table acl** { *acl-number* | *acl-name* } [**verbose**]命令查看通过指定基本访问控制列表过滤的路由。
- 使用 **display ip routing-table ip-prefix** *ip-prefix-name* [**verbose**]命令查看通过指定前缀列表过滤的路由。
- 使用 **display ip routing-table statistics** 命令查看路由表的综合信息。
- 使用 **display ip routing-table vpn-instance** *vpn-instance-name* 命令查看私网路由表摘要信息。
- 使用 **display ip routing-table vpn-instance** *vpn-instance-name* **verbose** 命令查看私网路由表详细信息。

----结束

1.1.2 路由管理模块的显示

使用路由管理模块的显示命令可以定位路由问题。

背景信息

使用路由管理模块的显示命令也是定位路由问题的一种方式，如下所示。**display** 命令可以在所有视图下使用。

操作步骤

- 使用 **display rm interface** [*interface-type interface-number*]命令查看接口的路由管理信息。

- 使用 **display rm interface vpn-instance** *vpn-instance-name* 命令查看私网接口的路由管理信息。

----结束

2 静态路由配置

关于本章

静态路由适用于结构比较简单的网络。合理的静态路由可以改进网络的性能，并可为重要的应用保证带宽。

2.1 静态路由概述
静态路由是一种需要管理员手工配置的特殊路由。

2.2 AR150/200 支持的静态路由特性
系统支持的静态路由特性包括：IPv4 静态路由、缺省路由、BFD for 静态路由和静态路由永久发布。

2.3 配置 IPv4 静态路由
在 IPv4 网络中，通过配置 IPv4 静态路由，可以准确地控制网络的路由选择。

2.4 配置 IPv6 静态路由
在 IPv6 网络中，通过配置 IPv6 静态路由，可以准确地控制网络的路由选择。

2.5 配置 BFD for 公网 IPv4 静态路由
IPv4 网络中，配置 BFD for 公网 IPv4 静态路由，可以提高路由的收敛速度，增强网络可靠性。

2.6 配置 NQA for IPv4 静态路由
在 IPv4 网络中，如果受互通设备不支持 BFD 功能的限制，无法配置 BFD for 公网 IPv4 静态路由对链路进行检测，这时可以配置 NQA for IPv4 静态路由。通过 NQA 测试例对链路状态进行检测，当链路发生故障时可以快速的进行链路切换，避免业务的长时间中断。

2.7 配置举例
静态路由配置举例包括组网需求、组网图、配置注意事项、配置思路和配置步骤。

2.1 静态路由概述

静态路由是一种需要管理员手工配置的特殊路由。

当网络结构比较简单时，只需配置静态路由就可以使网络正常工作。使用静态路由可以改进网络的性能，并可为重要的应用保证带宽。

静态路由的缺点在于：当网络发生故障或者拓扑发生变化后，静态路由不会自动改变，必须有管理员的介入。

2.2 AR150/200 支持的静态路由特性

系统支持的静态路由特性包括：IPv4 静态路由、缺省路由、BFD for 静态路由和静态路由永久发布。

IPv4 静态路由

IPv4 静态路由需要管理员手工配置，适合于一些结构比较简单的 IPv4 网络。

在配置 IPv4 静态路由时，如果指定的目的地址为 0.0.0.0（掩码长度为 0），则表示配置了一条 IPv4 缺省路由。如果报文的目的地址无法匹配路由表中的任何一项，路由器将选择 IPv4 缺省路由来转发 IPv4 报文。

AR150/200 支持普通静态路由，也支持与 VPN 实例关联的静态路由，后者主要用于 VPN 路由的管理。

有关 VPN 实例请参见《Huawei AR150&200 系列企业路由器 特性描述-VPN》。

IPv6 静态路由属性及功能

IPv6 静态路由与 IPv4 静态路由类似，也需要管理员手工配置，适合于一些结构比较简单的 IPv6 网络。

在配置 IPv6 静态路由时，如果指定的目的地址为::/0（掩码长度为 0），则表示配置了一条 IPv6 缺省路由。如果报文的目的地址无法匹配路由表中的任何一项，路由器将选择 IPv6 缺省路由来转发 IPv6 报文。

说明

IPv4 静态路由和 IPv6 静态路由之间的主要区别是目的地址和下一跳地址有所不同，IPv6 静态路由是使用 IPv6 地址为下一跳，而 IPv4 静态路由则使用 IPv4 地址为下一跳。

IPv6 静态路由功能使用 License 授权，缺省情况下，设备的 IPv6 静态路由功能受限无法使用。如果需要使用 IPv6 静态路由功能，请联系华为办事处申请并购买如下 License，

- AR150&200 数据业务增值包

缺省路由

缺省路由是另外一种特殊的路由。通常情况下，管理员可以通过手工方式配置缺省路由；但有些时候，也可以使动态路由协议生成缺省路由，如 OSPF 和 IS-IS。

简单来说，缺省路由是在没有找到匹配的路由表入口项时才使用的路由。可以通过命令 **display ip routing-table** 查看当前是否设置了缺省路由。

如果报文的目的地址不能与路由表的任何入口项相匹配，那么该报文将选取缺省路由。如果没有缺省路由，那么该报文将被丢弃，并向源端返回一个 ICMP 报文，报告该目的地址或网络不可达。

BFD for 静态路由

与动态路由协议不同，静态路由自身没有检测机制，当网络发生故障的时候，需要管理员介入。BFD for 静态路由特性可为公网静态路由绑定 BFD 会话，利用 BFD 会话来检测静态路由所在链路的状态，系统根据检测结果决定是否把静态路由加入 IP 路由表。

BFD for 静态路由可为每条静态路由绑定一个 BFD 会话。

- 当某条静态路由上的 BFD 会话检测到故障（由 Up 转为 Down），BFD 会将故障上报路由管理系统，由路由管理系统将这条路由设置为“非激活”状态（此条路由不可用，从 IP 路由表中删除）。
- 当某条静态路由上的 BFD 会话协商成功（由 Down 转为 Up），BFD 会上报路由管理系统，由路由管理系统将这条路由设置为“激活”状态（此路由可用，加入 IP 路由表）。

静态路由永久发布

静态路由永久发布可以为客户提供一种低成本、部署简单的链路检测机制，提高与其它厂商设备的兼容性。在客户希望确定业务流量的转发路径，不希望流量从其它路径传输时，可以通过 Ping 静态路由目的地址的方式实现链路检测，以极低的代价达到业务监控的目的。

2.3 配置 IPv4 静态路由

在 IPv4 网络中，通过配置 IPv4 静态路由，可以准确地控制网络的路由选择。

2.3.1 建立配置任务

在配置 IPv4 静态路由前了解此特性的应用环境、配置此特性的前置任务和数据准备，有助于快速、准确地完成配置任务。

应用环境

配置 IPv4 静态路由时，需要了解以下内容：

- 目的地址与掩码

  在 **ip route-static** 命令中，IPv4 地址为点分十进制格式，掩码可以用点分十进制表示，也可用掩码长度（即掩码中连续‘1’的位数）表示。
- 出接口和下一跳地址

  在配置静态路由时，可指定出接口 *interface-type interface-name*，也可指定下一跳地址 *nexthop-address*，是指定出接口还是指定下一跳地址要视具体情况而定。

  实际上，所有的路由项都必须明确下一跳地址。在发送报文时，首先根据报文的目的地址寻找路由表中与之匹配的路由。

  在某些情况下，如链路层被 PPP 封装，即使不知道对端地址，也可以在路由器配置时指定出接口。这样，即使对端地址发生了改变也无须改变该路由器的配置。
- 其它属性

对于不同的静态路由，可以为它们配置不同的优先级 **preference**，从而更灵活地应用路由管理策略。例如：配置到达相同目的地的多条路由，如果指定相同优先级，则可实现负载分担，如果指定不同优先级，则可实现路由备份。

在使用 **ip route-static** 配置静态路由时，如果将目的地址与掩码配置为全零（0.0.0.0 0.0.0.0），则表示配置的是缺省路由。

前置任务

在配置 IPv4 静态路由之前，需完成以下任务：

- 配置接口的链路层协议参数（和 IP 地址），使接口的链路协议状态为 Up

数据准备

在配置 IPv4 静态路由之前，需要准备以下数据。

序号	数据
1	目的网络地址和掩码
2	下一跳的 IPv4 地址或出接口
3	IPv4 静态路由的优先级

2.3.2 在公网上配置 IPv4 静态路由

配置 IPv4 静态路由需要注意目的地址、出接口和下一跳。

背景信息

请在需要配置静态路由的路由器上进行如下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **ip route-static** *ip-address* { *mask* | *mask-length* } { *nexthop-address* | *interface-type interface-number* [*nexthop-address*] | **vpn-instance** *vpn-instance-name nexthop-address* } [**preference** *preference* | **tag** *tag*] * [**description** *text*]，配置 IPv4 静态路由。

缺省情况下，没有配置 IPv4 静态路由。

----结束

2.3.3 （可选）配置 IPv4 静态路由的缺省优先级

配置 IPv4 静态路由缺省优先级可以影响路由的选路顺序。

背景信息

请在需要配置静态路由，并且需要改变静态路由缺省优先级的路由器上进行如下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **ip route-static default-preference** *preference*，配置静态路由的缺省优先级。

缺省情况下，静态路由的缺省优先级为 60。

在配置静态路由时，如果没有显式的指定优先级，就会使用缺省优先级。重新设置缺省优先级后，仅对新增的 IPv4 静态路由有效。

----结束

2.3.4 （可选）静态路由按迭代深度优先选择

配置静态路由按迭代深度优选之后，静态路由模块会选择迭代深度较小的静态路由作为活跃路由，并下发 FIB，其他路由为不活跃路由。

背景信息

配置静态路由后，系统中存在若干同一前缀的静态路由，迭代深度不同。配置了基于迭代深度优选之后，静态路由模块会选择迭代深度较小的静态路由作为活跃路由，并下发 FIB，其他路由为不活跃路由。

请在需要配置静态路由的路由器上进行如下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **ip route-static selection-rule relay-depth**，配置静态路由按迭代深度进行优先选择。

缺省情况下，静态路由不按迭代深度进行优选。

----结束

2.3.5 （可选）配置 IPv4 静态路由永久发布

通过配置静态路由支持永久发布，可以为客户提供一种低成本、部署简单的链路检测机制，提高与其它厂商设备的兼容性。

背景信息

链路有效性直接影响网络的稳定性和可用性，链路状态检测对网络维护具有重要意义。当用户希望确定业务流量的转发路径，不希望流量从其它路径穿越时，可以通过使用 Ping 方式来实现对业务转发链路状态的检测，以极低的代价达到业务监控的目的。

静态路由永久发布就是通过 Ping 静态路由目的地址的方式来检测链路的有效性。配置静态路由永久发布后，静态路由会一直生效，不受路由出接口状态的影响。这种情况下，系统只能通过指定路径转发 Ping 报文，而不会从其他路径绕行，从而真实反映指定路径的链路状态。

请在需要配置静态路由的路由器上进行如下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **ip route-static** *ip-address* { *mask* | *mask-length* } { *nexthop-address* | *interface-type interface-number* [*nexthop-address*] | **vpn-instance** *vpn-instance-name nexthop-address* } **permanent**，配置 IPv4 静态路由永久发布。

缺省情况下，没有配置 IPv4 静态路由永久发布。

----结束

2.3.6 检查配置结果

IPv4 静态路由配置成功后，可以查看路由的详细信息。

前提条件

已经完成 IPv4 静态路由的所有配置。

操作步骤

- 使用 **display ip routing-table** 命令查看 IPv4 路由表摘要信息。
- 使用 **display ip routing-table verbose** 命令查看 IPv4 路由表详细信息。

----结束

2.4 配置 IPv6 静态路由

在 IPv6 网络中，通过配置 IPv6 静态路由，可以准确地控制网络的路由选择。

2.4.1 建立配置任务

在配置 IPv6 静态路由前了解此特性的应用环境、配置此特性的前置任务和数据准备，有助于快速、准确地完成配置任务。

应用环境

在小型 IPv6 网络中，可以通过配置 IPv6 静态路由达到网络互连的目的。相对使用动态路由协议来说，可以节省带宽。

前置任务

在配置 IPv6 静态路由之前，需完成以下任务：

- 配置接口的链路层协议参数（和 IPv6 地址），使接口的链路协议状态为 Up

数据准备

在配置 IPv6 静态路由之前，需要准备以下数据。

序号	数据
1	目的地址和掩码
2	下一跳的 IPv6 地址或者出接口
3	IPv6 静态路由的优先级

2.4.2 在公网上配置 IPv6 静态路由

配置 IPv6 静态路由需要注意目的地址、出接口和下一跳。

背景信息

请在需要配置 IPv6 静态路由的路由器上进行如下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **ipv6 route-static** *dest-ipv6-address prefix-length* { *interface-type interface-number* [*nexthop-ipv6-address*] | *nexthop-ipv6-address* } [**preference** *preference* | **tag** *tag*] * [**description** *text*]，配置 IPv6 静态路由。

配置静态路由时，根据实际情况，或者指定出接口，或者指定下一跳地址。如果当输出接口为非 P2P 类型时，则必须指定下一跳地址。

如果没有指定 **preference** 参数，则缺省的优先级为 60。

缺省情况下，没有配置 IPv6 静态路由。

----结束

2.4.3 （可选）配置 IPv6 静态路由的缺省优先级

配置 IPv6 静态路由缺省优先级可以改变静态路由的优先级。

背景信息

请在需要配置 IPv6 静态路由，并且需要改变 IPv6 静态路由缺省优先级的路由器上进行如下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **ipv6 route-static default-preference** *preference*，配置 IPv6 静态路由的缺省优先级。

缺省情况下，IPv6 静态路由的缺省优先级为 60。

在配置 IPv6 静态路由时，如果没有显式的指定优先级，就会使用缺省优先级。重新设置缺省优先级后，仅对新增的 IPv6 静态路由有效。

----结束

2.4.4 检查配置结果

IPv6 静态路由配置成功后，可以查看路由的详细信息。

前提条件

已经完成 IPv6 静态路由的所有配置。

操作步骤

- 使用 **display ipv6 routing-table** 命令查看 IPv6 路由表摘要信息。
- 使用 **display ipv6 routing-table verbose** 命令查看 IPv6 路由表详细信息。

----结束

2.5 配置 BFD for 公网 IPv4 静态路由

IPv4 网络中，配置 BFD for 公网 IPv4 静态路由，可以提高路由的收敛速度，增强网络可靠性。

2.5.1 建立配置任务

在配置 BFD 检测静态路由前了解此特性的应用环境、配置此特性的前置任务和数据准备，可以快速、准确地完成配置任务。

应用环境

BFD 能提供 IPv4 转发故障的快速检测，使 IPv4 网络以良好 QoS 实现语音、视频及其它点播业务的传输，从而帮助服务提供商基于 IPv4 网络为客户提供所需的高可靠性、高适用性 VoIP 及其它实时业务。

在静态路由上绑定 BFD 会话，可利用 BFD 会话为公网 IPv4 静态路由提供链路检测机制。一条静态路由可以绑定一条 BFD 会话。

说明

目前，BFD 会话不会感知路由切换。如果绑定的对端 IP 地址改变引起路由切换到其他链路上，除非原链路转发不通，否则，BFD 不会重新协商。

前置任务

在配置 BFD for 公网 IPv4 静态路由之前，需完成以下任务：

- 配置接口的链路层协议参数和 IP 地址，使接口的链路协议状态为 Up

数据准备

在配置 BFD for 公网 IPv4 静态路由之前，需要准备以下数据。

序号	数据
1	目的网络地址和掩码
2	下一跳的 IPv4 地址或出接口
3	BFD 检测的对端 IP 地址
4	BFD 会话的本地标识符和远端标识符

2.5.2 在公网上配置 IPv4 静态路由

配置 IPv4 静态路由需要注意目的地址、出接口和下一跳。

背景信息

请在需要配置静态路由的路由器上进行如下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **ip route-static** *ip-address* { *mask* | *mask-length* } { *nexthop-address* | *interface-type interface-number* [*nexthop-address*] | **vpn-instance** *vpn-instance-name nexthop-address* } [**preference** *preference* | **tag** *tag*] * [**description** *text*]，配置 IPv4 静态路由。

缺省情况下，没有配置 IPv4 静态路由。

----结束

2.5.3 配置 BFD 会话

BFD 会话用于快速检测、监控网络中链路的转发连通状况。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **bfd**，使能全局 BFD 功能并进入 BFD 全局视图。

步骤 3 执行命令 **quit**，返回系统视图。

步骤 4 执行命令 **bfd** *cfg-name* **bind peer-ip** *peer-ip* [**vpn-instance** *vpn-instance-name*] [**interface** *interface-type interface-number*] [**source-ip** *source-ip*]，创建 BFD 会话。

- 在第一次创建 BFD 会话时，必须绑定对端的 IP 地址，且创建后不可修改。
- 在创建 BFD 配置项时，系统只检查 IP 地址是否符合 IP 地址格式，不检查其正确性。绑定错误的对端 IP 地址或源 IP 地址都将导致 BFD 会话无法建立。
- 如果同时指定了对端 IP 和本端接口，表示检测单跳链路，即检测以该接口为出接口、以 peer-ip 为下一跳地址的一条固定路由。如果只指定对端 IP，表示检测多跳路由。
- 当 BFD 与单播逆向路径转发 URPF（Unicast Reverse Path Forwarding）特性一起应用时，由于 URPF 会对接收到的报文进行源 IP 地址检查，用户在创建 BFD 绑定时，需

要使用 source-ip 选项手工指定正确的 BFD 报文的源 IP 地址，以免 BFD 报文被错误地丢弃。

步骤 5 配置标识符：

- 执行命令 **discriminator local** *discr-value*，配置本地标识符。
- 执行命令 **discriminator remote** *discr-value*，配置远端标识符。

说明

BFD 会话两端设备的本地标识符和远端标识符需要分别对应，即，本端的本地标识符与对端的远端标识符相同，否则会话无法正确建立。并且，本地标识符和远端标识符配置成功后不可修改。

步骤 6 执行命令 **commit**，提交配置。

说明

在创建 BFD 会话时，配置完必要的参数（例如本地标识符和远端标识符）后，必须执行 **commit** 命令才能成功创建会话。

----结束

2.5.4 配置静态路由绑定 BFD 会话

静态路由绑定 BFD 会话的时候，静态路由和 BFD 会话必须在同一条链路上。

背景信息

请在需要为静态路由绑定 BFD 会话的路由器上进行如下配置：

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **ip route-static** *ip-address* { *mask* | *mask-length* } { *nexthop-address* | *interface-type interface-number* [*nexthop-address*] } [**preference** *preference* | **tag** *tag*] * **track bfd-session** *cfg-name* [**description** *text*]，为公网 IPv4 静态路由绑定 BFD 会话。

说明

为静态路由绑定 BFD 会话的时候，请确保 BFD 会话和静态路由在同一链路上。

----结束

2.5.5 检查配置结果

BFD 检测静态路由配置成功后，可以查看 BFD 会话信息及 BFD 绑定静态路由的情况。

前提条件

已经完成 BFD for 公网 IPv4 静态路由的所有配置。

操作步骤

- 使用 **display bfd session** { **all** | **discriminator** *discr-value* } [**verbose**]命令查看 BFD 会话信息。
- 使用 **display current-configuration | include bfd** 命令查看 BFD for 静态路由的配置。

只有配置完 BFD 会话参数并成功建立会话后，才能查看到 BFD 会话信息。

如果 BFD 会话协商成功，可以看到 BFD 会话的状态为 Up，且在系统视图下执行 **display current-configuration | include bfd** 命令，可以查看到 BFD 会话已经绑定。

----结束

2.6 配置 NQA for IPv4 静态路由

在 IPv4 网络中，如果受互通设备不支持 BFD 功能的限制，无法配置 BFD for 公网 IPv4 静态路由对链路进行检测，这时可以配置 NQA for IPv4 静态路由。通过 NQA 测试例对链路状态进行检测，当链路发生故障时可以快速的进行链路切换，避免业务的长时间中断。

2.6.1 建立配置任务

在配置 NQA 与静态路由联动前了解此特性的应用环境、配置此特性的前置任务和数据准备，可以快速、准确地完成配置任务。

应用环境

在实际网络中，出于网络稳定性的考虑，通常要对链路状态进行实时检测，根据链路状态的变化进行链路的主备切换。常用的链路状态检测方案有 ARP 检测和 BFD 检测，也可以使用 IGP 收敛方案。但是在一些特殊的场合，上述方案均不适用，比如：

- 如果仅仅需要检测网络中某一条链路的状况，不必对每个用户进行检测，这时不宜使用 ARP 检测。
- 如果网络中存在不支持 BFD 检测的设备，无法使用 BFD 检测方案。
- 如果链路的两端存在二层设备，这样就无法配置动态路由协议，无法使用 IGP 协议收敛的方案。

而 NQA（Network Quality Analysis） for 静态路由则只要求互通设备的其中一端支持 NQA 即可，并不要求两端都支持，且不受二层设备的限制，正好可以解决上述问题。

在发生链路故障后，NQA 测试例可以快速的检测到这个变化，并且在 IP 路由表中把与该 NQA 测试例联动的静态路由删除，从而影响流量的转发。

前置任务

在配置 NQA for IPv4 静态路由之前，需完成以下任务：

- 配置接口的链路层协议参数（和 IP 地址），使接口的链路协议状态为 Up

数据准备

在配置 NQA for IPv4 静态路由之前，需要准备以下数据。

序号	数据
1	NQA 测试例的管理者和测试例名
2	NQA 检测的对端 IP 地址
3	目的网络地址和掩码

2.6.2 配置 ICMP 类型的 NQA 测试例

NQA 是网络故障诊断和定位的有效工具。

背景信息

NQA（Network Quality Analysis）可以测量网络上运行的各种协议的性能，使运营商能够实时采集到各种网络运行指标。例如，HTTP 的总时延、TCP 连接时延、DNS 解析时延、文件传输速率、FTP 连接时延、DNS 解析错误率等。对于这些业务特性的检测，NQA 是通过创建测试例来完成的。

NQA 把测试两端称为客户端和服务器端，NQA 的测试是由客户端发起。在客户端配置测试例后，NQA 把相应的测试类型放入到测试例队列中。在测试例启动后，根据返回的报文，可以对相关协议的运行状态提供数据信息。

ICMP 类型的 NQA 测试例用于检测 NQA 客户端到目的端的路由是否可达。ICMP 测试提供类似于普通命令行下的 **ping** 命令功能，但输出信息更为丰富。

- 缺省情况下能够保存最近 5 次的测试结果。
- 结果中能够显示平均时延，丢包率，最后一个报文正确接收的时间等信息。

ICMP 类型的 NQA 测试例最小发包间隔为 1 秒，这样 NQA 无论是在检测到链路故障，还是故障恢复后都能将检测结果通告给系统。NQA 的详细配置请参见《Huawei AR150&200 系列企业路由器 配置指南-网络管理》的“NQA 配置”。

请在 NQA 客户端进行下述配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **nqa test-instance** *admin-name test-name*，建立 NQA 测试例，并进入测试例视图。

步骤 3 执行命令 **test-type icmp**，配置测试例类型为 ICMP。

步骤 4 执行命令 **destination-address ipv4** *ip-address*，配置目的地址。

对于测试例而言，指定服务器端是通过 **destination-address** 命令配置 NQA 测试例的目的地址来实现的。

步骤 5 （可选）执行命令 **frequency** *interval*，配置 NQA 测试例的自动执行测试间隔。缺省情况下，没有配置自动测试间隔，即只进行一次测试。

步骤 6 （可选）执行命令 **probe-count** *number*，配置 NQA 测试例一次测试的探针数目。缺省情况下，测试探针数目是 3。

通过多次发送 NQA 测试例的测试探针，可以通过统计数据更加准确的评估网络质量。

步骤 7 执行命令 **start**，启动 NQA 测试。

命令 **start** 有多种形式，根据实际需要选择其中一种启动方式：

- 执行命令 **start now** [**end** { **at** [*yyyy*/*mm*/*dd*] *hh*:*mm*:*ss* | **delay** { **seconds** *second* | *hh*:*mm*:*ss* } | **lifetime** { **seconds** *second* | *hh*:*mm*:*ss* } }]，立即启动测试例。

- 执行命令 **start at** [*yyyy*/*mm*/*dd*] *hh*:*mm*:*ss* [**end** { **at** [*yyyy*/*mm*/*dd*] *hh*:*mm*:*ss* | **delay** { **seconds** *second* | *hh*:*mm*:*ss* } | **lifetime** { **seconds** *second* | *hh*:*mm*:*ss* } }]，在指定时刻启动测试例。
- 执行命令 **start delay** { **seconds** *second* | *hh*:*mm*:*ss* } [**end** { **at** [*yyyy*/*mm*/*dd*] *hh*:*mm*:*ss* | **delay** { **seconds** *second* | *hh*:*mm*:*ss* } | **lifetime** { **seconds** *second* | *hh*:*mm*:*ss* } }]，延迟指定时间后启动测试例。

----结束

2.6.3 配置静态路由与 NQA 测试例联动

配置静态路由与 NQA 测试例联动，当 NQA 检测到链路故障时，会撤销静态路由的发布，从而影响流量的转发。

背景信息

网络结构比较简单时，只需配置静态路由就可以使网络正常工作。或者当路由器不能通过动态路由协议建立到达目的网络的路由时，也可以使用静态路由。但是，与动态路由协议不同，静态路由自身没有检测机制，当网络发生故障时，静态路由无法感知，这样可能造成流量损失。

而 NQA for IPv4 静态路由特性可为静态路由绑定 NQA 测试例，利用 NQA 测试例的 Ping 功能来检测静态路由所在链路的状态，进而决定与其绑定的静态路由是否活跃，影响流量的转发。

请在需要配置静态路由的路由器上进行如下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **ip route-static** *ip-address* { *mask* | *mask-length* } { *nexthop-address* | *interface-type interface-number* [*nexthop-address*] } [**preference** *preference* | **tag** *tag*] * **track nqa** *admin-name test-name* [**description** *text*]，配置 IPv4 静态路由与 NQA 测试例联动。

说明

配置 NQA 测试例与静态路由联动时，不支持 NQA 检测的路由本身是所绑定的静态路由的情况。

配置同一条静态路由与其它 NQA 测试例联动时，会解除与前一个 NQA 测试例的联动关系。

----结束

2.6.4 检查配置结果

在静态路由与 NQA 测试例联动配置成功后，可以查看 NQA 测试的结果数据，以及静态路由与 NQA 测试例联动的信息。

前提条件

已经完成 NQA for IPv4 静态路由的所有配置。

说明

NQA 测试不会在终端自动显示测试结果，必须使用 **display nqa results** 命令查看测试结果。缺省情况下只能显示最近 5 次的测试结果。

操作步骤

步骤 1 执行 **display current-configuration | include nqa** 命令查看 NQA for 静态路由的配置。

步骤 2 执行 **display nqa results** [**test-instance** *admin-name test-name*]命令查看 NQA 测试结果。

----结束

任务示例

配置完成后，在系统视图下，执行 **display current-configuration | include nqa** 命令，可以看到静态路由已经绑定 NQA 测试例。例如：

```
<Huawei> display current-configuration | include nqa
 ip route-static 172.16.1.3 255.255.255.255 Ethernet1/0/0 track nqa admin icmp
nqa test-instance admin icmp
```

执行 **display nqa results** 命令，如果测试已经成功结束，可以看到以下信息。

- testflag is active
- testtype is icmp
- The test is finished
- Completion:success

例如：

```
<Huawei> display nqa results test-instance admin icmp
 NQA entry(admin, icmp) :testflag is active ,testtype is icmp
  1 . Test 206 result   The test is finished
   Send operation times: 15              Receive response times: 15
   Completion:success                 RTD OverThresholds number: 0
   Attempts number:1                     Drop operation number:0
   Disconnect operation number:0         Operation timeout number:0
   System busy operation number:0        Connection fail number:0
   Operation sequence errors number:0    RTT Stats errors number:0
   Destination ip address:172.16.1.2
   Min/Max/Average Completion Time: 30/50/35
   Sum/Square-Sum  Completion Time: 530/19900
   Last Good Probe Time: 2010-10-25 15:39:57.1
   Lost packet ratio: 0 %
```

对于 ICMP 类型的测试，还可以看到接收到响应报文的最小、最大、平均时间，即“Min/Max/Average Completion Time”。另外，还可以看到 NQA 测试的丢包率，即“Lost packet ratio: 0 %”，通过该项数值可以判断链路的状态。该例中丢包率为 0%，说明链路状态完好。

说明

只有 NQA 测试例配置了 **frequency** *interval* 命令，才会显示 **testflag is active**。如果 NQA 测试例没有配置 **frequency** *interval* 命令，则 NQA 只进行一次测试，并且显示结果为 **testflag is inactive**。

2.7 配置举例

静态路由配置举例包括组网需求、组网图、配置注意事项、配置思路和配置步骤。

2.7.1 配置 IPv4 静态路由示例

IPv4 网络中，利用 IPv4 静态路由，可以实现网络中任意两台设备之间的互通。

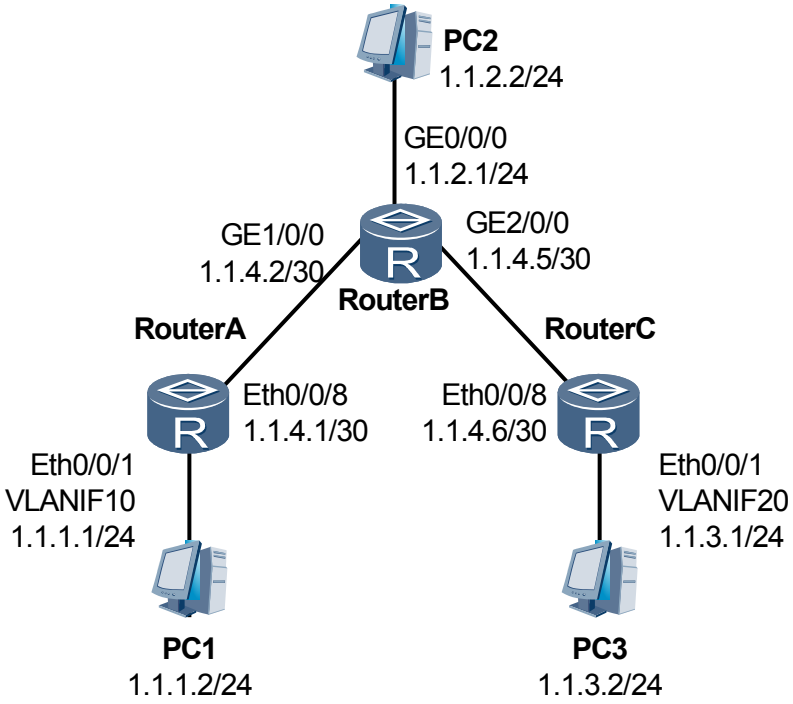
组网需求

路由器各接口及主机的 IP 地址和掩码如图 2-1 所示。要求采用静态路由，使图中任意两台主机之间都能互通。

说明

AR150/200 仅可作为 RouterA 或 RouterC。

图 2-1 配置 IPv4 静态路由组网图



配置思路

采用如下的思路配置 IPv4 静态路由：

1. 首先配置各路由器各接口的 IPv4 地址，使网络互通。
2. 在路由器上配置到目的地址的 IPv4 静态路由及缺省路由。
3. 在各主机上配置 IPv4 缺省网关，使任意两台主机可以互通。

数据准备

为完成此配置例，需准备如下的数据：

- RouterA 的下一跳为 1.1.4.2 的缺省路由。
- RouterB 的目的地址为 1.1.1.0，下一跳为 1.1.4.1 的静态路由。
- RouterB 的目的地址为 1.1.3.0，下一跳为 1.1.4.6 的静态路由。
- RouterC 的下一跳为 1.1.4.5 的缺省路由。
- 主机 PC1 的缺省网关 1.1.1.1，主机 PC2 的缺省网关 1.1.2.1，主机 PC3 的缺省网关 1.1.3.1。

操作步骤

步骤 1 配置各接口的 IP 地址（略）

步骤 2 配置静态路由

在 RouterA 上配置 IPv4 缺省路由。

```
[RouterA] ip route-static 0.0.0.0 0.0.0.0 1.1.4.2
```

在 RouterB 上配置两条 IPv4 静态路由。

```
[RouterB] ip route-static 1.1.1.0 255.255.255.0 1.1.4.1
[RouterB] ip route-static 1.1.3.0 255.255.255.0 1.1.4.6
```

在 RouterC 上配置 IPv4 缺省路由。

```
[RouterC] ip route-static 0.0.0.0 0.0.0.0 1.1.4.5
```

步骤 3 配置主机

配置主机 PC1 的缺省网关为 1.1.1.1，主机 PC2 的缺省网关为 1.1.2.1，主机 PC3 的缺省网关为 1.1.3.1。

步骤 4 查看配置结果

显示 RouterA 的 IP 路由表。

```
[RouterA] display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 8        Routes : 8

Destination/Mask    Proto   Pre  Cost      Flags NextHop         Interface

        0.0.0.0/0   Static 60   0           RD  1.1.4.2         Ethernet0/0/8
        1.1.1.0/24  Direct 0    0           D   1.1.1.1         Vlanif10
        1.1.1.1/32  Direct 0    0           D   127.0.0.1       InLoopBack0
        1.1.4.0/30  Direct 0    0           D   1.1.4.1         Ethernet0/0/8
        1.1.4.1/32  Direct 0    0           D   127.0.0.1       InLoopBack0
        1.1.4.2/32  Direct 0    0           D   1.1.4.2         Ethernet0/0/8
      127.0.0.0/8   Direct 0    0           D   127.0.0.1       InLoopBack0
      127.0.0.1/32  Direct 0    0           D   127.0.0.1       InLoopBack0
```

使用 Ping 命令验证连通性。

```
[RouterA] ping 1.1.3.1
  PING 1.1.3.1: 56  data bytes, press CTRL_C to break
    Reply from 1.1.3.1: bytes=56 Sequence=1 ttl=254 time=62 ms
    Reply from 1.1.3.1: bytes=56 Sequence=2 ttl=254 time=63 ms
    Reply from 1.1.3.1: bytes=56 Sequence=3 ttl=254 time=63 ms
    Reply from 1.1.3.1: bytes=56 Sequence=4 ttl=254 time=62 ms
    Reply from 1.1.3.1: bytes=56 Sequence=5 ttl=254 time=62 ms
  --- 1.1.3.1 ping statistics ---
    5 packet(s) transmitted
    5 packet(s) received
    0.00% packet loss
    round-trip min/avg/max = 62/62/63 ms
```

使用 Tracert 命令验证连通性。

```
[RouterA] tracert 1.1.3.1
 traceroute to  1.1.3.1(1.1.3.1), max hops: 30 ,packet length: 40,press CTRL_C to break
 1 1.1.4.2 31 ms  32 ms  31 ms
 2 1.1.4.6 62 ms  63 ms  62 ms
```

----结束

配置文件

- RouterA 的配置文件

```
#
 sysname RouterA
#
vlan batch 10
#
interface Vlanif10
ip address 1.1.1.1 255.255.255.0
#
interface Ethernet0/0/1
 port link-type access
 port default vlan 10
#
interface Ethernet0/0/8
 ip address 1.1.4.1 255.255.255.252
#
ip route-static 0.0.0.0 0.0.0.0 1.1.4.2
#
return
```

- RouterB 的配置文件

```
#
 sysname RouterB
#
interface GigabitEthernet0/0/0
 ip address 1.1.2.1 255.255.255.0
#
interface GigabitEthernet1/0/0
 ip address 1.1.4.2 255.255.255.252
#
interface GigabitEthernet2/0/0
 ip address 1.1.4.5 255.255.255.252
#
ip route-static 1.1.1.0 255.255.255.0 1.1.4.1
ip route-static 1.1.3.0 255.255.255.0 1.1.4.6
#
return
```

- RouterC 的配置文件

```
#
 sysname RouterC
#
vlan batch 20
#
interface Vlanif20
ip address 1.1.3.1 255.255.255.0
#
interface Ethernet0/0/1
 port link-type access
 port default vlan 20
#
interface Ethernet0/0/8
 ip address 1.1.4.6 255.255.255.252
#
ip route-static 0.0.0.0 0.0.0.0 1.1.4.5
#
return
```

2.7.2 配置 IPv6 静态路由示例

IPv6 网络中，利用 IPv6 静态路由，可以实现网络中任意两台设备之间的互通。

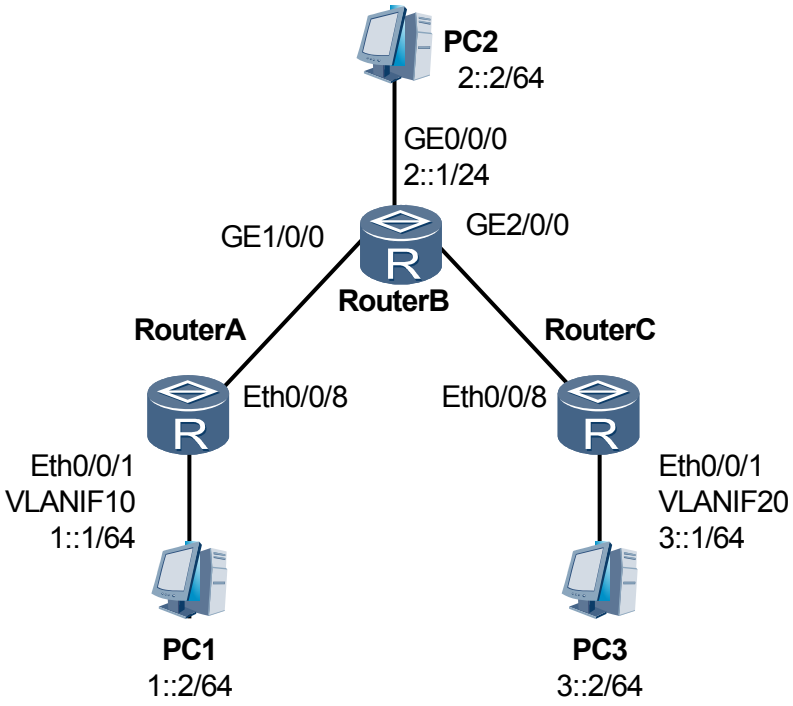
组网要求

如图 2-2 所示，图中所有 IPv6 地址的前缀长度都为 64。要求使所有主机和路由器之间互通。

说明

AR150/200 仅可作为 RouterA 或 RouterC。

图 2-2 配置 IPv6 静态路由组网图



由于图 2-2 中的 PC1、PC2 和 PC3 不能配置动态路由协议，所以该例采用静态路由。路由器与 PC 连接的接口使用 IPv6 链路本地地址。

配置思路

采用如下的思路配置 IPv6 静态路由：

1. 首先配置各路由器各接口的 IPv6 地址，使网络互通。
2. 在各路由器上配置到目的地址的 IPv6 静态路由及缺省路由。
3. 在各主机上配置 IPv6 缺省网关，使任意两台主机可以互通。

数据准备

为完成此配置例，需准备如下的数据：

- 各接口的链路本地地址：

Device	Interface	Link-local address

RouterA	Eth0/0/8	FE80::A19:A6FF:FECD:A897
RouterB	GE1/0/0	FE80::E0:FCD5:A2BF:401
RouterB	GE2/0/0	FE80::A19:A6FF:FECD:A896
RouterC	Eth0/0/8	FE80::A19:A6FF:FECD:A895

- RouterA 的出接口为 Eth0/0/8，下一跳为 FE80::E0:FCD5:A2BF:401 的缺省路由。
- RouterB 的目的地址为 1:: 64，出接口为 GE1/0/0，下一跳为 FE80::A19:A6FF:FECD:A897 的静态路由。
- RouterB 的目的地址为 3:: 64，出接口为 GE2/0/0，下一跳为 FE80::A19:A6FF:FECD:A895 的静态路由。
- RouterC 的出接口为 Eth0/0/8，下一跳为 FE80::A19:A6FF:FECD:A896 的缺省路由。
- 主机 PC1 的缺省网关 1::1，主机 PC2 的缺省网关 2::1，主机 PC3 的缺省网关 3::1。

操作步骤

步骤 1 配置各接口的 IPv6 地址（略）

步骤 2 配置 IPv6 静态路由

在 RouterA 上配置 IPv6 缺省路由。

```
[RouterA] ipv6 route-static :: 0 ethernet 0/0/8 FE80::E0:FCD5:A2BF:401
```

在 RouterB 上配置两条 IPv6 静态路由。

```
[RouterB] ipv6 route-static 1:: 64 gigabitethernet 1/0/0 FE80::A19:A6FF:FECD:A897
[RouterB] ipv6 route-static 3:: 64 gigabitethernet 2/0/0 FE80::A19:A6FF:FECD:A895
```

在 RouterC 上配置 IPv6 缺省路由。

```
[RouterC] ipv6 route-static :: 0 ethernet 0/0/8 FE80::A19:A6FF:FECD:A896
```

步骤 3 配置主机地址和网关

根据组网图配置好各主机的 IPv6 地址，并将 PC1 的缺省网关配置为 1::1，PC2 的缺省网关配置为 2::1，主机 3 的缺省网关配置为 3::1。

步骤 4 查看配置结果

查看 RouterA 的 IPv6 路由表。

```
[RouterA] display ipv6 routing-table
Routing Table : Public
         Destinations : 5        Routes : 5

 Destination  : ::                              PrefixLength : 0
 NextHop      : FE80::E0:FCD5:A2BF:401           Preference   : 60
 Cost         : 0                               Protocol     : Static
 RelayNextHop : ::                              TunnelID     : 0x0
 Interface    : Ethernet0/0/8                   Flags        : D

 Destination  : ::1                             PrefixLength : 128
```

```
NextHop      : ::1                          Preference   : 0
Cost         : 0                            Protocol     : Direct
RelayNextHop : ::                           TunnelID     : 0x0
Interface    : InLoopBack0                  Flags        : D

Destination  : 1::                          PrefixLength : 64
NextHop      : 1::1                         Preference   : 0
Cost         : 0                            Protocol     : Direct
RelayNextHop : ::                           TunnelID     : 0x0
Interface    : Vlanif10                     Flags        : D

Destination  : 1::1                         PrefixLength : 128
NextHop      : ::1                          Preference   : 0
Cost         : 0                            Protocol     : Direct
RelayNextHop : ::                           TunnelID     : 0x0
Interface    : Vlanif10                     Flags        : D

Destination  : FE80::                       PrefixLength : 10
NextHop      : ::                           Preference   : 0
Cost         : 0                            Protocol     : Direct
RelayNextHop : ::                           TunnelID     : 0x0
Interface    : NULL0                        Flags        : D
```

使用 Ping 进行验证。

```
[RouterA] ping ipv6 3::1
  PING 3::1 : 56  data bytes, press CTRL_C to break
    Reply from 3::1
    bytes=56 Sequence=1 hop limit=254  time = 63 ms
    Reply from 3::1
    bytes=56 Sequence=2 hop limit=254  time = 62 ms
    Reply from 3::1
    bytes=56 Sequence=3 hop limit=254  time = 62 ms
    Reply from 3::1
    bytes=56 Sequence=4 hop limit=254  time = 63 ms
    Reply from 3::1
    bytes=56 Sequence=5 hop limit=254  time = 63 ms
  --- 3::1 ping statistics ---
    5 packet(s) transmitted
    5 packet(s) received
    0.00% packet loss
    round-trip min/avg/max = 62/62/63 ms
```

使用 Tracert 进行验证。

```
[RouterA] tracert ipv6 3::1
traceroute to 3::1  30 hops max,60 bytes packet
 1 FE80::E0:FCD5:86D4:401 11 ms  3 ms  4 ms
 2 3::1 4 ms  3 ms  3 ms
```

----结束

配置文件

- RouterA 的配置文件

```
#
 sysname RouterA
#
 ipv6
#
vlan batch 10
#
interface Vlanif10
 ipv6 enable
 ipv6 address 1::1/64
#
interface Ethernet0/0/1
 port link-type access
```

```
 port default vlan 10
#
interface Ethernet0/0/8
 ipv6 enable
 ipv6 address auto link-local
#
ipv6 route-static :: 0 Ethernet 0/0/8 FE80::E0:FCD5:A2BF:401
#
return
```

- RouterB 的配置文件

```
#
 sysname RouterB
#
 ipv6
#
interface GigabitEthernet0/0/0
 ipv6 address 2::1/64
#
interface GigabitEthernet1/0/0
 ipv6 address auto link-local
#
interface GigabitEthernet2/0/0
 ipv6 address auto link-local
#
ipv6 route-static 1:: 64 GigabitEthernet 1/0/0 FE80::A19:A6FF:FECD:A897
ipv6 route-static 3:: 64 GigabitEthernet 2/0/0 FE80::A19:A6FF:FECD:A895
#
return
```

- RouterC 的配置文件

```
#
 sysname RouterC
#
 ipv6
#
vlan batch 20
#
interface Vlanif20
 ipv6 enable
 ipv6 address 3::1/64
#
interface Ethernet0/0/1
 port link-type access
 port default vlan 20
#
interface Ethernet0/0/8
 ipv6 enable
 ipv6 address auto link-local
#
ipv6 route-static :: 0 Ethernet 0/0/8 FE80::A19:A6FF:FECD:A896
#
return
```

3 RIP 配置

关于本章

RIP 可以发布和接收路由信息，影响路由器的数据转发途径，并提供网管功能，主要用于规模较小的网络中。

3.1 RIP 概述
RIP 的实现较为简单，在配置和维护管理方面也远比 OSPF 和 IS-IS 容易，在小型网络中有广泛的应用。

3.2 AR150/200 支持的 RIP 特性
AR150/200 中支持的 RIP 特性包括：RIP-1、RIP-2、水平分割、毒性逆转和多实例。

3.3 配置 RIP 的基本功能
配置 RIP 的基本功能主要包括启动 RIP、指定运行 RIP 的网段以及版本号，是能够使用 RIP 特性的前提。

3.4 配置 RIP 的路由属性
在实际应用中，可以通过配置 RIP 的路由属性改变 RIP 的选路策略，以满足复杂网络环境中的需要。

3.5 控制 RIP 路由信息的发布
对 RIP 路由信息的发布进行精确的控制，可以满足复杂网络环境中的需要。

3.6 控制 RIP 路由信息的接收
对 RIP 路由信息的接收进行精确的控制，可以满足复杂网络环境中的需要。

3.7 配置 RIP-2 特性
RIP-2 是 RIP version 2 的简称，它与 RIP-1 的不同点在于，RIP-2 支持 VLSM 和 CIDR，并支持验证功能，从而功能更加完善，安全性更高。

3.8 调整和优化 RIP 网络
在某些特殊的网络环境中配置 RIP 的一些特性功能，例如配置 RIP 定时器、报文的发送间隔、最大数量，可以对 RIP 网络的性能进行调整和优化。

3.9 配置 RIP GR
通过 RIP GR 解决 RIP 路由器重启后造成路由计算不准确、报文丢失的问题。

3.10 配置 BFD for RIP

当网络中运行高速率数据业务时，通过配置 BFD for RIP，可以实现 RIP 对网络中的故障快速做出响应。

3.11 配置静态 BFD for RIP

BFD 能够提供轻负荷、快速的链路故障检测，配置静态 BFD for RIP 是实现 BFD 检测功能的一种方式。

3.12 配置 RIP 的网管功能

通过配置 RIP 和 MIB 绑定，可以通过网管的环境来查看和配置 RIP。

3.13 维护 RIP

复位 RIP 连接、清除 RIP 的统计信息。

3.14 配置举例

在实际组网中，RIP 的版本不同以及是否引入外部路由将影响路由学习。

3.1 RIP 概述

RIP 的实现较为简单，在配置和维护管理方面也远比 OSPF 和 IS-IS 容易，在小型网络中有广泛的应用。

路由信息协议 RIP (Routing Information Protocol)是一种较为简单的内部网关协议 IGP (Interior Gateway Protocol)，主要用于规模较小的网络中，比如校园网以及结构较简单的地区性网络。对于更为复杂的环境和大型网络，一般不使用 RIP。

RIP 是一种基于距离矢量（Distance-Vector）算法的协议，它通过 UDP 报文进行路由信息的交换，使用的端口号为 520。

RIP 使用跳数（Hop Count）来衡量到达目的地址的距离，称为度量值。在 RIP 中，路由器到与它直接相连网络的跳数为 0，通过一个路由器可达的网络的跳数为 1，其余依此类推。为限制收敛时间，RIP 规定度量值取 0 ～ 15 之间的整数，大于或等于 16 的跳数被定义为无穷大，即目的网络或主机不可达。由于这个限制，使得 RIP 不可能在大型网络中得到应用。

为提高性能，防止产生路由循环，RIP 支持水平分割（Split Horizon）和毒性反转（Poison Reverse）功能。

- 水平分割指的是 RIP 从某个接口学到的路由，不会从该接口再发回给邻居设备。这样不但减少了带宽消耗，还可以防止路由环路。
- 毒性逆转指的是 RIP 从某个接口学到路由后，将该路由的开销设置为 16（即指明该路由不可达），并从原接口发回邻居设备。利用这种方式，可以清除对方路由表中的无用路由。

由于 RIP 的实现较为简单，在配置和维护管理方面也远比 OSPF 和 IS-IS 容易，因此在实际组网中仍有广泛的应用。

RIP 有两个版本：RIP-1 和 RIP-2。RIP-1 是有类别路由协议（Classful Routing Protocol）。RIP-2 是一种无分类路由协议（Classless Routing Protocol），并且使用 224.0.0.9 作为 RIP 路由器的组播地址。

与 RIP-1 相比，RIP-2 有以下优势：

- 支持外部路由标记（Route Tag），可以在路由策略中根据 Tag 对路由进行灵活的控制。
- 报文中携带掩码信息，支持路由聚合和 CIDR（Classless Inter-Domain Routing）。
- 支持指定下一跳，在广播网上可以选择到最优下一跳地址。
- 支持组播路由发送更新报文，只有支持 RIP-2 的设备才能收到协议报文，减少资源消耗。
- 支持对协议报文进行验证，并提供明文验证和 MD5 验证两种方式，增强安全性。

3.2 AR150/200 支持的 RIP 特性

AR150/200 中支持的 RIP 特性包括：RIP-1、RIP-2、水平分割、毒性逆转和多实例。

在 AR150/200 目前的实现中，支持以下 RIP 特性：

- 支持 RIP-1 和 RIP-2。
- 支持 RIP 多实例。

3.3 配置 RIP 的基本功能

配置 RIP 的基本功能主要包括启动 RIP、指定运行 RIP 的网段以及版本号，是能够使用 RIP 特性的前提。

3.3.1 建立配置任务

在配置 RIP 基本功能前了解此特性的应用环境、配置此特性的前置任务和数据准备，可以帮助您快速、准确地完成配置任务。

应用环境

RIP 基本功能的配置任务包括 RIP 特性配置中涉及的基本配置，完成本节的配置就可以使用 RIP 特性。

前置任务

在配置 RIP 的基本功能之前，需完成以下任务：

- 配置链路层协议。
- 配置接口的网络层地址，使相邻节点的网络层可达。

数据准备

在配置 RIP 的基本功能之前，需要准备以下数据。

序号	数据
1	RIP 进程号
2	RIP 接口所在的网段
3	RIP 版本号

3.3.2 启动 RIP

创建 RIP 进程是进行所有 RIP 配置的前提。

背景信息

如果在启动 RIP 前在接口视图下配置了 RIP 相关命令，这些配置只有在 RIP 启动后才会生效。

请在需要运行 RIP 协议的每台路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行 **rip** [*process-id*]，启动 RIP，进入 RIP 视图。

RIP 支持多实例，可以将 RIP 进程与 VPN 实例关联。此时，需执行 **rip** [*process-id*] **vpn-instance** *vpn-instance-name* 命令。

说明

为了方便管理，提高控制效率，RIP 支持多进程和多实例特性。多进程允许为一个指定的 RIP 进程关联一组接口，从而保证该进程进行的所有协议操作都仅限于这一组接口。而一个接口只能与一个 RIP 进程相关联。这样，就可以实现一台设备有多个 RIP 协议进程，每个进程负责唯一的一组接口。而且每个 RIP 进程的路由数据也是相互独立的，但进程之间可以相互引入路由。

对于支持 VPN 的设备，每个 RIP 进程都与一个指定的 VPN 实例相关联。这样，所有附加到该进程的接口都应与该进程相关联的 VPN 实例相关联。

----结束

3.3.3 在指定网段使能 RIP

RIP 只在指定网段上的接口运行。对于不在指定网段上的接口，RIP 既不在它上面接收和发送路由，也不将它的接口路由转发出去。因此，RIP 启动后必须指定其工作网段。

背景信息

缺省情况下，RIP 启动后在所有接口上禁用。

请在需要运行 RIP 协议的每台路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **rip** [*process-id*]，使能 RIP 路由进程，进入 RIP 视图。

步骤 3 执行命令 **network** *network-address*，在指定网段使能 RIP。

network-address 为自然网段的地址。

说明

一个接口只能与一个 RIP 进程相关联。

对于一个配置了多个子接口 IP 地址的物理接口，如果已经宣告该接口上的任一网段到某 RIP 进程，则该接口无法后续再和其他 RIP 进程相关联。

----结束

3.3.4 配置 RIP 的版本号

RIP 的版本包括 RIP-1 和 RIP-2 两种，它们的功能有所不同。

背景信息

请在运行 RIP 协议的路由器上进行以下配置。

操作步骤

- 配置全局 RIP 版本号
  1. 执行命令 **system-view**，进入系统视图。

2. 执行命令 **rip** [*process-id*]，使能 RIP 路由进程，进入 RIP 视图。
3. 执行 **version** { **1** | **2** }命令，指定全局 RIP 版本。

- 配置接口的 RIP 版本号
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **interface** *interface-type interface-number*，进入接口视图。
  3. 执行命令 **rip version** { **1** | **2** [**broadcast** | **multicast**] }，指定接口接收的 RIP 版本。

     说明

     缺省情况下，接口接收 RIP-1 和 RIP-2 的报文，只发送 RIP-1 报文。当配置接口版本为 RIP-2 时，同时可以指定报文的发送方式。如果没有配置接口的 RIP 版本号则以全局版本为准，接口下配置的版本号优先级高于全局版本号。

----结束

3.3.5 检查配置结果

RIP 基本功能配置成功后，您可以查看到 RIP 的当前运行状态、配置信息及路由信息。

前提条件

完成上述配置后，请执行下面的命令检查配置结果。

操作步骤

- 使用 **display rip** [*process-id* | **vpn-instance** *vpn-instance-name*]命令查看 RIP 的当前运行状态及配置信息。
- 使用 **display rip** *process-id* **route** 命令查看所有从其他路由器学习到的 RIP 路由。
- 使用 **display default-parameter rip** 命令查看 RIP 的缺省配置信息。
- 使用 **display rip** *process-id* **statistics interface** { **all** | *interface-type interface-number* [**verbose** | **neighbor** *neighbor-ip-address*] }命令查看 RIP 接口的数据信息。

----结束

3.4 配置 RIP 的路由属性

在实际应用中，可以通过配置 RIP 的路由属性改变 RIP 的选路策略，以满足复杂网络环境中的需要。

3.4.1 建立配置任务

RIP 的路由属性包括 RIP 协议的优先级、接口的附加度量值和最大等价路由条数。

应用环境

在实际应用中，可以通过配置 RIP 的路由属性改变 RIP 的选路策略，以满足复杂网络环境中的需要。通过本节的配置过程，你可以：

- 通过调整 RIP 接口的附加度量值来影响路由的选择；

- 当多个路由协议发现相同的路由时，通过配置 RIP 的协议优先级来改变路由协议的优先顺序；
- 使用多条等价路由进行负载分担。

前置任务

在配置 RIP 的路由属性之前，需完成以下任务：

- 配置接口的网络层地址，使相邻节点网络层可达。
- 配置 RIP 的基本功能。

数据准备

在配置 RIP 的路由属性之前，需要准备以下数据。

序号	数据
1	接口的附加度量值
2	RIP 协议优先级的值
3	最大等价路由条数

3.4.2 配置接口的附加度量值

附加路由度量值是在 RIP 路由原来度量值的基础上所增加的度量值（跳数）。对于 RIP 接收和发布路由，可通过不同的命令配置附加度量值。

背景信息

附加路由度量值是在 RIP 路由原来度量值的基础上所增加的度量值（跳数）。

- **rip metricin** 用于在接收到路由后，给其增加一个附加度量值，再加入路由表中，使得路由表中的度量值发生变化。运行该命令会影响到本地设备和其他设备的路由选择。
- **rip metricout** 用于自身路由的发布，发布时增加一个附加的度量值，但路由表中的度量值不会发生变化。运行该命令不会影响本地设备的路由选择，但是会影响其他设备的路由选择。

请在运行 RIP 协议的路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **interface** *interface-type interface-number*，进入接口视图。

步骤 3 执行命令 **rip metricin** *value*，设置接口在接收路由时增加的度量值。

步骤 4 执行命令 **rip metricout** { *value* | { *acl-number* | **acl-name** *acl-name* | **ip-prefix** *ip-prefix-name* } *value1* }，设置接口在发布路由时增加的度量值。

说明

当用 ACL 或 ip-prefix 方式来设置接口发送 RIP 路由增加的度量值时，指定 *value1* 为通过过滤策略的 RIP 路由增加的度量值，没有通过过滤的 RIP 路由增加的度量值为 1。

----结束

3.4.3 配置 RIP 协议优先级

当有多种协议的路由存在时，通过配置 RIP 的协议优先级，可以调整路由器选择最优路由。

背景信息

请在运行 RIP 协议的路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **rip** [*process-id*]，使能 RIP 路由进程，进入 RIP 视图。

步骤 3 执行命令 **preference** { *preference* | **route-policy** *route-policy-name* } *，设置 RIP 协议的优先级。

缺省情况下，RIP 协议的优先级为 100。

----结束

3.4.4 配置最大等价路由条数

通过配置 RIP 最大等价路由条数，可以调整进行负载分担的路由数目。

背景信息

请在运行 RIP 协议的路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **rip** [*process-id*]，使能 RIP 路由进程，进入 RIP 视图。

步骤 3 执行命令 **maximum load-balancing** *number*，设置 RIP 最大等价路由条数。

----结束

3.4.5 检查配置结果

RIP 的路由属性配置成功后，您可以查看到 RIP 的当前运行状态、配置信息及路由信息。

前提条件

完成上述配置后，请执行下面的命令检查配置结果。

操作步骤

- 使用 **display rip** [*process-id* | **vpn-instance** *vpn-instance-name*]命令查看 RIP 的当前运行状态及配置信息。
- 使用 **display rip** *process-id* **database** 命令查看 RIP 发布数据库的所有激活路由。
- 使用 **display rip** *process-id* **route** 命令查看所有从其他路由器学习到的 RIP 路由。

----结束

3.5 控制 RIP 路由信息的发布

对 RIP 路由信息的发布进行精确的控制，可以满足复杂网络环境中的需要。

3.5.1 建立配置任务

RIP 路由信息的发布可通过缺省路由、更新报文、引入外部路由信息的方式。

应用环境

在实际应用中，有时候需要对 RIP 路由信息的发布进行更为精确的控制，以满足网络需要。通过本节的配置过程，你可以：

- 向邻居发布缺省路由；
- 抑制接口发送 RIP 更新报文；
- 在多路由协议环境中引入外部路由并对发布的路由进行过滤。

前置任务

在控制 RIP 路由信息的发布之前，需要完成以下任务：

- 配置接口的网络层地址，使相邻节点网络层可达。
- **配置 RIP 的基本功能**。

数据准备

在控制 RIP 路由信息的发布之前，需要准备以下数据。

序号	数据
1	需要发布的缺省路由的度量值
2	需要抑制发送 RIP 更新报文的接口编号
3	要引入的外部路由协议名称和进程号

3.5.2 配置 RIP 发布缺省路由

缺省路由是指目的地址为 0.0.0.0 的路由。缺省情况下，RIP 不向其邻居发布缺省路由。

背景信息

请在运行 RIP 协议的路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **rip** [*process-id*]，使能 RIP 路由进程，进入 RIP 视图。

步骤 3 执行命令 **default-route originate** [**match default** [**avoid-learning**]] [**cost** *cost*]，使能当前路由器生成缺省路由或者将路由表中存在的缺省路由向 RIP 邻居发布，并且可设置该路由的度量值。

----结束

3.5.3 禁止接口发送更新报文

禁止接口发送更新报文，是预防路由环路的一种方式，可通过两种方法来实现。

背景信息

请在运行 RIP 协议的路由器上进行以下配置。

操作步骤

- 在 RIP 进程下配置（优先级高）
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **rip** [*process-id*]，使能 RIP 路由进程，进入 RIP 视图。
  3. 请根据需要，设置接口为抑制状态。
     - 执行命令 **silent-interface all**，设置所有接口为抑制状态。
     - 执行命令 **silent-interface** *interface-type interface-number*，禁止一个接口发送更新报文。

     可以设置接口为抑制状态，使其只接收报文，用来更新自己的路由表，但不能发送 RIP 报文。**silent-interface** 的优先级大于在接口下配置的 **rip output**，默认情况下为不抑制状态。
- 在接口视图下配置（优先级低）
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **interface** *interface-type interface-number*，进入接口视图。
  3. 执行命令 **undo rip output**，禁止接口发送 RIP 更新报文。

     可以对接口单独指定是否发送或接收 RIP 更新报文，其优先级小于 **silent-interface**。默认情况下允许发送 RIP 更新报文。

----结束

3.5.4 配置 RIP 引入外部路由信息

RIP 可以引入其他进程或其他协议学到的路由信息，从而丰富路由表项。

背景信息

请在运行 RIP 协议的路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **rip** [*process-id*]，使能 RIP 路由进程，进入 RIP 视图。

步骤 3 （可选）执行命令 **default-cost** *cost*，设定路由引入的缺省度量值。

如果在引入路由时没有指定度量值，则使用缺省度量值。

步骤 4 执行命令 **import-route bgp** [**cost** { *cost* | **transparent** } | **route-policy** *route-policy-name*] * 或 **import-route** { { **static** | **direct** | **unr** } | { { **rip** | **ospf** | **isis** } [*process-id*] } } [**cost** *cost* | **route-policy** *route-policy-name*] *，引入外部路由信息。

步骤 5 （可选）执行命令 **filter-policy** { *acl-number* | **acl-name** *acl-name* | **ip-prefix** *ip-prefix-name* } **export** [*protocol* [*process-id*]| *interface-type interface-number*]，对引入的路由信息向外发布时进行过滤。

由于 RIP 要发布的路由信息中，有可能是引入的其他路由协议的路由信息，所以可通过指定 *protocol* 参数来对这些特定的路由信息进行过滤。如果没有指定 *protocol* 参数，则对所有要发布的路由信息进行过滤，包括引入的路由和本地 RIP 路由（相当于直连路由）。

说明

RIP 协议规定的 Tag 字段长度为 16bits，其他路由协议的 Tag 字段长度为 32bits。如果在引入其他路由协议时，应用的路由策略中使用 Tag，则应确保 Tag 值不超过 65535，否则将导致路由策略失效或者产生错误的匹配结果。

----结束

3.5.5 检查配置结果

控制 RIP 路由信息的发布配置成功后，您可以查看到 RIP 的当前运行状态、配置信息及路由信息。

前提条件

完成上述配置后，请执行下面的命令检查配置结果。

操作步骤

- 使用 **display rip** [*process-id* | **vpn-instance** *vpn-instance-name*]命令查看 RIP 的当前运行状态及配置信息。
- 使用 **display rip** *process-id* **database** 命令查看 RIP 发布数据库的所有激活路由。
- 使用 **display rip** *process-id* **route** 命令查看所有从其他路由器学习到的 RIP 路由。

----结束

3.6 控制 RIP 路由信息的接收

对 RIP 路由信息的接收进行精确的控制，可以满足复杂网络环境中的需要。

3.6.1 建立配置任务

RIP 路由信息的接收可通过更新报文、接收主机路由等方式。

应用环境

在实际应用中，有时候需要对 RIP 路由信息的接收进行更为精确的控制，以满足复杂网络环境中的需要。通过本节的配置过程，你可以：

- 禁止接口接收 RIP 的更新报文；
- 对接收的路由信息进行过滤；
- 在多路由协议环境中引入外部路由并对接收的路由进行过滤。

前置任务

在控制 RIP 路由信息的接收之前，需要完成以下任务：

- 配置接口的网络层地址，使相邻节点网络层可达。
- **配置 RIP 的基本功能**。

数据准备

在控制 RIP 路由信息的接收之前，需要准备以下数据。

序号	数据
1	对路由信息过滤时所需要的相关过滤列表

3.6.2 禁止接口接收更新报文

禁止接口接收更新报文，是预防路由环路的一种方式。

背景信息

缺省情况下，允许接收 RIP 更新报文。

请在运行 RIP 协议的路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **interface** *interface-type interface-number*，进入接口视图。

步骤 3 执行命令 **undo rip input**，禁止接口接收 RIP 更新报文。

----结束

3.6.3 禁止 RIP 接收主机路由

禁止 RIP 接收主机路由功能，可以使路由器拒绝接收主机路由，防止路由器因接收到大量无意义路由而浪费网络资源。

背景信息

在某些特殊情况下，路由器会收到大量来自同一网段的主机路由，这些路由对于路由寻址没有多少作用，却占用了大量网络资源。配置了禁止主机路由功能后，路由器能够拒绝它所收到的主机路由。

缺省情况下，允许主机路由加到路由表里。

请在运行 RIP 协议的路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **rip** [*process-id*]，使能 RIP 路由进程，进入 RIP 视图。

步骤 3 执行命令 **undo host-route**，禁止主机路由加到路由表里。

说明

undo host-route 命令对 RIP-2 不起作用。缺省情况下，RIP-2 允许主机路由添加到路由表里。

----结束

3.6.4 配置 RIP 对接收的路由进行过滤

通过指定访问控制列表和地址前缀列表，可以配置入口过滤策略，对接收的路由进行过滤。在接收路由时，还可以指定只接收来自某个邻居的 RIP 报文。

背景信息

路由器提供路由信息过滤功能，通过指定访问控制列表和地址前缀列表，可以配置入口或出口过滤策略，对接收和发布的路由进行过滤。

在接收路由时，还可以指定只接收来自某个邻居的 RIP 报文。

请在运行 RIP 协议的路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **rip** [*process-id*]，使能 RIP 路由进程，进入 RIP 视图。

步骤 3 请根据需要，配置 RIP 对接收的路由进行过滤。

- 执行命令 **filter-policy** { *acl-number* | **acl-name** *acl-name* } **import**，基于 ACL 过滤学到的路由信息。
- 执行命令 **filter-policy gateway** *ip-prefix-name* **import**，基于目的地址前缀过滤邻居发布的路由信息。

- 执行命令 **filter-policy ip-prefix** *ip-prefix-name* [**gateway** *ip-prefix-name*] **import** [*interface-type interface-number*]，对指定接口学到的路由进行基于目的地址前缀的过滤和基于邻居的过滤。

说明

当需要对发布的路由进行过滤时，可以执行命令 **filter-policy** { *acl-number* | **acl-name** *acl-name* | **ip-prefix** *ip-prefix-name* } **export** [*protocol* [*process-id*] | *interface-type interface-number*]。

----结束

3.6.5 检查配置结果

控制 RIP 路由信息的接收配置成功后，您可以查看到 RIP 的当前运行状态、配置信息及路由信息。

前提条件

完成上述配置后，请执行下面的命令检查配置结果。

操作步骤

- 使用 **display rip** [*process-id* | **vpn-instance** *vpn-instance-name*]命令查看 RIP 的当前运行状态及配置信息。
- 使用 **display rip** *process-id* **database** [**verbose**]命令查看 RIP 发布数据库的所有激活路由。
- 使用 **display rip** *process-id* **interface** [*interface-type interface-number*] [**verbose**]命令查看 RIP 的接口信息。
- 使用 **display rip** *process-id* **neighbor** [**verbose**]命令查看 RIP 的邻居信息。
- 使用 **display rip** *process-id* **route** 命令查看所有从其他路由器学习到的 RIP 路由。

----结束

3.7 配置 RIP-2 特性

RIP-2 是 RIP version 2 的简称，它与 RIP-1 的不同点在于，RIP-2 支持 VLSM 和 CIDR，并支持验证功能，从而功能更加完善，安全性更高。

3.7.1 建立配置任务

在配置 RIP-2 特性之前，了解此特性的应用环境、配置此特性的前置任务和数据准备，可以帮助您快速、准确地完成配置任务。

应用环境

RIP-2 是一种有类别路由协议（Classless Routing Protocol），报文中带有子网掩码信息。因此部署 RIP-2 网络，可以节省 IP 地址，避免浪费 IP 地址。并且对于 IP 地址不连续的网络，不能够部署 RIP-1，此时只能部署 RIP-2。

RIP-2 特性专指一些仅针对 RIP-2 的配置，包括：

- RIP-2 的路由聚合

- RIP-2 报文的认证方式

前置任务

在配置 RIP-2 的基本功能之前，需完成以下任务：

- 配置链路层协议。
- 配置接口的网络层地址，使相邻节点的网络层可达。

数据准备

在配置 RIP-2 的基本功能之前，需要准备以下数据。

序号	数据
1	RIP-2 进程号
2	RIP-2 接口所在的网段

3.7.2 配置 RIP-2 的路由聚合

RIP-1 自然使能路由聚合功能，无需配置。RIP-2 支持 VLSM 和 CIDR，可以通过配置路由聚合，提高其灵活性。当需要将所有子网路由广播出去时，可关闭 RIP-2 的自动路由聚合功能。

背景信息

路由聚合是指：同一自然网段内的不同子网的路由在向外（其它网段）发送时聚合成一条自然掩码的路由发送。这一功能主要用于减小路由表的尺寸，进而减少网络上的流量。

路由聚合对 RIP-1 不起作用。RIP-2 支持 VLSM 和 CIDR。当需要将所有子网路由广播出去时，可关闭 RIP-2 的自动路由聚合功能。

请在运行 RIP-2 协议的路由器上进行以下配置。

说明

如果配置了水平分割或毒性反转，有类聚合将失效。因此在向自然网段边界外发送聚合路由时，相关视图下的水平分割和毒性反转功能都应关闭。

操作步骤

- 使能 RIP-2 自动路由聚合
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **rip** [*process-id*]，使能 RIP 路由进程，进入 RIP 视图。
  3. 执行命令 **version** 2，设置 RIP 版本为 RIP-2。
  4. 执行命令 **summary** [**always**]
     - 在未使能水平分割的基础上使能 RIP-2 自动路由聚合，则不用配置参数 **always**

- 不论水平分割是否使能，都使能 RIP-2 自动路由聚合，则需要配置参数 **always**

说明

在 RIP 视图下使用 **summary** 命令进行路由聚合，是使能 RIP-2 基于有类别网络的路由聚合。

- 配置 RIP-2 发布聚合地址
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **interface** *interface-type interface-number*，进入接口视图。
  3. 执行命令 **rip summary-address** *ip-address mask* [**avoid-feedback**]，配置 RIP-2 发布聚合的本地 IP 地址。

     说明

     在接口视图下使用 **rip summary-address** *ip-address mask* [**avoid-feedback**]命令进行路由聚合，是使能 RIP-2 基于无类别网络的路由聚合。

----结束

3.7.3 配置 RIP-2 报文的认证方式

RIP-2 支持对协议报文进行认证，并提供简单认证和 MD5 认证两种方式，增强安全性。

背景信息

RIP-2 支持两种认证方式：

- 简单认证。
- MD5 密文认证。

其中，简单认证使未加密的认证字随报文一同传送，不能提供安全保障，所以简单认证不能用于安全性要求较高的情况。

请在运行 RIP 协议的路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **interface** *interface-type interface-number*，进入接口视图。

步骤 3 配置 RIP-2 报文的认证方式。

- 执行命令 **rip authentication-mode simple** { [**plain**] *plain-text* | **cipher** *password-key* }，配置 RIP-2 报文为 **simple** 认证模式。
- 执行命令 **rip authentication-mode usual** { **plain** *plain-text* | [**cipher**] *password-key* }，配置 RIP-2 报文为 **MD5 usual** 认证模式。
- 执行命令 **rip authentication-mode nonstandard** { **keychain** *keychain-name* | { { **plain** *plain-text* | [**cipher**] *password-key* } *key-id* } }，配置 RIP-2 报文为 **MD5 nonstandard** 认证模式。

说明

如果配置 MD5 认证，则必须配置 MD5 的类型。**usual** 类型支持非标准认证报文格式，**nonstandard** 类型支持 IETF 标准认证报文格式。

符号^#^#和$@$@用来识别变长密码，^#^#作为新密码的前缀和后缀，$@$@作为老密码的前缀和后缀，所以不支持以“$@$@”或“^#^#”同时作为明文密码的起始和结束字符。

----结束

3.7.4 检查配置结果

RIP-2 特性配置成功后，您可以查看到 RIP 的当前运行状态、配置信息及路由信息。

前提条件

完成上述配置后，请执行下面的命令检查配置结果。

操作步骤

- 使用 **display rip** [*process-id* | **vpn-instance** *vpn-instance-name*]命令查看 RIP 的当前运行状态及配置信息。
- 使用 **display rip** *process-id* **database** [**verbose**]命令查看 RIP 发布数据库的所有激活路由。
- 使用 **display rip** *process-id* **route** 命令查看所有从其他路由器学习到的 RIP 路由。

----结束

3.8 调整和优化 RIP 网络

在某些特殊的网络环境中配置 RIP 的一些特性功能，例如配置 RIP 定时器、报文的发送间隔、最大数量，可以对 RIP 网络的性能进行调整和优化。

3.8.1 建立配置任务

在配置调整和优化 RIP 网络之前，了解一些特性的应用环境、配置此特性的前置任务和数据准备，可以帮助您快速、准确地完成配置任务。

应用环境

在某些特殊的网络环境中，需要配置 RIP 的一些特性功能，并需要对 RIP 网络的性能进行调整和优化。通过本节的配置过程，你可以：

- 通过调整 RIP 定时器来改变 RIP 网络的收敛速度；
- 通过调整接口发送更新报文的数量和时间间隔来减少对设备资源的消耗和对网络带宽的占用；
- 配置水平分割或毒性反转来防止路由循环；
- 使能 Replay-protect 功能，保证重启 RIP 进程后，邻居双方的正常通信。
- 在安全性较高网络环境中对报文进行有效性检查和验证；
- 在不支持广播或组播报文的链路上运行 RIP。

前置任务

在调整和优化 RIP 网络之前，需要完成以下任务：

- 配置接口的网络层地址，使相邻节点网络层可达。
- **配置 RIP 的基本功能**。

数据准备

在调整和优化 RIP 网络之前，需要准备以下数据。

序号	数据
1	各定时器的值
2	接口每次发送更新报文的数量与时间间隔
3	最大等价路由条数
4	报文验证的方式和密码
5	RIP 邻居的 IP 地址

3.8.2 配置 RIP 定时器

RIP 有三个定时器：Update、Age 和 Garbage-collect。改变这几个定时器的值，可以影响 RIP 的收敛速度。

背景信息

RIP 有三个定时器：Update、Age 和 Garbage-collect。改变这几个定时器的值，可以影响 RIP 的收敛速度。有关定时器的详细解释，请参见《Huawei AR150&200 系列企业路由器 特性描述 IP 路由》“定时器”。

请在运行 RIP 协议的路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **rip** [*process-id*]，使能 RIP 路由进程，进入 RIP 视图。

步骤 3 执行命令 **timers rip** *update age garbage-collect*，配置 RIP 定时器的值。

说明

- RIP 定时器的值在更改后立即生效。
- 如果这三个定时器的值如果配置不当，会引起路由不稳定。它们的配置值关系是：*update*<*age*，*update*<*garbage-collect*。例如，如果更新时间大于失效时间，那么在更新时间内，如果 RIP 路由发生变化，路由器将无法及时通知邻居。
- 定时器值的调整应考虑网络的性能，并在所有运行 RIP 的路由器上进行统一配置，以免增加不必要的网络流量或引起网络路由震荡。

缺省情况下，Update 定时器是 30 秒，Age 定时器是 180 秒，Garbage-collect 定时器则是 Update 定时器的 4 倍，即 120 秒。

在实际应用中，Garbage-collect 定时器的超时时间并不是固定的，当 Update 定时器设为 30 秒时，Garbage-collect 定时器可能在 90 到 120 秒之间。

这是因为：RIP 在将不可达路由从路由表中彻底删除前，将通过发送 4 次定时更新报文对外发布这条路由（发送时权值设为 16），从而使所有邻居了解这条路由已经处于不可达状态。由于路由变为不可达状态并不总是恰好在一个更新周期的开始，因此，Garbage-collect 定时器的实际时长是 Update 定时器的 3 ～ 4 倍。

----结束

3.8.3 配置报文的发送间隔和发送报文的最大数量

通过设置 RIP 发送更新报文的时间间隔和每次发送报文的最大数量，可以很好的控制路由器用于处理 RIP 更新报文的内存资源。

背景信息

请在运行 RIP 协议的路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **interface** *interface-type interface-number*，进入接口视图。

步骤 3 执行命令 **rip pkt-transmit** { **interval** *interval* | **number** *pkt-count* } *，在该接口上设置 RIP 发送更新报文的时间间隔和每次发送报文的最大数量。

----结束

3.8.4 配置水平分割和毒性反转

通过配置 RIP 的水平分割和毒性反转特性，可以有效的防止路由环路。

背景信息

如果同时配置了毒性反转和水平分割，则只使用毒性反转功能。

在帧中继和 X.25 等 NBMA 网络中，水平分割功能缺省为禁止状态。

请在运行 RIP 协议的路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **interface** *interface-type interface-number*，进入接口视图。

步骤 3 请根据需要选择配置。

- 执行命令 **rip split-horizon**，启动水平分割。
- 执行命令 **rip poison-reverse**，启动毒性反转。

----结束

3.8.5 使能 Replay-protect 功能

通过使能 Replay-protect 功能，可以得到接口 Down 之前所发送 RIP 报文的 Identification，避免双方的 RIP 路由信息不同步、丢失。

背景信息

假设运行 RIP 的接口状态变为 Down 之前发送的最后的 RIP 报文的 Identification 为 X，该接口状态变为 Up 后，再次发送 RIP 报文的 Identification 会变为 0。如果对方没有收到这个 Identification 为 0 的 RIP 报文，那么后续的 RIP 报文都将被丢弃，直到收到 Identification 为 X ＋ 1 的 RIP 报文。这样就会导致双方的 RIP 路由信息不同步、丢失。

通过使能 Replay-protect 功能，可以得到接口 Down 之前所发送 RIP 报文的 Identification，再次发送 RIP 报文的 Identification 会顺次加一，从而避免了上述情况的发生。

请在运行 RIP 协议的路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **interface** *interface-type interface-number*，进入接口视图。

步骤 3 执行命令 **rip authentication-mode md5 nonstandard** *password-key key-id*，配置 RIP-2 使用 MD5 密文的验证方式，验证报文使用非标准报文格式。

说明

配置 **rip replay-protect** 命令前需要先在 RIP 接口上配置 **rip authentication-mode md5 nonstandard**

步骤 4 执行命令 **rip replay-protect**，使能 replay-protect 功能。

说明

- Identification 是 IP 数据报中的标识字段，请参看特性描述 IP 业务部分。
- 在同一视图下多次配置此命令，只有最后一次配置生效。

----结束

3.8.6 配置 RIP 对更新报文进行有效性检查

RIP 对更新报文的检查包括 RIP-1 报文的零域检查和 RIP 更新报文的源地址检查两种，它们的功能和适用范围有所不同。

背景信息

请在运行 RIP 协议的路由器上进行以下配置。

操作步骤

- 对 RIP-1 报文中的零域进行检查
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **rip** [*process-id*]，使能 RIP 路由进程，进入 RIP 视图。
  3. 执行命令 **checkzero**，对 RIP-1 报文的零域进行检查。

RIP-1 报文中的有些字段必须为零，称之为零域。RIP-1 在接收报文时将对零域进行检查，零域的值不为零的 RIP-1 报文将不被处理。

由于 RIP-2 的报文没有零域，此项配置对 RIP-2 无效。

- 对 RIP 更新报文的源地址进行检查
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **rip** [*process-id*]，使能 RIP 路由进程，进入 RIP 视图。
  3. 执行命令 **verify-source**，对更新报文的源地址进行检查。

     RIP 在接收报文时将对源地址进行检查，没有通过检查的报文不被处理。缺省情况下进行源地址检查。

----结束

3.8.7 配置 RIP 邻居

通常情况下，RIP 使用广播或组播地址发送报文。如果在不支持广播或组播报文的链路上运行 RIP，则必须手工指定 RIP 的邻居。

背景信息

通常情况下，RIP 使用广播或组播地址发送报文。如果在不支持广播或组播报文的链路上运行 RIP，则必须在链路两端手工相互指定 RIP 的邻居。

请在运行 RIP 协议的路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **rip** [*process-id*]，使能 RIP 路由进程，进入 RIP 视图。

步骤 3 执行命令 **peer** *ip-address*，配置 RIP 邻居。

----结束

3.8.8 检查配置结果

调整和优化 RIP 网络配置成功后，您可以查看到 RIP 的当前运行状态、路由信息、邻居信息及接口信息。

前提条件

完成上述配置后，请执行下面的命令检查配置结果。

操作步骤

- 使用 **display rip** [*process-id* | **vpn-instance** *vpn-instance-name*]命令查看 RIP 的当前运行状态及配置信息。
- 使用 **display rip** *process-id* **database** [**verbose**]命令查看 RIP 发布数据库的所有激活路由。
- 使用 **display rip** *process-id* **interface** [*interface-type interface-number*] [**verbose**]命令查看 RIP 的接口信息。

- 使用 **display rip** *process-id* **neighbor** [**verbose**]命令查看 RIP 的邻居信息。
- 使用 **display rip** *process-id* **route** 命令查看所有从其他路由器学习到的 RIP 路由。

----结束

3.9 配置 RIP GR

通过 RIP GR 解决 RIP 路由器重启后造成路由计算不准确、报文丢失的问题。

3.9.1 建立配置任务

在实际应用中，为了实现业务转发不受主控板故障的影响，通常在双主控板的硬件环境下配置 RIP GR 才有意义。

应用场景

针对 RIP 协议，为了避免流量中断和主备板切换带来的路由震荡，可以使能 RIP 协议的 GR 特性。GR 是 Graceful Restart 的简称，又被称为平滑重启，是一种用于保证当路由协议重启时数据正常转发并且不影响关键业务的技术。

RIP 通过 GR 重启后，Restarter 路由器和 Helper 路由器之间重新建立邻居关系，更新路由表和转发表，从而实现网络流量不中断，保持网络拓扑稳定。在 GR 过程中，除了主备倒换设备的邻居外的其他路由器感知不到路由变化。

说明

在实际应用中，为了实现业务转发不受主板故障的影响，通常在双主控板的硬件环境下配置 RIP GR 才有意义。

AR150/200 只能作为 Helper 路由器，不能作为 Restarter 路由器。

前置任务

在配置 RIP GR 特性之前，需完成以下任务：

- 配置接口的网络层地址，使相邻节点网络层可达；
- **配置 RIP 的基本功能**，正常建立起邻居关系。

数据准备

在配置 RIP GR 之前，需要准备以下数据。

序号	数据
1	RIP 进程号
2	建立 GR 会话的参数

3.9.2 使能 RIP GR

为了避免流量中断和主备板切换带来的路由震荡，可以使能 RIP 协议的 GR 特性。

背景信息

请在需要使能 GR 特性的路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **rip** [*process-id*]，进入 RIP 视图。

步骤 3 执行命令 **graceful-restart** [**period** *period* | **wait-time** *time* | **planned-only** *time*] *，使能 RIP GR 功能。

当网络中多数路由器都不支持 RIP GR 时，建议配置较长的 **wait-time** *time* 时长，这样可以保证 Restart 路由器有足够的时间学习到正确的路由信息。

----结束

后续处理

如果 Restart 路由器在 **period** *period* 到期前完成了 GR 重启，该路由器会自动退出 GR。但如果 Restart 路由器没有在 **period** *period* 到期前完成 GR，它将被强制退出 GR。

3.9.3 检查配置结果

成功配置 RIP GR 后，您可以查看到 RIP GR 的状态。

前提条件

已经完成 RIP GR 的所有配置。

操作步骤

- 使用 **display rip** *process-id* **graceful-restart** [**verbose**]命令查看 RIP GR 的重启状态。

----结束

3.10 配置 BFD for RIP

当网络中运行高速率数据业务时，通过配置 BFD for RIP，可以实现 RIP 对网络中的故障快速做出响应。

背景信息

通常情况下，RIP 通过定时接收和发送更新报文来保持邻居关系，在老化定时器时间内没有收到邻居发送的更新报文则宣告邻居状态变为 Down。老化定时器的缺省值为 180s，如果出现链路故障，RIP 要经过 180s 才会检测到。如果网络中部署了高速数据业务，在此期间将导致数据大量丢失。

BFD 能够提供毫秒级别的故障检测机制，及时检测到被保护的链路或节点故障，并上报给 RIP 协议，提高 RIP 进程对网络拓扑变化做出响应的速度，从而实现 RIP 路由的快速收敛。

BFD for RIP 由 RIP 协议触发建立 BFD 会话，即 RIP 在建立邻居关系时，将邻居的检测参数通告给 BFD，BFD 根据收到的参数建立起会话。当有链路故障发生时，RIP 进程会

在毫秒级时间内收到邻居不可达的信息，此时，RIP 路由器删除路由表中邻居状态为 Down 的路由信息并启用备份路径来传送消息。

配置 BFD for RIP 有两种方式：

- **RIP 进程下使能 BFD**，当网络中大部分 RIP 接口需要使能 BFD for RIP 时，建议选择此方式。
- **RIP 接口下使能 BFD**，当网络中只有小部分 RIP 接口需要使能 BFD for RIP 时，建议选择此方式。

说明

目前，BFD 会话不会感知路由切换。如果绑定的对端 IP 地址改变引起路由切换到其他链路上，除非原链路转发不通，否则，BFD 不会重新协商。

前置任务

在配置 BFD for RIP 之前，需完成以下任务：

- 配置接口的网络层地址。
- **配置 RIP 的基本功能**。

数据准备

为完成此配置举例，需要准备以下数据：

序号	数据
1	启用 BFD 特性的 RIP 进程号。
2	启用 BFD 特性的接口的类型和编号。
3	（可选）BFD 会话的参数值。 **说明** 推荐使用 BFD 会话的缺省值。

操作步骤

- RIP 进程下使能 BFD
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **bfd**，使能全局 BFD 能力。
  3. 执行命令 **quit**，返回系统视图。
  4. 执行命令 **rip** *process-id*，进入 RIP 视图。
  5. 执行命令 **bfd all-interfaces enable**，打开 RIP 进程 BFD 特性的开关，建立 BFD 会话。

     当配置了全局 BFD 特性，且邻居状态为 Up 时，RIP 为该进程下所有满足上述条件的接口使用缺省的 BFD 参数值建立 BFD 会话。
  6. （可选）执行命令 **bfd all-interfaces** { **min-rx-interval** *min-receive-value* | **min-tx-interval** *min-transmit-value* | **detect-multiplier** *detect-multiplier-value* } *，配置 BFD 参数，指定用于建立 BFD 会话的各个参数值。

     具体参数如何配置取决于网路状况以及对网络可靠性的要求。

- 对于网络可靠性要求较高的链路，可以通过配置减小 BFD 报文实际发送时间间隔；
- 对于网络可靠性要求较低的链路，可以通过配置增大 BFD 报文实际发送时间间隔。

执行该命令后，所有 RIP 接口建立 BFD 会话的参数都会改变。BFD 报文实际发送间隔和检测倍数一般推荐使用缺省值。

7. （可选）执行以下步骤阻塞 RIP 进程下某些接口创建 BFD 会话的功能。
   - 执行命令 **quit**，返回系统视图。
   - 执行命令 **interface** *interface-type interface-number*，进入指定接口的接口视图。
   - 执行命令 **rip bfd block**，阻止接口创建 BFD 会话。

- RIP 接口下使能 BFD
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **bfd**，使能全局 BFD 能力。
  3. 执行命令 **quit**，返回系统视图。
  4. 执行命令 **interface** *interface-type interface-number*，进入指定接口的接口视图。
  5. 执行命令 **rip bfd enable**，打开接口 BFD 特性的开关，建立 BFD 会话。
  6. （可选）执行命令 **rip bfd** { **min-rx-interval** *min-receive-value* | **min-tx-interval** *min-transmit-value* | **detect-multiplier** *detect-multiplier-value* } *，配置 BFD 参数，指定用于建立 BFD 会话的各个参数值。

----结束

检查配置结果

当链路两端均使能 BFD for RIP 特性后，执行命令 **display rip bfd session** { **interface** *interface-type interface-number* | *neighbor-id* | **all** }，可以看到本地路由器上 BFDState 字段显示为 Up。

3.11 配置静态 BFD for RIP

BFD 能够提供轻负荷、快速的链路故障检测，配置静态 BFD for RIP 是实现 BFD 检测功能的一种方式。

背景信息

在 RIP 邻居间建立 BFD 会话可以快速检测链路故障，加快 RIP 进程对网络拓扑变化响应的速度。静态 BFD 可以实现以下两种功能：

- 单臂 BFD：在现网中存在大量设备不支持 BFD 功能，当支持 BFD 的设备与不支持 BFD 的设备对接时，可以通过配置静态 BFD 来实现单臂 BFD 检功能。
- 普通 BFD：在某些对故障响应速度要求高且两端设备都支持 BFD 的链路上，可以在两端配置静态 BFD 来实现普通 BFD 检测功能。

配置静态 BFD 会话需要通过命令行手工配置 BFD 检测，下发 BFD 会话建立请求。

前置任务

配置静态 BFD for RIP 的前置任务:

- 配置接口的网络层地址，使相邻节点网络层可达。
- **配置 RIP 的基本功能**。

数据准备

为完成此配置举例，需准备如下的数据：

序号	数据
1	RIP 进程号
2	启用 BFD 特性的接口的类型和编号

操作步骤

步骤 1 使能全局 BFD

1. 执行命令 **system-view**，进入系统视图。
2. 执行命令 **bfd**，使能全局 BFD 能力。
3. 执行命令 **quit**，返回系统视图。

说明

使能单臂 BFD 功能，请执行步骤 2，使能普通 BFD 功能，请执行步骤 3。

步骤 2 配置单臂 BFD 检测

1. 执行命令 **bfd** *cfg-name* **bind peer-ip** *peer-ip* **interface** *interface-type interface-number* **one-arm-echo**，创建 BFD 绑定。

   指定了对端 IP 和本端接口，表示检测单跳链路，即检测以该接口为出接口、以 **peer-ip** 为下一跳地址的一条固定路由。
2. 执行命令 **discriminator local** *discr-value*，配置本地标识符。
3. （可选）执行命令 **min-echo-rx-interval** *interval* 配置单臂 BFD 的最小接收间隔。
4. 执行命令 **commit**，提交配置。
5. 执行命令 **quit**，返回系统视图。

步骤 3 配置普通 BFD 检测

1. 执行命令 **bfd** *cfg-name* **bind peer-ip** *ip-address* [**interface** *interface-type interface-number*]，创建 BFD 绑定。

   指定了对端 IP 和本端接口，表示检测单跳链路，即检测以该接口为出接口、以 **peer-ip** 为下一跳地址的一条固定路由。
2. 配置标识符：
   - 执行命令 **discriminator local** *discr-value*，配置本地标识符。
   - 执行命令 **discriminator remote** *discr-value*，配置远端标识符。

   BFD 会话两端设备的本地标识符和远端标识符需要分别对应，否则会话无法正确建立。并且，本地标识符和远端标识符配置成功后不可修改。

说明

本地标识符 **local** *discr-value* 对应对端设备的远端标识符 **remote** *discr-value*，本地的远端标识符 **remote** *discr-value* 对应对端设备的本地标识符 **local** *discr-value*。

3. 执行命令 **commit**，提交配置。
4. 执行命令 **quit**，返回系统视图。

步骤 4 使能接口静态 BFD

1. 执行命令 **interface** *interface-type interface-number*，进入指定接口的接口视图。
2. 执行命令 **rip bfd static**，使能接口的静态 BFD 特性。
3. 执行命令 **quit**，返回系统视图。

----结束

检查配置结果

配置完成静态 BFD for RIP 之后，使用 **display rip** *process-id* **interface** [*interface-type interface-number*] **verbose** 命令可以查看指定接口上 BFD for RIP 的配置信息。

执行命令 **display rip** *process-id* **interface** *interface-type interface-number* **verbose**，可以看到在接口 GigabitEthernet1/0/0 上已经使能静态 BFD 特性。如：

```
<Huawei> display rip 1 interface ethernet1/0/0 verbose
 Ethernet1/0/0 (81.1.1.1)
  State    : UP        MTU : 500
  Metricin : 0
  Metricout : 1
  Input    : Enabled   Output      : Enabled
  Protocol : RIPv1 Compatible (Non-Standard)
  Send     : RIPv1 Packets
  Receive  : RIPv1 Packets, RIPv2 Multicast and Broadcast Packets
  Poison-reverse      : Disabled
  Split-Horizon       : Enabled
  Authentication type : None
  Replay Protection   : Disabled
  BFD                 : Enabled (Static)
  Summary Address (es):
    1.1.0.0/16
```

3.12 配置 RIP 的网管功能

通过配置 RIP 和 MIB 绑定，可以通过网管的环境来查看和配置 RIP。

3.12.1 建立配置任务

在配置 RIP 和 MIB 绑定之前，了解该特性的应用环境、配置此特性的前置任务和数据准备，可以帮助您快速、准确地完成配置任务。

应用环境

通过本节的配置过程，可以实现 RIP 和 MIB 的绑定。

前置任务

在配置 RIP 的网管功能之前，需要完成以下任务：

- 配置接口的网络层地址，使相邻节点网络层可达。
- **配置 RIP 的基本功能**。

数据准备

无

3.12.2 配置 RIP 和 MIB 绑定

配置 RIP 和 MIB 绑定时，需要指定 RIP 的进程号。

背景信息

请在运行 RIP 协议的路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **rip mib-binding** *process-id*，设置 RIP 和 MIB 绑定。

该命令用来设置 MIB 和 RIP 进程号的绑定关系，指定接受 SNMP 请求的 RIP 进程号。

----结束

3.12.3 检查配置结果

RIP 和 MIB 绑定成功后，您可以查看到 RIP 的当前配置信息里含有相关绑定信息。

前提条件

完成上述配置后，请执行下面的命令检查配置结果。

操作步骤

步骤 1 使用 **display current-configuration** 命令查看路由器当前生效的配置参数。

----结束

3.13 维护 RIP

复位 RIP 连接、清除 RIP 的统计信息。

3.13.1 复位 RIP

通过重启 RIP，达到复位的目的。

背景信息

复位 RIP 连接（执行 **reset rip** 命令）会导致路由器之间的 RIP 邻接关系中断。务必仔细确认是否必须执行复位 RIP 连接的操作。

如果需要复位 RIP 连接，可在用户视图下选择执行以下命令。

操作步骤

- 在用户视图下执行 **reset rip** *process-id* **configuration** 复位 RIP 特定进程的系统配置参数。当 RIP 进程启动时，所有配置参数将采用缺省值。

----结束

3.13.2 清除 RIP

清除 RIP 包括清除 RIP 的计数器统计信息。

背景信息

清除 RIP 的信息后，以前的信息将无法恢复，务必仔细确认。

在确认需要清除 RIP 的运行信息后，请在用户视图下执行以下命令。

操作步骤

- 在用户视图下执行 **reset rip** *process-id* **statistics** [**interface** { **all** | *interface-type interface-number* [**neighbor** *neighbor-ip-address*] }]命令清除由 RIP 进程维护的计数器的统计数据。

----结束

3.14 配置举例

在实际组网中，RIP 的版本不同以及是否引入外部路由将影响路由学习。

3.14.1 配置 RIP 版本示例

使用 RIP 之前，需要配置 RIP 的基本功能及版本。可以通过相关命令进行查看配置结果。

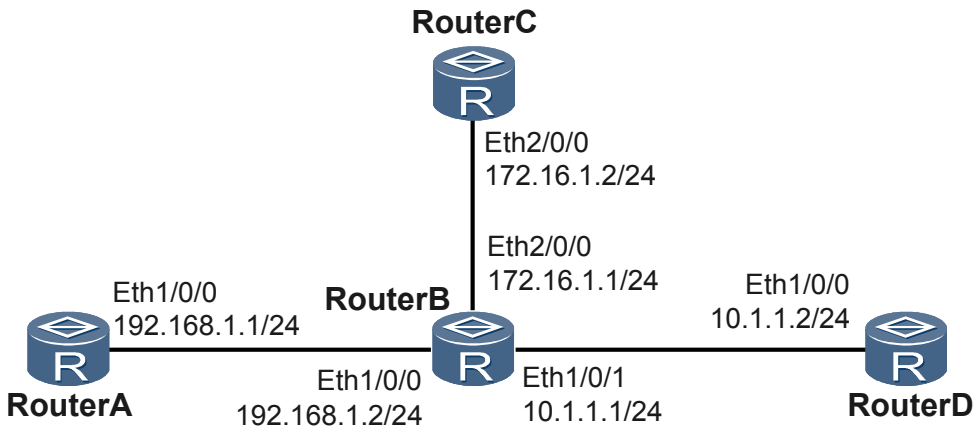
组网需求

如图 3-1 所示，要求在 RouterA、RouterB、RouterC 和 RouterD 的所有接口上使能 RIP，并使用 RIP-2 进行网络互连。

说明

AR150/200 仅可作为 RouterA、RouterC 或 RouterD。

图 3-1 配置 RIP 版本组网图



配置思路

采用如下的思路配置 RIP 的版本：

1. 配置各接口的 IP 地址，使网络可达。
2. 在各路由器上使能 RIP，配置 RIP 基本功能。
3. 在各路由器上配置 RIP-2 版本，查看精确的子网掩码信息。

数据准备

为完成此配置例，需准备如下的数据：

- 在 RouterA 上指定使能 RIP 的网段 192.168.1.0。
- 在 RouterB 上指定使能 RIP 的网段 192.168.1.0，172.16.0.0，10.0.0.0。
- 在 RouterC 上指定使能 RIP 的网段 172.16.0.0。
- 在 RouterD 上指定使能 RIP 的网段 10.0.0.0。
- 在 RouterA、RouterB、RouterC 和 RouterD 上配置 RIP-2 版本。

操作步骤

步骤 1 配置各接口的 IP 地址（略）

步骤 2 配置 RIP 基本功能

配置 RouterA。

```
[RouterA] rip
```

```
[RouterA-rip-1] network 192.168.1.0
[RouterA-rip-1] quit
```

配置 RouterB。

```
[RouterB] rip
[RouterB-rip-1] network 192.168.1.0
[RouterB-rip-1] network 172.16.0.0
[RouterB-rip-1] network 10.0.0.0
[RouterB-rip-1] quit
```

配置 RouterC。

```
[RouterC] rip
[RouterC-rip-1] network 172.16.0.0
[RouterC-rip-1] quit
```

配置 RouterD。

```
[RouterD] rip
[RouterD-rip-1] network 10.0.0.0
[RouterD-rip-1] quit
```

查看 RouterA 的 RIP 路由表。

```
[RouterA] display rip 1 route
 Route Flags: R - RIP
              A - Aging, S - Suppressed, G - Garbage-collect
------------------------------------------------------------------------------
 Peer 192.168.1.2  on Ethernet1/0/0
      Destination/Mask        Nexthop      Cost   Tag     Flags   Sec
         10.0.0.0/8           192.168.1.2      1     0        RA      14
         172.16.0.0/16        192.168.1.2      1     0        RA      14
```

从路由表中可以看出，RIP-1 发布的路由信息使用的是自然掩码。

步骤 3 配置 RIP 的版本

在 RouterA 上配置 RIP-2。

```
[RouterA] rip
[RouterA-rip-1] version 2
[RouterA-rip-1] quit
```

在 RouterB 上配置 RIP-2。

```
[RouterB] rip
[RouterB-rip-1] version 2
[RouterB-rip-1] quit
```

在 RouterC 上配置 RIP-2。

```
[RouterC] rip
[RouterC-rip-1] version 2
[RouterC-rip-1] quit
```

在 RouterD 上配置 RIP-2。

```
[RouterD] rip
[RouterD-rip-1] version 2
[RouterD-rip-1] quit
```

步骤 4 验证配置结果

查看 RouterA 的 RIP 路由表。

```
[RouterA] display rip 1 route
  Route Flags: R - RIP
               A - Aging, S - Suppressed, G - Garbage-collect
```

```
------------------------------------------------------------------------------
 Peer 192.168.1.2  on Ethernet1/0/0
      Destination/Mask        Nexthop      Cost   Tag      Flags   Sec
         10.1.1.0/24          192.168.1.2       1    0         RA       32
         172.16.1.0/24        192.168.1.2       1    0         RA       32
```

从路由表中可以看出，RIP-2 发布的路由中带有更为精确的子网掩码信息。

----结束

配置文件

- RouterA 的配置文件

```
#
 sysname RouterA
#
interface Ethernet1/0/0
 ip address 192.168.1.1 255.255.255.0
#
rip 1
 version 2
 network 192.168.1.0
#
return
```

- RouterB 的配置文件

```
#
 sysname RouterB
#
interface Ethernet1/0/0
 ip address 192.168.1.2 255.255.255.0
#
interface Ethernet1/0/1
 ip address 10.1.1.1 255.255.255.0
#
interface Ethernet2/0/0
 ip address 172.16.1.1 255.255.255.0
#
rip 1
 version 2
 network 192.168.1.0
 network 172.16.0.0
 network 10.0.0.0
#
return
```

- RouterC 的配置文件

```
#
 sysname RouterC
#
interface Ethernet2/0/0
 ip address 172.16.1.2 255.255.255.0
#
rip 1
 version 2
 network 172.16.0.0
#
return
```

- RouterD 的配置文件

```
#
 sysname RouterD
#
interface Ethernet1/0/0
 ip address 10.1.1.2 255.255.255.0
#
rip 1
 version 2
```

```
 network 10.0.0.0
#
return
```

4 RIPng 配置

关于本章

RIPng 是在 IPv6 网络中应用的 RIP 协议，并在原 RIP 协议基础上进行了一些扩展。

4.1 RIPng 概述
RIPng 协议是基于 D-V（Distance Vector，距离矢量）算法的路由协议，用跳数来衡量到达目的主机的距离。

4.2 AR150/200 支持的 RIPng 特性
AR150/200 中支持的 RIPng 特性包括：水平分割和毒性逆转。

4.3 配置 RIPng 的基本功能
配置 RIPng 的基本功能主要包括创建 RIPng 进程和在接口下使能 RIPng，是能够使用 RIPng 特性的前提。

4.4 配置 RIPng 的路由属性
通过配置 RIPng 路由属性，可以改变路由器或 RIPng 的选路策略。

4.5 控制 RIPng 路由信息的发布
对 RIPng 路由信息的发布进行精确的控制，可以满足复杂网络环境中的需要。

4.6 控制 RIPng 路由信息的接收
对 RIPng 路由信息的接收进行精确的控制，可以满足复杂网络环境中的需要。

4.7 调整优化 RIPng 网络
和 RIP 协议类似，您也可以配置 RIPng 定时器、水平分割、毒性逆转、零域检查，对 RIPng 网络的性能进行调整和优化。

4.8 维护 RIPng 配置
介绍如何清除 RIPng 的统计信息。

4.9 配置举例
在实际组网中，RIPng 的不同特性有不同的应用。

4.1 RIPng 概述

RIPng 协议是基于 D-V（Distance Vector，距离矢量）算法的路由协议，用跳数来衡量到达目的主机的距离。

RIPng（Routing Information Protocol Next Generation）是对原来的 IPv4 网络中 RIP-2 协议的扩展。大多数 RIP 的概念都可以用于 RIPng。

RIPng 对 RIP 协议的扩展

为了在 IPv6 网络中应用，RIPng 对原有的 RIP 协议进行了修改：

- UDP 端口号：使用 UDP 的 521 端口发送和接收路由信息。
- 组播地址：使用 FF02::9 作为链路本地范围内的 RIPng 路由器组播地址。
- 前缀长度：目的地址使用 128 比特的前缀长度（掩码长度）。
- 下一跳地址：使用 128 比特的 IPv6 地址。
- 源地址：使用链路本地地址作为源地址发送 RIPng 路由信息更新报文。

RIPng 工作机制

RIPng 协议是基于 D-V（Distance Vector，距离矢量）算法的路由协议。它通过 UDP 报文交换路由信息，使用的端口号为 521。RIPng 协议用跳数来衡量到达目的主机的距离（也称为度量值或开销）。在 RIPng 协议中，从一个路由器到其直连网络的跳数为 0，而通过另一台路由器到达一个网络的跳数为 1，如此类推。当跳数大于或等于 16 时，目的网络或主机就被定义为不可达。

缺省情况下，RIPng 每 30 秒发送一个路由刷新报文。如果在 180 秒内没有收到网络邻居的路由刷新报文，RIPng 将从邻居学到的所有路由标识为不可达。如果在 300 秒内没有收到邻居的路由刷新报文，RIPng 将从路由表中删除这些路由。

为了避免路由环路，RIPng 支持水平分割和毒性逆转。此外，RIPng 也可以从其它的路由协议中引入路由。

每台运行 RIPng 的路由器都管理着路由数据库，包括到达网络中所有可达目的地址的路由项。这些路由项包括下列信息：

- 目的地址：主机或网络的 IPv6 地址。
- 下一跳地址：要到达目的地址路由器所通过的下一个路由器地址。
- 接口：转发 IP 报文所通过的接口。
- 开销：到达目的地址所经过的跳数。整数形式，取值范围是 0 ～ 16，取 16 时目的网络或主机就被定义为不可达。
- 定时器：从上次更改路由项到现在的时长。如果更改路由项，定时器将重置为 0。
- 路由标记：用来将内部路由协议与外部路由协议区别开来的标签。

4.2 AR150/200 支持的 RIPng 特性

AR150/200 中支持的 RIPng 特性包括：水平分割和毒性逆转。

AR150/200 可以通过配置 RIPng 的路由属性改变 RIPng 的选路策略，并对 RIPng 路由信息的发布和接收进行更为精确的控制，以满足复杂网络环境中的需要。在某些特殊的网络环境中可以配置 RIPng 的一些特性功能，从而对 RIPng 网络的性能进行调整和优化。

说明

RIPng 功能使用 License 授权，缺省情况下，设备的 RIPng 功能受限无法使用。如果需要使用 RIPng 功能，请联系华为办事处申请并购买如下 License，

- AR150&200 数据业务增值包

4.3 配置 RIPng 的基本功能

配置 RIPng 的基本功能主要包括创建 RIPng 进程和在接口下使能 RIPng，是能够使用 RIPng 特性的前提。

4.3.1 建立配置任务

为了让路由器学到接口所在网段的路由，必须保证接口的链路状态为 Up。

应用环境

RIPng 基本能力的配置任务包括 RIPng 特性配置中涉及的基本配置，完成本节的配置就可以使用 RIPng 特性。

在 RIPng 的配置中，应该首先在系统视图下使能 RIPng。因为即使在接口视图下可以配置 RIPng 相关的命令，但这些命令只有在系统视图下使能了 RIPng 后才会生效。

前置任务

在配置 RIPng 基本功能之前，需完成以下任务：

- 使能路由器的 IPv6 能力。
- 配置接口的网络层地址，使相邻节点的网络层可达。

数据准备

在配置 RIPng 的基本功能之前，需准备以下数据。

序号	数据
1	RIPng 进程号
2	使能 RIPng 的接口

4.3.2 使能 RIPng 并进入 RIPng 视图

创建 RIPng 进程是进行 RIPng 相关配置的前提。通过此步骤，您还可以进入 RIPng 视图，进行相关配置。

背景信息

请在需要运行 RIPng 协议的每台路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **ripng** [*process-id*]，使能 RIPng 路由进程，进入 RIPng 视图。

如果只运行一个 RIPng 路由进程，则通常在命令中不指定参数 *process-id*，即默认 *process-id* 为 1。

该进程取消后，在接口上配置的有关于 **ripng** *process-id* **enable** 的命令要重新配置。

----结束

4.3.3 在接口下使能 RIPng

把接口使能到 RIPng 进程中，可以通过 RIPng 交换该接口的路由信息。

背景信息

请在需要运行 RIPng 的每台路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **interface** *interface-type interface-number*，进入接口视图。

这里的接口是路由器的网络侧的接口，即与其他设备互连的接口。为了让路由器学到接口所在网段的路由，必须保证接口的链路状态为 Up。

步骤 3 执行命令 **ripng** *process-id* **enable**，在指定接口上使能 RIPng。

说明

在接口视图下，如果没有使能 IPv6 功能，则此命令不可执行。

在 ATM 接口上不支持此命令。

如果一台路由器有多个接口与其他设备互连，请重复步骤 2 到步骤 3。

----结束

4.3.4 检查配置结果

RIPng 基本功能配置成功后，您可以查看到 RIPng 的配置信息及路由信息。

前提条件

完成上述配置后，请执行下面的命令检查配置结果。

操作步骤

- 使用 **display ripng** [*process-id*]命令查看 RIPng 进程的配置信息。

- 使用 **display ripng** *process-id* **route** 命令查看所有从其他路由器学习到的 RIPng 路由。
- 使用 **display default-parameter ripng** 命令查看 RIPng 进程的缺省配置信息。
- 使用 **display ripng** *process-id* **statistics interface** { **all** | *interface-type interface-number* [**verbose** | **neighbor** *neighbor-ipv6-address*] }命令查看 RIPng 接口的数据信息。

----结束

4.4 配置 RIPng 的路由属性

通过配置 RIPng 路由属性，可以改变路由器或 RIPng 的选路策略。

4.4.1 建立配置任务

RIPng 的路由属性包括 RIPng 协议的优先级、接口的度量值。

应用环境

在实际应用中，可以通过配置 RIPng 的路由属性改变 RIPng 的选路策略，以满足复杂网络环境中的需要。通过本节的配置过程，你可以：

- 通过调整 RIPng 接口的附加度量值来影响路由的选择；
- 当多个路由协议发现相同的路由时，通过配置 RIPng 的协议优先级来改变路由协议的优先顺序；
- 使用多条等价路由进行负载分担。

前置任务

在配置 RIPng 的路由属性之前，需完成以下任务：

- 配置接口的网络层地址，使相邻节点网络层可达。
- **配置 RIPng 的基本功能**。

数据准备

在配置 RIPng 的路由属性之前，需要准备以下数据。

序号	数据
1	接口的附加度量值
2	RIPng 协议优先级的值
3	最大等价路由条数

4.4.2 配置 RIPng 的协议优先级

当有多种协议的路由存在时，通过配置 RIPng 的协议优先级，可以调整路由器选择最优路由。

背景信息

任何路由协议都具备特有的协议优先级，可使路由策略在不同的协议中选择最佳路由。可以手工设置 RIPng 协议的优先级，设置的值越大，其优先级就会越低。

请在运行 RIPng 的路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **ripng** [*process-id*]，使能 RIPng 路由进程，进入 RIPng 视图。

步骤 3 执行命令 **preference** { *preference* | **route-policy** *route-policy-name* } *，设置 RIPng 优先级。

----结束

4.4.3 配置接口的附加度量值

对于 RIPng 接收和发布路由，可通过不同的命令配置附加度量值。

背景信息

附加路由度量值是在 RIPng 路由原来度量值的基础上所增加的度量值（跳数）。

- **ripng metricin** 用于在接收到路由后，给其增加一个附加度量值，再加入路由表中，使得路由表中的度量值发生变化。运行该命令会影响到本地设备和其他设备的路由选择。
- **ripng metricout** 用于自身路由的发布，发布时增加一个附加的度量值，但路由表中的度量值不会发生变化。运行该命令不会影响本地设备的路由选择，但是会影响其他设备的路由选择。

当用 ipv6-prefix 方式来设置接口发送 RIPng 路由增加的度量值时，指定 *value1* 为通过过滤策略的 RIPng 路由增加的度量值，没有通过过滤的 RIPng 路由增加的度量值为 1。

请在运行 RIPng 的路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **interface** *interface-type interface-number*，进入接口视图。

步骤 3 执行命令 **ripng metricin** *value*，设置接收 RIPng 报文时的路由附加度量值。

步骤 4 执行命令 **ripng metricout** { *value* | **ipv6-prefix** *ipv6-prefix-name value1* }，设置发送 RIPng 报文时的路由附加度量值。

说明

如果路由器有多个接口与其他 RIPng 路由器连接，则需重复步骤 2 到步骤 4，直到配置完所有链路的开销。

----结束

4.4.4 配置 RIPng 的最大等价路由条数

通过配置 RIPng 最大等价路由条数，可以调整进行负载分担的路由数目。

背景信息

请在运行 RIPng 的路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **ripng** [*process-id*]，进入 RIPng 视图。

步骤 3 执行命令 **maximum load-balancing** *number*，配置 RIPng 的最大等价路由条数。

----结束

4.4.5 检查配置结果

RIPng 的路由属性配置成功后，您可以查看到 RIPng 的配置信息及路由信息。

前提条件

完成上述配置后，请执行下面的命令检查配置结果。

操作步骤

- 使用 **display ripng** [*process-id*]命令查看 RIPng 进程的当前运行状态及配置信息。
- 使用 **display ripng** *process-id* **database** 命令查看 RIPng 发布数据库中的所有激活路由。
- 使用 **display ripng** *process-id* **route** 命令查看所有从其他路由器学习到的 RIPng 路由。

----结束

4.5 控制 RIPng 路由信息的发布

对 RIPng 路由信息的发布进行精确的控制，可以满足复杂网络环境中的需要。

4.5.1 建立配置任务

RIPng 路由信息的发布可通过路由聚合、发布缺省路由、引入外部路由信息的方式。

应用环境

在实际应用中，有时候需要对 RIPng 路由信息的发布进行更为精确的控制，以满足复杂网络环境中的需要。通过本节的配置过程，你可以：

- 向邻居发布缺省路由；
- 抑制接口发送 RIPng 更新报文；

- 在多路由协议环境中引入外部路由并对发布的路由进行过滤。

前置任务

在控制 RIPng 路由信息的发布之前，需完成以下任务：

- 配置接口的网络层地址，使相邻节点网络层可达。
- **配置 RIPng 的基本功能**。

数据准备

在控制 RIPng 路由信息的发布之前，需要准备以下数据。

序号	数据
1	需要发布的缺省路由的度量值
2	要引入的外部路由协议名称和进程号

4.5.2 配置 RIPng 路由聚合

配置 RIPng 路由器在接口发布聚合 IPv6 地址，可以节省路由表中 RIPng 路由的空间。您还可以通过配置参数，禁止从接口学习到相同的聚合路由。

背景信息

配置 RIPng 路由器在接口上发布聚合后的 IPv6 前缀信息，而不是具体路由。

请在运行 RIPng 的路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **interface** *interface-type interface-number*，进入接口视图。

步骤 3 执行命令 **ripng summary-address** *ipv6-address prefix-length* [**avoid-feedback**]，配置 RIPng 路由聚合。

----结束

4.5.3 配置 RIPng 发布缺省路由

RIPng 缺省路由的发布有两种方式，您可以根据组网的实际情况配置发布缺省路由。您还可以同时指定发布的缺省路由的开销值。

背景信息

请在运行 RIPng 的路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **interface** *interface-type interface-number*，进入接口视图。

步骤 3 执行命令 **ripng default-route** { **only** | **originate** } [**cost** *cost*]，配置 RIPng 发布缺省路由。

请根据组网的实际情况配置发布缺省路由。

- **only**：只发布 IPv6 缺省路由（::/0），抑制其它路由的发布。
- **originate**：发布 IPv6 缺省路由（::/0），但不影响其它路由的发布。

生成的 RIPng 缺省路由将强制通过指定接口的路由更新报文发布出去，该路由的发布不考虑其是否已经存在于 IPv6 路由表中。

----结束

4.5.4 配置 RIPng 引入外部路由的缺省权值

通过此配置，可以在 RIPng 从其他路由协议引入路由但没有指定权值的情况下，为引入的外部路由设置缺省 RIPng 权值。

背景信息

请在运行 RIPng 的路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **ripng** [*process-id*]，进入 RIPng 视图。

步骤 3 执行命令 **default-cost** *cost*，设置引入的外部路由的缺省权值。

该命令用于从其他路由协议引入路由但没有指定权值的情况下，为引入的外部路由设置缺省 RIPng 权值。

----结束

4.5.5 配置 RIPng 引入外部路由

和 RIP 协议类似，RIPng 也可以引入外部路由，提供更多路由信息。

背景信息

请在运行 RIPng 的路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **ripng** [*process-id*]，进入 RIPng 视图。

步骤 3 （可选）执行命令 **default-cost** *cost*，设置引入的外部路由的缺省权值。

步骤 4 执行命令 **import-route** *protocol* [*process-id*] [**cost** *cost* | **route-policy** *route-policy-name*] *，引入外部路由。

如果在路由引入过程中没有显式的指定路由开销，将采用缺省路由开销。

步骤 5 （可选）执行命令 **filter-policy ipv6-prefix** *ipv6-prefix-name* **export** [*protocol* [*process-id*]]，配置 RIPng 对引入的路由信息进行过滤。

RIPng 可以通过 IPv6 前缀列表对引入的路由进行过滤，只将符合条件的路由发布（Export）给邻居 RIPng 路由器。如果在此命令中没有指定 *protocol*，则对所有要发布的路由信息进行过滤，包括已引入的路由和本地 RIPng 路由（相当于直连路由）。

----结束

4.5.6 检查配置结果

控制 RIPng 路由信息的发布配置成功后，您可以查看到 RIPng 的路由信息。

前提条件

完成上述配置后，请执行下面的命令检查配置结果。

操作步骤

- 使用 **display ripng** *process-id* **database** 命令查看 RIPng 发布数据库中的所有激活路由。
- 使用 **display ripng** *process-id* **route** 命令查看所有从其他路由器学习到的 RIPng 路由。

----结束

4.6 控制 RIPng 路由信息的接收

对 RIPng 路由信息的接收进行精确的控制，可以满足复杂网络环境中的需要。

4.6.1 建立配置任务

在配置控制 RIPng 路由信息的接收之前，了解此特性的应用环境、配置此特性的前置任务和数据准备，可以帮助您快速、准确地完成配置任务。

应用环境

在实际应用中，有时候需要对 RIPng 路由信息的接收进行更为精确的控制，以满足复杂网络环境中的需要。通过本节的配置过程，你可以：

- 禁止接口接收 RIPng 的更新报文；
- 对接收的路由信息进行过滤；
- 在多路由协议环境中引入外部路由并对接收的路由进行过滤。

前置任务

在控制 RIPng 路由信息的接收之前，需要完成以下任务：

- 配置接口的网络层地址，使相邻节点网络层可达。
- **配置 RIPng 的基本功能**。

数据准备

在控制 RIPng 路由信息的接收之前，需要准备以下数据。

序号	数据
1	对路由信息过滤时所需要的相关过滤列表

4.6.2 配置 RIPng 对接收的路由信息进行过滤

通过 IPv6 前缀列表对接收的路由信息过滤，可以有选择的接收路由信息。

背景信息

请在运行 RIPng 的路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **ripng** [*process-id*]，进入 RIPng 视图。

步骤 3 执行命令 **filter-policy ipv6-prefix** *ipv6-prefix-name* **import**，对接收的路由信息进行过滤。

可以使用 IPv6 前缀列表对接收的路由信息进行过滤，只有通过过滤的路由才能被加入到 RIPng 路由表。

----结束

4.6.3 检查配置结果

控制 RIPng 路由信息的接收配置成功后，您可以查看到 RIPng 的路由信息。

前提条件

完成上述配置后，请执行下面的命令检查配置结果。

操作步骤

- 使用 **display ripng** *process-id* **database** 命令查看 RIPng 发布数据库中的所有激活路由。
- 使用 **display ripng** *process-id* **route** 命令查看所有从其他路由器学习到的 RIPng 路由。

----结束

4.7 调整优化 RIPng 网络

和 RIP 协议类似，您也可以配置 RIPng 定时器、水平分割、毒性逆转、零域检查，对 RIPng 网络的性能进行调整和优化。

4.7.1 建立配置任务

在配置调整和优化 RIPng 网络之前，了解一些特性的应用环境、配置此特性的前置任务和数据准备，可以帮助您快速、准确地完成配置任务。

应用环境

在某些特殊的网络环境中，需要配置 RIPng 的一些特性功能，并需要对 RIPng 网络的性能进行调整和优化。通过本节的配置过程，你可以：

- 通过调整 RIPng 定时器来改变 RIPng 网络的收敛速度；
- 配置水平分割或毒性反转来防止路由循环。

前置任务

在调整和优化 RIPng 网络之前，需要完成以下任务：

- 配置接口的网络层地址，使相邻节点网络层可达。
- **配置 RIPng 的基本功能**。

数据准备

在调整和优化 RIPng 网络之前，需要准备以下数据。

序号	数据
1	各定时器的值

4.7.2 配置 RIPng 定时器

RIPng 有三个定时器：Update、Age 和 Garbage-collect。请注意 RIPng 三个定时器的值如果配置不当，会引起路由不稳定。

背景信息

说明

请注意 RIPng 三个定时器的值如果配置不当，会引起路由不稳定。它们的关系是 *update<age*，*update<garbage-collect*。例如，如果更新时间大于失效时间，那么在更新时间内，如果 RIPng 路由发生变化，路由器将无法及时通知邻居。

缺省情况下，Update 定时器的值为 30 秒，Age 定时器的值为 180 秒，Garbage-collect 定时器的值为 120 秒。

请在运行 RIPng 的路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **ripng** [*process-id*]，进入 RIPng 视图。

步骤 3 执行命令 **timers ripng** *update age garbage-collect*，设置 RIPng 定时器。

----结束

4.7.3 配置报文的发送间隔和发送报文的最大数量

通过配置报文的发送间隔和发送报文的最大数量，可以优化 RIPng 性能。

背景信息

请在运行 RIPng 协议的路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **interface** *interface-type interface-number*，进入接口视图。

步骤 3 执行命令 **ripng pkt-transmit** { **interval** *interval* | **number** *pkt-count* }*，在该接口上设置 RIPng 发送更新报文的时间间隔和每次发送报文的最大数量。

----结束

4.7.4 配置水平分割和毒性反转

通过配置 RIPng 的水平分割和毒性反转特性，可以有效的防止路由环路。

背景信息

水平分割，即从某接口学习到的路由将不会再由该接口发送出去，这在某种程度上避免了路由循环。当路由器传递路由需要经过帧中继或 X.25 等 NBMA 网络时，水平分割功能缺省情况下为禁止状态。

当毒性反转被使能时，从一个接口学到的路由还可以从这个接口向外发布，但此时这些路由的 metric 值已设置为 16，即不可达。

当毒性反转和水平分割都使能的情况下，只采用毒性反转功能。

请在运行 RIPng 的路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **interface** *interface-type vlan-id*，进入接口视图。

步骤 3 请根据需要选择配置。

- 执行命令 **ripng split-horizon** ，使能水平分割功能。

- 执行命令 **ripng poison-reverse**，使能毒性反转功能。

----结束

4.7.5 使能 RIPng 报文的零域检查

RIPng 报文中的一些字段必须配置为 0，也称为零域。如果零域中的值不为零，这些报文将被忽略，不做处理。

背景信息

请在运行 RIPng 的路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **ripng** [*process-id*]，进入 RIPng 视图。

步骤 3 执行命令 **checkzero**，使能 RIPng 报文的零域检查。

----结束

4.7.6 检查配置结果

调整和优化 RIP 网络配置成功后，您可以查看到 RIPng 的路由信息、邻居信息及接口信息。

前提条件

完成上述配置后，请执行下面的命令检查配置结果。

操作步骤

- 使用 **display ripng** [*process-id*]命令查看 RIPng 进程的配置信息。
- 使用 **display ripng** *process-id* **database** [**verbose**]命令查看 RIPng 发布数据库中的所有激活路由。
- 使用 **display ripng** *process-id* **interface** [*interface-type interface-number*] [**verbose**]命令查看 RIPng 的接口信息。
- 使用 **display ripng** *process-id* **neighbor** [**verbose**]命令查看 RIPng 的邻居信息。
- 使用 **display ripng** *process-id* **route** 命令查看所有从其他路由器学习到的 RIPng 路由。

----结束

4.8 维护 RIPng 配置

介绍如何清除 RIPng 的统计信息。

4.8.1 清除 RIPng

介绍如何清楚 RIPng 的相关配置信息，清除 RIPng 包括清除 RIPng 的计数器统计信息。

背景信息

注意

清除 RIPng 的信息后，之前的信息将无法恢复，务必仔细确认。

在确认需要清除 RIPng 的运行信息后，请在用户视图下执行以下命令。

操作步骤

- 在用户视图下执行命令 **reset ripng** *process-id* **statistics** [**interface** { **all** | *interface-type interface-number* [**neighbor** *neighbor-ip-address*] }]清除由 RIPng 进程维护的计数器的统计数据。

----结束

4.9 配置举例

在实际组网中，RIPng 的不同特性有不同的应用。

4.9.1 配置 RIPng 的基本功能示例

使用 RIPng 之前，需要配置 RIPng 的基本功能。

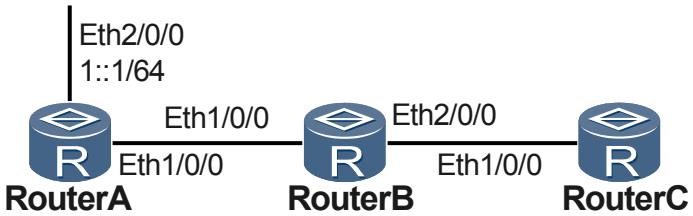
组网要求

如图 4-1 所示，图中所有 IPv6 地址的前缀长度都为 64，且相邻路由器之间使用 IPv6 链路本地地址连接。要求所有路由器通过 RIPng 来学习网络中的 IPv6 路由信息。

说明

AR150/200 仅可作为 RouterC。

图 4-1 配置 RIPng 基本功能组网图



配置思路

采用如下的思路配置 RIPng：

1. 配置各接口的 IPv6 地址。
2. 在各路由器上使能 RIPng 基本功能，使各路由器互通。

数据准备

为完成此配置例，需准备如下的数据：

- 在各路由器上使能进程 RIPng1。

操作步骤

步骤 1 配置各接口的 IPv6 地址（略）

步骤 2 配置 RIPng 的基本功能

配置 RouterA。

```
[RouterA] ripng 1
[RouterA-ripng-1] quit
[RouterA] interface ethernet 2/0/0
[RouterA-Ethernet2/0/0] ripng 1 enable
[RouterA-Ethernet2/0/0] quit
[RouterA] interface ethernet 1/0/0
[RouterA-Ethernet1/0/0] ripng 1 enable
[RouterA-Ethernet1/0/0] quit
```

配置 RouterB。

```
[RouterB] ripng 1
[RouterB-ripng-1] quit
[RouterB] interface ethernet 1/0/0
[RouterB-Ethernet1/0/0] ripng 1 enable
[RouterB-Ethernet1/0/0] quit
[RouterB] interface ethernet 2/0/0
[RouterB-Ethernet2/0/0] ripng 1 enable
[RouterB-Ethernet2/0/0] quit
```

配置 RouterC。

```
[RouterC] ripng 1
[RouterC-ripng-1] quit
[RouterC] interface ethernet 1/0/0
[RouterC-Ethernet1/0/0] ripng 1 enable
[RouterC-Ethernet1/0/0] quit
```

查看 RouterB 的 RIPng 路由表。

```
[RouterB] display ripng 1 route
   Route Flags: A - Aging, G - Garbage-collect
 ------------------------------------------------------------------
 Peer FE80::A19:A6FF:FECE:7D4C on Ethernet1/0/0
 Dest 1::/64,
     via FE80::A19:A6FF:FECE:7D4C, cost  1, tag 0, A, 25 Sec
```

查看 RouterC 的 RIPng 路由表。

```
[RouterC] display ripng 1 route
   Route Flags: A - Aging, G - Garbage-collect
 ------------------------------------------------------------------
 Peer FE80::2E0:FCFF:FE01:9 on Ethernet1/0/0
 Dest 1::/64,
     via FE80::2E0:FCFF:FE01:9, cost  2, tag 0, A, 4 Sec
```

----结束

配置文件

- RouterA 的配置文件

```
#
```

```
 sysname RouterA
#
 ipv6
#
interface Ethernet1/0/0
 ipv6 enable
 ipv6 address auto link-local
 ripng 1 enable
#
interface Ethernet2/0/0
 ipv6 enable
 ipv6 address 1::1/64
 ripng 1 enable
#
ripng 1
#
return
```

- RouterB 的配置文件

```
#
 sysname RouterB
#
 ipv6
#
interface Ethernet1/0/0
 ipv6 enable
 ipv6 address auto link-local
 ripng 1 enable
#
interface Ethernet2/0/0
 ipv6 enable
 ipv6 address auto link-local
 ripng 1 enable
#
ripng 1
#
return
```

- RouterC 的配置文件

```
#
 sysname RouterC
#
 ipv6
#
interface Ethernet1/0/0
 ipv6 enable
 ipv6 address auto link-local
 ripng 1 enable
#
ripng 1
#
return
```

5 OSPF 配置

关于本章

OSPF 是 IETF 组织开发的一个基于链路状态的内部网关协议，广泛应用于接入网和城域网中。

5.1 OSPF 概述
OSPF 是一个基于链路状态的内部网关协议，目前针对 IPv4 协议使用的是 OSPFv2。

5.2 AR150/200 中支持的 OSPF 特性
AR150/200 中支持的 OSPF 特性包括：多进程、验证、Smart-Discover、GR、VPN 多实例、伪连接、BFD、OSPF-BGP 联动和 GTSM。

5.3 配置 OSPF 的基本功能
通过使能 OSPF 和建立邻居，配置 OSPF 的基本功能，实现 OSPF 网络中各节点的互通。

5.4 在 NBMA 网络和 P2MP 网络中配置 OSPF
在 NBMA 网络和 P2MP 网络中配置 OSPF 协议和调整属性，可以灵活组建 OSPF 网络。

5.5 调整 OSPF 的选路
配置 OSPF 的选路策略，满足复杂网络环境中的组网需求。

5.6 控制 OSPF 的路由信息
控制 OSPF 的路由信息的发布与接收，并引入其他协议的路由。

5.7 配置 OSPF 的 STUB 区域
通过将位于自治系统边缘的非骨干区域配置成 STUB 区域，不传播来自 OSPF 网络其它区域的外部路由和自治系统外部的路由，这样可以避免大量外部路由对路由器带宽和存储资源的消耗，缩减其路由表规模，减少需要传递的路由信息数量。

5.8 配置 OSPF 的 NSSA 区域
通过将位于自治系统边缘的非骨干区域配置成 NSSA 区域后，不传播来自 OSPF 网络其它区域的外部路由，但引入自治系统外部的路由，这样可以避免大量外部路由对路由器带宽和存储资源的消耗，可以缩减其路由表规模，减少需要传递的路由信息数量。

5.9 配置 BFD for OSPF
配置 BFD for OSPF 特性，当路由器检测到链路故障时，能够快速的感知并将故障通告给 OSPF 进程或 OSPF 接口，触发 OSPF 重新计算路由，提高 OSPF 的收敛速度。

5.10 配置 OSPF GR
配置 OSPF GR 可以避免流量中断和主备板切换带来的路由震荡。

5.11 提高 OSPF 网络的安全性

在对安全性较高的网络中，可以通过配置 OSPF GTSM 机制和验证方式来提高 OSPF 网络的安全性。

5.12 配置 OSPF 网管功能

OSPF 同时支持网管功能，可以配置 OSPF MIB 与某一进程绑定，以及发送 Trap 消息和日志功能。

5.13 维护 OSPF

维护 OSPF，包括复位、清除 OSPF。

5.14 配置举例

介绍 OSPF 配置举例。请结合配置流程图了解配置过程。配置示例中包括组网需求、配置注意事项、配置思路等。

5.1 OSPF 概述

OSPF 是一个基于链路状态的内部网关协议，目前针对 IPv4 协议使用的是 OSPFv2。

OSPF 是 IETF 组织开发的一个基于链路状态的内部网关协议。

说明

本章若没有特别说明，下文中所提到的 OSPF 均指 OSPFv2。

OSPF 的特性

OSPF 有以下几条重要的特点：

- 适应范围广：支持各种规模的网络，可支持几百台路由器。
- 快速收敛：在网络的拓扑结构发生变化后立即发送更新报文，使这一变化在自治系统中同步。
- 无自环：由于 OSPF 根据收集到的链路状态用最短路径树算法计算路由，从算法本身保证了不会生成自环路由。
- 区域划分：允许自治系统的网络被划分成区域来管理，路由器的链路状态数据库仅需和所在区域的其他路由器保持一致。链路状态数据库的减小降低了对路由器内存的占用和 CPU 的消耗。同时，需要在区域间传送的路由信息的减小，降低了网络带宽的占用。
- 等价路由：支持到同一目的地址的多条等价路由。
- 路由分级：使用 4 类不同的路由，按优先顺序分别是：区域内路由、区域间路由、第一类外部路由、第二类外部路由。
- 支持验证：支持基于区域和接口的报文验证，以保证报文交互的安全性。
- 组播发送：在某些类型的链路上以组播地址发送协议报文，减少对其他未使能 OSPF 设备的干扰。

OSPF 协议路由的计算过程

OSPF 协议路由的计算过程可简单描述如下：

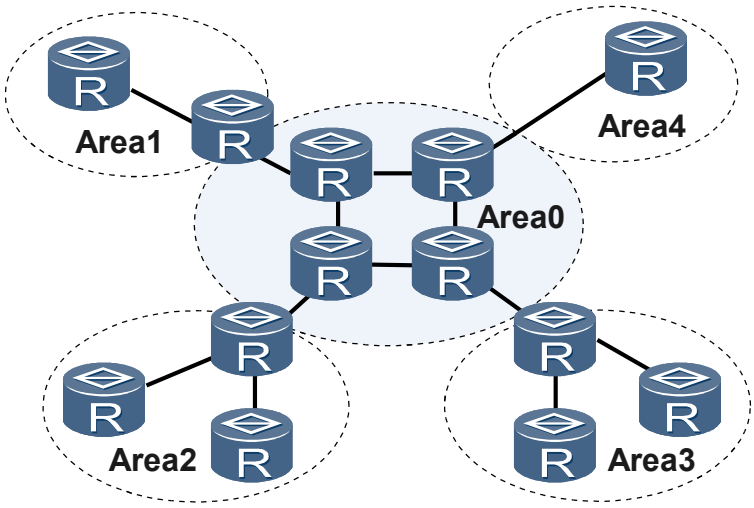
1. 每台 OSPF 设备根据自己周围的网络拓扑结构生成链路状态通告 LSA（Link State Advertisement），并通过更新报文将 LSA 发送给网络中的其它 OSPF 设备。
2. 每台 OSPF 设备都会收集其它路由器发来的 LSA，所有的 LSA 放在一起便组成了链路状态数据库 LSDB（Link State Database）。LSA 是对设备周围网络拓扑结构的描述，LSDB 则是对整个自治系统的网络拓扑结构的描述。
3. OSPF 设备将 LSDB 转换成一张带权的有向图，这张图便是对整个网络拓扑结构的真实反映。同一区域内各个设备得到的有向图是完全相同的。
4. 每台路由器根据有向图，使用 SPF 算法计算出一棵以自己为根的最短路径树，这棵树给出了到自治系统中各节点的路由。

OSPF 区域划分

随着网络规模日益扩大，路由器数量的增多会导致 LSDB 非常庞大，导致路由器负担很重。OSPF 协议通过将自治系统划分成不同的区域（Area）来解决上述问题。区域是从逻辑上将路由器划分为不同的组，每个组用区域号（Area ID）来标识。区域的边界是路

由器，而不是链路。一个网段（链路）只能属于一个区域，或者说每个运行 OSPF 的接口必须指明属于哪一个区域。如图 5-1 所示。

图 5-1 OSPF 区域划分



划分区域后，可以在区域边界路由器上进行路由聚合，减少通告到其他区域的 LSA 数量。另外，还可以最小化由于网络拓扑变化带来的影响。

路由器的类型

OSPF 路由器根据在 AS 中的不同位置，可以分为以下四类：

- 区域内路由器（Internal Routers）

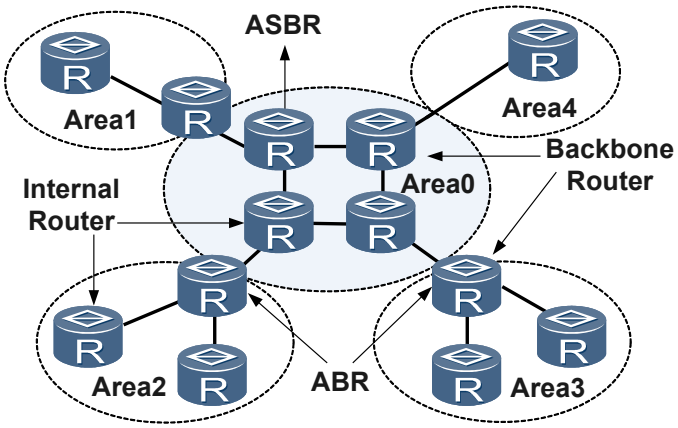
  该类路由器设备的所有接口都属于同一个 OSPF 区域。
- 区域边界路由器 ABR（Area Border Routers）

  该类路由器设备可以同时属于两个以上的区域，但其中一个必须是骨干区域。ABR 用来连接骨干区域和非骨干区域，它与骨干区域之间既可以是物理连接，也可以是逻辑上的连接。
- 骨干路由器（Backbone Routers）

  该类路由器设备至少有一个接口属于骨干区域。因此，所有的 ABR 和位于 Area0 的内部节点都是骨干路由器。
- 自治系统边界路由器 ASBR（AS Boundary Routers）

  与其他 AS 交换路由信息的路由器设备称为 ASBR。ASBR 并不一定位于 AS 的边界，它可能是区域内路由器，也可能是 ABR。只要一台 OSPF 设备引入了外部路由的信息，它就成为 ASBR。

图 5-2 OSPF 路由器的类型



OSPF 的网络类型

OSPF 根据链路层协议类型将网络分为下列四种类型：

- 广播（Broadcast）类型：当链路层协议是 Ethernet、FDDI 时，OSPF 缺省认为网络类型是 Broadcast。在该类型的网络中：
  - 以组播形式（224.0.0.5：含义是 OSPF 路由器的预留 IP 组播地址）发送 Hello 报文及所有源自 DR 的报文；
  - 以组播形式（224.0.0.6：含义是 OSPF DR（Designated Router，指定路由器）的预留 IP 组播地址）发送 LSU 报文，进而 DR 将该 LSU 报文发送到 224.0.0.5；
  - 以单播形式发送 DD 报文、LSR 报文和所有重传报文；
  - 正常情况下，以组播形式（224.0.0.5）发送 LSAck 报文。当路由器收到重复的 LSA 或达到最大生存时间的 LSA 被删除时，LSAck 以单播形式发送。
- NBMA（Non-Broadcast Multi-Access）类型：当链路层协议是帧中继、ATM 或 X.25 时，OSPF 缺省认为网络类型是 NBMA。在该类型的网络中，以单播形式发送协议报文（Hello 报文、DD 报文、LSR 报文、LSU 报文、LSAck 报文）。
- 点到多点 P2MP（point-to-multipoint）类型：P2MP 类型是由其他的网络类型强制更改的。在该类型的网络中，以组播形式（224.0.0.5）发送 Hello 报文，以单播形式发送 DD 报文、LSR 报文、LSU 报文、LSAck 报文。
- 点到点 P2P（point-to-point）类型：当链路层协议是 PPP、HDLC 和 LAPB 时，OSPF 缺省认为网络类型是 P2P。在该类型的网络中，以组播形式（224.0.0.5）发送协议报文（Hello 报文、DD 报文、LSR 报文、LSU 报文、LSAck 报文）。

5.2 AR150/200 中支持的 OSPF 特性

AR150/200 中支持的 OSPF 特性包括：多进程、验证、Smart-Discover、GR、VPN 多实例、伪连接、BFD、OSPF-BGP 联动和 GTSM。

多进程

OSPF 支持多进程，在同一台路由器上可以运行多个不同的 OSPF 进程，它们之间互不影响，彼此独立。不同 OSPF 进程之间的路由交互相当于不同路由协议之间的路由交互。

路由器的一个接口只能属于某一个 OSPF 进程。

OSPF 多进程的一个典型应用就是在 VPN 场景中 PE 和 CE 之间运行 OSPF 协议，同时 VPN 骨干网上的 IGP 也采用 OSPF。在 PE 上，这两个 OSPF 进程互不影响。

验证功能

OSPF 支持报文验证功能，只有通过验证的 OSPF 报文才能接收，否则将不能正常建立邻居。AR150/200 支持两种验证方式：

- 区域验证方式
- 接口验证方式

当两种验证方式都存在时，优先使用接口验证方式。

Smart-discover

通常情况下，路由器会周期性地从运行 OSPF 协议的接口上发送 Hello 报文。通过 Hello 报文，路由器之间可以建立和维持邻居关系，并且选举出该多址网络（广播型或 NBMA）上的 DR、BDR。当建立邻居关系或者选举多址网络上的 DR、BDR 时，接口每次都需要等到 Hello 定时器到时才能发送 Hello 报文，从而影响了建立邻居关系和选举 DR、BDR 的速度。

说明

- 接口发送 Hello 报文的时间间隔取决于在接口上配置的发送 Hello 报文的时间间隔。
- Hello 报文时间间隔的缺省值因网络类型而异。

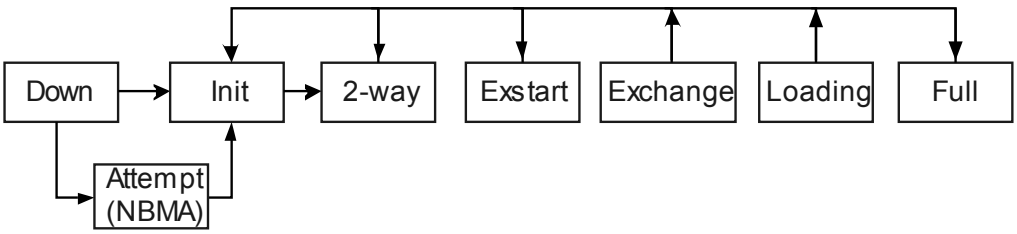
通过 Smart-discover 功能，可以解决以上问题：

- 在广播型和 NBMA 网络中，快速的建立起邻居关系和选举出该网络上的 DR、BDR。

  当邻居状态首次到达 2-way 状态，或者由 2-way 及更高状态回到 Init 状态时，如图 5-3 所示，使能了 Smart-discover 功能的接口一旦收到邻居的 Hello 报文，发现邻居状态发生了变化后，会立刻主动的向邻居发送 Hello 报文，而不必等到 Hello 定时器到时再发送。

  该多址网络上的 DR、BDR 的接口状态发生变化时，使能了 Smart-discover 功能的接口会主动在该网段发送 Hello 报文，参与到 DR 或者 BDR 的选举中。

  图 5-3 邻居状态机的变化



- 在 P2P 或 P2MP 网络中，快速的建立起邻接关系，其原理同广播型和 NBMA 网络。

OSPF GR

当一台路由器进行重启或因为各种原因进行主备倒换时，自身会直接老化 FIB（Forward Information Base）表中的所有路由表项，造成路由中断。而且与它邻接的路由器会把它从邻居列表中删除，并通知给其他路由器，这样整个网络就会重新进行 SPF 计算。如果这台路由器很快又恢复，就会造成邻居关系震荡，从而导致路由震荡。

路由器使能平滑重启 OSPF GR（Graceful Restart）功能后，如果仅是因为异常重启，路由器仍可以保证流量转发不中断，网络不会因为路由器的短时重启而震荡。

说明

若没有特殊说明，本文中的“协议重启”专指以 Graceful Restart 方式重启 OSPF 协议。

路由器在进行协议重启时，GR Restarter 不老化转发信息。同时，GR Helper 在一段时间内仍保留从 GR Restarter 得到的拓扑信息或路由。这样就保证了发生协议重启时，流量转发不中断。

OSPF VPN 多实例

OSPF 支持多实例，可以运行在 VPN 网络中的 PE-CE 之间。

在 BGP MPLS VPN 网络中，属于同一个 VPN 的多个 Site 可以使用 OSPF 作为内部路由协议。然而，它们会被看作属于不同的自治系统来处理。这样，在一个节点学到的 OSPF 路由，将被作为外部路由传送给另一节点。这种处理方式导致了比较高的 OSPF 路由协议流量，并带来了一些原来可以避免的网络管理问题。

在 AR150/200 的实现中，可以在 PE 上通过配置域 ID 来区分 Site 所在的 VPN。属于同一个 VPN 的不同 Site 之间彼此看做是直接相连的。这样，PE 路由器之间交换 OSPF 路由信息时就好象是通过一条专线相连，改善了网络管理并使 OSPF 的应用更为有效。

说明

此特性的相关配置请参考《AR150/200 企业路由器 配置指南 VPN》。

OSPF 伪连接

OSPF 伪连接（sham link）是 MPLS VPN 骨干网上两个 PE 路由器之间的点到点链路，这些链路使用 unnumbered 的地址。

通常情况下，BGP 对等体之间通过 BGP 扩展团体属性在 MPLS VPN 骨干网上承载路由信息。另一端 PE 上运行的 OSPF 可利用这些信息来生成 PE 到 CE 的 Type-3 summary LSA，这些路由是区域间路由。

但是，如果路由器和它同一区域内的 PE 路由器相连，且建立到达特定目的地址的内部区域路由（后门路由），那么 VPN 流量就将总是穿越这条后门路由，而不是骨干路由。这是因为在路由表中建立的 OSPF 内部区域路由的优先级较高。为了避免这一异常现象，可以在 PE 路由器之间配置一条 unnumbered 的点到点伪连接。这样，就可以通过一条低开销的内部区域路由到达 PE 路由器。

说明

OSPF 伪连接的相关配置请参考《AR150/200 企业路由器 配置指南 VPN》。

BFD for OSPF

缺省情况下，在广播网络中，OSPF 发送 Hello 报文的时间间隔为 10 秒钟；在 NBMA 网络中，发送 Hello 报文的时间间隔为 30 秒钟。并且，宣告邻居 Down 掉的时间即相邻

路由器失效的时间一般配置为 Hello 报文间隔的 4 倍。若在相邻路由器失效时间内没有收到邻居发来的 Hello 报文，将会删除邻居。即路由器感知到邻居故障的时间最短也是秒级。在高速的网络环境中，这将导致报文大量丢失。

双向转发检测 BFD（Bidirectional Forwarding Detection）就是为解决现有检测机制的不足而产生的。通过配置 BFD 可以设置毫秒级的时间检测间隔。使用 BFD 并不是代替 OSPF 协议本身的 Hello 机制，只是配合 OSPF 协议更快的发现邻接方面出现的故障，并及时通知 OSPF 重新计算相关路由以便正确指导报文的转发。

路由管理模块 RM（Routing Management Module）为 OSPF 提供与 BFD 模块交互的相关服务。OSPF 通过 RM 通知 BFD 来动态创建或删除 BFD session，同时 BFD 的事件消息也通过 RM 传递给 OSPF。

BFD 会话建立与删除的过程如下：

- 创建 BFD 会话的过程：配置了全局 BFD 功能，并使能接口或者进程的 BFD 特性，且 OSPF 邻居的状态为 Full，当满足以上条件时，OSPF 将通过 RM 模块通知 BFD 模块自动建立 BFD 会话并协商 BFD 的相关参数。
- 删除 BFD 会话的过程：当 BFD 检测到链路发生故障时，BFD 产生 Down 事件通过 RM 模块通知上层协议，此时 OSPF 响应这个事件并马上取消该链路上的邻接关系。这时邻居状态不再为 Full，因为不满足 BFD 会话建立的条件，所以 OSPF 通过 RM 模块通知 BFD 模块删除该 BFD 会话

OSPF 支持在 Broadcast、P2P、P2MP 和 NBMA 链路上动态的创建/删除 BFD 会话。

请根据网络环境配置 BFD，如果时间参数设置不正确将会导致网络震荡。

OSPF-BGP 联动

如果有新的路由器加入到网络中或者路由器重启时，可能会出现在 BGP 收敛期间内网络流量丢失的现象。这是由于 IGP 收敛速度比 BGP 快而造成的。

在存在备份链路的情况下，OSPF-BGP 联动特性可以使得重启路由器或者新加入的路由器在 BGP-OSPF 联动期间启动 stub Router 定时器，在设定的联动时间内保持为 stub 路由器的形式（通过增大该路由器所生成的 LSA 中的链路的度量值到 65535），从而告知其它 OSPF 路由器不要使用这个 Stub 路由器来转发数据。由此确保该路由器不会被用作穿越路由器，从而避免了 BGP 在链路回切时因为其路由收敛速度滞后于 OSPF 路由收敛速度而造成的流量丢失现象。

GTSM 简介

GTSM（Generalized TTL Security Mechanism），即通用 TTL 安全保护机制。GTSM 通过检查 IP 报文头中的 TTL 值是否在一个预先定义好的范围内，对 IP 层以上业务进行保护。在实际应用中，主要用于保护建立在 TCP/IP 基础上的控制层面（路由协议等）免受 CPU 利用（CPU-utilization）类型的攻击，如 CPU 过载（CPU overload）。

5.3 配置 OSPF 的基本功能

通过使能 OSPF 和建立邻居，配置 OSPF 的基本功能，实现 OSPF 网络中各节点的互通。

5.3.1 建立配置任务

在 OSPF 的各项配置任务中，必须先使能 OSPF、运行 OSPF 的进程与区域、建立邻居关系后，才能配置其它的功能特性。

应用环境

在同一区域内配置多台路由器时，大多数的配置数据（如定时器、过滤、聚合等）都应该以区域为单位进行统一规划。错误的配置可能会导致相邻路由器之间无法相互传递信息，甚至导致路由信息的阻塞或者自环。

在接口视图下配置的 OSPF 命令不受 OSPF 是否使能的限制。在关闭 OSPF 后，原来在接口下配置的相关命令仍然存在。

前置任务

在配置 OSPF 的基本功能之前，需完成以下任务：

- 配置链路层协议。
- 配置接口的网络层地址，使各相邻节点网络层可达。

数据准备

在配置 OSPF 的基本功能之前，需要准备以下数据。

序号	数据
1	路由器的 Router ID
2	OSPF 进程号
3	如果配置了 OSPF 多实例，需准备 VPN 实例名
4	接口所属的区域编号
5	接口所在网段的 IP 地址

5.3.2 使能 OSPF

创建 OSPF 进程，指定路由器的 Router ID，使能 OSPF。配置完成后，在区域中指定运行 OSPF 协议的接口和接口所属的区域，达到在自治区域中发现并计算路由的目的。

背景信息

路由器如果要运行 OSPF 协议，必须存在 Router ID。Router ID 是一个 32 比特无符号整数，是路由器在自治系统中的唯一标识。为保证 OSPF 运行的稳定性，在进行网络规划时应该确定 Router ID 的划分并手工配置。

OSPF 协议通过将自治系统（AS）划分成不同的区域（Area）来解决 LSDB 增大的问题。区域是从逻辑上将路由器划分为不同的组，每个组用区域号（Area ID）来标识。区域的边界是路由器，不是链路，一个网段（链路）只能属于一个区域，或者说每个运行 OSPF 的接口必须指明属于哪一个区域。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **ospf** [*process-id* | **router-id** *router-id* | **vpn-instance** *vpn-instance-name*] *，启动 OSPF 进程，进入 OSPF 视图。

- *process-id* 为进程号，缺省值为 1。AR150/200 支持 OSPF 多进程，可以根据业务类型划分不同的进程。进程号不影响与其它路由器之间的报文交换。因此，不同的路由器之间，即使进程号不同也可以进行报文交换。
- 每个 OSPF 进程的 Router ID 要保证全网唯一，否则会导致邻居不能正常建立、路由信息不正确的问题。缺省情况下，路由器系统会从当前接口的 IP 地址中自动选取一个作为 Router ID。人为配置 Router ID 时，必须保证自治系统中任意两台 Router ID 都不相同，通常的做法是将 Router ID 配置为与该设备某个接口的 IP 地址一致。
- 如果指定了 VPN 实例，那么此 OSPF 进程属于指定的 VPN 实例，如果未指定则属于公网实例。

步骤 3 执行命令 **area** *area-id*，进入 OSPF 区域视图。

OSPF 区域分为骨干区域（Area 0）和非骨干区域。骨干区域负责区域之间的路由，非骨干区域之间的路由信息必须通过骨干区域来转发。

步骤 4 执行命令 **network** *ip-address wildcard-mask* [**description** *text*]，配置区域所包含的网段。其中，**description** 字段用来为 OSPF 指定网段配置描述信息。

满足下面两个条件，接口上才能正常运行 OSPF 协议：

- 接口的 IP 地址掩码长度≥**network** 命令指定的掩码长度。
- 接口的主 IP 地址必须在 **network** 命令指定的网段范围内。

缺省情况下，OSPF 以 32 位主机路由的方式对外发布 Loopback 接口的 IP 地址，与 Loopback 接口上配置的掩码长度无关。如果要发布 Loopback 接口的网段路由，需要在接口下配置网络类型为 NBMA 或广播型。请参考**配置 OSPF 接口上网络类型**。

----结束

5.3.3 （可选）创建虚连接

建立 OSPF 骨干区域之间的逻辑链路，保证 OSPF 网络的互通性。

背景信息

在划分 OSPF 区域之后，非骨干区域之间的 OSPF 路由更新是通过骨干区域来交换完成的。因此，OSPF 要求所有非骨干区域必须与骨干区域保持连通，并且骨干区域之间也要保持连通。但在实际应用中，因为各方面条件的限制，可能无法满足这个要求，这时可以通过配置 OSPF 虚连接解决。

请在运行 OSPF 协议的路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **ospf** [*process-id*]，进入 OSPF 进程视图。

步骤 3 执行命令 **area** *area-id*，进入 OSPF 区域视图。

步骤 4 执行命令 **vlink-peer** *router-id* [**smart-discover** | **hello** *hello-interval* | **retransmit** *retransmit-interval* | **trans-delay** *trans-delay-interval* | **dead** *dead-interval* | [**simple** [**plain** *plain-text* | **cipher** *cipher-text*] | { **md5** | **hmac-md5** } [*key-id* { **plain** *plain-text* | **cipher** *cipher-text* }] | **authentication-null** | **keychain** *keychain-name*]] *，创建并配置虚连接。

在虚连接的另一端也需要配置此命令。

----结束

后续处理

建立虚连接后，不同的设备制造商可能会使用不同的 MTU（maximum transmission unit）缺省设置。为了保证一致，应该设置接口发送 DD 报文时 MTU 值为缺省值 0。参见**使能在 DD 报文中填充接口的实际 MTU**。

5.3.4 （可选）配置路由器的路由选路规则

根据实际设备的路由选路规则，选择 RFC1583 或 RFC2328 定义的规则进行路由选择。

背景信息

由于 RFC2328 与 RFC1583 定义的路由选路规则不同，因此使能 OSPF 后，根据实际设备支持的路由选路的定义情况（支持 RFC2328 或支持 RFC1583）配置 OSPF 域的路由选路规则。默认支持 RFC1583，当设备支持的是 RFC2328 时，需要将 RFC1583 配置成 RFC2328，使 OSPF 路由域中的所有设备配置为同一种路由选路规则。

请在运行 OSPF 协议的路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **ospf** [*process-id*]，进入 OSPF 进程视图。

步骤 3 执行命令 **undo rfc1583 compatible**，将 RFC1583 配置成 RFC2328，配置 OSPF 域的路由选路规则。

缺省情况下，路由器支持 RFC1583 的选路规则。

----结束

5.3.5 （可选）配置 OSPF 的协议优先级

当多个路由协议发现相同的路由时，通过配置 OSPF 的协议优先级来改变路由的优先选择顺序。

背景信息

由于路由器上可能同时运行多个动态路由协议，就存在各个路由协议之间路由信息共享和选择的问题。系统为每一种路由协议设置一个优先级。在不同协议发现同一条路由时，优先级高的路由将被优选。

请在运行 OSPF 协议的路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **ospf** [*process-id*]，进入 OSPF 进程视图。

步骤 3 执行命令 **preference** [**ase**] { *preference* | **route-policy** *route-policy-name* } *，配置 OSPF 协议的优先级。

- **ase** 表示设置 AS-External 路由的优先级。
- *preference* 表示 OSPF 协议路由的优先级，值越小，优先级越高。
- *route-policy-name* 表示对特定的路由通过路由策略设置优先级。

缺省情况下，OSPF 路由的优先级为 10。当指定 ASE 时，缺省优先级为 150。

----结束

5.3.6 （可选）配置对 OSPF 更新 LSA 的泛洪限制

配置对 OSPF 更新 LSA 的泛洪限制，即更新报文的数量和更新的时间间隔，使 LSA 泛洪控制在一定范围内，避免邻居因为需要处理大量 LSA 而忽略 hello 报文，导致邻居的中断。

背景信息

当邻居数量或者需要泛洪的 LSA 报文数量较多时，邻居路由器会在短时间内收到大量的更新报文。如果邻居路由器不能及时处理这些突发的大量报文，则有可能因为忙于处理更新报文而丢弃了维护邻居关系的 Hello 报文，造成邻居断开。而重建邻居时，需要交互的报文数量将会更大，由此导致报文数量过大的情况进一步恶化。通过对 OSPF 更新 LSA 的泛洪限制可以有效避免以上情况的发生，起到了维护邻居关系的目的。

请在运行 OSPF 协议的路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **ospf** [*process-id*]，进入 OSPF 进程视图。

步骤 3 执行命令 **flooding-control** [**number** *transmit-number* | **timer-interval** *transmit-interval*] *，配置对 OSPF 更新 LSA 的泛洪限制。

缺省情况下，每次泛洪更新 LSA 的数量的缺省值是 50，泛洪更新 LSA 的时间间隔是 30 秒。

配置 **flooding-control** 命令后，控制对 OSPF 更新 LSA 的泛洪的这个功能立刻生效。

若没有配置 **flooding-control** 命令，当邻居数量超过 256 个时自动使能该功能。

----结束

5.3.7 （可选）配置报文重传的次数

当 DD 报文、Update 报文或 Request 报文收不到相应的确认报文时，使能重传特性，并限制报文重传的次数，确保报文在重传次数内到达。

背景信息

如果达到重传次数，但仍未收到确认报文，此时会断开邻居。

缺省情况下，不使能重传限制功能。

请在运行 OSPF 协议的路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **ospf** [*process-id*]，进入 OSPF 进程视图。

步骤 3 执行命令 **retransmission-limit** [*max-number*]，配置 OSPF 重传限制功能。

max-number 是最大重传限制次数，缺省值是 30。

----结束

5.3.8 （可选）配置邻接路由器重传 LSA 的间隔

配置邻接路由器的重传 LSA 的间隔可以根据网络情况控制重传的节奏，提高收敛速度。

背景信息

当一台路由器向它的邻居发送一条 LSA 后，需要等到对方的确认报文。若在重传间隔时间内没有收到对方的确认报文，就会向邻居重传这条 LSA。

请在运行 OSPF 协议的路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **interface** *interface-type interface-number*，进入接口视图。

步骤 3 执行命令 **ospf timer retransmit** *interval*，设置邻接路由器重传 LSA 的间隔。

相邻路由器重传 LSA 时间间隔的值不要设置得太小，否则将会引起不必要的重传。通常应该大于一个报文在两台路由器之间传送一个来回的时间。

缺省情况下，重传间隔时间为 5 秒。一般情况下使用缺省值。

----结束

5.3.9 （可选）使能在 DD 报文中填充接口的实际 MTU

通过使能在 DD 报文中填充接口的实际 MTU，将使用接口的实际 MTU 值填写 DD 报文的 Interface MTU 字段。

背景信息

一般不需要配置 MTU（maximum transmission unit）值，缺省情况下是 0。

建立虚连接后，不同的设备制造商可能会使用不同的 MTU 缺省设置。为了保证一致，应该设置接口发送 DD 报文时 MTU 值为缺省值 0。

注意

当配置 DD 报文 MTU 值后，会引起邻居关系重新建立。

请在运行 OSPF 协议的路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **interface** *interface-type interface-number*，进入接口视图。

步骤 3 执行命令 **ospf mtu-enable**，使能接口发送 DD 报文时填 MTU 值，同时还会检查邻居 DD 报文所携带的 MTU 是否超过本端的 MTU 值。

----结束

5.3.10 检查配置结果

OSPF 基本功能配置成功后，您可以查看到 LSDB、各区域邻居和路由表的信息。

前提条件

已经完成 OSPF 基本功能的所有配置。

操作步骤

- 使用 **display ospf** [*process-id*] **peer** 命令查看 OSPF 邻居的信息。
- 使用 **display ospf** [*process-id*] **routing** 命令查看 OSPF 路由表的信息。
- 使用 **display ospf** [*process-id*] **lsdb** 命令查看 OSPF 的 LSDB 信息。

----结束

5.4 在 NBMA 网络和 P2MP 网络中配置 OSPF

在 NBMA 网络和 P2MP 网络中配置 OSPF 协议和调整属性，可以灵活组建 OSPF 网络。

5.4.1 建立配置任务

在 NBMA 网络和 P2MP 网络中配置 OSPF，实现 OSPF 协议的功能。

应用环境

OSPF 根据链路层协议类型将网络分为四种不同的类型。如表 5-1 所示。

说明

本章仅体现在 NBMA 网络和 P2MP 网络中有差异的 OSPF 配置。OSPF 的其他功能适用于四种网络类型。

表 5-1 OSPF 的网络类型和特点

网络类型	特点	缺省选择
广播类型（Broadcast）	在该类型的网络中，通常以组播形式发送 Hello 报文、LSU 报文和 LSAck 报文，以单播形式发送 DD 报文和 LSR 报文。	当链路层协议是 Ethernet、FDDI 时，缺省情况下，OSPF 认为网络类型是 Broadcast。

网络类型	特点	缺省选择
NBMA 类型（Non-broadcast multiple access）	在该类型的网络中，以单播形式发送 Hello 报文、DD 报文、LSR 报文、LSU 报文、LSAck 报文。 NBMA 网络必须是全连通的，即网络中任意两台路由器之间都必须直接可达。	当链路层协议是 ATM 时，缺省情况下，OSPF 认为网络类型是 NBMA。
点到点 P2P 类型（point-to-point）	在该类型的网络中，以组播形式发送 Hello 报文、DD 报文、LSR 报文、LSU 报文、LSAck 报文。	当链路层协议是 PPP、HDLC 和 LAPB 时，缺省情况下，OSPF 认为网络类型是 P2P。
点到多点 P2M 类型（Point-to-Multipoint）	在该类型的网络中：以组播形式发送 Hello 报文，以单播形式发送 DD 报文、LSR 报文、LSU 报文、LSAck 报文。 P2MP 网络中的掩码长度必须一致。	没有一种链路层协议会被缺省的认为是 P2MP 类型，P2MP 必须是由其他的网络类型强制更改的。

由**表 5-1** 可以看出，OSPF 协议在上述四种网络类型中的差异主要集中在发送报文形式不同，因此，在四种网络类型中配置的 OSPF 协议，配置差异主要体现在协议报文的发送形式的配置。

前置任务

在配置 OSPF 在不同网络类型中的属性之前，需完成以下任务：

- 配置接口的网络层地址，使相邻节点之间网络层可达。
- **配置 OSPF 的基本功能**。

数据准备

在配置 OSPF 在不同网络类型中的属性之前，需要准备以下数据。

序号	数据
1	运行 OSPF 的接口编号
2	要使用的网络类型
3	接口的 DR 优先级
4	NBMA 网络邻居的 IP 地址
5	NBMA 网络发送轮询报文的时间间隔

5.4.2 配置接口的网络类型

OSPF 根据链路层协议类型将网络分为四种不同的类型。通过配置接口的网络类型，可以强制改变接口的网络类型。

背景信息

缺省情况下，接口的网络类型根据物理接口选择：

- 以太网接口的网络类型为广播（Broadcast）。
- ATM 接口的网络类型为 NBMA。

说明

P2MP 网络类型必须是由其他的网络类型强制更改的。

一般情况下，链路两端的 OSPF 接口的网络类型必须一致，否则双方不可以建立起邻居关系。当且仅当链路两端的 OSPF 接口的网络类型一端是广播网而另一端是 P2P 时，双方仍可以正常的建立起邻居关系，但互相学不到路由信息。

请在运行 OSPF 协议的路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **interface** *interface-type interface-number*，进入接口视图。

步骤 3 执行命令 **ospf network-type** { **broadcast** | **nbma** | **p2mp** | **p2p** }，配置 OSPF 接口的网络类型。

当用户为接口配置了新的网络类型后，原接口的网络类型将被替换。

根据实际情况配置接口的网络类型，例如：

- 如果接口的网络类型是广播，但在广播网络上有不支持组播地址的路由器，可以将接口的网络类型改为 NBMA 网络。
- 如果接口的网络类型是 NBMA，且网络是全连通的，即任意两台路由器都直接可达。此时，可以将接口类型改为广播网络，并且不必再配置邻居路由器。
- 如果接口的网络类型是 NBMA，但网络不是全连通的，必须将接口的网络类型改为 P2MP。这样，两台不能直接可达的路由器就可以通过一台与两者都直接可达的路由器来交换路由信息。接口的网络类型改为 P2MP 网络后，不必再配置邻居路由器。
- 如果同一网段内只有两台路由器运行 OSPF 协议，建议将接口的网络类型改为 P2MP 网络。

说明

OSPF 协议不支持 NULL 接口的配置。

----结束

5.4.3 配置 NBMA 网络属性

配置 NBMA 网络属性，实现 OSPF 协议的功能。

操作步骤

步骤 1 （可选）配置 NBMA 网络类型。

由于 NBMA 网络必须是全连通的，所以网络中任意两台路由器之间都必须直接可达。但在很多情况下，这个要求无法满足，此时必须通过命令强制改变网络的类型为 P2MP。详细描述参见**配置接口的网络类型**。

1. 执行命令 **system-view**，进入系统视图。
2. 执行命令 **interface** *interface-type interface-number*，进入接口视图。
3. 执行命令 **ospf network-type nbma**，配置 OSPF 接口的网络类型为 NBMA。

步骤 2 （可选）配置 NBMA 网络发送轮询报文的时间间隔。

在 NBMA 网络上，当邻居失效后，路由器将按设置的轮询时间间隔定期地发送 Hello 报文。

1. 执行命令 **ospf timer poll** *interval*，在 NBMA 接口上配置发送轮询报文的时间间隔。

   缺省情况下，时间间隔 *interval* 为 120 秒。

步骤 3 配置 NBMA 网络的邻居。

网络类型为 NBMA 的接口，无法通过广播 Hello 报文的形式发现邻居路由器，必须在接口上手工配置邻居路由器的 IP 地址和邻居路由器是否有选举权。

1. 执行命令 **quit**，退出接口视图。
2. 执行命令 **ospf** [*process-id*]，进入 OSPF 进程视图。
3. 执行命令 **peer** *ip-address* [**dr-priority** *priority*]，配置 NBMA 网络的邻居。

----结束

5.4.4 配置 P2MP 网络属性

配置 P2MP 网络属性，实现 OSPF 协议的功能。

背景信息

请在运行 OSPF 协议的路由器上进行以下配置。

操作步骤

步骤 1 配置忽略对网络掩码的检查。

在 P2MP 网络上，掩码长度不一致的设备不可以建立邻居关系。通过配置设备间忽略对 Hello 报文中网络掩码的检查，就可以正常建立 OSPF 邻居关系。

1. 执行命令 **system-view**，进入系统视图。
2. 执行命令 **interface** *interface-type interface-number*，进入接口视图。
3. 执行命令 **ospf network-type p2mp**，配置 OSPF 接口的网络类型。

   P2MP 网络类型必须是由其他的网络类型强制更改的。详细描述参见**配置接口的网络类型**。
4. 执行命令 **ospf p2mp-mask-ignore** 命令，配置在 P2MP 网络上忽略对网络掩码的检查。

步骤 2（可选）配置对发送的 LSA 进行过滤。

当两台路由器之间存在多条链路时，通过对出方向的 LSA 进行过滤可以在某些链路上过滤 LSA 的传送，减少不必要的重传，节省带宽资源。

1. 执行命令 **quit**，退出接口视图。
2. 执行命令 **ospf** [*process-id*]，进入 OSPF 进程视图。
3. 执行命令 **filter-lsa-out peer** *ip-address* { **all** | { **summary** [**acl**{ *acl-number* | *acl-name* }] | **ase** [**acl**{ *acl-number* | *acl-name* }] | **nssa** [**acl**{ *acl-number* | *acl-name* }] } * }，配置在 P2MP 网络中对发送的 LSA 进行过滤。

   缺省情况下，不对 LSA 进行过滤。

----结束

5.4.5 检查配置结果

OSPF 在 NBMA 网络和 P2MP 网络中的属性配置成功后，您可以查看到统计信息、LSDB、邻居和接口信息。

前提条件

已经完成 OSPF 在 NBMA 网络和 P2MP 网络中的属性的所有配置。

操作步骤

- 使用以下命令查看 OSPF 的 LSDB 信息：
  - **display ospf** [*process-id*] **lsdb** [**brief**]
  - **display ospf** [*process-id*] **lsdb** [**router** | **network** | **summary** | **asbr** | **ase** | **nssa** | **opaque-link** | **opaque-area** | **opaque-as**] [*link-state-id*] [**originate-router** [*advertising-router-id*] | **self-originate**] [**age** { **min-value** *min-age-value* | **max-value** *max-age-value* } *]
- 使用 **display ospf** [*process-id*] **peer** [[*interface-type interface-number*] *neighbor-id* | **brief** | **last-nbr-down**]命令查看 OSPF 的邻接点信息。
- 使用 **display ospf** [*process-id*] **nexthop** 查看 OSPF 的下一跳信息。
- 使用以下命令查看 OSPF 的路由表信息：
  - **display ospf** [*process-id*] **routing** [*ip-address* [*mask* | *mask-length*]] [**interface** *interface-type interface-number*] [**nexthop** *nexthop-address*]
  - **display ospf** [*process-id*] **routing router-id** [*router-id*]
- 使用 **display ospf** [*process-id*] **interface** [**all** | *interface-typeinterface-number*] [**verbose**]查看 OSPF 的接口信息。

----结束

5.5 调整 OSPF 的选路

配置 OSPF 的选路策略，满足复杂网络环境中的组网需求。

5.5.1 建立配置任务

在配置 OSPF 路由属性前了解此特性的应用环境、配置此特性的前置任务和数据准备，可以帮助您快速、准确地完成配置任务。

应用背景

在实际应用中，为了满足复杂网络环境中的需要，可以通过配置 OSPF 的路由属性改变 OSPF 的选路策略，主要有以下几种方式：

- 设置接口的开销值，优选开销值小的链路传输路由。
- 配置等价路由，形成负载分担。
- 在进行升级等维护操作时配置 stub 路由器，确保关键路由的稳定传输。
- 配置抑制接口发送或接收报文，达到优选路由的目的。

前置任务

在配置 OSPF 的路由属性之前，需完成以下任务：

- 配置接口的网络层地址，使相邻节点之间网络层可达。
- **配置 OSPF 的基本功能**。

数据准备

在配置 OSPF 的路由属性之前，需要准备以下数据。

序号	数据
1	接口开销
2	最大等价路由条数
3	等价路由的优先级

5.5.2 配置接口开销

通过配置 OSPF 的接口开销值，调整和优化路由的选路。

背景信息

配置 OSPF 的接口开销值，可以优选开销值小的链路传输路由，从而达到调整和优化路由的选路规则。

OSPF 的接口开销值可以直接配置，也可以通过接口带宽自动计算。

请在运行 OSPF 协议的路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **interface** *interface-type interface-number*，进入接口视图。

步骤 3 执行命令 **ospf cost** *cost*，设置 OSPF 接口的开销值。

ospf cost 可以直接配置 OSPF 接口的开销值。一般情况下，路由器选择接口开销值小的路径传输。

如果没有直接配置 OSPF 的接口开销值，也可以通过接口带宽自动计算接口开销值。计算公式为：接口开销=带宽参考值/接口带宽，取计算结果的整数部分作为接口开销值（当结果小于 1 时取 1）。缺省情况下，带宽参考值为 100Mbit/s。此时，改变带宽参考值就可以间接的改变 OSPF 接口的开销值。

配置带宽参考值的方法如下：

1. 执行命令 **system-view**，进入系统视图。
2. 执行命令 **ospf** [*process-id*]，进入 OSPF 进程视图。
3. 执行命令 **bandwidth-reference** *value*，配置带宽参考值。

   在配置带宽参考值时请注意，必须保证该进程中所有路由器的带宽参考值一致。

----结束

5.5.3 配置等价路由

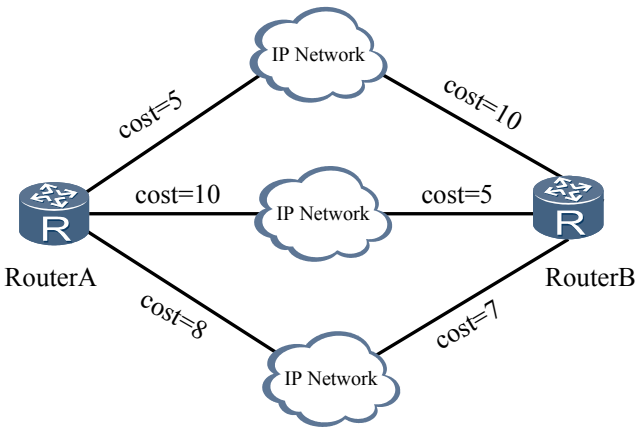
通过配置 OSPF 等价路由的条数和优先级，实现负载分担，调整和优化路由的选路。

背景信息

当网络中到达同一目的地存在同一路由协议发现的多条路由，且这几条路由的开销值也相同，那么这些路由就是等价路由，可以实现负载分担。

例如，如图 5-4 所示。路由器 A 和路由器 B 之间的三条路由都运行 OSPF 协议，且几条路由的开销值也相同，那么这三条路由就是等价路由，形成了负载分担。

图 5-4 等价路由组网图



请在运行 OSPF 协议的路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **ospf** [*process-id*]，进入 OSPF 进程视图。

步骤 3 执行命令 **maximum load-balancing** *number*，配置最大等价路由数量。

说明

OSPF 协议支持最大等价路由的数量是 4，缺省值是 4。

步骤 4 （可选）执行命令 **nexthop** *ip-address* **weight** *value*，配置 OSPF 的负载分担优先级。

当组网中存在的等价路由数量大于 **maximum load-balancing** 命令配置的等价路由数量时，会随机选取有效路由进行负载分担。如果需要指定负载分担的有效路由，可以通过 **nexthop** 命令配置路由的优先级，将需要指定的有效路由的优先级设置为高。

weight 值越小，路由优先级越高。**weight** 的缺省值是 255，表示等价路由间进行负载分担，不区分优先级。

----结束

5.5.4 配置 Stub 路由器

进行升级等维护操作会引起路由震荡或不稳定，为了避免某条路径的路由不中断，配置该条路径的路由器为 Stub 路由器，避免路由选取 Stub router 的路径。

背景信息

配置 Stub 路由器是一种特殊的路由选路，配置了 stub router 的路径不被优选。实现方法是将度量值设为最大（65535），尽量避免数据从此路由器转发。用于保护此路由器链路，通常使用在升级等维护操作的场景。

请在运行 OSPF 协议的路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **ospf** [*process-id*]，进入 OSPF 进程视图。

步骤 3 执行命令 **stub-router** [**on-startup** [*interval*]]，配置 Stub 路由器。

缺省情况下，没有路由器为 Stub 路由器。

如果配置了 Stub 路由器，缺省情况下，路由器保持为 Stub 路由器的时间间隔是 500 秒。

说明

通过此命令配置的 Stub 路由器与 Stub 区域里的路由器没有必然联系。

----结束

5.5.5 抑制接口接收和发送报文

配置抑制接口接收和发送 OSPF 报文是一种特殊的路由选路，配置后，OSPF 路由信息不被某一网络中的路由器获得且使本地路由器不接收网络中其他路由器发布的路由更新信息。

背景信息

通过抑制接口接收和发送的 OSPF 报文，使路由信息不被某一网络中的路由器获得且使本地路由器不接收网络中其他路由器发布的路由更新信息，从而达到优先保证某条路由的目的。

请在运行 OSPF 协议的路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **ospf** [*process-id*]，进入 OSPF 进程视图。

步骤 3 执行命令 **silent-interface** { **all** | *interface-type interface-number* }，抑制接口接收和发送 OSPF 报文。

不同的进程可以对同一接口抑制发送和接收 OSPF 报文，但 **silent-interface** 命令只对本进程已经使能的 OSPF 接口起作用，对其它进程的接口不起作用。

将运行 OSPF 协议的接口指定为 Silent 状态后，该接口的直连路由仍可以发布出去，但接口的 Hello 报文将被阻塞，接口上无法建立邻居关系。这样可以增强 OSPF 的组网适应能力，减少系统资源的消耗。

----结束

5.5.6 检查配置结果

调整 OSPF 的选路，您可以查看到 OSPF 路由表、接口和下一跳路由信息。

前提条件

已经完成 OSPF 选路的所有配置。

操作步骤

- 使用 **display ospf** [*process-id*] **routing** [*ip-address* [*mask* | *mask-length*]] [**interface** *interface-type interface-number*] [**nexthop** *nexthop-address*]命令查看 OSPF 的路由表信息。
- 使用 **display ospf** [*process-id*] **interface** [**all** | *interface-type interface-number*] [**verbose**]命令查看 OSPF 的接口信息。

----结束

5.6 控制 OSPF 的路由信息

控制 OSPF 的路由信息的发布与接收，并引入其他协议的路由。

5.6.1 建立配置任务

在控制 OSPF 的路由信息前了解此特性的应用环境、配置此特性的前置任务和数据准备，可以帮助您快速、准确地完成配置任务。

应用环境

通过本节的配置，可以控制 OSPF 的路由信息的发布与接收，并引入其他协议的路由。

前置任务

在控制 OSPF 路由信息之前，需完成以下任务：

- 配置接口的网络层地址，使相邻节点之间网络层可达。
- 配置 OSPF 的基本功能。

数据准备

在控制 OSPF 路由信息之前，需要准备以下数据。

序号	数据
1	链路开销
2	如果对路由信息进行过滤，则需要对应的过滤列表
3	要引入的路由协议名称和进程号、缺省值

5.6.2 配置引入外部路由

通过引入其他路由协议路由，可以扩充 OSPF 路由信息。

背景信息

当 OSPF 网络中的设备需要访问运行其他协议的网络中的设备时，需要将其他协议的路由引入到 OSPF 网络中。

OSPF 是一个无自环的动态路由协议，但这是针对域内路由和域间路由而言的，其对引入的外部路由环路没有很好的防范机制，所以在配置 OSPF 引入外部路由时一定要慎重，防止人为配置引起的环路。具体情况请参见《Huawei AR150&200 系列企业路由器特性描述-VPN》的“OSPF VPN 扩展”部分内容。

请在运行 OSPF 协议的自治系统边界路由器 ASBR 上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **ospf** [*process-id*]，进入 OSPF 进程视图。

步骤 3 执行命令 **import-route** { **limit** *limit-number* | { **bgp** [**permit-ibgp**] | **direct** | **unr** | **rip** [*process-id-rip*] | **static** | **isis** [*process-id-isis*] | **ospf** [*process-id-ospf*] } [**cost** *cost* | **type** *type* | **tag** *tag* | **route-policy** *route-policy-name*] * }，引入其它协议的路由信息。

步骤 4 （可选）执行命令 **default** { **cost** { *cost* | **inherit-metric** } | **limit** *limit* | **tag** *tag* | **type** *type* } *，配置引入路由时的参数缺省值（开销、路由数量、标记、类型）。

当 OSPF 引入外部路由时，可以配置一些额外参数的缺省值，如开销、路由数量、标记和类型。路由标记可以用来标识协议相关的信息，如 OSPF 接收 BGP 时用来区分自治系统的编号。

缺省情况下，OSPF 引入外部路由的缺省度量值为 1，一次可引入外部路由数量的上限为 2147483647，引入的外部路由类型为 Type2，设置缺省标记值为 1。

说明

可以通过以下三条命令设置引入路由的开销值，其优先级依次递减：

- 通过 **apply cost** 命令设置的路由开销值。
- 通过 **import-route** 命令设置的引入路由开销值。
- 通过 **default** 命令设置引入路由的缺省开销值。

步骤 5 （可选）执行命令 **filter-policy** { *acl-number* | **acl-name** *acl-name* | **ip-prefix** *ip-prefix-name* } **export** [*protocol* [*process-id*]]，配置对步骤 3 引入的路由进行过滤，通过过滤的路由才能被发布出去。

OSPF 对引入的路由进行过滤，是指 OSPF 只将满足条件的外部路由转换为 Type5 LSA 并发布出去。

用户可以通过指定 *protocol* [*process-id*]对特定的某一种协议或某一进程的路由信息进行过滤。如果没有指定 *protocol* [*process-id*]，则 OSPF 将对所有引入的路由信息进行过滤。

import-route 命令不能引入外部路由的缺省路由。

----结束

5.6.3 配置引入缺省路由

缺省路由具有减小路由表容量，实现路由信息屏蔽的功能，在 OSPF 组网中具有广泛的应用。

背景信息

OSPF 实际组网应用中，区域边界和自治系统边界通常都是由多个路由器组成的多出口冗余备份或者负载分担。此时，为了减少路由表的容量，可以配置缺省路由，保证网络的高可用性。

OSPF 缺省路由通常应用于下面两种情况：

1. 由区域边界路由器（ABR）发布 Type-3 LSA, 用来指导区域内路由器进行区域之间报文的转发。
2. 由自治系统边界路由器（ASBR）发布 Type-5 LSA 或 Type-7 LSA，用来指导 OSPF 路由域内路由器进行域外报文的转发。

当路由器无精确匹配的路由时，就可以通过缺省路由进行报文转发。

Type-3 LSA 缺省路由的优先级要高于 Type-5 LSA 或 Type-7 LSA 路由。

OSPF 缺省路由的发布方式取决于引入缺省路由的区域类型。如表 5-2 所示。

表 5-2 缺省路由发布方式

区域类型	产生条件	发布方式	产生 LSA 的类型	泛洪范围
普通区域	通过 **default-route-advertise** 命令配置。	ASBR 发布	Type-5 LSA	普通区域
STUB 区域	自动产生	ABR 发布	Type-3 LSA	STUB 区域
NSSA 区域	通过 **nssa**[**default-route-advertise**]	ASBR 发布	Type-7 LSA	NSSA 区域
完全 NSSA 区域	自动产生	ABR 发布	Type-3 LSA	NSSA 区域

请在运行 OSPF 协议的自治系统边界路由器 ASBR 上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **ospf** [*process-id*]，进入 OSPF 进程视图。

步骤 3 执行命令 **default-route-advertise** [[**always** | **permit-calculate-other**] | **cost** *cost* | **type** *type* | **route-policy** *route-policy-name*] *，引入缺省路由到 OSPF 进程中。

如果要指定 Type-3 summary LSA 的缺省开销值，则配置 *cost* 参数，必须首先使能 VPN。

OSPF 路由域中在通告缺省路由前，会比较缺省路由的优先级，如果在其中某 OSPF 路由器上同时配置了静态缺省路由，要使 OSPF 通告的缺省路由加入到当前的路由表中，则必须保证所配置的静态缺省路由的优先级比 OSPF 通告的缺省路由的优先级低。

配置 NSSA 区域的缺省路由，请参见**配置 OSPF 的 NSSA 区域**。

----结束

5.6.4 配置路由聚合

当大规模部署 OSPF 网络时，为了避免 OSPF 路由表中条目过多从而占用过多系统内存的情况，可以配置路由聚合，减小路由表的规模。

背景信息

当 OSPF 网络规模较大时，配置路由聚合，可以有效减少路由表中的条目，减小对系统资源的占用，不影响系统的性能。此外，如果被聚合的 IP 地址范围内的某条链路频繁 Up 和 Down，该变化并不会通告到被聚合的 IP 地址范围外的设备。因此，可以避免网络中的路由翻动，在一定程度上提高了网络的稳定性。

请在运行 OSPF 协议的路由器上进行以下配置。

操作步骤

- 配置 ABR 路由聚合
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **ospf** [*process-id*]，进入 OSPF 进程视图。
  3. 执行命令 **area** *area-id*，进入 OSPF 区域视图。
  4. 执行命令 **abr-summary** *ip-address mask* [[**advertise** | **not-advertise**] | **cost** *cost*] *，配置 OSPF 的 ABR 路由聚合。
- 配置 ASBR 路由聚合
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **ospf** [*process-id*]，进入 OSPF 进程视图。
  3. 执行命令 **asbr-summary** *ip-address mask* [**not-advertise** | **tag***tag* | **cost** *cost* | **distribute-delay** *interval*] *，配置 OSPF 的 ASBR 路由聚合。

  说明

  在配置路由聚合后，本地 OSPF 设备的路由表保持不变。但是其他 OSPF 设备的路由表中将只有一条聚合路由，没有具体路由。直到网络中被聚合的路由都出现故障而消失时，该聚合路由才会消失。

----结束

5.6.5 配置 OSPF 对接收的路由进行过滤

通过对 OSPF 接收的路由进行过滤，设置路由信息的过滤条件，只有满足过滤条件的路由才能被添加到路由表中。

背景信息

请在运行 OSPF 协议的路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **ospf** [*process-id*]，进入 OSPF 进程视图。

步骤 3 执行命令 **filter-policy** { *acl-number* | **acl-name***acl-name* | **ip-prefix** *ip-prefix-name* } **import**，配置对 OSPF 接收的路由进行过滤。

由于 OSPF 是基于链路状态的动态路由协议，路由信息隐藏在链路状态中，所以不能使用 **filter-policy import** 命令对发布和接收的 LSA 进行过滤。该命令实际上是对 OSPF 计算出来的路由进行过滤，只有通过过滤的路由才被添加到路由表中。因此，接收到的路由无论是否通过过滤，都不会对 LSDB 有影响。

----结束

5.6.6 配置对发送的 LSA 进行过滤

通过对发送的 LSA 进行过滤可以不向邻居发送无用的 LSA，从而减少邻居 LSDB 的大小，提高网络收敛速度。

背景信息

当两台路由器之间存在多条链路时，通过对发送的 LSA 进行过滤可以在某些链路上过滤 LSA 的传送，减少不必要的重传，节省带宽资源。

请在运行 OSPF 协议的路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **interface***interface-typeinterface-number*，进入接口视图。

步骤 3 执行命令 **ospf filter-lsa-out** { **all** | { **summary** [**acl** { *acl-number* | *acl-name* }] | **ase** [**acl** { *acl-number* | *acl-name* }] | **nssa** [**acl** { *acl-number* | *acl-name* }] } * }，配置对出方向的 LSA 进行过滤。

缺省情况下，不对发送的 LSA 进行过滤。

----结束

5.6.7 （可选）配置对区域内的 LSA 进行过滤

通过对区域内的 LSA 进行过滤可以不向邻居发送无用的 LSA，从而减少 LSDB 的大小，提高网络收敛速度。

背景信息

通过对区域内出方向的 Type-3 LSA（Summary LSA）设置过滤条件，只有通过过滤的信息才能被接收、发布。

此功能仅在 ABR 上配置。

请在运行 OSPF 协议的路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **ospf** [*process-id*]，进入 OSPF 进程视图。

步骤 3 执行命令 **area** *area-id*，进入 OSPF 区域视图。

步骤 4 配置对区域内入方向或出方向的 Type-3 LSA 进行过滤。

- 配置对区域内入方向的 Type-3 LSA 进行过滤。

  执行命令 **filter** { *acl-number* | **acl-name** *acl-name* | **ip-prefix** *ip-prefix-name* | **route-policy** *route-policy-name* } **export**，配置对区域内入方向的 Type-3 LSA 进行过滤。
- 配置对区域内出方向的 Type-3 LSA 进行过滤。

  执行命令 **filter** { *acl-number* | **acl-name** *acl-name* | **ip-prefix** *ip-prefix-name* | **route-policy** *route-policy-name* } **import**，配置对区域内出方向的 Type-3 LSA 进行过滤。

----结束

5.6.8 （可选）使能 Mesh-Group 特性

使能 Mesh-Group 特性，避免重复泛洪，节省资源。

背景信息

当路由器和邻居存在并行链路时，使能 Mesh-Group 特性，可以减轻链路的压力。

Mesh-Group 时以邻居的 Router-id 唯一标识一个 Group。是几条并行的 LSA，合并为一个组，只泛洪一次。只有同时满足以下三个条件的接口才能属于同一个 Mesh-Group：

- 属于相同区域和 OSPF 进程
- 接口状态大于 Exchange
- 只连着同一个邻居

请在运行 OSPF 协议的路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **ospf** [*process-id*]，进入 OSPF 进程视图。

步骤 3 执行命令 **mesh-group enable**，使能 Mesh-Group 特性。

缺省情况下，不使能 Mesh-Group 特性。

----结束

5.6.9 配置 LSDB 中 External LSA 的最大数量

通过配置 OSPF 的 LSDB 中 External LSA 的最大条目数，保证路由条数在一个合理的范围内。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **ospf** [*process-id*]，进入 OSPF 进程视图。

步骤 3 执行命令 **lsdb-overflow-limit** *number*，配置 LSDB 中 External LSA 的最大数量。

----结束

5.6.10 检查配置结果

调整 OSPF 的路由信息后，您可以查看到 OSPF 路由表、接口信息和 ASBR 聚合信息。

前提条件

已经完成控制 OSPF 路由信息的所有配置。

操作步骤

- 使用以下命令查看 OSPF 路由表信息：
  - **display ospf** [*process-id*] **routing** [*ip-address* [*mask* | *mask-length*]] [**interface** *interface-type interface-number*] [**nexthop** *nexthop-address*]
  - **display ospf** [*process-id*] **routing router-id** [*router-id*]

- 使用 **display ospf** [*process-id*] **interface** [**all** | *interface-type interface-number*] [**verbose**]命令查看 OSPF 的接口信息。
- 使用 **display ospf** [*process-id*] **asbr-summary** [*ip-address mask*]命令查看 OSPF ASBR 聚合信息。

----结束

5.7 配置 OSPF 的 STUB 区域

通过将位于自治系统边缘的非骨干区域配置成 STUB 区域，不传播来自 OSPF 网络其它区域的外部路由和自治系统外部的路由，这样可以避免大量外部路由对路由器带宽和存储资源的消耗，缩减其路由表规模，减少需要传递的路由信息数量。

应用环境

OSPF 划分区域可以减少网络中 LSA 的数量。对于位于自治系统边界的非骨干区域，为了更多的缩减其路由表规模和降低 LSA 的数量，可以将它们配置为 STUB 区域。

STUB 区域是一种可选的配置属性。通常来说，STUB 区域位于自治系统的边界，例如，只有一个 ABR 的非骨干区域。在这些区域中，路由器的路由表规模以及路由信息传递的数量都会大量减少。

配置 STUB 区域时需要注意以下几点：

- 骨干区域（Area0）不能配置成 STUB 区域。
- 如果要将一个区域配置成 STUB 区域，则该区域中的所有路由器都要配置 STUB 区域属性。
- STUB 区域内不能存在 ASBR，即自治系统外部的路由不能在 STUB 区域内传播。
- STUB 区域内不能存在虚连接。

前置任务

在配置 OSPF 的 STUB 区域之前，需完成以下任务：

- 配置接口的网络层地址，使相邻节点之间网络层可达。
- **配置 OSPF 的基本功能**。

数据准备

在配置 OSPF 的 STUB 区域之前，需要准备以下数据。

序号	数据
1	（可选）发送到 STUB 区域缺省路由的开销 **说明** 缺省情况下，发送到 STUB 区域的缺省路由的开销为 1。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **ospf** [*process-id*]，进入 OSPF 进程视图。

步骤 3 执行命令 **area** *area-id*，进入 OSPF 区域视图。

步骤 4 执行命令 **stub**，配置当前区域为 STUB 区域。

说明

- 所有连接到 STUB 区域的路由器必须使用 **stub** 命令将该区域配置成 STUB 区域属性。
- 配置或取消 STUB 属性，可能会触发区域更新。只有在上一次区域更新完成后，才能进行再次配置或取消配置操作。

步骤 5 （可选）执行命令 **stub** [**no-summary**]，配置禁止 ABR 向 STUB 区域内发送 Type-3 LSA（Summary LSA）。

步骤 6 （可选）执行命令 **default-cost** *cost*，配置发送到 STUB 区域缺省路由的开销。

当区域配置为 STUB 区域后，为保证到自治系统外的路由可达，STUB 区域的 ABR 将生成一条缺省路由，并发布给 STUB 区域中的其他路由器。

缺省情况下，发送到 STUB 区域的缺省路由的开销为 1。

----结束

检查配置结果

使用以下命令查看 OSPF 的 LSDB 信息：

- **display ospf** [*process-id*] **lsdb** [**brief**]
- **display ospf** [*process-id*] **lsdb** [**router** | **network** | **summary** | **asbr** | **ase** | **nssa** | **opaque-link** | **opaque-area** | **opaque-as**] [*link-state-id*] [**originate-router** [*advertising-router-id*] | **self-originate**] [**age** { **min-value** *min-age-value* | **max-value** *max-age-value* } *]

使用以下命令查看 OSPF 路由表的信息：

- **display ospf** [*process-id*] **routing** [*ip-address* [*mask* | *mask-length*]] [**interface** *interface-type interface-number*] [**nexthop** *nexthop-address*]
- **display ospf** [*process-id*] **routing router-id** [*router-id*]

使用 **display ospf** [*process-id*] **abr-asbr** [*router-id*]命令查看 OSPF ABR 及 ASBR 信息。

5.8 配置 OSPF 的 NSSA 区域

通过将位于自治系统边缘的非骨干区域配置成 NSSA 区域后，不传播来自 OSPF 网络其它区域的外部路由，但引入自治系统外部的路由，这样可以避免大量外部路由对路由器带宽和存储资源的消耗，可以缩减其路由表规模，减少需要传递的路由信息数量。

应用环境

NSSA 区域适用于既需要引入外部路由又要避免外部路由带来的资源消耗的场景。

OSPF NSSA（Not-So-Stubby Area）区域是 OSPF 特殊的区域类型。NSSA 区域与 STUB 区域有许多相似的地方，两者都不传播来自 OSPF 网络其它区域的外部路由。差别在于 STUB 区域是不能引入外部路由，NSSA 区域能够将自治域外部路由引入并传播到整个 OSPF 自治域中。

在 NSSA 区域中使用 Type-7 LSA 描述引入的外部路由信息。Type-7 LSA 由 NSSA 区域的自治域边界路由器（ASBR）产生，其扩散范围仅限于边界路由器所在的 NSSA 区

域。NSSA 区域的区域边界路由器（ABR）收到 Type-7 LSA 时，会有选择地将其转化为 Type-5 LSA，以便将外部路由信息通告到 OSPF 网络的其它区域。

如果要将一个区域配置成 NSSA 区域，则该区域中的所有路由器都要配置 NSSA 区域属性。

前置任务

在配置 OSPF 的 NSSA 区域之前，需完成以下任务：

- 配置接口的网络层地址，使相邻节点之间网络层可达。
- **配置 OSPF 的基本功能**。

数据准备

在配置 OSPF 的 NSSA 区域之前，需要准备以下数据。

序号	数据
1	（可选）发送到 NSSA 区域缺省路由的开销 **说明** 缺省情况下，发送到 NSSA 区域的缺省路由的开销为 1。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **ospf** [*process-id*]，进入 OSPF 进程视图。

步骤 3 执行命令 **area** *area-id*，进入 OSPF 区域视图。

步骤 4 执行命令 **nssa** [**default-route-advertise** | **flush-waiting-timer** *interval-value* | **no-import-route** | **no-summary** | **set-n-bit** | **suppress-forwarding-address** | **translator-always** | **translator-interval** *interval-value* | **zero-address-forwarding**] *，配置当前区域为 NSSA 区域。

说明

- 所有连接到 NSSA 区域的路由器必须使用 **nssa** 命令将该区域配置成 NSSA 属性。
- 配置或取消 NSSA 属性，可能会触发区域更新。只有在上一次区域更新完成后，才能进行再次配置或取消配置操作。

nssa 命令参数的使用场景如下：

- **default-route-advertise** 用来在 ABR 或者 ASBR 上配置产生缺省的 Type-7 LSA 到 NSSA 区域。

  在 ABR 上无论路由表中是否存在路由 0.0.0.0，都会产生 Type-7 LSA 缺省路由。在 ASBR 上只有当路由表中存在路由 0.0.0.0，才会产生 Type-7 LSA 缺省路由。
- 当 ASBR 所在的区域被配置成 NSSA 时，在 LSA 洪泛区域中的其他路由器上仍会保留已经没用的 Type-5 LSA，这些 LSA 必须等到老化时间到达 3600 秒后才会被删除。由于大量的 LSA 会占用路由器内存，所以对设备的性能造成了一定影响。此时，通过配置 **flush-waiting-timer** 参数产生老化时间被置为最大值（3600 秒）的 Type-5 LSA，及时清除其他路由器上已经没用的 Type-5 LSA。

说明

- 当 LSA 报文头部的 LS age（老化时间）达到 3600 秒时，该 LSA 会被删除。
- 当 ASBR 同时还是 ABR 时，**flush-waiting-timer** 功能不会生效，防止删除非 NSSA 区域的 Type-5 LSA。

- 当 ASBR 同时还是 ABR 时，通过配置 **no-import-route** 参数使 OSPF 通过 **import-route** 命令引入的外部路由不被通告到 NSSA 区域。
- 为了继续减少发送到 NSSA 区域的 LSA 的数量，可以配置 ABR 的 **no-summary** 属性，禁止 ABR 向 NSSA 区域内发送 Summary LSA（Type 3）。
- 设置了 **set-n-bit** 关键字后，路由器会和本 NSSA 区域内的相邻路由器重新建立邻居关系。
- 当 NSSA 区域中有多个 ABR 时，系统会根据规则自动选择一个 ABR 作为转换器（通常情况下 NSSA 区域选择 Router ID 最大的设备），将 Type-7 LSA 转换为 Type-5 LSA。通过在 ABR 上配置 **translator-always** 参数，可以将某一个 ABR 指定为转换器。如果需要指定某两个 ABR 进行负载分担，可以通过配置 **translator-always** 来指定两个转换器同时工作。如果需要某一个固定的转换器，防止由于转换器变动引起的 LSA 重新泛洪，可以预先使用此命令指定。
- **translator-interval** 参数主要用于转换器切换过程，保障切换平滑进行。所以 *interval-value* 参数的缺省间隔要大于泛洪的时间。

步骤 5 （可选）执行命令 **default-cost** *cost*，配置发送到 NSSA 区域缺省路由的开销。

当区域配置为 NSSA 区域后，为保证到自治系统外的路由可达，NSSA 区域的 ABR 将生成一条缺省路由，并发布给 NSSA 区域中的其他路由器。

缺省路由也可以通过 Type-7 LSA 来表示，用于指导流量流向其它自治域。

在 NSSA 区域中，可能同时存在多个边界路由器。为了防止路由环路产生，边界路由器之间不计算对方发布的缺省路由。

缺省情况下，发送到 NSSA 区域的缺省路由的开销为 1。

----结束

检查配置结果

使用以下命令查看 OSPF 的 LSDB 信息：

- **display ospf** [*process-id*] **lsdb** [**brief**]
- **display ospf** [*process-id*] **lsdb** [**router** | **network** | **summary** | **asbr** | **ase** | **nssa** | **opaque-link** | **opaque-area** | **opaque-as**] [*link-state-id*] [**originate-router** [*advertising-router-id*] | **self-originate**]

使用以下命令查看 OSPF 的路由表信息：

- **display ospf** [*process-id*] **routing** [*ip-address* [*mask* | *mask-length*]] [**interface** *interface-type interface-number*] [**nexthop** *nexthop-address*]
- **display ospf** [*process-id*] **routing router-id** [*router-id*]

使用 **display ospf** [*process-id*] **interface** [**all** | *interface-type interface-number*] [**verbose**] 命令查看 OSPF 的接口信息。

5.9 配置 BFD for OSPF

配置 BFD for OSPF 特性，当路由器检测到链路故障时，能够快速的感知并将故障通告给 OSPF 进程或 OSPF 接口，触发 OSPF 重新计算路由，提高 OSPF 的收敛速度。

5.9.1 建立配置任务

介绍配置 BFD for OSPF 特性的应用环境、前置任务和数据准备，可以帮助您快速、准确地完成配置任务。

应用场景

OSPF 通过周期性的向邻居发送 Hello 报文来实现邻居检测，检测到故障所需时间比较长，超过 1 秒钟。随着科技的发展，语音、视频及其它点播业务应用广泛，而这些业务对于丢包和延时非常敏感，当数据达到吉比特速率级时，较长的检测时间会导致大量数据丢失，无法满足电信级网络高可靠性的需求。

为了解决上述问题，配置指定进程或指定接口的 BDF for OSPF 特性，可以快速检测链路的状态，故障检测时间可以达到毫秒级，提高链路状态变化时 OSPF 的收敛速度。

说明

目前，BFD 会话不会感知路由切换。如果绑定的对端 IP 地址改变引起路由切换到其他链路上，除非原链路转发不通，否则，BFD 不会重新协商。

前置任务

在配置 BFD for OSPF 特性之前，需完成以下任务：

- 配置接口的网络层地址，使相邻节点之间网络层可达。
- **配置 OSPF 的基本功能**。

数据准备

在配置 BFD for OSPF 之前，需要准备以下数据。

序号	数据
1	使能 BFD for OSPF 特性的指定进程的进程号。
2	使能 BFD for OSPF 特性的指定接口的类型和编号。
3	（可选）BFD 会话的参数值。 **说明** 推荐使用 BFD 会话的缺省值。

5.9.2 配置指定进程的 BFD for OSPF

配置指定进程的 BDF for OSPF 特性，可以快速检测链路的状态，提高链路状态变化时 OSPF 的收敛速度。

背景信息

配置 BDF for OSPF 特性后，当 BFD（Bidirectional Forwarding Detection）检测到链路故障时，能够将故障快速的通告给链路两端的路由器，触发 OSPF 的快速收敛。当邻居关系为 Down 时，则动态删除 BFD 会话。

OSPF 创建 BFD 会话需要先使能全局 BFD 功能。

如果对某个 OSPF 进程下所有的接口配置 BFD 特性，请在链路两端建立 BFD 会话的路由器上均进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **bfd**，配置全局 BFD 功能。

步骤 3 执行命令 **quit**，返回系统视图。

步骤 4 执行命令 **ospf** [*process-id*]，进入 OSPF 进程视图。

步骤 5 执行命令 **bfd all-interfaces enable**，配置 BFD for OSPF 特性，使用缺省参数值建立 BFD 会话。

当配置了全局 BFD 特性，且邻居状态为 Full 时，OSPF 为进程下所有的接口都建立了使用缺省参数值的 BFD 会话。

步骤 6 （可选）执行命令 **bfd all-interfaces** { **min-rx-interval** *receive-interval* | **min-tx-interval** *transmit-interval* | **detect-multiplier** *multiplier-value* } *，修改 BFD 会话的参数。

BFD 报文实际发送时间间隔和检测倍数一般推荐使用缺省值，即不执行该命令。

具体参数如何配置取决于网路状况以及对网络可靠性的要求，对于网络可靠性要求较高链路，可以配置减小 BFD 报文实际发送时间间隔；对于网络可靠性要求较低的链路，可以配置增大 BFD 报文实际发送时间间隔。

说明

- 本地 BFD 报文实际发送时间间隔 = MAX { 本地配置的发送时间间隔 *transmit-interval*，对端配置的接收时间间隔 *receive-interval* }
- 本地 BFD 报文实际接收时间间隔 = MAX { 对端配置的发送时间间隔 *transmit-interval*，本地配置的接收时间间隔 *receive-interval* }
- 本实 BFD 报文际地检测时间 = 本地实际接收时间间隔 × 对端配置的 BFD 检测倍数 *multiplier-value*

例如，

- 本地配置的发送时间间隔为 200ms，本地配置的接收时间间隔为 300ms，本地检测倍数为 4。
- 对端配置的发送时间间隔为 100ms，对端配置的接收时间间隔为 600ms，对端检测倍数为 5。

则：

- 本地实际的发送时间间隔为 MAX { 200ms，600ms } = 600ms，本地实际接收时间间隔为 MAX { 100ms，300ms } = 300ms，本地实际检测时间间隔为 300ms × 5 = 1500ms。
- 对端实际的发送时间间隔为 MAX { 100ms，300ms } = 300ms，对端实际接收时间间隔为 MAX { 200ms，600ms } = 600ms，对端实际检测时间间隔为 600ms × 4 = 2400ms。

步骤 7 （可选）阻止指定接口创建 BFD 会话。

配置 BDF for OSPF 特性后，OSPF 进程下所有邻居状态为 Full 的接口都将创建 BFD 会话。如果不希望某些接口使能 BFD 特性，可以阻止指定接口创建 BFD 会话。

1. 执行命令 **quit**，返回系统视图。
2. 执行命令 **interface** *interface-type interface-number*，进入接口视图。
3. 执行命令 **ospf bfd block**，阻止指定接口创建 BFD 会话。

----结束

5.9.3 配置指定接口的 BFD for OSPF

配置指定接口的 BFD for OSPF 特性，可以提高某些接口故障时的 OSPF 收敛速度。

背景信息

如果希望单独只对某些指定的接口配置 BFD for OSPF 特性，当这些接口的链路发生故障时，路由器可以快速的感知，并及时通知 OSPF 重新计算路由，从而提高 OSPF 的收敛速度。当邻居关系为 Down 时，则动态删除 BFD 会话。

OSPF 创建 BFD 会话需要先使能全局 BFD 功能。

请在指定接口配置 BFD 会话的路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **bfd**，配置全局 BFD 功能。

步骤 3 执行命令 **quit**，返回系统视图。

步骤 4 执行命令 **interface** *interface-type interface-number*，进入接口视图。

步骤 5 执行命令 **ospf bfd enable**，配置 BFD for OSPF 特性，使用缺省参数值建立 BFD 会话。

当配置了全局 BFD 特性，且邻居状态为 Full 时，OSPF 为指定的接口建立了使用缺省参数值的 BFD 会话。

说明

接口上配置 BFD for OSPF 特性的优先级高于进程中配置 BFD for OSPF 特性的优先级。

步骤 6 （可选）执行命令 **ospf bfd** { **min-rx-interval** *receive-interval* | **min-tx-interval** *transmit-interval* | **detect-multiplier** *multiplier-value* } *，修改 BFD 会话的参数。

BFD 报文实际发送时间间隔和检测倍数一般推荐使用缺省值，即不执行该命令。

具体参数如何配置取决于网络状况以及对网络可靠性的要求，对于网络可靠性要求较高链路，可以配置减小 BFD 报文实际发送时间间隔；对于网络可靠性要求较低的链路，可以配置增大 BFD 报文实际发送时间间隔。

说明

- 本地 BFD 报文实际发送时间间隔 = MAX { 本地配置的发送时间间隔 *transmit-interval*，对端配置的接收时间间隔 *receive-interval* }
- 本地 BFD 报文实际接收时间间隔 = MAX { 对端配置的发送时间间隔 *transmit-interval*，本地配置的接收时间间隔 *receive-interval* }
- 本实 BFD 报文实际检测时间 = 本地实际接收时间间隔 × 对端配置的 BFD 检测倍数 *multiplier-value*

例如，

- 本地配置的发送时间间隔为 200ms，本地配置的接收时间间隔为 300ms，本地检测倍数为 4。
- 对端配置的发送时间间隔为 100ms，对端配置的接收时间间隔为 600ms，对端检测倍数为 5。

则：

- 本地实际的发送时间间隔为 MAX { 200ms，600ms } = 600ms，本地实际接收时间间隔为 MAX { 100ms，300ms } = 300ms，本地实际检测时间间隔为 300ms × 5 = 1500ms。
- 对端实际的发送时间间隔为 MAX { 100ms，300ms } = 300ms，对端实际接收时间间隔为 MAX { 200ms，600ms } = 600ms，对端实际检测时间间隔为 600ms × 4 = 2400ms。

----结束

5.9.4 检查配置结果

成功配置 BFD for OSPF 特性后，您可以查看到 OSPF 的 BFD 会话信息。

前提条件

已经完成 BFD for OSPF 的所有配置。

操作步骤

- 使用 **display ospf** [*process-id*] **bfd session** *interface-type interface-number* [*router-id*] 或 **display ospf** [*process-id*] **bfd session** { *router-id* | **all** }命令查看 OSPF 的 BFD 会话信息。

----结束

5.10 配置 OSPF GR

配置 OSPF GR 可以避免流量中断和主备板切换带来的路由震荡。

5.10.1 建立配置任务

介绍配置 OSPF GR 特性的应用环境、前置任务和数据准备，可以帮助您快速、准确地完成配置任务。

应用场景

对于 OSPF 协议，为了避免流量中断和主备板切换带来的路由震荡，可以使能 OSPF 协议的 GR 特性。

OSPF 通过 GR 重启后，Restarter 路由器和 Helper 路由器之间重新建立邻居关系，交换路由信息并同步数据库，更新路由表和转发表，从而实现 OSPF 快速收敛，保持网络拓扑稳定。

说明

在实际应用中，为了实现业务转发不受主板故障的影响，通常在双主板的硬件环境下配置 OSPF GR 才有意义。

AR150/200 只能作为 Helper 路由器，不能作为 Restarter 路由器。

前置任务

在配置 OSPF GR 特性之前，需完成以下任务：

- 配置接口的网络层地址，使相邻节点之间网络层可达。
- **配置 OSPF 的基本功能**。

数据准备

在配置 OSPF GR 之前，需要准备以下数据。

序号	数据
1	OSPF 进程号
2	（可选）建立 GR 会话的参数 **说明** 推荐使用缺省值。

5.10.2 使能 OSPF GR

使能 OSPF GR，实现 OSPF 快速收敛，保持网络拓扑稳定。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **ospf** [*process-id*]，进入 OSPF 视图。

步骤 3 执行命令 **opaque-capability enable**，使能 opaque-LSA 功能。

因为 OSPF 中通过 Type-9 类 LSA 对 OSPF GR 支持，所以需要首先使能 OSPF 的 opauqe-LSA 特性。

步骤 4 执行命令 **graceful-restart**，使能 OSPF GR 特性。

----结束

5.10.3 （可选）配置 Restarter 端 GR 的会话参数

Restarter 端 GR 的会话参数包括 GR 周期，Planned GR 和 Totally GR。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **ospf** [*process-id*]，进入 OSPF 视图。

步骤 3 执行命令 **graceful-restart [period** *period* **| planned-only | partial]** *，配置 Restarter 端 GR 的会话参数。

- **period** 用来配置 Restarter 端 GR 的周期。缺省情况下，重启的时间为 120 秒。
- **planned-only** 用来配置 Restarter 只支持 Planned GR。缺省情况下，Restarter 支持 Planned GR 和 Unplanned GR。
- **partial** 用来配置 Restarter 支持 Partial GR。缺省情况下，Restarter 支持 Totally GR。

----结束

5.10.4 （可选）配置 Helper 端 GR 的会话参数

Helper 端 GR 的会话参数包括过滤策略，外部 LSA 检查和 Planned GR。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **ospf** [*process-id*]，进入 OSPF 视图。

步骤 3 执行命令 **graceful-restart helper-role** { [{ **ip-prefix** *ip-prefix-name* | **acl-number** *acl-number* | **acl-name** *acl-name* } | **ignore-external-lsa** | **planned-only**] * | **never** }，配置 Helper 端 GR 的会话参数。

- ACL 参数用来配置过滤策略，只有通过过滤器策略后才能进入 Helper 模式。
- **ignore-external-lsa** 用来配置 Helper 不对自治系统外部的 LSA（AS-external LSA）进行检查。缺省情况下，执行外部 LSA 检查。
- **planned-only** 用来配置 Helper 只支持 Planned GR。缺省情况下，Helper 支持 Planned GR 和 Unplanned GR。
- **never** 用来配置路由器不支持 Helper 模式。

----结束

5.10.5 检查配置结果

成功配置 OSPF GR 后，您可以查看到 OSPF GR 的状态。

前提条件

已经完成 OSPF GR 的所有配置。

操作步骤

- 使用 **display ospf** [*process-id*] **graceful-restart** [**verbose**]命令查看 OSPF GR 的重启状态。

----结束

5.11 提高 OSPF 网络的安全性

在对安全性较高的网络中，可以通过配置 OSPF GTSM 机制和验证方式来提高 OSPF 网络的安全性。

5.11.1 建立配置任务

在提高 OSPF 网络的安全性前了解此特性的应用环境、配置此特性的前置任务和数据准备，可以帮助您快速、准确地完成配置任务。

应用环境

在安全性较高的网络中，可以通过配置 OSPF 验证特性和 GTSM 机制来提高 OSPF 网络的安全性。GTSM 机制通过 TTL 的检测来达到防止攻击的目的，如果攻击者模拟真实的 OSPF 协议单播报文，对一台路由器不断的发送报文，路由器接口板收到这些报文后，发现是本机报文，则直接上送控制层面的 OSPF 协议处理，而不加辨别其“合法性”，这样导致路由器控制层面因为处理这些“合法”报文，系统异常繁忙，CPU 占用率高。配置 GTSM 功能，通过检测 IP 报文头中的 TTL 值是否在一个预先定义好的特定范围内来对路由器进行保护，增强系统的安全性。

说明

因为 GTSM 只支持单播地址，因此在 OSPF 中 GTSM 的作用范围主要是虚连接和伪连接。

前置任务

在提高 OSPF 网络的安全性之前，需完成以下任务：

- 配置接口的网络层地址，使相邻节点之间网络层可达。
- **配置 OSPF 的基本功能**。

数据准备

在提高 OSPF 网络的安全性之前，需要准备以下数据。

序号	数据
1	OSPF 进程号
2	（可选）OSPF 的 VPN 实例名称
3	（可选）需要检测的 TTL 值
4	需要配置验证的 OSPF 区域编号
5	需要配置验证的 OSPF 接口编号
6	验证方式及密码

5.11.2 配置 OSPF GTSM 功能

GTSM 机制通过 TTL 的检测来达到防止攻击的目的。

背景信息

应用 GTSM 功能，需要在 OSPF 连接的两端都使能 GTSM。

被检测的报文的 TTL 值有效范围为 [255 - *hops*+1, 255]。

GTSM 只会对匹配 GTSM 策略的报文进行 TTL 检查。对于未匹配策略的报文，可以设置为通过或丢弃。如果配置 GTSM 缺省报文动作为丢弃，就需要在 GTSM 中配置所有可能的路由器连接情况，没有配置的路由器发送的报文将被丢弃，无法建立连接。因此，在保证安全性的同时会损失一些易用性。

对于丢弃的报文，可以通过 LOG 信息开关，控制是否对报文被丢弃的情况记录日志，以方便故障的定位。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **ospf valid-ttl-hops** *hops* [**vpn-instance***vpn-instance-name*]，配置 OSPF GTSM 功能。

说明

ospf valid-ttl-hops 命令有两个功能，一是使能 OSPF GTSM 特性，二是配置需要检测的 TTL 值。**vpn-instance** 参数只对后一个功能有效。因此，如果仅配置私网策略或仅配置公网策略，建议将未匹配 GTSM 策略的报文的缺省动作设置为 **pass**，以免其他实例的 OSPF 报文被错误地丢弃。

步骤 3 （可选）执行命令 **gtsm default-action** { **drop** | **pass** }，设置未匹配 GTSM 策略的报文的缺省动作。

缺省情况下，未匹配 GTSM 策略的报文可以通过过滤。

说明

如果仅仅配置了缺省动作，但没有配置 GTSM 策略时，GTSM 不起作用。

步骤 4 （可选）执行命令 **gtsm log drop-packet all**，打开指定单板的 LOG 信息的开关，在单板 GTSM 丢弃报文时记录 LOG 信息。

----结束

5.11.3 配置验证方式

OSPF 支持报文验证功能，只有通过验证的报文才能接收，否则将不能正常建立邻居关系。

背景信息

使用区域验证时，一个区域中所有的路由器在该区域下的验证模式和口令必须一致。例如，在 Area0 内所有路由器上配置验证模式为简单验证，口令为 abc。

接口验证方式用于在相邻的路由器之间设置验证模式和口令，优先级高于区域验证方式。

操作步骤

- 配置区域验证方式
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **ospf** [*process-id*]，进入 OSPF 进程视图。
  3. 执行命令 **area** *area-id*，进入 OSPF 区域视图。
  4. 请根据需求，配置 OSPF 区域的验证模式。

- 执行命令 **authentication-mode simple** [[**plain**] *plain-text* | **cipher** *cipher-text*]，配置 OSPF 区域的简单验证模式。
- 执行命令 **authentication-mode** { **md5** | **hmac-md5** } [*key-id* { **plain** *plain-text* | [**cipher**] *cipher-text* }]，配置 OSPF 区域的 md5 验证模式。
- 执行命令 **authentication-mode keychain** *keychain-name*，配置 OSPF 区域的 Keychain 验证模式。

  说明

  使用 Keychain 验证模式，需要在系统视图下配置 Keychain 信息。必须保证本端 ActiveSendKey 和对端 ActiveRecvKey 的 **key-id**、**algorithm**、**key-string** 相同，才能建立 OSPF 邻居。

- 配置接口验证方式
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **interface** *interface-type interface-number*，进入接口视图。
  3. 请根据需求，配置接口验证方式。
     - 执行命令 **ospf authentication-mode simple** [[**plain**] *plain-text* | **cipher** *cipher-text*]，配置 OSPF 接口的简单验证模式。
     - 执行命令 **ospf authentication-mode** { **md5** | **hmac-md5** } [*key-id* { **plain** *plain-text* | [**cipher**] *cipher-text* }]，配置 OSPF 接口的 MD5 验证模式。
     - 执行命令 **ospf authentication-mode null**，不对 OSPF 接口进行验证。
     - 执行命令 **ospf authentication-mode keychain** *keychain-name*，配置 OSPF 区域的 Keychain 验证模式。

       说明

       使用 Keychain 验证模式，需要在系统视图下配置 Keychain 信息。必须保证本端 ActiveSendKey 和对端 ActiveRecvKey 的 **key-id**、**algorithm**、**key-string** 相同，才能建立 OSPF 邻居。

  除 Keychain 验证模式外，同一网段的接口的验证模式和口令必须相同，不同网段可以不同。

----结束

5.11.4 检查配置结果

通过配置 OSPF 的各种性能，提高 OSPF 网络的安全性后，您可以查看到 GTSM 和概要统计信息。

前提条件

已经完成提高 OSPF 网络的安全性的所有配置。

操作步骤

- 使用 **display gtsm statistics all** 命令查看 GTSM 的统计信息。
- 使用 **display ospf** [*process-id*] **request-queue** [*interface-type interface-number*] [*neighbor-id*]命令查看 OSPF 请求列表。
- 使用 **display ospf** [*process-id*] **retrans-queue** [*interface-type interface-number*] [*neighbor-id*]命令查看 OSPF 重传列表。
- 使用以下命令查看 OSPF 的错误信息：

- **display ospf** [*process-id*] **error**[**lsa**]
- **display ospf error** [**packet** [*number*]]

----结束

5.12 配置 OSPF 网管功能

OSPF 同时支持网管功能，可以配置 OSPF MIB 与某一进程绑定，以及发送 Trap 消息和日志功能。

5.12.1 建立配置任务

在配置 OSPF 网管功能前了解此特性的应用环境、配置此特性的前置任务和数据准备，可以帮助您快速、准确地完成配置任务。

应用环境

OSPF 同时支持网管功能，可以配置 OSPF MIB 与某一进程绑定，以及发送 Trap 消息和日志功能。

前置任务

在配置 OSPF 的网管功能之前，需要完成以下任务：

- 配置接口的网络层地址，使相邻节点网络层可达。
- **配置 OSPF 的基本功能**。

数据准备

在配置 OSPF 网管功能之前，需要准备以下数据。

序号	数据
1	OSPF 进程号

5.12.2 配置 OSPF MIB 绑定

OSPF MIB 是一个虚拟的数据库，是在被管理设备端维护的设备状态信息集。

背景信息

当启动了多个 OSPF 进程时，可以配置 OSPF MIB 对哪个进程进行处理，即绑定在哪个进程。

请在运行 OSPF 协议的路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **ospf mib-binding** *process-id*，配置 OSPF MIB 绑定。

----结束

5.12.3 配置 OSPF TRAP 功能

告警是系统检测到故障而产生的通知，告警中携带对应的故障信息。Trap 信息是路由器发送到网管设备的信息。

背景信息

请在运行 OSPF 协议的路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **snmp-agent trap enable feature-name ospf** [**trap-name** { **ospfifauthfailure** | **ospfifconfigerror** | **ospfifrxbadpacket** | **ospfifstatechange** | **ospflsdbapproachingoverflow** | **ospflsdboverflow** | **ospfmaxagelsa** | **ospfnbrrestarthelperstatuschange** | **ospfnbrstatechange** | **ospfnssatranslatorstatuschange** | **ospforiginatelsa** | **ospfrestartstatuschange** | **ospftxretransmit** | **ospfvirtifauthfailure** | **ospfvirtifconfigerror** | **ospfvirtifrxbadpacket** | **ospfvirtifstatechange** | **ospfvirtiftxretransmit** | **ospfvirtnbrrestarthelperstatuschange** | **ospfvirtnbrstatechange** }]，打开 OSPF 模块的告警开关。

如果只打开某个或几个事件的告警开关时，请选择 **trap-name**。

----结束

5.12.4 配置 OSPF 日志信息功能

日志信息记录了用户对路由器的操作（比如命令配置）以及特定事件（比如网络连接失效）等信息。

背景信息

请在运行 OSPF 协议的路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **ospf** [*process-id*]，进入 OSPF 进程视图。

步骤 3 执行命令 **enable log** [**config** | **error** | **state** | **snmp-trap**]，使能日志信息。

----结束

5.12.5 检查配置结果

通过配置 OSPF 的网管功能，您可以查看到信息通道的内容、信息中心记录的各项信息、日志缓冲区记录和告警缓冲区记录的信息。

前提条件

已经完成 OSPF 网管功能的所有配置。

操作步骤

- 使用 **display ospf** [*process-id*] **brief** 命令查看 OSPF MIB 绑定信息。
- 使用 **display snmp-agent trap feature-name ospf all** 命令查看 OSPF 模块的所有告警信息。

----结束

5.13 维护 OSPF

维护 OSPF，包括复位、清除 OSPF。

5.13.1 复位 OSPF

通过重启 OSPF，达到复位的目的。可以选择以 GR 的方式复位 OSPF。

背景信息

复位 OSPF 连接（执行 **reset ospf** 命令）会导致路由器之间的 OSPF 邻接关系中断。务必仔细确认是否必须执行复位 OSPF 连接的操作。

如果需要复位 OSPF 连接，可在用户视图下选择执行以下命令。

操作步骤

- 在用户视图下执行 **reset ospf** [*process-id*] **process** [**flush-waiting-timer** *time*]命令重启 OSPF 进程。
- 在用户视图下执行 **reset ospf** [*process-id*] **process** [**graceful-restart**]命令以 GR 方式重启 OSPF 进程。

----结束

5.13.2 清除 OSPF

清除 OSPF 包括清除 OSPF 的计数器、引入的路由和单板上的 GTSM 统计信息。

背景信息

清除 OSPF 的信息后，以前的信息将无法恢复，务必仔细确认。

在确认需要清除 OSPF 的运行信息后，请在用户视图下执行以下命令。

操作步骤

- 在用户视图下执行 **reset ospf** [*process-id*] **counters** [**neighbor** [*interface-type interface-number*] [*router-id*]]命令清除 OSPF 计数器。
- 在用户视图下执行 **reset ospf** [*process-id*] **redistribution** 命令清除 OSPF 引入的路由。
- 在用户视图下执行 **reset gtsm statistics all** 命令清除单板上的 GTSM 统计信息。

----结束

5.14 配置举例

介绍 OSPF 配置举例。请结合配置流程图了解配置过程。配置示例中包括组网需求、配置注意事项、配置思路等。

5.14.1 配置 OSPF 基本功能示例

介绍 OSPF 基本功能的配置过程，包括在各路由器上使能 OSPF、指定不同区域内的网段。

组网需求

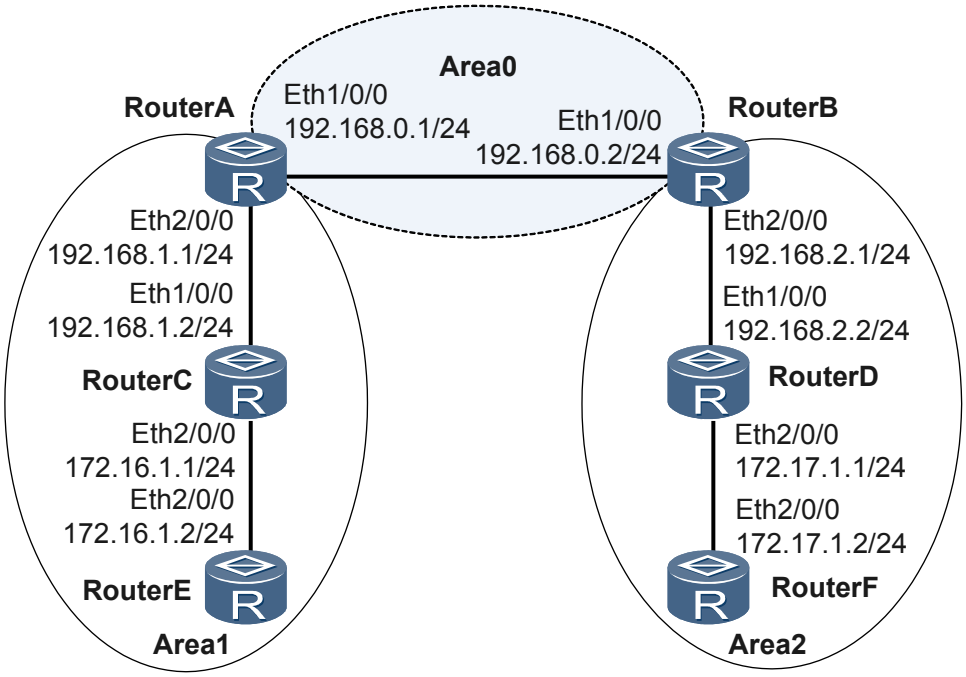
如图 5-5 所示，所有的路由器都运行 OSPF，并将整个自治系统划分为 3 个区域，其中 RouterA 和 RouterB 作为 ABR 来转发区域之间的路由。

配置完成后，每台路由器都应学到 AS 内的到所有网段的路由。

说明

AR150/200 仅可作为 RouterE 或 RouterF。

图 5-5 配置 OSPF 基本功能组网图



配置思路

采用如下的思路配置 OSPF 基本功能：

1. 在各路由器上使能 OSPF。
2. 指定不同区域内的网段。

数据准备

为完成此配置例，需准备如下的数据：

- RouterA 的 router id 1.1.1.1，运行的 OSPF 进程号 1，在区域 0 的网段 192.168.0.0/24，在区域 1 的网段 192.168.1.0/24。
- RouterB 的 router id 2.2.2.2，运行的 OSPF 进程号 1，在区域 0 的网段 192.168.0.0/24，在区域 2 的网段 192.168.2.0/24。
- RouterC 的 router id 3.3.3.3，运行的 OSPF 进程号 1，在区域 1 的网段 192.168.1.0/24，172.16.1.0/24。
- RouterD 的 router id 4.4.4.4，运行的 OSPF 进程号 1，在区域 2 的网段 192.168.2.0/24，172.17.1.0/24。
- RouterE 的 router id 5.5.5.5，运行的 OSPF 进程号 1，在区域 1 的网段 172.16.1.0/24。
- RouterF 的 router id 6.6.6.6，运行的 OSPF 进程号 1，在区域 2 的网段 172.17.1.0/24。

操作步骤

步骤 1 配置各接口的 IP 地址（略）

步骤 2 配置 OSPF 基本功能

配置 RouterA。

```
[RouterA] router id 1.1.1.1
[RouterA] ospf
[RouterA-ospf-1] area 0
[RouterA-ospf-1-area-0.0.0.0] network 192.168.0.0 0.0.0.255
[RouterA-ospf-1-area-0.0.0.0] quit
[RouterA-ospf-1] area 1
[RouterA-ospf-1-area-0.0.0.1] network 192.168.1.0 0.0.0.255
[RouterA-ospf-1-area-0.0.0.1] quit
```

配置 RouterB。

```
[RouterB] router id 2.2.2.2
[RouterB] ospf
[RouterB-ospf-1] area 0
[RouterB-ospf-1-area-0.0.0.0] network 192.168.0.0 0.0.0.255
[RouterB-ospf-1-area-0.0.0.0] quit
[RouterB-ospf-1] area 2
[RouterB-ospf-1-area-0.0.0.2] network 192.168.2.0 0.0.0.255
[RouterB-ospf-1-area-0.0.0.2] quit
```

配置 RouterC。

```
[RouterC] router id 3.3.3.3
[RouterC] ospf
[RouterC-ospf-1] area 1
[RouterC-ospf-1-area-0.0.0.1] network 192.168.1.0 0.0.0.255
[RouterC-ospf-1-area-0.0.0.1] network 172.16.1.0 0.0.0.255
```

```
[RouterC-ospf-1-area-0.0.0.1] quit
```

配置 RouterD。

```
[RouterD] router id 4.4.4.4
[RouterD] ospf
[RouterD-ospf-1] area 2
[RouterD-ospf-1-area-0.0.0.2] network 192.168.2.0 0.0.0.255
[RouterD-ospf-1-area-0.0.0.2] network 172.17.1.0 0.0.0.255
[RouterD-ospf-1-area-0.0.0.2] quit
```

配置 RouterE。

```
[RouterE] router id 5.5.5.5
[RouterE] ospf
[RouterE-ospf-1] area 1
[RouterE-ospf-1-area-0.0.0.1] network 172.16.1.0 0.0.0.255
[RouterE-ospf-1-area-0.0.0.1] quit
```

配置 RouterF。

```
[RouterF] router id 6.6.6.6
[RouterF] ospf
[RouterF-ospf-1] area 2
[RouterF-ospf-1-area-0.0.0.2] network 172.17.1.0 0.0.0.255
[RouterF-ospf-1-area-0.0.0.2] quit
```

步骤 3 验证配置结果

查看 RouterA 的 OSPF 邻居。

```
[RouterA] display ospf peer
          OSPF Process 1 with Router ID 1.1.1.1
                  Neighbors
 Area 0.0.0.0 interface 192.168.0.1(Ethernet1/0/0)'s neighbors
Router ID: 2.2.2.2      Address: 192.168.0.2
State: Full  Mode:Nbr is  Master  Priority: 1
   DR: 192.168.0.2  BDR: 192.168.0.1   MTU: 0
   Dead timer due in 36  sec
   Retrans timer interval: 5
   Neighbor is up for 00:15:04
   Authentication Sequence: [ 0 ]
                  Neighbors
 Area 0.0.0.1 interface 192.168.1.1(Ethernet2/0/0)'s neighbors
Router ID: 3.3.3.3       Address: 192.168.1.2
State: Full  Mode:Nbr is  Master  Priority: 1
   DR: 192.168.1.2  BDR: 192.168.1.1   MTU: 0
   Dead timer due in 39  sec
   Retrans timer interval: 5
   Neighbor is up for 00:07:32
   Authentication Sequence: [ 0 ]
```

显示 RouterA 的 OSPF 路由信息。

```
[RouterA] display ospf routing
          OSPF Process 1 with Router ID 1.1.1.1
                   Routing Tables
 Routing for Network
 Destination      Cost  Type         NextHop         AdvRouter       Area
 172.16.1.0/24     2    Transit      192.168.1.2     3.3.3.3         0.0.0.1
 172.17.1.0/24     3    Inter-area   192.168.0.2     2.2.2.2         0.0.0.0
 192.168.0.0/24    1    Stub         192.168.0.1     1.1.1.1         0.0.0.0
 192.168.1.0/24    1    Stub         192.168.1.1     1.1.1.1         0.0.0.1
 192.168.2.0/24    2    Inter-area   192.168.0.2     2.2.2.2         0.0.0.0
 Total Nets: 5
 Intra Area: 3  Inter Area: 2  ASE: 0  NSSA: 0
```

显示 RouterA 的 LSDB。

```
[RouterA] display ospf lsdb
```

```
          OSPF Process 1 with Router ID 1.1.1.1
                  Link State Database
                          Area: 0.0.0.0
 Type      LinkState ID    AdvRouter          Age  Len   Sequence   Metric
 Router    2.2.2.2         2.2.2.2            317  48    80000003   1
 Router    1.1.1.1         1.1.1.1            316  48    80000002   1
 Sum-Net   172.16.1.0      1.1.1.1            250  28    80000001   2
 Sum-Net   172.17.1.0      2.2.2.2            203  28    80000001   2
 Sum-Net   192.168.2.0     2.2.2.2            237  28    80000002   1
 Sum-Net   192.168.1.0     1.1.1.1            295  28    80000002   1
                          Area: 0.0.0.1
Type      LinkState ID    AdvRouter          Age  Len   Sequence   Metric
 Router    5.5.5.5         5.5.5.5            214  36    80000004   1
 Router    3.3.3.3         3.3.3.3            217  60    80000008   1
 Router    1.1.1.1         1.1.1.1            289  48    80000002   1
 Sum-Net   172.17.1.0      1.1.1.1            202  28    80000002   3
 Network   172.16.1.1      3.3.3.3            670  32    80000001   0
 Sum-Net   172.17.1.0      1.1.1.1            202  28    80000001   3
 Sum-Net   192.168.2.0     1.1.1.1            242  28    80000001   2
 Sum-Net   192.168.0.0     1.1.1.1            300  28    80000001   1
```

查看 RouterD 的路由表，并使用 Ping 进行测试连通性。

```
[RouterD] display ospf routing
          OSPF Process 1 with Router ID 4.4.4.4
                  Routing Tables
 Routing for Network
 Destination        Cost  Type       NextHop     AdvRouter       Area
 172.16.1.0/24      4   Inter-area 192.168.2.1     2.2.2.2         0.0.0.2
 172.17.1.0/24      1   Transit    172.17.1.1      4.4.4.4         0.0.0.2
 192.168.0.0/24     2   Inter-area 192.168.2.1     2.2.2.2         0.0.0.2
 192.168.1.0/24     3   Inter-area 192.168.2.1     2.2.2.2         0.0.0.2
 192.168.2.0/24     1   Stub       192.168.2.2     4.4.4.4         0.0.0.2
 Total Nets: 5
 Intra Area: 2  Inter Area: 3  ASE: 0  NSSA: 0
[RouterD] ping 172.16.1.1
  PING 172.16.1.1: 56  data bytes, press CTRL_C to break
    Reply from 172.16.1.1: bytes=56 Sequence=1 ttl=253 time=62 ms
    Reply from 172.16.1.1: bytes=56 Sequence=2 ttl=253 time=16 ms
    Reply from 172.16.1.1: bytes=56 Sequence=3 ttl=253 time=62 ms
    Reply from 172.16.1.1: bytes=56 Sequence=4 ttl=253 time=94 ms
    Reply from 172.16.1.1: bytes=56 Sequence=5 ttl=253 time=63 ms
  --- 172.16.1.1 ping statistics ---
    5 packet(s) transmitted
    5 packet(s) received
    0.00% packet loss
    round-trip min/avg/max = 16/59/94 ms
```

----结束

配置文件

- RouterA 的配置文件

```
#
 sysname RouterA
#
router id 1.1.1.1
#
interface Ethernet1/0/0
 ip address 192.168.0.1 255.255.255.0
#
interface Ethernet2/0/0
ip address 192.168.1.1 255.255.255.0
#
ospf 1
 area 0.0.0.0
  network 192.168.0.0 0.0.0.255
 area 0.0.0.1
```

```
  network 192.168.1.0 0.0.0.255
#
return
```

- RouterB 的配置文件

```
#
 sysname RouterB
#
router id 2.2.2.2
#
interface Ethernet1/0/0
ip address 192.168.0.2 255.255.255.0
#
interface Ethernet2/0/0
 ip address 192.168.2.1 255.255.255.0
#
ospf 1
 area 0.0.0.0
  network 192.168.0.0 0.0.0.255
 area 0.0.0.2
  network 192.168.2.0 0.0.0.255
#
return
```

- RouterC 的配置文件

```
#
 sysname RouterC
#
router id 3.3.3.3
#
interface Ethernet1/0/0
 ip address 192.168.1.2 255.255.255.0
#
interface Ethernet2/0/0
 ip address 172.16.1.1 255.255.255.0
#
ospf 1
 area 0.0.0.1
  network 192.168.1.0 0.0.0.255
  network 172.16.1.0 0.0.0.255
#
return
```

- RouterD 的配置文件

```
#
 sysname RouterD
#
router id 4.4.4.4
#
interface Ethernet1/0/0
 ip address 192.168.2.2 255.255.255.0
#
interface Ethernet2/0/0
 ip address 172.17.1.1 255.255.255.0
#
ospf 1
 area 0.0.0.2
  network 192.168.2.0 0.0.0.255
  network 172.17.1.0 0.0.0.255
#
return
```

- RouterE 的配置文件

```
#
 sysname RouterE
#
router id 5.5.5.5
#
interface Ethernet2/0/0
```

```
 ip address 172.16.1.2 255.255.255.0
#
ospf 1
 area 0.0.0.1
  network 172.16.1.0 0.0.0.255
#
return
```

- RouterF 的配置文件

```
#
 sysname RouterF
#
router id 6.6.6.6
#
interface Ethernet2/0/0
 ip address 172.17.1.2 255.255.255.0
#
ospf 1
 area 0.0.0.2
  network 172.17.1.0 0.0.0.255
#
return
```

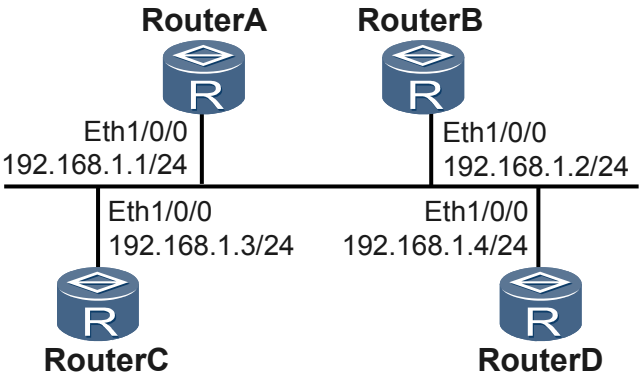
5.14.2 配置 OSPF 的 DR 选择示例

介绍在广播网络中，配置接口上的 DR 优先级进行 DR 选择的过程。

组网需求

在图 5-6 中，RouterA 的优先级为 100，它是网络上的最高优先级，所以 RouterA 被选为 DR；RouterC 是优先级第二高的，被选为 BDR；RouterB 的优先级为 0，这意味着它将无法成为 DR 或 BDR；RouterD 没有配置优先级，取缺省值 1。

图 5-6 配置 OSPF 的 DR 选择组网图



配置思路

采用如下的思路配置 OSPF 的 DR 选择：

1. 配置各路由器上 router id，使能 OSPF，指定网段。
2. 在缺省优先级情况下，查看各路由器 DR/BDR 状态。
3. 配置接口上的 DR 优先级，查看 DR/BDR 状态。

数据准备

为完成此配置例，需准备如下的数据：

- RouterA 的 Router ID 1.1.1.1，DR 优先级 100。
- RouterB 的 Router ID 2.2.2.2，DR 优先级 0。
- RouterC 的 Router ID 3.3.3.3，DR 优先级 2。
- RouterD 的 Router ID 4.4.4.4，DR 优先级取缺省值 1。

操作步骤

步骤 1 配置各接口的 IP 地址（略）

步骤 2 配置 OSPF 基本功能

配置 RouterA。

```
[RouterA] router id 1.1.1.1
[RouterA] ospf
[RouterA-ospf-1] area 0
[RouterA-ospf-1-area-0.0.0.0] network 192.168.1.0 0.0.0.255
[RouterA-ospf-1-area-0.0.0.0] quit
```

配置 RouterB。

```
[RouterB] router id 2.2.2.2
[RouterB] ospf
[RouterB-ospf-1] area 0
[RouterB-ospf-1-area-0.0.0.0] network 192.168.1.0 0.0.0.255
[RouterB-ospf-1-area-0.0.0.0] quit
```

配置 RouterC。

```
[RouterC] router id 3.3.3.3
[RouterC] ospf
[RouterC-ospf-1] area 0
[RouterC-ospf-1-area-0.0.0.0] network 192.168.1.0 0.0.0.255
[RouterC-ospf-1-area-0.0.0.0] quit
```

配置 RouterD。

```
[RouterD] router id 4.4.4.4
[RouterD] ospf
[RouterD-ospf-1] area 0
[RouterD-ospf-1-area-0.0.0.0] network 192.168.1.0 0.0.0.255
[RouterD-ospf-1-area-0.0.0.0] quit
```

查看 DR/BDR 的状态。

```
[RouterA] display ospf peer
          OSPF Process 1 with Router ID 1.1.1.1
                  Neighbors
 Area 0.0.0.0 interface 192.168.1.1(Ethernet1/0/0)'s neighbors
 Router ID: 2.2.2.2      Address: 192.168.1.2
State: Full  Mode:Nbr is  Master  Priority: 1
DR: 192.168.1.4  BDR: 192.168.1.3  MTU: 0
   Dead timer due in 32  sec
   Retrans timer interval: 5
   Neighbor is up for 00:04:21
   Authentication Sequence: [ 0 ]
 Router ID: 3.3.3.3      Address: 192.168.1.3
State: Full  Mode:Nbr is  Master  Priority: 1
DR: 192.168.1.4  BDR: 192.168.1.3  MTU: 0
   Dead timer due in 37  sec
   Retrans timer interval: 5
```

```
   Neighbor is up for 00:04:06
   Authentication Sequence: [ 0 ]
 Router ID: 4.4.4.4      Address: 192.168.1.4
State: Full  Mode:Nbr is  Master  Priority: 1
DR: 192.168.1.4  BDR: 192.168.1.3  MTU: 0
   Dead timer due in 37  sec
   Retrans timer interval: 5
   Neighbor is up for 00:03:53
   Authentication Sequence: [ 0 ]
```

查看 RouterA 的邻居信息，可以看到 DR 优先级（缺省为 1）以及邻居状态，此时 RouterD 为 DR，RouterC 为 BDR。

说明

当优先级相同时，router-id 高的为 DR。若 DR、BDR 已经选择完毕，当一台新路由器加入后，即使它的 DR 优先级值最大，也不会立即成为该网段中的 DR。

步骤 3 配置接口上的 DR 优先级

配置 RouterA。

```
[RouterA] interface ethernet 1/0/0
[RouterA-Ethernet1/0/0] ospf dr-priority 100
[RouterA-Ethernet1/0/0] quit
```

配置 RouterB。

```
[RouterB] interface ethernet 1/0/0
[RouterB-Ethernet1/0/0] ospf dr-priority 0
[RouterB-Ethernet1/0/0] quit
```

配置 RouterC。

```
[RouterC] interface ethernet 1/0/0
[RouterC-Ethernet1/0/0] ospf dr-priority 2
[RouterC-Ethernet1/0/0] quit
```

查看 DR/BDR 的状态。

```
[RouterD] display ospf peer
          OSPF Process 1 with Router ID 4.4.4.4
                  Neighbors
 Area 0.0.0.0 interface 192.168.1.4(Ethernet1/0/0)'s neighbors
 Router ID: 1.1.1.1      Address: 192.168.1.1
   State: Full  Mode:Nbr is  Slave  Priority: 100
DR: 192.168.1.4  BDR: 192.168.1.3  MTU: 0
   Dead timer due in 31  sec
   Retrans timer interval: 5
   Neighbor is up for 00:11:17
   Authentication Sequence: [ 0 ]
 Router ID: 2.2.2.2      Address: 192.168.1.2
   State: Full  Mode:Nbr is  Slave  Priority: 0
DR: 192.168.1.4  BDR: 192.168.1.3  MTU: 0
   Dead timer due in 35  sec
   Retrans timer interval: 5
   Neighbor is up for 00:11:19
   Authentication Sequence: [ 0 ]
 Router ID: 3.3.3.3      Address: 192.168.1.3
   State: Full  Mode:Nbr is  Slave  Priority: 2
DR: 192.168.1.4  BDR: 192.168.1.3  MTU: 0
   Dead timer due in 33  sec
   Retrans timer interval: 5
   Neighbor is up for 00:11:15
   Authentication Sequence: [ 0 ]
```

步骤 4 重启 OSPF 进程

在各路由器的用户视图下，同时执行命令 **reset ospf 1 process**，以重启 OSPF 进程。

步骤 5 验证配置结果

查看 OSPF 邻居状态。

```
[RouterD] display ospf peer
          OSPF Process 1 with Router ID 4.4.4.4
                  Neighbors
 Area 0.0.0.0 interface 192.168.1.4(Ethernet1/0/0)'s neighbors
 Router ID: 1.1.1.1      Address: 192.168.1.1
State: Full  Mode:Nbr is  Slave  Priority: 100
DR: 192.168.1.1  BDR: 192.168.1.3  MTU: 0
   Dead timer due in 35  sec
   Retrans timer interval: 5
   Neighbor is up for 00:07:19
   Authentication Sequence: [ 0 ]
 Router ID: 2.2.2.2      Address: 192.168.1.2
State: Full  Mode:Nbr is  Master  Priority: 0
DR: 192.168.1.1  BDR: 192.168.1.3  MTU: 0
   Dead timer due in 35  sec
   Retrans timer interval: 5
   Neighbor is up for 00:07:19
   Authentication Sequence: [ 0 ]
 Router ID: 3.3.3.3      Address: 192.168.1.3
State: Full  Mode:Nbr is  Slave  Priority: 2
DR: 192.168.1.1  BDR: 192.168.1.3  MTU: 0
   Dead timer due in 37  sec
   Retrans timer interval: 5
   Neighbor is up for 00:07:17
   Authentication Sequence: [ 0 ]
```

查看 OSPF 接口的状态。

```
[RouterA] display ospf interface
          OSPF Process 1 with Router ID 1.1.1.1
                  Interfaces
 Area: 0.0.0.0
 IP Address  Type        State    Cost  Pri   DR              BDR
 192.168.1.1 Broadcast   DR       1     100 192.168.1.1 192.168.1.3
[RouterB] display ospf interface
          OSPF Process 1 with Router ID 2.2.2.2
                  Interfaces
 Area: 0.0.0.0
 IP Address      Type         State    Cost  Pri   DR              BDR
 192.168.1.2     Broadcast    DROther  1     0 192.168.1.1 192.168.1.3
```

如果邻居的状态是 Full，这说明它和邻居之间形成了邻接关系；如果停留在 2-Way 的状态，则说明他们都不是 DR 或 BDR，两者之间不需要交换 LSA。

如果 OSPF 接口的状态是 DROther，则说明它既不是 DR，也不是 BDR。

----结束

配置文件

- RouterA 的配置文件

```
#
 sysname RouterA
#
router id 1.1.1.1
#
interface Ethernet1/0/0
 ip address 192.168.1.1 255.255.255.0
 ospf dr-priority 100
#
ospf 1
```

```
 area 0.0.0.0
  network 192.168.1.0 0.0.0.255
#
return
```

- RouterB 的配置文件

```
#
 sysname RouterB
#
router id 2.2.2.2
#
interface Ethernet1/0/0
 ip address 192.168.1.2 255.255.255.0
 ospf dr-priority 0
#
ospf 1
 area 0.0.0.0
  network 192.168.1.0 0.0.0.255
#
return
```

- RouterC 的配置文件

```
#
 sysname RouterC
#
router id 3.3.3.3
#
interface Ethernet1/0/0
 ip address 192.168.1.3 255.255.255.0
 ospf dr-priority 2
#
ospf 1
 area 0.0.0.0
  network 192.168.1.0 0.0.0.255
#
return
```

- RouterD 的配置文件

```
#
 sysname RouterD
#
router id 4.4.4.4
#
interface Ethernet1/0/0
 ip address 192.168.1.4 255.255.255.0
#
ospf 1
 area 0.0.0.0
  network 192.168.1.0 0.0.0.255
#
return
```

5.14.3 配置 OSPF 的 Stub 区域示例

介绍引入了静态路由的 Stub 区域的配置过程，用来减少通告到此区域内的 LSA 数量，但不影响路由的可达性。

组网需求

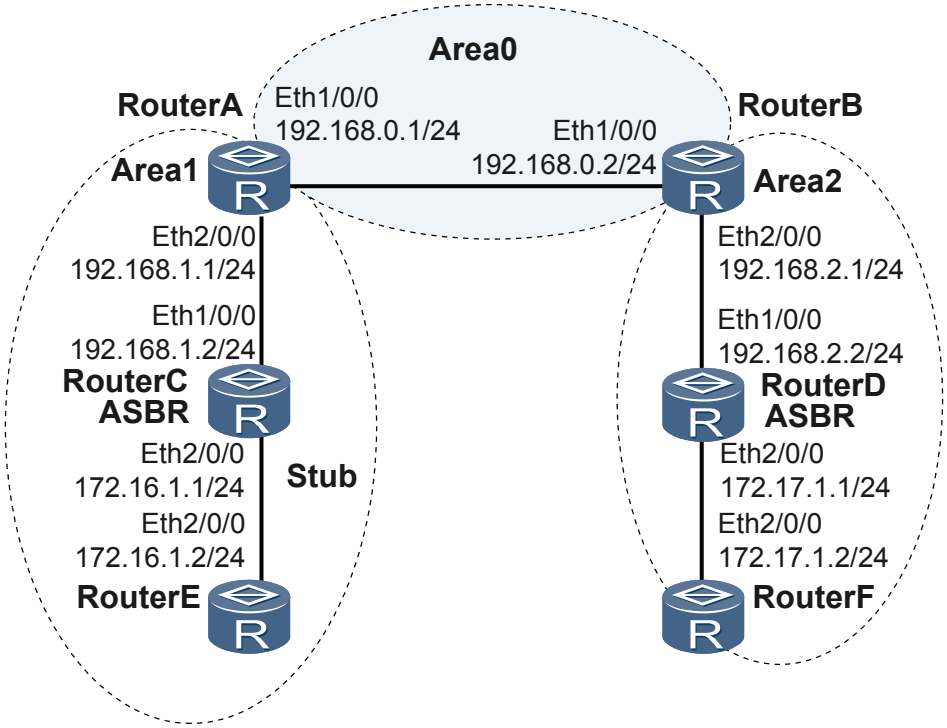
如图 5-7 所示，所有的路由器都运行 OSPF，整个自治系统划分为 3 个区域。其中 RouterA 和 RouterB 作为 ABR 来转发区域之间的路由，RouterD 作为 ASBR 引入了外部路由（静态路由）。

要求将 Area1 配置为 Stub 区域，减少通告到此区域内的 LSA 数量，但不影响路由的可达性。

说明

AR150/200 仅可作为 RouterE 或 RouterF。

图 5-7 配置 OSPF Stub 区域组网图



配置思路

采用如下的思路配置 OSPF 的 Stub 区域：

1. 在各路由器上使能 OSPF，配置 OSPF 基本功能。
2. 在 RouterD 上配置静态路由，并在 OSPF 中引入。
3. 配置 Area1 为 Stub 区域（需要在 Area1 内所有的路由器上配置 **Stub** 命令），在 RouterC 上查看 OSPF 路由信息。
4. 在 RouterA 上配置禁止向 Stub 区域通告 Type3 LSA，在 RouterC 上查看 OSPF 路由信息。

数据准备

为完成此配置例，需准备如下的数据：

- RouterA 的 router id 1.1.1.1，运行的 OSPF 进程号 1，区域 0 的网段 192.168.0.0/24，区域 1 的网段 192.168.1.0/24。
- RouterB 的 router id 2.2.2.2，运行的 OSPF 进程号 1，区域 0 的网段 192.168.0.0/24，区域 2 的网段 192.168.2.0/24。
- RouterC 的 router id 3.3.3.3，运行的 OSPF 进程号 1，区域 1 的网段 192.168.1.0/24，172.16.1.0/24。

- RouterD 的 router id 4.4.4.4，运行的 OSPF 进程号 1，区域 2 的网段 192.168.2.0/24，172.17.1.0/24。
- RouterE 的 router id 5.5.5.5，运行的 OSPF 进程号 1，区域 1 的网段 172.16.1.0/24。
- RouterF 的 router id 6.6.6.6，运行的 OSPF 进程号 1，区域 2 的网段 172.17.1.0/24。

操作步骤

步骤 1 配置接口的 IP 地址（略）

步骤 2 配置 OSPF 基本功能（请参见举例**配置 OSPF 的基本功能**）

步骤 3 配置 RouterD 引入静态路由

```
[RouterD] ip route-static 200.0.0.0 8 null 0
[RouterD] ospf
[RouterD-ospf-1] import-route static type 1
[RouterD-ospf-1] quit
```

查看 RouterC 的 ABR/ASBR 信息。

```
[RouterC] display ospf abr-asbr
          OSPF Process 1 with Router ID 3.3.3.3
                Routing Table to ABR and ASBR
 RtType         Destination     Area        Cost    Nexthop          Type
 Intra-area     1.1.1.1         0.0.0.1     1       192.168.1.1      ABR
 Inter-area     4.4.4.4         0.0.0.1     3       192.168.1.1      ASBR
```

查看 RouterC 的 OSPF 路由表。

说明

当 RouterC 所在区域为普通区域时，可以看到路由表中存在 AS 外部的路由。

```
[RouterC] display ospf routing
     OSPF Process 1 with Router ID 3.3.3.3
                  Routing Tables
 Routing for Network
 Destination        Cost  Type         NextHop         AdvRouter       Area
 172.16.1.0/24      1     Transit      172.16.1.1      3.3.3.3         0.0.0.1
 172.17.1.0/24      4     Inter-area   192.168.1.1     1.1.1.1         0.0.0.1
 192.168.0.0/24     2     Inter-area   192.168.1.1     1.1.1.1         0.0.0.1
 192.168.1.0/24     1     Stub         192.168.1.2     3.3.3.3         0.0.0.1
 192.168.2.0/24     3     Inter-area   192.168.1.1     1.1.1.1         0.0.0.1
 Routing for ASEs
 Destination    Cost      Type       Tag         NextHop         AdvRouter
 200.0.0.0/8    4         Type1      1           192.168.1.1     4.4.4.4
 Total Nets: 6
 Intra Area: 2  Inter Area: 3  ASE: 1  NSSA: 0
```

步骤 4 配置 Area1 为 Stub 区域

配置 RouterA。

```
[RouterA] ospf
[RouterA-ospf-1] area 1
[RouterA-ospf-1-area-0.0.0.1] stub
[RouterA-ospf-1-area-0.0.0.1] quit
```

配置 RouterC。

```
[RouterC] ospf
[RouterC-ospf-1] area 1
[RouterC-ospf-1-area-0.0.0.1] stub
[RouterC-ospf-1-area-0.0.0.1] quit
```

配置 RouterE。

```
[RouterE] ospf
[RouterE-ospf-1] area 1
[RouterE-ospf-1-area-0.0.0.1] stub
[RouterE-ospf-1-area-0.0.0.1] quit
```

显示 RouterC 的路由表。

说明

当把 RouterC 所在区域配置为 Stub 区域时，已经看不到 AS 外部的路由，取而代之的是一条缺省路由。

```
[RouterC] display ospf routing
          OSPF Process 1 with Router ID 3.3.3.3
                Routing Tables
 Routing for Network
 Destination        Cost  Type       NextHop         AdvRouter       Area
 0.0.0.0/0          2     Inter-area  192.168.1.1     1.1.1.1         0.0.0.1
 172.16.1.0/24      1     Transit     172.16.1.1      3.3.3.3         0.0.0.1
 172.17.1.0/24      4     Inter-area  192.168.1.1     1.1.1.1         0.0.0.1
 192.168.0.0/24     2     Inter-area  192.168.1.1     1.1.1.1         0.0.0.1
 192.168.1.0/24     1     Stub        192.168.1.2     3.3.3.3         0.0.0.1
 192.168.2.0/24     3     Inter-area  192.168.1.1     1.1.1.1         0.0.0.1
 Total Nets: 6
 Intra Area: 2  Inter Area: 4  ASE: 0  NSSA: 0
```

步骤 5 # 配置禁止向 Stub 区域通告 Type3 LSA。

```
[RouterA] ospf
[RouterA-ospf-1] area 1
[RouterA-ospf-1-area-0.0.0.1] stub no-summary
[RouterA-ospf-1-area-0.0.0.1] quit
```

步骤 6 验证配置结果

查看 RouterC 的 OSPF 路由表。

```
[RouterC] display ospf routing
          OSPF Process 1 with Router ID 3.3.3.3
                Routing Tables
 Routing for Network
 Destination        Cost  Type       NextHop         AdvRouter       Area
 0.0.0.0/0          2     Inter-area  192.168.1.1     1.1.1.1         0.0.0.1
 172.16.1.0/24      1     Transit     172.16.1.1      3.3.3.3         0.0.0.1
 192.168.1.0/24     1     Stub        192.168.1.2      3.3.3.3         0.0.0.1
 Total Nets: 3
 Intra Area: 2  Inter Area: 1  ASE: 0  NSSA: 0
```

说明

禁止向 Stub 区域通告 Summary LSA 后，Stub 路由器的路由表项进一步减少，只保留了一条通往区域外部的缺省路由。

----结束

配置文件

说明

RouterB 和 RouterF 的配置文件与前例相同，此处省略。

- RouterA 的配置文件

```
#
 sysname RouterA
#
router id 1.1.1.1
#
interface Ethernet1/0/0
 ip address 192.168.0.1 255.255.255.0
#
```

```
interface Ethernet2/0/0
 ip address 192.168.1.1 255.255.255.0
#
ospf 1
 area 0.0.0.0
  network 192.168.0.0 0.0.0.255
 area 0.0.0.1
  network 192.168.1.0 0.0.0.255
  stub no-summary
#
return
```

- RouterC 的配置文件

```
#
 sysname RouterC
#
router id 3.3.3.3
#
interface Ethernet1/0/0
 ip address 192.168.1.2 255.255.255.0
#
interface Ethernet2/0/0
 ip address 172.16.1.1 255.255.255.0
#
ospf 1
 area 0.0.0.1
  network 192.168.1.0 0.0.0.255
  network 172.16.1.0 0.0.0.255
  stub
#
return
```

- RouterD 的配置文件

```
#
 sysname RouterD
#
router id 4.4.4.4
#
interface Ethernet1/0/0
 ip address 192.168.2.2 255.255.255.0
#
interface Ethernet2/0/0
 ip address 172.17.1.1 255.255.255.0
#
ospf 1
 import-route static type 1
 area 0.0.0.2
  network 192.168.2.0 0.0.0.255
  network 172.17.1.0 0.0.0.255
#
ip route-static 200.0.0.0 255.0.0.0 NULL0
#
return
```

- RouterE 的配置文件

```
#
 sysname RouterE
#
router id 5.5.5.5
#
interface Ethernet2/0/0
 ip address 172.16.1.2 255.255.255.0
#
ospf 1
 area 0.0.0.1
  network 172.16.1.0 0.0.0.255
  stub
#
return
```

6 OSPFv3 配置

关于本章

OSPFv3 是 OSPF Version 3 的简称，通过组建 OSPFv3 网络，在自治域内发现并计算路由信息。OSPFv3 可以应用于大规模网络，最多可支持几百台路由器。

6.1 OSPFv3 概述
OSPFv3 使用了与 OSPFv2 相同的基本实现机制，但并不兼容 OSPFv2。

6.2 AR150/200 支持的 OSPFv3 特性
AR150/200 中支持的 OSPFv3 特性包括：多进程和 GR。

6.3 配置 OSPFv3 基本功能
只有配置了基本功能，才可组建 OSPFv3 网络。

6.4 建立或维持 OSPFv3 邻居或邻接关系
通过建立、维持 OSPFv3 邻居或邻接关系，可以组建 OSPFv3 网络。

6.5 配置 OSPFv3 的区域属性
OSPFv3 支持 Stub 区域和虚连接的配置，其原理及应用环境与 OSPFv2 相同。

6.6 配置 OSPFv3 的 NSSA 区域
通过配置 NSSA 区域，允许引入外部路由，同时增加一类新的 LSA，Type7 NSSA-LSA。

6.7 配置 OSPFv3 的路由属性
通过配置 OSPFv3 的路由属性改变 OSPFv3 的选路策略，以满足复杂网络环境中的需要。

6.8 控制 OSPFv3 的路由信息
控制 OSPFv3 路由信息包括路由聚合，对路由的过滤和引入外部路由。

6.9 调整和优化 OSPFv3 网络
在特殊的网络环境中配置 OSPFv3 的一些特性功能，对 OSPFv3 网络的性能进行调整和优化。

6.10 配置 OSPFv3 GR
通过 OSPFv3 GR 解决 OSPFv3 路由器重启后造成路由计算不准确、报文丢失的问题。

6.11 配置 OSPFv3 网管功能
OSPFv3 同时支持网管功能，可以配置 OSPFv3 MIB 与某一进程绑定。

6.12 维护 OSPFv3

维护 OSPFv3 是指复位 OSPFv3 和调试 OSPFv3。

6.13 配置举例

介绍 OSPFv3 配置举例。请结合配置流程图了解配置过程。配置示例中包括组网需求、配置注意事项、配置思路等。

6.1 OSPFv3 概述

OSPFv3 使用了与 OSPFv2 相同的基本实现机制，但并不兼容 OSPFv2。

OSPFv3 是 OSPF 版本 3 的简称，主要提供对 IPv6 的支持，遵循的标准为 RFC2740（OSPF for IPv6）。

OSPFv3 和 OSPFv2 在很多方面是相同的：

- Router ID，Area ID，LSA Link State ID 仍然是 32 位的。
- 相同类型的报文：Hello 报文、DD 报文、LSR 报文、LSU 报文和 LSAck 报文。
- 相同的邻居发现机制和邻接（Adjacency）形成机制。
- 相同的 LSA 扩散（Flooding）机制和老化（Aging）机制。
- 基本相同的 LSA 类型。

OSPFv3 和 OSPFv2 的不同主要有：

- OSPFv3 是基于链路（Link）运行，OSPFv2 是基于网段（Network）运行。
- OSPFv3 在同一条链路上可以运行多个实例。
- OSPFv3 的拓扑关系和 IPv6 地址前缀没有关系。
- 使用 IPv6 的链路本地（Link-local）地址标识邻接的邻居。
- 新增的 3 种不同的 LSA 扩散范围。

6.2 AR150/200 支持的 OSPFv3 特性

AR150/200 中支持的 OSPFv3 特性包括：多进程和 GR。

在 AR150/200 目前的实现中，支持以下 OSPFv3 特性：

- 支持 RFC2740 规定的基本特性；
- 支持 OSPFv3 STUB 区域；
- 支持 OSPFv3 多进程（Multi-Process），可以在一台路由器上运行多个 OSPFv3 进程；
- 支持 OSPFv3 GR：
  - 当一台路由器进行重启或因为各种原因进行主备倒换时，自身会直接老化 FIB（Forward Information Base）表中的所有路由表项，造成路由中断。而且与它邻接的路由器会把它从邻居列表中删除，并通知给其他路由器，这样整个网络就会重新进行 SPF 计算。如果这台路由器很快又恢复，就会造成邻居关系震荡，从而导致路由震荡。
  - 如果仅是因为路由器异常重启，在使能平滑重启 OSPFv3 GR（Graceful Restart）功能后，仍可以保证流量转发不中断，网络不会因为路由器的短时重启而震荡。

说明

OSPFv3 功能使用 License 授权，缺省情况下，设备的 OSPFv3 功能受限无法使用。如果需要使用 OSPFv3 功能，请联系华为办事处申请并购买如下 License，

- AR150&200 数据业务增值包

6.3 配置 OSPFv3 基本功能

只有配置了基本功能，才可组建 OSPFv3 网络。

6.3.1 建立配置任务

在各项配置任务中，必须先启动 OSPFv3，指定接口与区域号后，才能配置其它的功能特性。

应用环境

配置 OSPFv3 时，必须先启动 OSPFv3 进程，并指定 Router ID，之后其它的功能才能配置或生效。

在各项配置任务中，必须先启动 OSPFv3，指定接口与区域号后，才能配置其它的功能特性。而配置与接口相关的功能特性不受 OSPFv3 是否使能的限制。

前置任务

在配置 OSPFv3 的基本功能之前，需完成以下任务：

- 各相邻节点网络层可达
- 使能 IPv6 能力

数据准备

在配置 OSPFv3 的基本功能之前，需要准备以下数据。

序号	数据
1	Router ID
2	OSPFv3 进程号
3	需要使能 OSPFv3 的接口及其所属区域

6.3.2 启动 OSPFv3

创建 OSPFv3 进程是配置所有 OSPFv3 特性的首要步骤。通过创建 OSPFv3 进程，还可以手工指定路由器的 Router ID。

背景信息

OSPFv3 支持多进程，一台路由器上启动的多个 OSPFv3 进程之间由不同的进程号区分。OSPFv3 进程号在启动 OSPFv3 时进行设置，它只在本地有效，不影响与其它路由器之间的报文交换。

Router ID 是一个 32 比特无符号整数，采用 IPv4 地址形式，是一台路由器在自治系统中的唯一标识。OSPFv3 的 Router ID 必须手工配置，如果没有配置 ID 号，OSPFv3 无法正常运行。

手工配置 Router ID 时，必须保证自治系统中任意两台路由器的 Router ID 都不相同。如果在同一台路由器上运行了多个 OSPFv3 进程，必须为不同的进程指定不同的 Router ID。

为保证 OSPFv3 运行的稳定性，在进行网络规划时，应确定路由器 ID 的划分并手工配置。

请在需要运行 OSPFv3 协议的每台路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **ospfv3** [*process-id*]，启动 OSPFv3，进入 OSPFv3 视图。

步骤 3 执行命令 **router-id** *router-id*，配置 Router ID。

----结束

6.3.3 在接口上使能 OSPFv3

由于接口多实例化，所以在将接口使能到 OSPFv3 时，需要指定是哪个接口实例被使能到 OSPFv3 进程中。

背景信息

在系统视图使能 OSPFv3 后，需要在接口使能 OSPFv3。

由于接口多实例化，所以在将接口使能到 OSPFv3 时，需要指定是哪个接口实例被使能到 OSPFv3 进程中，如果不指定实例 ID，则缺省为 0。建立邻居的接口上使能的实例必须相同。

请在需要运行 OSPFv3 协议的每台路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **interface** *interface-type interface-number*，进入接口视图。

步骤 3 执行命令 **ospfv3** *process-id* **area** *area-id* [**instance** *instance-id*]，在接口上使能 OSPFv3。

区域 ID 可以采用十进制整数或 IPv4 地址形式输入，但显示时使用 IPv4 地址形式。

步骤 4 （可选）执行命令 **ospfv3 network-type** { **broadcast** | **nbma** | **p2mp** [**non-broadcast**] | **p2p** } [**instance** *instance-id*]，配置接口的网络类型。

当接口支持多实例化，在将接口使能到 OSPFv3 时，必须指定是哪个接口实例被使能到 OSPFv3 进程中，即必须指定 *instance-id*。如果不指定实例 ID，则缺省为 0，会出现配置的接口的网络类型与实际接口的网络类型不匹配的情况。此时该步骤为必选步骤。

----结束

6.3.4 进入 OSPFv3 区域视图

OSPFv3 协议通过将自治系统划分成不同的区域，和区域中指定运行 OSPFv3 协议的接口和接口所属的区域，达到在自治区域中发现并计算路由的目的。

背景信息

在配置同一区域内的 OSPFv3 路由器时，应注意：大多数配置数据都应该对区域统一考虑，否则可能会导致相邻路由器之间无法交换信息，甚至导致路由信息的阻塞或者产生路由环。

请在需要运行 OSPFv3 协议的每台路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **ospfv3** [*process-id*]，进入 OSPFv3 视图。

步骤 3 执行命令 **area** *area-id*，进入 OSPFv3 区域视图。

区域 ID 可以采用十进制整数或 IPv4 地址形式输入，但显示时使用 IPv4 地址形式。

OSPFv3 的区域不能直接删除，当区域视图下的所有配置都删除，且该区域中相关接口的状态都为 DOWN，此区域就会被系统自动删除。

----结束

6.3.5 检查配置结果

OSPFv3 基本功能配置成功后，您可以查看到概要、LSDB、邻居和 OSPFv3 路由表信息。

前提条件

已经完成 OSPFv3 基本功能的所有配置。

操作步骤

- 使用 **display ospfv3** [*process-id*]命令查看 OSPFv3 进程的概要信息。
- 使用 **display ospfv3** [*process-id*] **interface** [**area** *area-id*] [*interface-type interface-number*]命令查看 OSPFv3 接口信息。
- 使用以下命令查看 OSPFv3 的 LSDB 信息:
  - **display ospfv3** [*process-id*] **lsdb** [**area** *area-id*] [**originate-router** *advertising-router-id* | **self-originate**] [{ **router** | **network** | **inter-router** [**asbr-router** *asbr-router-id*] | { **inter-prefix** | **nssa** } [*ipv6-address prefix-length*] | **link** | **intra-prefix** | **grace** } [*link-state-id*]]
  - **display ospfv3** [*process-id*] **lsdb** [**originate-router** *advertising-router-id* | **self-originate**] **external** [*ipv6-address prefix-length*] [*link-state-id*]
- 使用 **display ospfv3** [*process-id*] [**area** *area-id*] **peer** [*interface-type interface-number*] [**verbose**]或 **display ospfv3** [*process-id*] [**area** *area-id*] **peer** *neighbor-id* [**verbose**]命令查看 OSPFv3 邻居信息。
- 使用以下命令来查看 OSPFv3 路由表信息：
  - **display ospfv3** [*process-id*] **routing**
  - **display ospfv3** [*process-id*] **routing** [**abr-routes** | **asbr-routes** | **statistics** | *ipv6-address prefix-length* | **intra-routes** | **inter-routes** | **ase-routes** | **nssa-routes**]
- 使用 **display ospfv3** [*process-id*] **path** 命令查看 OSPFv3 的路径信息。

- 使用 **display default-parameter ospfv3** 命令查看 OSPFv3 缺省配置信息。

----结束

6.4 建立或维持 OSPFv3 邻居或邻接关系

通过建立、维持 OSPFv3 邻居或邻接关系，可以组建 OSPFv3 网络。

6.4.1 建立配置任务

在接口上配置的各种参数要和邻接路由器的参数保持一致。

应用环境

在实际应用中，建立或维持 OSPFv3 邻居关系是组建 OSPFv3 网络的重要前提，通过本节的配置，你可以：

- 通过改变 OSPFv3 报文的定时器，可以调整 OSPFv3 网络的收敛速度以及协议报文带来的网络负荷。
- 通过配置 OSPFv3 重传限制可以实现在 OSPFv3 重传报文时，如果超过了设定的重传次数，OSPFv3 将断开邻居，避免在邻居收不到报文的情况下，一直重传造成的死循环。
- 通过调整 LSA 更新和接收的时间间隔，可以提高 OSPFv3 网络的收敛速度。

前置任务

在建立或维持 OSPFv3 邻居或邻接关系之前，需完成以下任务：

- 使能 IPv6 能力
- **配置 OSPFv3 基本功能**。

数据准备

在建立或维持 OSPFv3 邻居或邻接关系之前，需要准备以下数据。

序号	数据
1	发送 Hello 报文的时间间隔
2	相邻路由器间失效时间
3	相邻路由器重传 LSA 的时间间隔
4	LSA 传送延迟时间

6.4.2 配置接口发送 Hello 报文的时间间隔

通过调整 OSPFv3 邻居之间的 Hello 定时器的时间间隔，可以改变邻居建立的速度，从而影响网络收敛的速度。

背景信息

Hello 报文周期性地被发送至邻居路由器，用于发现与维持邻居关系、选举 DR 与 BDR。根据 RFC2328 的规定，要保持网络邻居间的 Hello 时间间隔一致。需要注意的是，Hello 定时器的值与路由收敛速度、网络负荷大小成反比。

请在运行 OSPFv3 协议的路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **interface** *interface-type interface-number*，进入接口视图。

步骤 3 执行命令 **ospfv3 timer hello** *interval* [**instance** *instance-id*]，配置接口发送 Hello 报文的时间间隔。

----结束

6.4.3 配置相邻路由器失效的时间

在相邻路由器失效时间间隔内，若未收到邻居的 Hello 报文，就认为该邻居已失效。

背景信息

在一定时间间隔内，如果路由器未收到对方的 Hello 报文，则认为对端路由器失效，这个时间间隔被称为相邻路由器间的失效时间。在同一接口上路由器的失效时间应至少为 Hello 间隔时间的 4 倍。

请在运行 OSPFv3 协议的路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **interface** *interface-type interface-number*，进入接口视图。

步骤 3 执行命令 **ospfv3 timer dead** *interval* [**instance** *instance-id*]，配置相邻路由器间失效时间。

----结束

6.4.4 配置邻接路由器重传 LSA 的间隔

当一台路由器向它的邻居发送一条 LSA 后，需要等到对方的确认报文。若在重传间隔时间内没有收到对方的确认报文，就会向邻居重传这条 LSA。

背景信息

请在运行 OSPFv3 协议的路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **interface** *interface-type interface-number*，进入接口视图。

步骤 3 执行命令 **ospfv3 timer retransmit** *interval* [**instance** *instance-id*]，配置相邻路由器重传 LSA 的时间间隔。

重传间隔的值必须大于一个报文在两台路由器之间传送一个来回的时间。

说明

邻接路由器重传 LSA 时间间隔的值不要设置得太小，否则将会引起不必要的重传。

----结束

6.4.5 配置接口的 LSA 传送延迟时间

因为 OSPFv3 报文在链路传送时需要花费时间，所以当路由器的一个接口发送 LSA 时，会在其老化时间（age）上增加一定的延迟时间。

背景信息

由于 LSA 在本路由器的链路状态数据库 LSDB 中会随时间老化，但在网络的传输过程中却不会，所以有必要在发送之前将 LSA 的老化时间增加一定的延迟时间。对于低速网络，该项配置尤为重要。

请在运行 OSPFv3 协议的路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **interface** *interface-type interface-number*，进入接口视图。

步骤 3 执行命令 **ospfv3 trans-delay** *interval* [**instance** *instance-id*]，配置接口的 LSA 传送延迟时间。

----结束

6.4.6 检查配置结果

OSPFv3 邻居或邻接关系稳定后，您可以查看到运行 OSPFv3 的接口和邻居信息。

前提条件

已经完成建立或维持 OSPFv3 邻居或邻接关系的所有配置。

操作步骤

- 使用 **display ospfv3** [*process-id*] **interface** [**area** *area-id*] [*interface-type interface-number*]命令查看 OSPFv3 接口信息。

----结束

6.5 配置 OSPFv3 的区域属性

OSPFv3 支持 Stub 区域和虚连接的配置，其原理及应用环境与 OSPFv2 相同。

6.5.1 建立配置任务

STUB 区域是一种可选的配置属性，但并不是每个区域都符合配置的条件。通常来说，STUB 区域位于自治系统的边界，并且只有一个 ABR 的非骨干区域。

应用环境

OSPFv3 划分区域后，可以减少网络中 LSA 的数量，OSPFv3 的扩展性也得以增强。对于位于 AS 边缘的一些非骨干区域，为了更多的缩减其路由表规模和降低 LSA 的数量，可以将它们配置为 STUB 区域。

前置任务

在配置 OSPFv3 的区域特性之前，需完成以下任务：

- 使能 IPv6 能力
- **配置 OSPFv3 基本功能**

数据准备

在配置 OSPFv3 区域之前，需要准备以下数据。

序号	数据
1	确定哪些区域需要配置为 STUB 区域
2	发布到 STUB 区域缺省路由的开销值

6.5.2 配置 OSPFv3 的 Stub 区域

Stub 区域是一种特定的区域，Stub 区域的 ABR 不传播它们接收到的自治系统外部路由，LSA 的数量大大减少。

背景信息

请在运行 OSPFv3 协议的 Stub 域内的每台路由器上进行如下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **ospfv3** [*process-id*]，进入 OSPFv3 视图。

步骤 3 执行命令 **area** *area-id*，进入 OSPFv3 区域视图。

步骤 4 执行命令 **stub** [**no-summary**]，配置一个区域为 Stub 区域。

步骤 5 （可选）执行命令 **default-cost** *cost*，配置发送到 Stub 区域缺省路由的开销值。

缺省情况下，发送到 Stub 区域缺省路由的开销值为 1。

仅在 Stub 区域的 ABR 上配置发送到 Stub 区域缺省路由的开销值，不需要在 Stub 区域中的其他路由器上配置。

stub 命令中的参数 **no-summary** 也只有在 ABR 上配置时才生效，如果使用了这一参数，则此 ABR 只向区域内发布一条缺省路由的 Summary-LSA，不生成任何其它 Summary-LSA。这种既没有 AS-external-LSA，也没有 Summary-LSA 的 Stub 区域，又称为 Totally Stub 区域。

----结束

6.5.3 配置 OSPFv3 虚连接

对于没有和骨干区域直接相连的非骨干区域，或者不连续的骨干区域来说，可以建立逻辑上的连通性。

背景信息

在划分 OSPFv3 区域之后，非骨干区域之间的 OSPFv3 路由更新是通过骨干区域来交换完成的。对此，OSPFv3 要求所有非骨干区域必须与骨干区域保持连通，并且骨干区域自身也要保持连通。但在实际应用中，可能会因为各方面条件的限制，无法满足这个要求。这时可以通过配置 OSPFv3 虚连接予以解决。

虚连接必须在两端同时配置方可生效。

请在运行 OSPFv3 协议的路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **ospfv3** [*process-id*]，进入 OSPFv3 视图。

步骤 3 执行命令 **area** *area-id*，进入 OSPFv3 区域视图。

步骤 4 执行命令 **vlink-peer** *router-id* [**hello** *hello-interval* | **retransmit** *retransmit-interval* | **trans-delay** *trans-delay-interval* | **dead** *dead-interval* | **instance** *instance-id*] *，创建并配置虚连接。

----结束

6.5.4 检查配置结果

成功配置 OSPFv3 的区域属性后，您可以查看到 LSDB、OSPFv3 路由表和虚连接信息。

前提条件

已经完成 OSPFv3 区域属性的所有配置。

操作步骤

- 使用以下命令查看 OSPFv3 的 LSDB 信息:
  - **display ospfv3** [*process-id*] **lsdb** [**area** *area-id*] [**originate-router** *advertising-router-id* | **self-originate**] [{ **router** | **network** | **inter-router** [**asbr-router** *asbr-router-id*] | { **inter-prefix** | **nssa** } [*ipv6-address prefix-length*] | **link** | **intra-prefix** | **grace** } [*link-state-id*]]
  - **display ospfv3** [*process-id*] **lsdb** [**originate-router** *advertising-router-id* | **self-originate**] **external** [*ipv6-address prefix-length*] [*link-state-id*]

- 使用以下命令来查看 OSPFv3 路由表信息：
  - **display ospfv3** [*process-id*] **routing**
  - **display ospfv3** [*process-id*] **routing** [**abr-routes** | **asbr-routes** | **statistics** | *ipv6-address prefix-length* | **intra-routes** | **inter-routes** | **ase-routes** | **nssa-routes**]
- 使用 **display ospfv3** [*process-id*] **vlink** 命令查看 OSPFv3 虚连接信息。

----结束

6.6 配置 OSPFv3 的 NSSA 区域

通过配置 NSSA 区域，允许引入外部路由，同时增加一类新的 LSA，Type7 NSSA-LSA。

6.6.1 建立配置任务

STUB 区域不能引入外部路由，为此又产生了 NSSA 区域的概念。

应用环境

NSSA 区域中允许 Type7 LSA 的传播。Type7 LSA 由 NSSA 区域的 ASBR 产生，NSSA 负责将 Type7 LSA 转换为 Type5 LSA，并通告到其他区域。

前置任务

在配置 OSPFv3 的 NSSA 区域之前，需完成以下任务：

- 配置接口的网络层地址，使相邻节点之间网络层可达。
- **配置 OSPFv3 的基本功能**。

数据准备

在配置 OSPFv3 的 NSSA 区域之前，需要准备以下数据。

序号	数据
1	发送到 NSSA 区域缺省路由的开销值

6.6.2 配置当前区域为 NSSA 区域

NSSA 区域是 Stub 区域的一种变形，该区域允许自治系统外部路由的引入，由 ASBR 发布 Type 7 LSA 通告给本区域。

背景信息

请在运行 OSPFv3 协议的路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **ospfv3** [*process-id*]，进入 OSPFv3 进程视图。

步骤 3 执行命令 **area** *area-id*，进入 OSPFv3 区域视图。

步骤 4 执行命令 **nssa** [**default-route-advertise** [**cost** *cost* | **type** *type* | **tag** *tag*] * | **no-import-route** | **no-summary** | **translator-always** | **translator-interval** *translator-interval* | **set-n-bit**] *，配置一个区域为 NSSA 区域。

----结束

后续处理

所有连接到 NSSA 区域的路由器，必须使用 **nssa** 命令将该区域配置成 NSSA 属性。

配置或取消 NSSA 属性，可能会触发区域更新。只有在上一次区域更新完成后，才能进行再次配置或取消配置操作。

6.6.3 检查配置结果

OSPFv3 的 NSSA 区域配置成功后，您可以查看到 OSPFv3 路由表信息。

前提条件

已经完成 OSPFv3 的 NSSA 区域的所有配置。

操作步骤

- 使用 **display ospfv3 area** 命令用来查看 OSPFv3 的区域信息。
- 使用以下命令来查看 OSPFv3 路由表信息。
  - **display ospfv3** [*process-id*] **routing**
  - **display ospfv3** [*process-id*] **routing** [**abr-routes** | **asbr-routes** | **statistics** | *ipv6-address prefix-length* | **intra-routes** | **inter-routes** | **ase-routes** | **nssa-routes**]

----结束

6.7 配置 OSPFv3 的路由属性

通过配置 OSPFv3 的路由属性改变 OSPFv3 的选路策略，以满足复杂网络环境中的需要。

6.7.1 建立配置任务

在配置 OSPFv3 的路由属性前了解此特性的应用环境、配置此特性的前置任务和数据准备，可以帮助您快速、准确地完成配置任务。

应用背景

在实际应用中，可以通过配置 OSPFv3 的路由属性改变 OSPFv3 的选路策略，以满足复杂网络环境中的需要。通过本节的配置过程，你可以：

- 设置 OSPFv3 接口的开销值；
- 使用多条等价路由进行负载分担。

前置任务

在配置 OSPFv3 的路由属性之前，需完成以下任务：

- 使能 IPv6 能力
- **配置 OSPFv3 基本功能**

数据准备

在配置 OSPFv3 的路由属性之前，需要准备以下数据。

序号	数据
1	链路开销
2	最大等价路由条数

6.7.2 配置 OSPFv3 接口的开销值

OSPFv3 会根据该接口的带宽自动计算其链路开销值，但也可以通过命令配置。

背景信息

用户可以在不同接口上配置 OSPFv3 的链路开销值，从而影响路由的计算。

请在运行 OSPFv3 协议的路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **interface** *interface-type interface-number*，进入接口视图。

步骤 3 执行命令 **ospfv3 cost** *cost* [**instance** *instance-id*]，设置 OSPFv3 接口的开销。

缺省情况下，OSPFv3 接口的链路开销为 1。

----结束

6.7.3 配置 OSPFv3 最大等价路由条数

当到达同一目的地存在同一路由协议发现的多条路由时，且这几条路由的开销值也相同，那么这些路由间就形成了负载分担的关系。

背景信息

请在运行 OSPFv3 协议的路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **ospfv3** [*process-id*]，进入 OSPFv3 视图。

步骤 3 执行命令 **maximum load-balancing** *number*，配置最大等价路由条数。

OSPFv3 协议支持的取值范围是 1 ～ 32，缺省值是 32。

----结束

6.7.4 检查配置结果

配置 OSPFv3 的路由属性后，您可以查看到运行 OSPFv3 的接口、LSDB 和 OSPFv3 路由表信息。

前提条件

已经完成 OSPFv3 路由属性的所有配置。

操作步骤

- 使用 **display ospfv3** [*process-id*] **interface** [**area** *area-id*] [*interface-type interface-number*]命令查看 OSPFv3 接口信息。
- 使用以下命令查看 OSPFv3 的 LSDB 信息：
  - **display ospfv3** [*process-id*] **lsdb** [**area** *area-id*] [**originate-router** *advertising-router-id* | **self-originate**] [{ **router** | **network** | **inter-router** [**asbr-router** *asbr-router-id*] | { **inter-prefix** | **nssa** } [*ipv6-address prefix-length*] | **link** | **intra-prefix** | **grace** } [*link-state-id*]]
  - **display ospfv3** [*process-id*] **lsdb** [**originate-router** *advertising-router-id* | **self-originate**] **external** [*ipv6-address prefix-length*] [*link-state-id*]
- 使用以下命令来查看 OSPFv3 路由表信息：
  - **display ospfv3** [*process-id*] **routing**
  - **display ospfv3** [*process-id*] **routing** [**abr-routes** | **asbr-routes** | **statistics** | *ipv6-address prefix-length* | **intra-routes** | **inter-routes** | **ase-routes** | **nssa-routes**]

----结束

6.8 控制 OSPFv3 的路由信息

控制 OSPFv3 路由信息包括路由聚合，对路由的过滤和引入外部路由。

6.8.1 建立配置任务

在控制 OSPFv3 的路由信息前了解此特性的应用环境、配置此特性的前置任务和数据准备，可以帮助您快速、准确地完成配置任务。

应用环境

通过本节的配置，可以控制 OSPFv3 的路由信息的发布与接收，并引入外部路由。

前置任务

在控制 OSPFv3 的路由信息之前，需完成以下任务：

- 使能 IPv6 能力

- 配置 OSPFv3 基本功能

数据准备

在控制 OSPFv3 的路由信息之前，需要准备以下数据。

序号	数据
1	聚合后的 IPv6 路由前缀
2	对路由过滤时所应用的过滤列表编号或名称
3	OSPFv3 接口的链路开销
4	可用的最大等价路由条数
5	要引入的外部路由名称、进程号、开销值

6.8.2 配置 OSPFv3 路由聚合

ABR 将具有相同前缀的路由聚合成一条 LSA 并发布到其他区域，ASBR 也可以将具有相同前缀的引入路由聚合成一条 LSA 并发布到其他区域，这样可以大幅度减少 LSDB 的规模。

背景信息

如果该区域中存在多个连续的网段，则可以使用 **abr-summary** 命令将它们聚合成一个网段，ABR 只发送一条聚合后的 LSA，所有落入本命令指定的聚合网段范围的 LSA 将不再会被单独发送出去，这样可减少其它区域中 LSDB 的规模。

当大量路由被引入时，可以使用 **asbr-summary** 命令对引入的路由进行聚合，同时，可以设置发布聚合路由的延迟时间。这样可以确保每次发布的聚合路由信息携带更多的有效路由，避免由于不正确的路由信息造成的网络振荡。

操作步骤

- 在 ABR 上配置路由聚合

  请在运行 OSPFv3 协议的 ABR 上进行以下配置。

  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **ospfv3** [*process-id*]，进入 OSPFv3 视图。
  3. 执行命令 **area** *area-id*，进入 OSPFv3 区域视图。
  4. 执行命令 **abr-summary** *ipv6-address prefix-length* [**cost** *cost* | **not-advertise**]*，配置 OSPFv3 区域路由聚合。

     cost *cost* 参数设置聚合路由的开销值。缺省情况下，所有参与聚合的路由的最大开销值为聚合路由的开销值。取值范围是 1 ～ 16777214。

     如果在命令中使用了关键字 **not-advertise**，则属于这一网段的路由信息将不会被发布出去。

- 在 ASBR 上配置路由聚合。

请在运行 OSPFv3 协议的 ASBR 上进行以下配置。

1. 执行命令 **system-view**，进入系统视图。
2. 执行命令 **ospfv3** [*process-id*]，进入 OSPFv3 视图。
3. 执行命令 **asbr-summary** *ipv6-address prefix-length* [**cost** *cost* | **tag** *tag* | **not-advertise** | **distribute-delay** *interval*] *，配置 OSPFv3 的 ASBR 路由聚合。

   cost *cost* 参数设置聚合路由的开销。缺省情况下，所有参与聚合的路由的最大开销值为聚合路由的开销值。取值范围是 1 ～ 16777214。

   tag *tag* 用来通过路由策略控制路由发布的标签。取值范围是 1 ～ 4294967295。

   如果在命令中使用了关键字 **not-advertise**，则表示不通告匹配指定 IPv6 前缀或前缀长度的聚合 IPv6 路由。

   通过 **distribute-delay** *interval* 参数设置发布聚合路由的延迟时间。

----结束

6.8.3 配置 OSPFv3 对接收的路由进行过滤

通过路由设置路由信息的过滤条件，只有通过过滤的信息才能被发布或接收。

背景信息

OSPFv3 接收到 LSA 后，可以根据一定的过滤条件来决定是否将计算后得到的路由信息加入到本地路由表中。

请在运行 OSPFv3 协议的路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **ospfv3** [*process-id*]，进入 OSPFv3 视图。

步骤 3 执行命令 **filter-policy ipv6-prefix** *ipv6-prefix-name* **import**，配置对接收的路由信息进行过滤。

filter-policy 命令只对 OSPFv3 计算出来的路由进行过滤，没有通过过滤的路由将不被加入到本地路由表中，从而不能指导转发。

----结束

6.8.4 配置 OSPFv3 引入外部路由

通过引入其他路由协议路由，可以扩充 OSPFv3 路由信息。

背景信息

由于 OSPFv3 是基于链路状态的路由协议，不能直接对发布的 LSA 进行过滤，所以只能在 OSPFv3 引入路由时进行过滤，只有符合条件的路由才能变成 LSA 发布出去。

请在运行 OSPFv3 协议的路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **ospfv3** [*process-id*]，进入 OSPFv3 视图。

步骤 3 执行命令 **default** { **cost** *cost* | **tag** *tag* | **type** *type* } *，配置引入路由的缺省开销值。

步骤 4 执行命令 **import-route** *protocol* [*process-id*] [**cost** *cost* | **type** *type* | **tag** *tag* | **route-policy** *route-policy-name*] *，引入外部路由信息。

步骤 5 （可选）执行命令 **default-route-advertise** [**always** | **cost** *cost* | **type** *type* | **tag** *tag* | **route-policy** *route-policy-name*] *，将缺省路由通告到 OSPFv3 路由区域。

步骤 6 （可选）执行命令 **filter-policy ipv6-prefix** *ipv6-prefix-name* **export** [*protocol* [*process-id*]]，对引入的外部路由信息进行过滤。

在 OSPFv3 路由器上配置 **import-route** 命令引入外部路由信息后，这台 OSPFv3 路由器就成为 ASBR。

用户可以通过指定 *protocol* 对特定的某一种路由信息进行过滤。如果没有指定 *protocol*，则 OSPFv3 将对所有引入的路由信息进行过滤。

说明

filter-policy 命令只对本机使用 **import-route** 命令引入的路由（即当本机 OSPFv3 路由器成为 ASBR 时）起作用，它在 OSPF 引入路由时对其进行过滤，被过滤掉的路由也就不会变成 LSA 被 OSPF 发布出去。如果没有配置 **import-route** 命令来引入其它外部路由（包括不同进程的 OSPFv3 路由），则 **filter-policy** 命令失效。

----结束

6.8.5 检查配置结果

控制 OSPFv3 的路由信息后，您可以查看到运行 OSPFv3 的接口、LSDB 和 OSPFv3 路由表信息。

前提条件

已经完成控制 OSPFv3 路由信息的所有配置。

操作步骤

- 使用以下命令查看 OSPFv3 的路由聚合信息：
  - **display ospfv3** [*process-id*] **abr-summary-list** [*ipv6-address prefix-length*]
  - **display ospfv3** [*process-id*] **asbr-summary** [*ipv6-address prefix-length*] [**verbose**]
- 使用以下命令查看 OSPFv3 的 LSDB 信息：
  - **display ospfv3** [*process-id*] **lsdb** [**area** *area-id*] [**originate-router** *advertising-router-id* | **self-originate**] [{ **router** | **network** | **inter-router** [**asbr-router** *asbr-router-id*] | { **inter-prefix** | **nssa** } [*ipv6-address prefix-length*] | **link** | **intra-prefix** | **grace** } [*link-state-id*]]
  - **display ospfv3** [*process-id*] **lsdb** [**originate-router** *advertising-router-id* | **self-originate**] **external** [*ipv6-address prefix-length*] [*link-state-id*]
- 使用以下命令来查看 OSPFv3 路由表信息：

- **display ospfv3** [*process-id*] **routing**
- **display ospfv3** [*process-id*] **routing** [**abr-routes** | **asbr-routes** | **statistics** | *ipv6-address prefix-length* | **intra-routes** | **inter-routes** | **ase-routes** | **nssa-routes**]

----结束

6.9 调整和优化 OSPFv3 网络

在特殊的网络环境中配置 OSPFv3 的一些特性功能，对 OSPFv3 网络的性能进行调整和优化。

6.9.1 建立配置任务

在调整和优化 OSPFv3 网络前了解此特性的应用环境、配置此特性的前置任务和数据准备，可以帮助您快速、准确地完成配置任务。

应用环境

通过改变 OSPFv3 的报文定时器，可以调整 OSPFv3 网络的收敛速度以及协议报文带来的网络负荷。在一些低速链路上，需要考虑接口传送 LSA 的延迟时间。通过调整 SPF 计算间隔时间，可以抑制由于网络频繁变化带来的资源消耗问题。

对于广播网，通过配置接口的 DR 优先级来影响 DR/BDR 的选择。

前置任务

在调整和优化 OSPFv3 网络之前，需完成以下任务：

- 使能 IPv6 能力
- **配置 OSPFv3 基本功能**

数据准备

在调整和优化 OSPFv3 网络之前，需要准备以下数据。

序号	数据
1	OSPFv3 报文定时器的值
2	SPF 定时器的值
3	接口的 DR 优先级

6.9.2 配置 SPF 定时器

通过调整 SPF 计算间隔时间，可以抑制由于网络频繁变化带来的资源消耗问题。

背景信息

当 OSPFv3 的链路状态数据库 LSDB 发生改变时，需要重新计算最短路径，如果每次改变都立即计算最短路径，将占用大量资源，并会影响路由器的效率，通过调整 *delay-*

interval 和 *hold-interval* 的值可以避免由网络频繁变化造成的带宽耗尽和路由消耗问题。**intelligent-timer** 参数用来调度 SPF 计算的时间间隔（毫秒级），从而可以加快网络收敛的速度。

请在运行 OSPFv3 协议的路由器上进行以下配置。

操作步骤

- 配置 SPF 常用定时器

  请在运行 OSPFv3 协议的路由器上进行以下配置：

  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **ospfv3** [*process-id*]，进入 OSPFv3 视图。
  3. 执行命令 **spf timers** *delay-interval hold-interval*，设置 SPF 常用定时器。

- 配置 SPF 智能定时器

  请在运行 OSPFv3 协议的路由器上进行以下配置：

  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **ospfv3** [*process-id*]，进入 OSPFv3 视图。
  3. 执行命令 **spf-schedule-interval** { *delay-interval hold-interval* | **intelligent-timer** *max-interval start-interval hold-interval-1* }，设置 SPF 智能定时器。

     说明
     SPF 常用定时器和 SPF 智能定时器是互斥的。

----结束

6.9.3 配置接收 LSA 的时间间隔

通过配置接收 LSA 的时间间隔，可以避免冗余的 LSA 更新信息。

背景信息

当网络变得不稳定的时候，可以控制接收同一条 LSA 更新信息的最小时间间隔。为避免由于网络变化造成的冗余 LSA 更新信息，缺省情况下，OSPFv3 将接收同一条 LSA 更新信息的时间间隔设置为 1000 毫秒。

请在运行 OSPFv3 协议的路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **ospfv3** [*process-id*]，进入 OSPFv3 视图。

步骤 3 执行命令 **lsa-arrival-interval** *arrival-interval*，设置接收 LSA 的时间间隔。

arrival-interval 的取值范围是 1 ～ 10000，单位是毫秒。缺省情况下，接收 LSA 的时间间隔是 1000 毫秒。

----结束

6.9.4 配置生成 LSA 的智能定时器

通过配置生成 LSA 的智能定时器，可以加快网络的收敛。

背景信息

如果把重新生成同一 LSA 实例的时间间隔设置为毫秒级，将加快网络收敛的速度。当网络变得不稳定的时候，可以通过限制生成 LSA 的智能定时器来延迟重新生成 LSA 的时间间隔。

请在运行 OSPFv3 协议的路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **ospfv3** [*process-id*]，进入 OSPFv3 视图。

步骤 3 执行命令，**lsa-originate-interval intelligent-timer** *max-interval start-interval hold-interval*，生成相同 LSA 的智能定时器的时间间隔设置完成。

- *max-interval* 用来设置 LSA 更新的最大间隔时间。取值范围是 1 ～ 10000，单位是毫秒。
- *start-interval* 用来设置 LSA 更新的初始间隔时间。取值范围是 0 ～ 1000，单位是毫秒。
- *hold-interval* 用来设置 LSA 更新的抑制时间间隔。取值范围是 1 ～ 5000，单位是毫秒。

缺省情况下，设置 LSA 更新的最大间隔时间为 5000 毫秒，初始间隔时间为 500 毫秒，抑止时间间隔为 1000 毫秒。

----结束

6.9.5 抑制接口接收和发送 OSPFv3 报文

通过抑制使能了 OSPFv3 的接口接收和发送 OSPFv3 报文，OSPFv3 路由信息能够不被某一网络中的路由器获得且不接收其他路由器的路由信息。

背景信息

如果要使 OSPFv3 路由信息不被某一网络中的路由器获得且不接收其他路由器的路由信息，可以抑制使能了 OSPFv3 的接口接收和发送 OSPFv3 报文。

请在运行 OSPFv3 协议的路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **ospfv3** [*process-id*]，进入 OSPFv3 视图。

步骤 3 执行命令 **silent-interface** *interface-type interface-number*，抑制接口收发 OSPFv3 报文。

----结束

后续处理

不同的进程可以对同一接口抑制接收/发送 OSPFv3 报文，但 **silent-interface** 命令只对本进程已经使能的 OSPFv3 接口起作用，不对其它进程的接口起作用。

当运行 OSPFv3 协议的接口被配置为 Silent 状态后，该接口的直连路由仍可以由同一路由器的 Intra-Area-Prefix-LSA 发布，但接口上不会建立 OSPFv3 邻居关系。这一特性可以增强 OSPFv3 的组网适应能力。

6.9.6 配置接口的 DR 优先级

当网络类型为广播网或 NBMA 类型时，可以通过配置接口的 DR 优先级来影响网络中 DR/BDR 的选择。

背景信息

路由器接口的 DR 优先级将影响接口在选举 DR 时所具有的资格，优先级为 0 的路由器不会被选举为 DR 或 BDR。

请在运行 OSPFv3 协议的路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **interface** *interface-type interface-number*，进入接口视图。

步骤 3 执行命令 **ospfv3 dr-priority** *priority* [**instance** *instance-id*]，设置接口在选举 DR 时的优先级。

----结束

后续处理

改变优先级后，可以利用下面两种方法重新进行 DR/BDR 的选择，但是这会导致路由器之间的 OSPFv3 邻接关系中断，一般情况下不推荐使用。

- 重启所有路由器。
- 在建立了 OSPFv3 邻居的接口上配置 **shutdown**/**undo shutdown** 命令。

6.9.7 配置 Stub 路由器

Stub 路由器用来控制流量，它告知其他 OSPFv3 路由器不要使用这个 Stub 路由器来转发数据，但可以拥有一个到 Stub 路由器的路由。

背景信息

Stub 路由器用来控制流量，它告知其他 OSPFv3 路由器不要使用这个 Stub 路由器来转发数据，但可以拥有一个到 Stub 路由器的路由。

请在运行 OSPFv3 协议的路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **ospfv3** [*process-id*]，进入 OSPFv3 进程视图。

步骤 3 执行命令 **stub-router** [**on-startup** [*interval*]]，配置 Stub 路由器。

说明

通过此命令配置的 Stub 路由器与 Stub 区域里的路由器没有必然联系。

----结束

6.9.8 忽略 DD 报文中的 MTU 检查

通过忽略对 DD 报文中 MTU 字段进行检查，OSPFv3 路由器可以接收 MTU 值为 0 的报文。

背景信息

请在运行 OSPFv3 协议的路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **interface** *interface-type interface-number*，进入接口视图。

步骤 3 执行命令 **ospfv3 mtu-ignore** [**instance** *instance-id*]，忽略 DD 报文中的 MTU 检查。

配置此命令后，接口对接收的 DD 报文中的 MTU 字段不再进行检查。

----结束

6.9.9 检查配置结果

调整和优化 OSPFv3 网络后，您可以查看到运行 OSPFv3 的接口、LSDB 和 OSPFv3 路由表信息。

前提条件

已经完成调整和优化 OSPFv3 网络的所有配置。

操作步骤

- 使用 **display ospfv3**[*process-id*] **interface** [**area** *area-id*] [*interface-type interface-number*]命令查看 OSPFv3 接口信息。
- 使用以下命令查看 OSPFv3 的 LSDB 信息：
  - **display ospfv3** [*process-id*] **lsdb** [**area** *area-id*] [**originate-router** *advertising-router-id* | **self-originate**] [{ **router** | **network** | **inter-router** [**asbr-router** *asbr-router-id*] | { **inter-prefix** | **nssa** } [*ipv6-address prefix-length*] | **link** | **intra-prefix** | **grace** } [*link-state-id*]]
  - **display ospfv3** [*process-id*] **lsdb** [**originate-router** *advertising-router-id* | **self-originate**] **external** [*ipv6-address prefix-length*] [*link-state-id*]

- 使用以下命令来查看 OSPFv3 路由表信息：
  - **display ospfv3** [*process-id*] **routing**
  - **display ospfv3** [*process-id*] **routing** [**abr-routes** | **asbr-routes** | **statistics** | *ipv6-address prefix-length* | **intra-routes** | **inter-routes** | **ase-routes** | **nssa-routes**]

----结束

6.10 配置 OSPFv3 GR

通过 OSPFv3 GR 解决 OSPFv3 路由器重启后造成路由计算不准确、报文丢失的问题。

6.10.1 建立配置任务

缺省情况下，OSPFv3 协议的 GR 和 Helper 能力都被禁止。

应用环境

针对 OSPFv3 协议，为了避免协议重启带来的路由震荡和流量转发中断，可以使能 OSPFv3 协议的 GR 特性。

协议重启后，GR Restarter 和邻接 GR Helper 之间继续保持邻居关系，交换路由信息并同步数据库，更新路由表和转发表，从而实现 OSPFv3 快速收敛。

说明

AR150/200 只能作为 Helper 路由器，不能作为 Restarter 路由器。

前置任务

在配置 OSPFv3 GR 之前，需完成以下任务：

- **配置 OSPFv3 基本功能**

数据准备

在配置 OSPFv3 GR 之前，需准备以下数据。

序号	数据
1	OSPFv3 进程号
2	OSPFv3 邻居的 Helper 能力过滤规则

6.10.2 使能 OSPFv3 协议的 GR 能力

OSPF 通过 GR 重启后，Restarter 路由器只和 Helper 路由器之间重新建立邻居关系，交换路由信息并同步数据库，更新路由表和转发表，从而实现 OSPF 快速收敛，保持网络拓扑稳定。

背景信息

请在运行 OSPFv3 协议的路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **ospfv3** *process-id*，进入 OSPFv3 视图。

步骤 3 执行命令 **graceful-restart** [**period** *period* | **ack-time** *time* | **retransmit-interval** *interval* | **lsa-checking-ignore** | **planned-only**] *，使能 OSPFv3 协议的 GR 能力。

缺省情况下，OSPFv3 协议的 GR 能力被禁止。

ack-time 是个可选参数，可以使得 Restarter 端在 *ack-time* 时间间隔内发现更多的邻居。

----结束

6.10.3 使能 OSPFv3 GR 的 Helper 能力

GR Helper 是 Restarter 的邻居，能够识别 GR 信令，在 GR Restarter 进行主备倒换时保持和 GR Restarter 的邻接关系不变，协助 GR Restarter 进行网络拓扑关系的恢复。

背景信息

请在运行 OSPFv3 协议的路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **ospfv3** *process-id*，进入 OSPFv3 视图。

步骤 3 执行命令 **helper-role** [{ **ip-prefix** *ip-prefix-name* | **acl-number** *acl-number* | **acl-name** *acl-name* } | **max-grace-period** *period* | **planned-only** | **lsa-checking-ignore**] *，使能 OSPFv3 协议的 GR 能力。

缺省情况下，OSPFv3 GR 的 Helper 能力被禁止。

----结束

6.10.4 检查配置结果

配置 OSPFv3 GR 后，您可以查看到 GR 信息。

前提条件

已经完成 OSPFv3 GR 的所有配置。

操作步骤

- 使用 **display ospfv3** [*process-id*] **graceful-restart-information** 命令查看 OSPFv3 GR 的状态。

----结束

6.11 配置 OSPFv3 网管功能

OSPFv3 同时支持网管功能，可以配置 OSPFv3 MIB 与某一进程绑定。

6.11.1 建立配置任务

在配置 OSPFv3 网管功能前了解此特性的应用环境、配置此特性的前置任务和数据准备，可以帮助您快速、准确地完成配置任务。

应用环境

OSPFv3 同时支持网管功能，可以配置 OSPFv3 MIB 与某一进程绑定。

前置任务

在配置 OSPFv3 的网管功能之前，需要完成以下任务：

- 配置接口的网络层地址，使相邻节点网络层可达。
- **配置 OSPFv3 基本功能**。

数据准备

无

6.11.2 配置 OSPFv3 MIB 绑定

MIB 是一个虚拟的数据库，是在被管理设备端维护的设备状态信息集。

背景信息

当启动了多个 OSPFv3 进程时，可以配置 OSPFv3 MIB 对哪个进程进行处理，即绑定在哪个进程。

请在运行 OSPFv3 协议的路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **ospfv3 mib-binding** *process-id*，配置 OSPFv3 MIB 绑定。

----结束

6.11.3 配置 OSPFv3 TRAP 功能

OSPFv3 MIB 是一个虚拟的数据库，是在被管理设备端维护的设备状态信息集。

背景信息

请在运行 OSPFv3 协议的路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **snmp-agent trap enable feature-name ospfv3** [**trap-name** { **ifconfigerror** | **ifrxbadpacket** | **ifstatechange** | **nbrrestarthelperstatuschange** | **nbrstatechange** | **nssatranslatorstatuschange** | **restartstatuschange** | **virtifconfigerror** | **virtifrxbadpacket** | **virtifstatechange** | **virtnbrrestarthelperstatuschange** | **virtnbrstatechange** }]，打开 OSPFv3 模块的告警开关。

----结束

6.11.4 检查配置结果

通过配置 OSPFv3 的网管功能，您可以查看到信息通道的内容、息中心记录的各项信息、日志缓冲区记录和告警缓冲区记录的信息。

前提条件

已经完成 OSPFv3 网管功能的所有配置。

操作步骤

- 使用 **display current-configuration** 命令查看路由器当前生效的配置参数。

----结束

6.12 维护 OSPFv3

维护 OSPFv3 是指复位 OSPFv3 和调试 OSPFv3。

6.12.1 复位 OSPFv3

通过重启 OSPFv3，达到复位的目的。可以选择以 GR 的方式复位 OSPFv3。

背景信息

复位 OSPFv3 连接（执行 **reset ospfv3** 命令）会导致路由器之间的 OSPFv3 邻接关系中断。务必仔细确认是否必须执行复位 OSPFv3 连接的操作。

当 OSPFv3 路由策略或协议发生变化后，需要通过复位 OSPFv3 连接使新的配置生效。如果需要复位 OSPFv3 连接，可在用户视图下选择执行以下命令。

操作步骤

- 通过以下命令，重启 OSPFv3 进程。
  - **reset ospfv3** { *process-id* | **all** } [**graceful-restart** [**extend-period** *period*]]

- **reset ospfv3** { *process-id* | **all** } **counters** [**neighbor** [*interface-type interface-number*] [*router-id*]]

----结束

6.13 配置举例

介绍 OSPFv3 配置举例。请结合配置流程图了解配置过程。配置示例中包括组网需求、配置注意事项、配置思路等。

6.13.1 配置 OSPFv3 区域示例

介绍 OSPFv3 基本功能的配置过程，包括在各路由器上使能 OSPFv3、指定不同区域内的网段。

组网需求

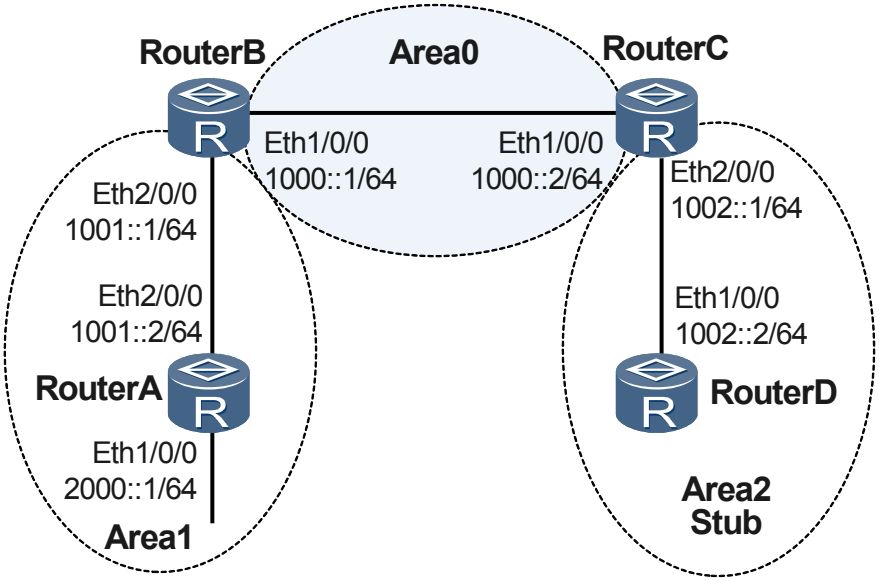
如图 6-1 所示，所有的路由器都运行 OSPFv3，整个自治系统划分为 3 个区域。其中 RouterB 和 RouterC 作为 ABR 来转发区域之间的路由。

要求将 Area2 配置为 Stub 区域，减少通告到此区域内的 LSA 数量，但不影响路由的可达性。

说明

AR150/200 仅可作为 RouterD。

图 6-1 配置 OSPFv3 区域组网图



配置思路

采用如下的思路配置 OSPFv3 区域：

1. 在各路由器上使能 OSPFv3 的基本功能。
2. 配置 Area2 为 Stub 区域（需要在 Area2 内所有的路由器上配置 **stub** 命令），查看 RouterD 的 OSPFv3 路由表信息。
3. 配置 Area2 为 Totally Stub 区域，查看 RouterD 的 OSPFv3 路由表信息。

数据准备

为完成此配置例，需准备如下的数据：

- RouterA 的 router id 1.1.1.1，所在的区域号 Area1。
- RouterB 的 router id 2.2.2.2，所在的区域号 Area0 和 Area1。
- RouterC 的 router id 3.3.3.3，所在的区域号 Area0 和 Area2。
- RouterD 的 router id 4.4.4.4，所在的区域号 Area2。

操作步骤

步骤 1 配置各接口的 IPv6 地址（略）

步骤 2 配置 OSPFv3 基本功能

配置 RouterA。

```
[RouterA] ipv6
[RouterA] ospfv3
[RouterA-ospfv3-1] router-id 1.1.1.1
[RouterA-ospfv3-1] quit
[RouterA] interface ethernet 1/0/0
[RouterA-Ethernet1/0/0] ospfv3 1 area 1
[RouterA-Ethernet1/0/0] quit
[RouterA] interface ethernet 2/0/0
[RouterA-Ethernet2/0/0] ospfv3 1 area 1
[RouterA-Ethernet2/0/0] quit
```

配置 RouterB。

```
[RouterB] ipv6
[RouterB] ospfv3
[RouterB-ospfv3-1] router-id 2.2.2.2
[RouterB-ospfv3-1] quit
[RouterB] interface ethernet 1/0/0
[RouterB-Ethernet1/0/0] ospfv3 1 area 0
[RouterB-Ethernet1/0/0] quit
[RouterB] interface ethernet 2/0/0
[RouterB-Ethernet2/0/0] ospfv3 1 area 1
[RouterB-Ethernet2/0/0] quit
```

配置 RouterC。

```
[RouterC] ipv6
[RouterC] ospfv3
[RouterC-ospfv3-1] router-id 3.3.3.3
[RouterC-ospfv3-1] quit
[RouterC] interface ethernet 1/0/0
[RouterC-Ethernet1/0/0] ospfv3 1 area 0
[RouterC-Ethernet1/0/0] quit
[RouterC] interface ethernet 2/0/0
[RouterC-Ethernet2/0/0] ospfv3 1 area 2
[RouterC-Ethernet2/0/0] quit
```

配置 RouterD。

```
[RouterD] ipv6
```

```
[RouterD] ospfv3
[RouterD-ospfv3-1] router-id 4.4.4.4
[RouterD-ospfv3-1] quit
[RouterD] interface ethernet 1/0/0
[RouterD-Ethernet1/0/0] ospfv3 1 area 2
[RouterD-Ethernet1/0/0] quit
```

查看 RouterB 的 OSPFv3 邻居状态。

```
[RouterB] display ospfv3 peer
OSPFv3 Process (1)
OSPFv3 Area (0.0.0.1)
Neighbor ID     Pri   State           Dead Time   Interface  Instance ID
1.1.1.1           1   Full/ -         00:00:34    Ethernet2/0/0          0
OSPFv3 Area (0.0.0.0)
Neighbor ID     Pri   State           Dead Time   Interface  Instance ID
3.3.3.3           1   Full/ -         00:00:32    Ethernet1/0/0          0
```

查看 RouterC 的 OSPFv3 邻居状态。

```
[RouterC] display ospfv3 peer
OSPFv3 Process (1)
OSPFv3 Area (0.0.0.0)
Neighbor ID     Pri   State           Dead Time   Interface  Instance ID
2.2.2.2           1   Full/ -         00:00:37    Ethernet1/0/0          0
OSPFv3 Area (0.0.0.2)
Neighbor ID     Pri   State           Dead Time   Interface  Instance ID
4.4.4.4           1   Full/ -         00:00:33    Ethernet2/0/0          0
```

查看 RouterD 的 OSPFv3 路由表信息。

```
[RouterD] display ospfv3 routing
Codes : E2 - Type 2 External, E1 - Type 1 External, IA - Inter-Area,
N - NSSA, U - Uninstalled
OSPFv3 Process (1)
   Destination                                  Metric
     Next-hop
  IA 1000::/64                                   2
          via FE80::1572:0:5EF4:1, Ethernet1/0/0
  IA 1001::/64                                   3
          via FE80::1572:0:5EF4:1, Ethernet1/0/0
     1002::/64                                   1
          directly-connected, Ethernet1/0/0
  IA 2000::/64                                   4
          via FE80::1572:0:5EF4:1, Ethernet1/0/0
```

步骤 3 配置 Stub 区域

配置 RouterD 的 Stub 区域。

```
[RouterD] ospfv3
[RouterD-ospfv3-1] area 2
[RouterD-ospfv3-1-area-0.0.0.2] stub
[RouterD-ospfv3-1-area-0.0.0.2] quit
```

配置 RouterC 的 Stub 区域，设置发送到 Stub 区域的缺省路由的开销为 10。

```
[RouterC] ospfv3
[RouterC-ospfv3-1] area 2
[RouterC-ospfv3-1-area-0.0.0.2] stub
[RouterC-ospfv3-1-area-0.0.0.2] default-cost 10
[RouterC-ospfv3-1-area-0.0.0.2] quit
```

查看 RouterD 的 OSPFv3 路由表信息，可以看到路由表中多了一条缺省路由，它的开销值为直连路由的开销和所配置的开销值之和。

```
[RouterD] display ospfv3 routing
Codes : E2 - Type 2 External, E1 - Type 1 External, IA - Inter-Area,
N - NSSA, U - Uninstalled
```

```
OSPFv3 Process (1)
OSPFv3 Process (1)
   Destination                                  Metric
     Next-hop
  IA ::/0                                       11
          via FE80::1572:0:5EF4:1, Ethernet1/0/0
  IA 1000::/64                                  2
          via FE80::1572:0:5EF4:1, Ethernet1/0/0
  IA 1001::/64                                  3
          via FE80::1572:0:5EF4:1, Ethernet1/0/0
     1002::/64                                  1
          directly-connected, Ethernet1/0/0
  IA 2000::/64                                  4
          via FE80::1572:0:5EF4:1, Ethernet1/0/0
```

步骤 4 配置 Totally Stub 区域

配置 RouterC，设置 Area2 为 Totally Stub 区域。

```
[RouterC] ospfv3
[RouterC-ospfv3-1] area 2
[RouterC-ospfv3-1-area-0.0.0.2] stub no-summary
[RouterC-ospfv3-1-area-0.0.0.2] quit
```

步骤 5 验证配置结果

查看 RouterD 的 OSPFv3 路由表，可以发现路由表项数目减少了，其他非直连路由都被抑制，只有缺省路由被保留。

```
[RouterD] display ospfv3 routing
Codes : E2 - Type 2 External, E1 - Type 1 External, IA - Inter-Area,
N - NSSA, U - Uninstalled
OSPFv3 Process (1)
OSPFv3 Process (1)
   Destination                                  Metric
     Next-hop
  IA ::/0                                       11
          via FE80::1572:0:5EF4:1, Ethernet1/0/0
     1002::/64                                  1
          directly-connected, Ethernet1/0/0
```

----结束

配置文件

- RouterA 的配置文件

```
#
 sysname RouterA
#
 ipv6
#
interface Ethernet1/0/0
 ipv6 enable
 ipv6 address 2000::1/64
 ospfv3 1 area 0.0.0.1
#
interface Ethernet2/0/0
 ipv6 enable
 ipv6 address 1001::2/64
 ospfv3 1 area 0.0.0.1
#
ospfv3 1
 router-id 1.1.1.1
 area 0.0.0.1
#
return
```

- RouterB 的配置文件

```
#
 sysname RouterB
#
 ipv6
#
interface Ethernet1/0/0
 ipv6 enable
 ipv6 address 1000::1/64
 ospfv3 1 area 0.0.0.0
#
interface Ethernet2/0/0
 ipv6 enable
 ipv6 address 1001::1/64
 ospfv3 1 area 0.0.0.1
#
ospfv3 1
 router-id 2.2.2.2
 area 0.0.0.0
 area 0.0.0.1
#
return
```

- RouterC 的配置文件

```
#
 sysname RouterC
#
 ipv6
#
interface Ethernet1/0/0
 ipv6 enable
 ipv6 address 1000::2/64
 ospfv3 1 area 0.0.0.0
#
interface Ethernet2/0/0
 ipv6 enable
 ipv6 address 1002::1/64
 ospfv3 1 area 0.0.0.2
#
ospfv3 1
 router-id 3.3.3.3
 area 0.0.0.0
 area 0.0.0.2
  stub no-summary
  default-cost 10
#
return
```

- RouterD 的配置文件

```
#
 sysname RouterD
#
 ipv6
#
interface Ethernet1/0/0
 ipv6 enable
 ipv6 address 1002::2/64
 ospfv3 1 area 0.0.0.2
#
ospfv3 1
 router-id 4.4.4.4
 area 0.0.0.2
  stub
#
return
```

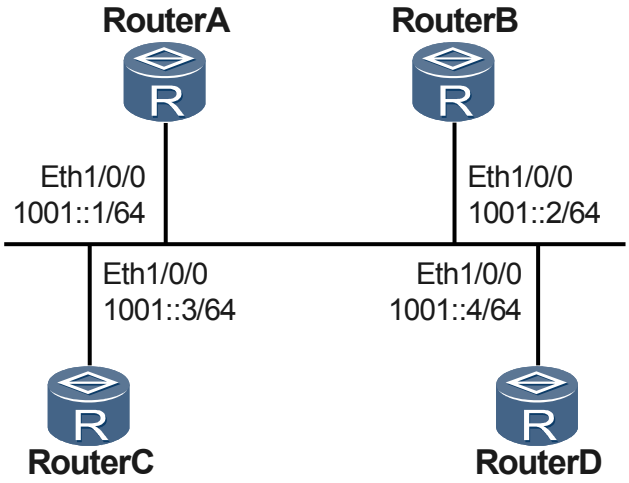
6.13.2 配置 OSPFv3 的 DR 选择示例

介绍在广播网络中，配置接口上的 DR 优先级进行 DR 选择的过程。

组网需求

在图 6-2 中，RouterA 的优先级为 100，它是网络上的最高优先级，所以 RouterA 被选为 DR；RouterC 是优先级第二高的，被选为 BDR；RouterB 的优先级为 0，这意味着它将无法成为 DR；RouterD 没有配置优先级，取缺省值 1。

图 6-2 配置 OSPFv3 的 DR 选择组网图



配置思路

采用如下的思路配置 OSPFv3 的 DR 选择：

1. 配置各路由器上 router id，使能 OSPFv3，指定网段。
2. 缺省优先级情况下，查看各路由器 DR/BDR 状态。
3. 配置接口上的 DR 优先级，查看 DR/BDR 状态。

数据准备

为完成此配置例，需准备如下的数据：

- RouterA 的 router id 1.1.1.1，DR 优先级 100。
- RouterB 的 router id 2.2.2.2，DR 优先级 0。
- RouterC 的 router id 3.3.3.3，DR 优先级 2。
- RouterD 的 router id 4.4.4.4，DR 优先级取缺省值 1。

操作步骤

步骤 1 配置各接口的 IPv6 地址（略）

步骤 2 配置 OSPFv3 基本功能

配置 RouterA，启动 OSPFv3，并设置其 Router ID 为 1.1.1.1。

```
[RouterA] ipv6
[RouterA] ospfv3
[RouterA-ospfv3-1] router-id 1.1.1.1
[RouterA-ospfv3-1] quit
[RouterA] interface ethernet 1/0/0
[RouterA-Ethernet1/0/0] ospfv3 1 area 0
[RouterA-Ethernet1/0/0] quit
```

配置 RouterB，启动 OSPFv3，并设置其 Router ID 为 2.2.2.2。

```
[RouterB] ipv6
[RouterB] ospfv3
[RouterB-ospfv3-1] router-id 2.2.2.2
[RouterB-ospfv3-1] quit
[RouterB] interface ethernet 1/0/0
[RouterB-Ethernet1/0/0] ospfv3 1 area 0
[RouterB-Ethernet1/0/0] quit
```

配置 RouterC，启动 OSPFv3，并设置其 Router ID 为 3.3.3.3。

```
[RouterC] ipv6
[RouterC] ospfv3
[RouterC-ospfv3-1] router-id 3.3.3.3
[RouterC-ospfv3-1] quit
[RouterC] interface ethernet 1/0/0
[RouterC-Ethernet1/0/0] ospfv3 1 area 0
[RouterC-Ethernet1/0/0] quit
```

配置 RouterD，启动 OSPFv3，并设置其 Router ID 为 4.4.4.4。

```
[RouterD] ipv6
[RouterD] ospfv3
[RouterD-ospfv3-1] router-id 4.4.4.4
[RouterD-ospfv3-1] quit
[RouterD] interface ethernet 1/0/0
[RouterD-Ethernet1/0/0] ospfv3 1 area 0
[RouterD-Ethernet1/0/0] quit
```

查看 RouterA 的邻居信息，可以看到 DR 优先级（缺省为 1）以及邻居状态，此时 RouterD 为 DR，RouterC 为 BDR。

说明

当优先级相同时，router-id 高的为 DR。如果路由器的某个 Ethernet 接口成为 DR 之后，则这台路由器的其他广播接口在进行后续的 DR 选择时，具有高优先权。即选择已经是 DR 的路由器作为 DR，DR 不可抢占。

```
[RouterA] display ospfv3 peer
OSPFv3 Process (1)
OSPFv3 Area (0.0.0.0)
Neighbor ID     Pri   State           Dead Time   Interface  Instance ID
2.2.2.2           1   2-Way/DROther   00:00:32    Ethernet1/0/0         0
3.3.3.3           1   Full/Backup     00:00:36    Ethernet1/0/0         0
4.4.4.4           1   Full/DR         00:00:38    Ethernet1/0/0         0
```

查看 RouterD 的邻居信息，可以看到 RouterD 和其他邻居之间的邻居状态都为 Full。

```
[RouterD] display ospfv3 peer
OSPFv3 Process (1)
OSPFv3 Area (0.0.0.0)
Neighbor ID     Pri   State           Dead Time   Interface  Instance ID
1.1.1.1           1   Full/DROther    00:00:32    Ethernet1/0/0         0
2.2.2.2           1   Full/DROther    00:00:35    Ethernet1/0/0         0
3.3.3.3           1   Full/Backup     00:00:30    Ethernet1/0/0         0
```

步骤 3 配置接口的 DR 优先级

配置 RouterA 的 DR 优先级为 100。

```
[RouterA] interface ethernet 1/0/0
[RouterA-Ethernet1/0/0] ospfv3 dr-priority 100
[RouterA-Ethernet1/0/0] quit
```

配置 RouterB 的 DR 优先级为 0。

```
[RouterB] interface ethernet 1/0/0
[RouterB-Ethernet1/0/0] ospfv3 dr-priority 0
[RouterB-Ethernet1/0/0] quit
```

配置 RouterC 的 DR 优先级为 2。

```
[RouterC] interface ethernet 1/0/0
[RouterC-Ethernet1/0/0] ospfv3 dr-priority 2
[RouterC-Ethernet1/0/0] quit
```

显示 RouterA 的邻居信息，可以看到 DR 优先级已经更新，但 DR/BDR 并未改变。

```
[RouterA] display ospfv3 peer
OSPFv3 Process (1)
OSPFv3 Area (0.0.0.0)
Neighbor ID     Pri   State           Dead Time   Interface  Instance ID
2.2.2.2           0   2-Way/DROther   00:00:34    Ethernet1/0/0         0
3.3.3.3           2   Full/Backup     00:00:38    Ethernet1/0/0         0
4.4.4.4           1   Full/DR         00:00:31    Ethernet1/0/0         0
```

显示 RouterD 的邻居信息，可以看到 RouterD 仍然为 DR。

```
[RouterD] display ospfv3 peer
OSPFv3 Process (1)
OSPFv3 Area (0.0.0.0)
Neighbor ID     Pri   State           Dead Time   Interface  Instance ID
1.1.1.1         100   Full/DROther    00:00:36    Ethernet1/0/0         0
2.2.2.2           0   Full/DROther    00:00:30    Ethernet1/0/0         0
3.3.3.3           2   Full/Backup     00:00:36    Ethernet1/0/0         0
```

步骤 4 重新进行 DR/BDR 选择

重启所有路由器（或者在建立了 OSPFv3 邻居的接口上配置 **shutdown** 或 **undo shutdown** 命令），使 OSPFv3 重新进行 DR/BDR 的选择。

步骤 5 验证配置结果

查看 RouterA 的邻居信息，可以看到 RouterC 为 BDR。

```
[RouterA] display ospfv3 peer
OSPFv3 Process (1)
OSPFv3 Area (0.0.0.0)
Neighbor ID     Pri   State           Dead Time   Interface  Instance ID
2.2.2.2           0   Full/DROther    00:00:31    Ethernet1/0/0         0
3.3.3.3           2   Full/Backup     00:00:36    Ethernet1/0/0         0
4.4.4.4           1   Full/DROther    00:00:39    Ethernet1/0/0         0
```

查看 RouterD 的邻居信息，可以看到 RouterA 为 DR。

```
[RouterD] display ospfv3 peer
OSPFv3 Process (1)
OSPFv3 Area (0.0.0.0)
Neighbor ID     Pri   State           Dead Time   Interface  Instance ID
1.1.1.1         100   Full/DR         00:00:39    Ethernet1/0/0         0
2.2.2.2           0   2-Way/DROther   00:00:35    Ethernet1/0/0         0
3.3.3.3           2   Full/Backup     00:00:39    Ethernet1/0/0         0
```

----结束

配置文件

- RouterA 的配置文件

```
#
 sysname RouterA
#
 ipv6
#
interface Ethernet1/0/0
 ipv6 enable
 ipv6 address 1001::1/64
 ospfv3 1 area 0.0.0.0
 ospfv3 dr-priority 100
#
ospfv3 1
 router-id 1.1.1.1
#
return
```

- RouterB 的配置文件

```
#
 sysname RouterB
#
 ipv6
#
interface Ethernet1/0/0
 ipv6 enable
 ipv6 address 1001::2/64
 ospfv3 1 area 0.0.0.0
 ospfv3 dr-priority 0
#
ospfv3 1
 router-id 2.2.2.2
#
return
```

- RouterC 的配置文件

```
#
 sysname RouterC
#
 ipv6
#
interface Ethernet1/0/0
 ipv6 address 1001::3/64
 ospfv3 1 area 0.0.0.0
 ospfv3 dr-priority 2
#
ospfv3 1
 router-id 3.3.3.3
#
return
```

- RouterD 的配置文件

```
#
 sysname RouterD
#
 ipv6
#
interface Ethernet1/0/0
 ipv6 enable
 ipv6 address 1001::4/64
 ospfv3 1 area 0.0.0.0
#
ospfv3 1
 router-id 4.4.4.4
#
return
```

7 IS-IS 配置

关于本章

介绍 IS-IS 协议的基本原理、配置过程和配置举例。

7.1 IS-IS 基本概念
IS-IS 属于内部网关协议，用于自治系统内部。IS-IS 是一种链路状态协议，使用最短路径优先算法进行路由计算。

7.2 AR150/200 支持的 IS-IS 特性
AR150/200 支持的 IS-IS 特性包括：多实例和多进程、热备份、多拓扑、本地 MT、GR、TE 与 DS-TE、管理标记、LSP 分片扩展、动态主机名交换、快速收敛、BFD 和三次握手机制。

7.3 配置 IS-IS 的基本功能(IPv4)
配置 IS-IS 的基本功能(IPv4)可以实现基于 IPv4 地址族的 IS-IS 网络中各节点的互通。配置步骤主要包括配置 IS-IS 进程和配置 IS-IS 接口。

7.4 建立或维持 IS-IS 邻居或邻接关系
针对影响 IS-IS 邻居关系保持的参数配置进行介绍。

7.5 调整 IS-IS 的选路(IPv4)
通过调整 IS-IS 选路，可以实现对路由选择的精确控制。

7.6 配置 IS-IS 路由聚合(IPv4)
当大规模部署 IS-IS 网络时，为了避免 IS-IS 路由表中条目过多而降低路由查找速度的现象以及降低管理的复杂度，可以配置路由聚合，减小路由表的规模。

7.7 配置 IS-IS 与其他路由协议交互(IPv4)
在网络中同时部署了 IS-IS 和其他路由协议时，需要配置 IS-IS 与其他路由协议的路由交互，才能使不同协议的网络正常通信。

7.8 调整 IS-IS 路由的收敛速度(IPv4)
提高对 IS-IS 网络中故障的响应速度，加快出现网络故障时的路由收敛速度，可以提高 IS-IS 网络的可靠性。

7.9 配置静态 IPv4 BFD for IS-IS
BFD 能够提供轻负荷、快速（毫秒级）的通道故障检测，配置静态 IPv4 BFD for IS-IS 是实现 BFD 检测功能的一种方式。

7.10 配置动态 IPv4 BFD for IS-IS

如果对数据传输有较高要求，需要提高链路状态变化时 IS-IS 的收敛速度，可以在运行 IS-IS 的链路上配置动态 IPv4 BFD。

7.11 配置 IS-IS 的基本功能(IPv6)
配置 IS-IS 的基本功能(IPv6)可以实现基于 IPv6 地址族的 IS-IS 网络中各节点的互通。配置步骤主要包括配置 IS-IS 进程和配置 IS-IS 接口。

7.12 调整 IS-IS 的选路(IPv6)
通过调整 IS-IS 选路，可以实现对路由选择的精确控制。

7.13 配置 IS-IS 路由聚合(IPv6)
当大规模部署 IS-IS 网络时，为了避免 IS-IS 路由表中条目过多而降低路由查找速度的现象以及降低管理的复杂度，可以配置路由聚合，减小路由表的规模。

7.14 配置 IS-IS 与其他路由协议交互(IPv6)
在网络中同时部署了 IS-IS 和其他路由协议时，需要配置 IS-IS 与其他路由协议的路由交互，才能使不同协议的网络正常通信。

7.15 调整 IS-IS 路由的收敛速度(IPv6)
提高对 IS-IS 网络中故障的响应速度，加快出现网络故障时的路由收敛速度，可以提高 IS-IS 网络的可靠性。

7.16 配置 IS-IS GR
通过配置 IS-IS GR，可以使路由器平滑重启，避免出现暂时的“黑洞”。

7.17 维护 IS-IS 配置
维护 IS-IS，包括复位和清除 IS-IS。

7.18 配置举例
介绍 IS-IS 配置举例。请结合配置流程图了解配置过程。配置示例中包括组网需求、配置注意事项、配置思路等。

7.1 IS-IS 基本概念

IS-IS 属于内部网关协议，用于自治系统内部。IS-IS 是一种链路状态协议，使用最短路径优先算法进行路由计算。

IS-IS 最初是国际标准化组织 ISO（the International Organization for Standardization）为它的无连接网络协议 CLNP（ConnectionLess Network Protocol）设计的一种动态路由协议。

为了提供对 IP 的路由支持，IETF 在 RFC1195 中对 IS-IS 进行了扩充和修改，使它能够同时应用在 TCP/IP 和 OSI 环境中，称为集成化 IS-IS（Integrated IS-IS 或 Dual IS-IS）。

IS-IS 属于内部网关协议 IGP（Interior Gateway Protocol），用于自治系统内部。IS-IS 是一种链路状态协议，使用最短路径优先 SPF（Shortest Path First）算法进行路由计算，与 OSPF 协议有很多相似之处。

IS-IS 区域

为了支持大规模的路由网络，IS-IS 在路由域内采用两级的分层结构。一个大的路由域被分成一个或多个区域（Areas）。区域内的路由通过 Level-1 路由器管理，区域间的路由通过 Level-2 路由器管理。

图 7-1 所示为一个运行 IS-IS 协议的网络，它与 OSPF 的多区域网络拓扑结构非常相似。其中 Area1 是骨干区域，该区域中的所有设备均是 Level-2 路由器。另外 4 个区域为非骨干区域，它们都通过 Level-1-2 路由器与骨干路由器相连。

图 7-1 IS-IS 拓扑结构图之一

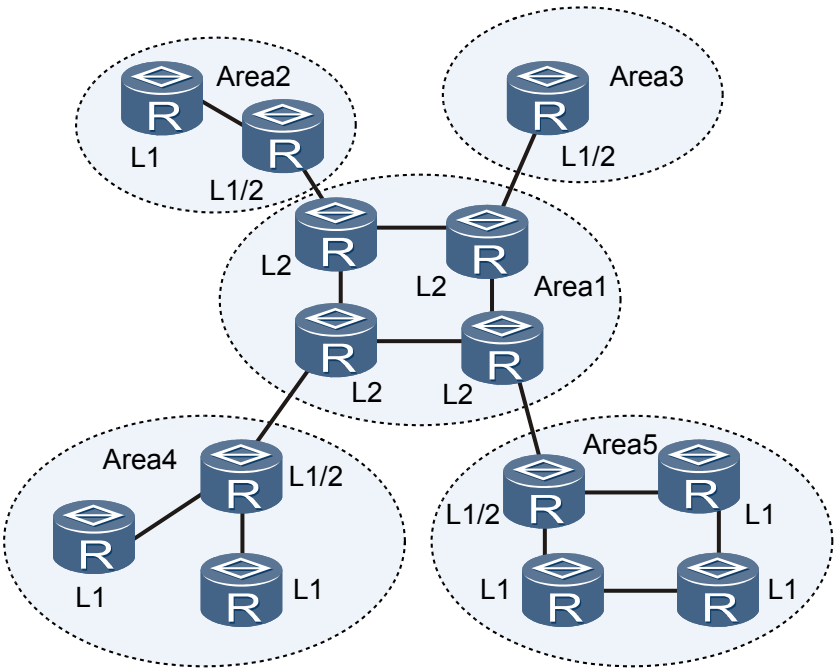
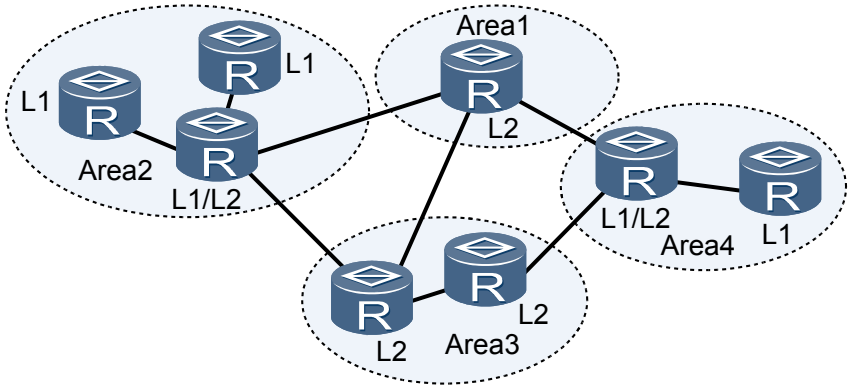


图 7-2 是 IS-IS 的另外一种拓扑结构图。其中 Level-1-2 路由器不只用来连接 Level-1 和 Level-2 路由器，而且还与其他 Level-2 路由器一起构成了 IS-IS 的骨干网。在这个拓扑

中，并没有规定哪个区域是骨干区域。所有 Level-2 路由器构成了 IS-IS 的骨干网，他们可以属于不同的区域，但必须是连续的。

图 7-2 IS-IS 拓扑结构图之二



说明

IS-IS 的骨干网（Backbone）指的不是一个特定的区域。

这种组网方案也体现出 IS-IS 与 OSPF 的不同点。在 OSPF 中，区域之间的路由需要通过骨干区域转发，只有在同一个区域内才使用 SPF 算法。而 IS-IS 不论是 Level-1 还是 Level-2 路由，都采用 SPF 算法，分别生成最短路径树 SPT（Shortest Path Tree）。

网络类型

IS-IS 只支持两种类型的网络，根据物理链路不同可分为：

- 广播链路：如 Ethernet、Token-Ring 等。
- 点到点链路：如 PPP、HDLC 等。

说明

对于 NBMA（Non-Broadcast Multi-Access）网络，如 ATM，需对其配置子接口，并注意子接口类型不能为 P2MP。IS-IS 不能在点到多点链路 P2MP（Point to MultiPoint）上运行。

7.2 AR150/200 支持的 IS-IS 特性

AR150/200 支持的 IS-IS 特性包括：多实例和多进程、热备份、多拓扑、本地 MT、GR、TE 与 DS-TE、管理标记、LSP 分片扩展、动态主机名交换、快速收敛、BFD 和三次握手机制。

说明

IS-IS for IPv6 功能使用 License 授权，缺省情况下，设备的 IS-IS for IPv6 功能受限无法使用。如果需要使用 IS-IS for IPv6 功能，请联系华为办事处申请并购买如下 License，

- AR150&200 数据业务增值包

多实例和多进程

为了方便管理，提高控制效率，IS-IS 支持多进程和多实例特性。

- 多进程

为一个指定的 IS-IS 进程关联一组接口，从而保证该进程所有操作都仅限于这一组接口。这样，就可以实现一台路由器有多个 IS-IS 进程，每个进程负责唯一的一组接口。

- 多实例

  对于支持 VPN 的路由器，每个 IS-IS 进程都与一个指定的 VPN 实例相关联，所有附加到该进程的接口都应与这个 VPN 实例相关联。

IS-IS GR

Graceful Restart 指的是平缓重启路由器的一种功能，可以保证流量转发不中断，网络不会因为路由器的短时间重启而引起路由震荡。

若 IS-IS 协议不以 GR 方式重启，会重置 IS-IS 会话，重新生成链路状态协议数据报文 LSP（Link State Protocol Data Unit）和泛洪 LSP，进而在整个区域引发 SPF 计算，引起整个区域的路由震荡和转发中断。IETF 针对这种情况为 IS-IS 制定了 GR 规范（RFC3847），对保留 FIB 表和未保留 FIB 表的协议重启都进行了处理。

说明

有关 IS-IS GR 的详细介绍请参见《Huawei AR150&200 系列企业路由器 特性描述-IP 路由》中“IS-IS”。

管理标记

管理标记简化了管理，允许在 IS-IS 域中通过 IP 地址前缀发布进行控制。该标记用来携带关于 IP 地址前缀的管理信息，其用途包括控制不同级别和不同区域间的路由引入，各种路由协议，同一路由器上运行的 IS-IS 多实例，以及 TAG 的承载。

管理标记值与某些属性相关联。如果 IS-IS 要发布可达的 IP 地址前缀具有该属性，IS-IS 就会将管理标记加入到该前缀的 IP 可达信息 TLV 中。这样，管理标记就会随着前缀发布到整个路由域。

LSP 分片扩展

当 IS-IS 要发布的 LSP 中的信息量变大时，以同一系统的多个 LSP 分片的形式发布。LSP 分片由 LSP 的 LSP 标识符字段进行标识，这个字段的长度是 1 字节，因此，IS-IS 路由器可产生的分片数最大为 256。

IS-IS LSP 分片扩展特性可使 IS-IS 路由器生成更多的 LSP 分片。该特性可通过使能网络管理器，为路由器配置附加的系统 ID 实现。每个系统 ID 都代表一个虚拟系统，每个虚拟系统都可生成 256 个 LSP 分片。通过增加附加的系统 ID（最多可配置 50 个虚拟系统），IS-IS 路由器可最多生成 13056 个 LSP 分片。

- 相关术语
  - 初始系统（Originating System）

    初始系统是实际运行 IS-IS 协议的路由器。本手册描述的方法允许一个单独的 IS-IS 进程像多个虚拟路由器一样发布 LSP，而“Originating System”指的是那个“真正”的 IS-IS 进程。
  - 系统 ID（Normal System-ID）

    初始系统的系统 ID。
  - 附加系统 ID（Additional System-ID）

附加系统 ID 由网络管理器分配。每个附加系统 ID 都允许生成 256 个额外的或扩展的 LSP 分片。附加系统 ID 和普通系统 ID 一样，在整个路由域中必须唯一。

- 虚拟系统（Virtual System）

  由附加系统 ID 标识的系统，用来生成扩展 LSP 分片。这些分片在其 LSP ID 中携带附加系统 ID。

- 操作模式

  IS-IS 路由器可以在两种模式下运行 LSP 分片扩展特性：

  - mode-1：用于网络中的部分路由器不支持 LSP 分片扩展特性的情况。在该模式下，初始系统向 LSP 中的每个虚拟系统发布一条链路，然后每个虚拟系统也向初始系统发布一条链路。这些虚拟系统就好象网络中与初始系统相连的路由器。这种模式有一个局限就是虚拟系统的 LSP 报文只能发布路由信息。
  - mode-2：用于网络中所有路由器都支持 LSP 分片扩展特性的情况。在该模式下，网络中所有路由器都知道虚拟系统生成的 LSP 实际属于初始系统，而且对虚拟系统发布的 LSP 报文的链路状态信息没有限制。

动态主机名交换机制

动态主机名交换机制是为了方便对 IS-IS 网络的维护和管理而引入的，它为 IS-IS 路由器提供了一种从主机名到 System ID 映射的服务。这个动态的主机名信息在 LSP 中以一个动态主机名 TLV 的形式发布。

这个机制同时还提供将主机名与广播网中的 DIS（Designated Intermediate System）相关联的服务，并将此信息通过伪节点的 LSP 以动态主机名 TLV 的形式发布出去。

在维护和管理中，使用主机名比使用 System ID 会更直观，也更容易记忆。配置此功能后，当在路由器上使用显示命令查看 IS-IS 相关信息时，看到的是路由器的主机名，而不再是 System ID。

IS-IS 路由聚合

IS-IS 路由聚合是指将多条具有相同 IP 前缀的 IS-IS 路由聚合成一条路由。

当 IS-IS 网络规模较大时，路由聚合可以有效减少路由表中的条目，减小对系统资源的占用，方便管理。

此外，如果被聚合的 IP 地址范围内的某条链路频繁 Up 和 Down，该变化并不会通告到被聚合的 IP 地址范围外的设备。因此，可以避免网络中的路由振荡，在一定程度上提高了网络的稳定性。

路由器支持无类别网络的路由聚合。

IS-IS 负载分担

当 IS-IS 网络中有多条冗余链路时，可能会出现多条等价路由。

IS-IS 负载分担是指将流量均匀的分配在多条等价路径上，可以提高网络中链路的利用率及减少某些链路负担过重造成阻塞发生的情况。但是由于对流量转发具有一定的随机性，因此可能不利于对业务流量的管理。

IS-IS 下一跳优先级

当 IS-IS 网络中有多条冗余链路时，可能会出现多条等价路由。

路由器支持为等价路由中的每条路由配置优先级，优先级高的路由将被优选，优先级低的路由可以作为备用链路。

该特性可以在不修改原有配置的基础上，指定某条路由被优选，便于业务的管理，同时提高网络的可靠性。

IS-IS 快速收敛

- I-SPF（Incremental SPF）

  I-SPF 是指增量路由计算，它每次只对变化的一部分路由进行计算，而不是对全部路由重新计算。

  在 ISO-10589 中定义使用 Dijkstra 算法进行路由计算。当网络拓扑中有一个节点发生变化时，这种算法需要重新计算网络中的所有节点，计算时间长，占用过多的 CPU 资源，影响整个网络的收敛速度。

  I-SPF 改进了这个算法，除了第一次计算时需要计算全部节点外，每次只计算受影响的节点，而最后生成的最短路径树 SPT 与原来的算法所计算的结果相同，大大降低了 CPU 的占用率，提高了网络收敛速度。

- PRC（Partial Route Calculation）

  部分路由计算 PRC 的原理与 I-SPF 相同，都是只计算变化的那一部分。但 PRC 不需要计算节点路径，而是根据 I-SPF 算出来的 SPT 来更新叶子（路由）。

  在路由计算中，叶子代表路由，节点则代表路由器。如果 I-SPF 计算后的 SPT 改变，PRC 会只处理那个变化的节点上的所有叶子；如果经过 I-SPF 计算后的 SPT 并没有变化，则 PRC 只处理变化的叶子信息。

  比如一个节点使能一个 IS-IS 接口，则整个网络拓扑的 SPT 是不变的，这时 PRC 只更新这个节点的接口路由，从而节省 CPU 占用率。

  PRC 和 I-SPF 配合使用可以将网络的收敛性能进一步提高，它是原始 SPF 算法的改进，所以已经代替了原有的算法。

  说明

  在 AR150/200 的实现中，仅使用 I-SPF 和 PRC 作为 IS-IS 路由计算的算法。

- LSP 快速扩散

  按 RFC 协议的实现，当 IS-IS 收到其它路由器发来的 LSP 时，如果此 LSP 比自己 LSDB 中的 LSP 要新，则是用一个定时器定期将 LSDB 内的 LSP 扩散出去，所以 LSDB 的同步会比较缓慢。

  LSP 快速扩散特性改进了这种方式，配置此特性的路由器收到一个或多个比较新的 LSP 时，在路由计算之前，先将小于指定数目的 LSP 扩散出去，加快 LSDB 的同步过程。这种方式在很大程度上可以提高整个网络的收敛速度。

- 智能定时器

  改进了路由算法后，如果触发路由计算的间隔较长，同样会影响网路的收敛速度。使用毫秒级定时器可以缩短这个间隔时间，但如果网络变化比较频繁，又会造成过度占用 CPU 资源。SPF 智能定时器既可以对少量的外界突发事件进行快速响应，又可以避免过度的占用 CPU。

  通常情况下，一个正常运行的 IS-IS 网络是稳定的，发生大量的网络变动的几率很小，IS-IS 路由器不会频繁的进行路由计算，所以第一次触发的时间可以设置的非常短（毫秒级）。如果拓扑变化比较频繁，智能定时器会随着计算次数的增加，间隔时间也会逐渐延长，避免占用大量的 CPU 资源。

  与 SPF 智能定时器类似的还有 LSP 生成智能定时器。在 IS-IS 协议中，当 LSP 生成定时器到期时，系统会根据当前拓扑重新生成一个自己的 LSP。原有的实现机制是采用间隔时间定长的定时器，不能同时满足快速收敛和低 CPU 占用率的需要。为

此将 LSP 生成定时器也设计成智能定时器，使其可以对于突发事件（如接口 Up/Down）快速响应，加快网络的收敛速度。同时，当网络变化频繁时，智能定时器的间隔时间会自动延长，避免过度占用 CPU 资源。

说明

请根据网络的实际情况和路由器的性能，谨慎配置此类定时器。

BFD for IS-IS

AR150/200 支持使用 BFD 对 IS-IS 邻居关系进行检测。BFD 能够快速检测到 IS-IS 邻居间的链路故障，并上报给 IS-IS 协议，从而实现 IS-IS 协议的快速收敛。

说明

由于 IS-IS 只能建立单跳邻居，BFD 只对 IS-IS 邻居间的单跳链路进行检测。

- 静态 BFD

  静态 BFD 是指通过命令行手工配置 BFD 会话参数，包括配置本地标识符和远端标识符等，然后手工下发 BFD 会话建立请求。

  这种方式的缺点是建立和删除 BFD 会话时都需要手工触发，缺乏灵活性。而且有可能造成人为的配置错误，比如配置了错误的本地标识符或者远端标识符时，BFD 会话将不能正常工作。

  AR150/200 支持用静态 BFD for IS-IS 检测 IPv4 网络。

- 动态 BFD

  动态 BFD 会话指的是由路由协议动态触发建立 BFD 会话。

  路由协议在建立了新的邻居关系时，将邻居的参数及检测参数（包括目的地址、源地址等）通告给 BFD，BFD 根据收到的参数建立起会话。动态 BFD 比静态 BFD 更具有灵活性。

  通常情况下，IS-IS 设定发送 Hello 报文的时间间隔为 10 秒，宣告邻居失效的时间即邻居保持时间一般配置为 Hello 报文间隔的 3 倍，所以邻居故障感知时间为秒级，由此可能会出现高速的网络环境中大量报文丢失的问题。

  动态 BFD（Bidirectional Forwarding Detection）能够提供轻负荷、快速（毫秒级）的通道故障检测。使用动态 BFD 并不是代替 IS-IS 协议本身的 Hello 机制，而是配合 IS-IS 协议更快的发现邻居设备或链路出现的故障，并及时通知 IS-IS 重新计算相关路由以便正确指导报文的转发。

  AR150/200 支持采用动态 BFD for IS-IS 检测 IPv4 和 IPv6 网络。

  说明

  有关 BFD for IS-IS 的详细介绍请参见《Huawei AR150&200 系列企业路由器 特性描述-IP 路由》中“IS-IS”。

三次握手机制（3-Way Handshake）

IS-IS 协议在点到点链路上需要一个可靠的链路层协议。ISO 10589 中的 IS-IS 的 2 次握手机制（2-Way Handshake）使用 Hello 报文来建立相邻路由器间点到点链路的邻接关系。这种机制中，只要路由器收到对端发来的 Hello 报文，就宣布邻居为 Up 状态，建立邻接关系。

这种机制存在明显缺陷。例如，邻接关系建立后，当链路状态反复波动引起 CSNP 报文丢失，导致 LSDB 在一个完整的 LSP 更新周期内不能达到同步；当路由器间存在两条及以上的链路时，如果某条链路上到达对端的单向状态为 Down，而另一条链路同方向的状态为 Up，这样路由器之间还是能建立起邻接关系。SPF 在计算时会使用另一条链路

上的参数，这导致没有检测到故障的路由器在转发报文时仍然试图通过状态为 Down 的链路。

三次握手机制解决了上述不可靠点到点链路中存在的问题。这种方式下，路由器只有在知道邻居路由器也接收到它的报文时，才宣布邻居路由器处于 Up 状态，从而建立邻接关系。同时，三次握手机制中使用 32 比特的扩展 Circuit ID，打破了目前由本地 8 比特 Circuit ID 字段限制的 255 个点到点链路。

7.3 配置 IS-IS 的基本功能(IPv4)

配置 IS-IS 的基本功能(IPv4)可以实现基于 IPv4 地址族的 IS-IS 网络中各节点的互通。配置步骤主要包括配置 IS-IS 进程和配置 IS-IS 接口。

7.3.1 建立配置任务

在配置 IS-IS 基本功能(IPv4)前了解此特性的应用环境、配置此特性的前置任务和数据准备，可以帮助您快速、准确地完成配置任务。

应用环境

在 IPv4 网络中部署 IS-IS 协议时，首先需要配置 IS-IS 的基本功能，实现网络中节点的路由互通。

只有完成基本功能的配置，才能配置其他 IS-IS 特性。

配置 IS-IS 的基本功能(IPv4)的步骤主要包括：

1. 创建 IS-IS 进程(IPv4)
2. 使能 IS-IS 接口(IPv4)

前置任务

在配置 IS-IS 的基本功能(IPv4)之前，需完成以下任务：

- 配置链路层协议
- 配置接口的网络层地址，使相邻节点网络层可达

数据准备

在配置 IS-IS 的基本功能(IPv4)之前，需要准备以下数据。

序号	数据
1	IS-IS 进程号
2	IS-IS 进程的网络实体名称
3	设备及接口的 Level 级别

7.3.2 创建 IS-IS 进程(IPv4)

配置 IS-IS 的基本功能（IPv4）首先需要创建 IS-IS 进程（IPv4），然后才能使能 IS-IS 接口（IPv4）。

背景信息

IS-IS 进程的配置包括：

- **创建 IS-IS 进程并配置设备的 NET**
- **（可选）配置设备的 Level 级别**

  缺省情况下，设备的 Level 级别为 **level-1-2**。

  建议根据网络规划的需要，配置设备的 Level 级别。否则，IS-IS 会为 Level-1 和 Level-2 分别建立邻居，维护两份相同的 LSDB，造成对设备资源的过多占用。
- **（可选）配置 IS-IS 主机名映射**

  配置 IS-IS 主机名映射后，使用显示命令查看 IS-IS 的相关信息时，会以配置的动态名称代替设备的 System ID，从而提高 IS-IS 网络的可维护性。
- **（可选）打开 IS-IS 的邻接状态开关**

  在本地 terminal monitor 开关已开启的情况下，当打开邻接状态输出开关后，IS-IS 邻接状态的变化会输出到配置终端上，直至邻接状态输出开关被关闭。

操作步骤

- 创建 IS-IS 进程并配置设备的 NET
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **isis** [*process-id*]，创建 IS-IS 进程，进入 IS-IS 视图。

     参数 *process-id* 用来指定一个 IS-IS 进程。如果不指定参数 *process-id*，则系统默认的进程为 1。IS-IS 进程可以与 VPN 实例相关联，此时需要执行命令 **isis** [*process-id*] [**vpn-instance** *vpn-instance-name*] 。
  3. 执行命令 **network-entity** *net*，设置网络实体名称。

  注意

  建议将 Loopback 接口的地址转化为 NET，保证 NET 在网络中的唯一性。如果网络中的 NET 不唯一，容易引发路由振荡，因此要做好前期网络规划。

  IS-IS 在建立 Level-2 邻居时，不检查区域地址是否相同，而在建立 Level-1 邻居时，区域地址必须相同，否则无法建立邻居。
- （可选）配置设备的 Level 级别
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **isis** [*process-id*]，创建 IS-IS 进程，进入 IS-IS 视图。
  3. 执行命令 **is-level** { **level-1** | **level-1-2** | **level-2** }，设置路由器的 Level 级别。
- （可选）配置 IS-IS 主机名映射
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **isis** [*process-id*]，创建 IS-IS 进程，进入 IS-IS 视图。

3. 执行命令 **is-name** *symbolic-name*，配置 IS-IS 动态主机名映射，为本地设备配置主机名称。

   该配置属于动态配置，即配置的主机名称 *symbolic-name* 以 LSP 报文的形式发布给区域中的其它 IS-IS 设备。

   在其他设备上使用 IS-IS 相关显示命令查看 IS-IS 信息那时，系统 ID 将被 *symbolic-name* 代替。

4. 执行命令 **is-name map** *system-id symbolic-name*，配置 IS-IS 静态主机名映射，为远端 IS-IS 设备配置主机名称。

   该配置属于静态配置，即只在本地设备生效，配置的主机名称 *symbolic-name* 不会通过 LSP 报文发送出去。

   因此，如果网络中的对应的 IS-IS 设备配置了动态主机名映射，那么该映射关系将覆盖本地路由器的静态映射。

- （可选）打开 IS-IS 的邻接状态开关
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **isis** [*process-id*]，创建 IS-IS 进程，进入 IS-IS 视图。
  3. 执行命令 **log-peer-change**，打开邻接状态输出开关。

----结束

7.3.3 使能 IS-IS 接口(IPv4)

只有在相应的接口下使能 IS-IS，IS-IS 才能通过该接口发送 Hello 报文建立邻居、扩散 LSP 报文。

背景信息

IS-IS 设备的 Level 级别和接口的 Level 级别共同决定了建立邻居关系的 Level 级别。两台 Level-1-2 设备建立邻居关系时，缺省情况下，会分别建立 Level-1 和 Level-2 邻居关系。如果只希望建立 Level-1 或者 Level-2 的邻居关系，可以通过修改接口的 Level 级别实现。

接口下使能 IS-IS 后，该接口会主动发送 Hello 报文尝试与对端建立邻居。如果对端不是 IS-IS 设备，或者只是希望将该接口所在网段的路由发布出去，并不希望通过该接口建立邻居，可以配置抑制该接口。配置后，该接口所在网段的路由仍然可以被发布出去，且并不发送 Hello 报文，减少对链路带宽的占用。

操作步骤

- 使能 IS-IS 接口
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **interface** *interface-type interface-number*，进入接口视图。
  3. 执行命令 **isis enable** [*process-id*]，使能 IS-IS 接口。

     配置该命令后，IS-IS 将通过该接口建立邻居、扩散 LSP 报文。

     说明

     由于 Loopback 接口不需要建立邻居，因此如果在 Loopback 接口下使能 IS-IS，只会将该接口所在的网段路由通过其他 IS-IS 接口发布出去。

- （可选）配置 IS-IS 接口的 Level 级别

1. 执行命令 **system-view**，进入系统视图。
2. 执行命令 **interface** *interface-type interface-number*，进入接口视图。
3. 执行命令 **isis circuit-level** [**level-1** | **level-1-2** | **level-2**]，设置接口的 Level 级别。

   缺省情况下，接口的 Level 级别为 **level-1-2**。

   说明

   只有当 IS-IS 设备的 Level 级别为 Level-1-2 时，改变接口的 Level 级别才有意义，否则将由 IS-IS 设备的 Level 级别决定所能建立的邻接关系层次。

- （可选）配置 IS-IS 接口为抑制状态
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **interface** *interface-type interface-number*，进入接口视图。
  3. 执行命令 **isis silent**，配置 IS-IS 接口为抑制状态。

     IS-IS 接口为抑制状态时，此接口不再接收或发送 IS-IS 报文，但接口所在网段的路由仍可以被发布到域内的其他 IS-IS 设备。

----结束

7.3.4 (可选)配置 IS-IS 接口的开销(IPv4)

配置 IS-IS 的接口开销可以控制 IS-IS 的路由选择，请根据网络规划适当配置接口的开销。

背景信息

IS-IS 有三种方式来确定接口的开销，按照优先级由高到低分别是：

- 接口开销：为单个接口设置开销。
- 全局开销：为所有接口设置开销。
- 自动计算开销：根据接口带宽自动计算开销。

如果没有显式的配置任何命令，则 IS-IS 接口的默认开销为 10，开销类型是 narrow。

操作步骤

- 配置 IS-IS 开销的类型
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **isis** [*process-id*]，进入 IS-IS 视图。
  3. 执行命令 **cost-style** { **narrow** | **wide** | **wide-compatible** | { { **narrow-compatible** | **compatible** } [**relax-spf-limit**] } }，设置 IS-IS 开销的类型。

     对于不同的开销类型，其接口开销的取值范围有所不同，接收到的路由开销取值范围也有所不同。

     - narrow 类型：接口开销取值范围为 1 ～ 63。接收到的路由开销值最大为 1023。
     - narrow-compatible 和 compatible 类型：接口开销取值范围为 1 ～ 63。接收到的路由开销值和参数 **relax-spf-limit** 有关。
       - 不设置 **relax-spf-limit** 参数：

         如果路由开销值小于等于 1023，且该路由经过的所有接口的开销值都小于等于 63：这条路由的开销值按照实际值接收。

如果路由开销值小于等于 1023，但该路由经过的所有接口中有的接口链路开销值大于 63：IS-IS 设备只能学到该接口所在网段的路由和接口所引入的路由，这条路由的开销值按照实际值接收，之后通过此接口转发的路由将被丢弃。

如果路由开销值大于 1023：IS-IS 设备只能学到路由开销值第一次超过 1023 的那个接口（该接口之前的所有接口的链路开销值小于等于 63）的路由。该接口所在网段的路由和引入的路由均可以被学习到，路由的开销值按照 1023 接收，之后通过此接口转发的路由将被丢弃。

- 设置 **relax-spf-limit** 参数：

  对接口的链路开销值和路由开销值均没有限制，按照实际的路由开销值正常接收该路由。

- wide 和 wide-compatible 类型：接口开销取值范围是 1 ～ 16777215。配置为最大值 16777215 时，该链路上生成的邻居 TLV（cost 为 16777215）不能用于路由计算，仅用于传递 TE 相关信息。接收到的路由开销值最大为 0xFFFFFFFF。

- 配置 IS-IS 接口的开销

  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **interface** *interface-type interface-number*，进入接口视图。
  3. 执行命令 **isis cost** *cost* [**level-1** | **level-2**]，设置 IS-IS 接口的开销。

     使用此命令可以单独为某接口配置开销值。

- 配置 IS-IS 的全局开销

  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **isis** [*process-id*]，进入 IS-IS 视图。
  3. 执行命令 **circuit-cost** *cost* [**level-1** | **level-2**]，设置 IS-IS 全局开销。

     使用此命令可以一次性改变所有接口的开销值。

- 使能 IS-IS 自动计算接口的开销

  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **isis** [*process-id*]，进入 IS-IS 视图。
  3. 执行命令 **bandwidth-reference** *value*，配置计算带宽的参考值。缺省情况下，带宽参考值为 100，单位是 Mbit/s。
  4. 执行命令 **auto-cost enable**，使能自动计算接口的开销值。

  只有当开销类型为 wide 或 wide-compatible 时，步骤 3 配置的带宽参考值才是有效的，此时各接口的开销值=(bandwidth-reference/接口带宽值)×10。

  当开销类型为 narrow、narrow-compatible 或 compatible 时，各个接口的开销值根据表 7-1 来确定。

表 7-1 IS-IS 接口开销和带宽对应关系表

开销值	接口带宽范围
60	接口带宽≤10Mbit/s
50	10Mbit/s<接口带宽≤100Mbit/s
40	100Mbit/s<接口带宽≤155Mbit/s

开销值	接口带宽范围
30	155Mbit/s<接口带宽≤622Mbit/s
20	622Mbit/s<接口带宽≤2.5Gbit/s
10	2.5Gbit/s<接口带宽

说明

要改变 Loopback 接口的开销，只能在接口视图下使用 **isis cost** 命令配置。

----结束

7.3.5 (可选)配置不同网络类型接口的 IS-IS 属性(IPv4)

针对不同网络类型的接口，可以配置不同的 IS-IS 属性。

背景信息

由于 IS-IS 在广播网中和 P2P 网络中建立邻居的方式不同，因此，针对不同类型的接口，可以配置不同的 IS-IS 属性。

在广播网中，IS-IS 需要选择 DIS，因此通过配置 IS-IS 接口的 DIS 优先级，可以使拥有接口优先级最高的设备优选为 DIS。

链路两端的 IS-IS 接口的网络类型必须一致，否则双方不可以建立起邻居关系。如果对端设备的接口类型为 P2P 接口，可以将本地设备的广播网接口改为 P2P 接口，以满足与对端建立邻居的需要。

在 P2P 网络中，IS-IS 不需要选择 DIS，因此无需配置接口的 DIS 优先级。但是为了保证 P2P 链路的可靠性，可以配置 IS-IS 使用 P2P 接口在建立邻居时采用 3-way 模式，以检测单向链路故障。

操作步骤

- 配置接口的 DIS 优先级
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **interface** *interface-type interface-number*，进入接口视图。
  3. 执行命令 **isis dis-priority** *priority* [**level-1** | **level-2**]，设置用来选举 DIS 的优先级，数值越大优先级越高。
- 配置 IS-IS 接口的网络类型
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **interface** *interface-type interface-number*，进入接口视图。
  3. 执行命令 **isis circuit-type p2p**，设置接口的网络类型为 P2P。

     缺省情况下，接口网络类型根据物理接口决定。

     在使能 IS-IS 的接口上，当接口网络类型发生改变时，相关配置发生改变，具体如下。

     - 使用 **isis circuit-type p2p** 命令将广播网接口模拟成 P2P 接口时，接口发送 Hello 报文的间隔时间、宣告邻居失效前 IS-IS 没有收到的邻居 Hello 报文数

目、点到点链路上 LSP 报文的重传间隔时间以及 IS-IS 各种认证均恢复为缺省配置，而 DIS 优先级、DIS 名称、广播网络上发送 CSNP 报文的间隔时间等配置均失效。

- 使用 **undo isis circuit-type** 命令恢复接口的网络类型时，接口发送 Hello 报文的间隔时间、宣告邻居失效前 IS-IS 没有收到的邻居 Hello 报文数目、点到点链路上 LSP 报文的重传间隔时间、IS-IS 各种认证、DIS 优先级和广播网络上发送 CSNP 报文的间隔时间均恢复为缺省配置。

- 配置 P2P 链路中的邻居建立协商模型
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **interface** *interface-type interface-number*，进入接口视图。
  3. 执行命令 **isis ppp-negotiation** { **2-way** | **3-way** [**only**] }，指定接口使用的协商模型。

     缺省情况下，使用 **3-way** 协商模式。

     此命令只适用于 P2P 链路上建立邻居。对于广播链路，可以通过命令 **isis circuit-type p2p** 更改链路类型为 P2P，然后使用此命令配置邻居建立方式。
- 配置 PPP 链路协议接口的 OSICP 协商检查
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **interface** *interface-type interface-number*，进入接口视图。
  3. 执行命令 **isis ppp-osicp-check**，指定 PPP 链路接口进行 OSICP 状态检查。

     缺省情况下，PPP 链路协议的 OSICP 状态不影响 IS-IS 接口的状态。

     此命令只适用于 PPP 链路协议的接口，对于运行其他链路协议的点对点接口，配置命令无效。

     配置此命令后，PPP 链路协议的 OSI 网络协商状态会影响 IS-IS 接口状态。当 PPP 协议感知 OSI 网络不通时，IS-IS 接口的链路状态将会被设为 Down，到接口网段的路由就不会在 LSP 中发布。
- 配置 IS-IS 对接收的 Hello 报文不作 IP 地址检查
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **interface** *interface-type interface-number*，进入接口视图。
  3. 执行命令 **isis peer-ip-ignore**，配置对接收的 Hello 报文不作 IP 地址检查。

----结束

7.3.6 检查配置结果

配置好 IS-IS 的基本功能(IPv4)后，可以查看 IS-IS 的邻居、接口和路由信息。

前提条件

已经完成 IS-IS 的基本功能(IPv4)的所有配置。

操作步骤

步骤 1 使用 **display isis name-table** [*process-id* | **vpn-instance** *vpn-instance-name*]命令查看本地设备名称到系统 ID 的映射关系表。

步骤 2 使用 **display isis peer** [**verbose**] [*process-id* | **vpn-instance** *vpn-instance-name*]命令查看 IS-IS 的邻居信息。

步骤 3 使用 **display isis interface** [**verbose**] [*process-id* | **vpn-instance** *vpn-instance-name*]命令查看使能了 IS-IS 的接口信息。

步骤 4 使用 **display isis route** [*process-id* | **vpn-instance** *vpn-instance-name*] [**ipv4**] [**verbose** | [**level-1** | **level-2**] | *ip-address* [*mask* | *mask-length*]] *命令查看 IS-IS 的路由信息。

----结束

7.4 建立或维持 IS-IS 邻居或邻接关系

针对影响 IS-IS 邻居关系保持的参数配置进行介绍。

7.4.1 建立配置任务

在配置影响 IS-IS 邻居关系建立和维持的各种特性前了解此特性的应用环境、配置此特性的前置任务和数据准备，可以帮助您快速、准确地完成配置任务。

应用环境

本节主要介绍建立或维持 IS-IS 邻居或邻接关系的配置，具体有：

- 调整各种 IS-IS 报文的定时器，包括 Hello 报文、CSNP 报文和 LSP 等。
- 调整 LSP 的各种参数。

前置任务

在建立或维持 IS-IS 邻居或邻接关系之前，需完成以下任务：

- 配置接口的网络层地址，使相邻节点网络层可达。
- **配置 IS-IS 的基本功能(IPv4)**。

数据准备

在建立或维持 IS-IS 邻居或邻接关系之前，需完成以下任务：

序号	数据
1	IS-IS 报文定时器的参数
2	LSP 的参数

7.4.2 配置 IS-IS 报文定时器

Hello 报文定时器、CSNP 报文定时器和 LSP 报文定时器。

背景信息

请在运行 IS-IS 协议的路由器上进行以下配置。

操作步骤

- 配置 Hello 报文发送间隔
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **interface** *interface-type interface-number*，进入接口视图。
  3. 执行命令 **isis timer hello** *hello-interval* [**level-1** | **level-2**]，设置接口上 Hello 报文发送间隔。

     广播链路上存在 Level-1 和 Level-2 两种 Hello 报文，不同类型的报文可以设置不同的值。如果不指定级别，则默认为 Level-1 和 Level-2 同时配置。在点到点链路上，只有一种 Hello 报文，不需要使用参数 **level-1** 和 **level-2**。

     说明

     参数 **level-1** 和 **level-2** 仅在广播接口上是可配置的。
- 配置 Hello 报文失效数目
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **interface** *interface-type interface-number*，进入接口视图。
  3. 执行命令 **isis timer holding-multiplier** *number* [**level-1** | **level-2**]，设置 Hello 报文失效数目。

     如果命令中不指定级别，则默认为 Level-1 和 Level-2 同时配置。

     说明

     参数 **level-1** 和 **level-2** 仅在广播接口上是可见的。

  IS-IS 协议通过 Hello 报文的收发来维护与相邻路由器的邻居关系，当本端路由器在一段时间内没有收到对端发送的 Hello 报文时，将认为邻居路由器已经失效。

  在 IS-IS 中，本端路由器与相邻路由器保持为邻居关系的时间长短可以通过设置 Hello 报文的失效数目和 Hello 报文的时间间隔来控制。
- 配置 CSNP 报文发送间隔
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **interface** *interface-type interface-number*，进入接口视图。
  3. 执行命令 **isis timer csnp** *csnp-interval* [**level-1** | **level-2**]，设置接口上 CSNP 报文发送间隔。

  CSNP 报文是 DIS（Designated IS）在广播型网络上同步链路状态数据库 LSDB 所发送的报文。如果命令中不指定 Level-1 或 Level-2，则默认为设置当前级别的 CSNP 报文广播间隔。
- 配置接口的 LSP 重传间隔
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **interface** *interface-type interface-number*，进入接口视图。
  3. 执行命令 **isis circuit-type p2p**，设置接口网络类型为 P2P 类型。
  4. 执行命令 **isis timer lsp-retransmit** *retransmit-interval*，设置 LSP 在点到点链路上的重传间隔。

  在点到点的链路中，本端发送的 LSP 如果一段时间内没有收到应答，则认为原先发送的 LSP 丢失或被丢弃，为保证发送的可靠性，本端路由器会根据重传间隔将原先的 LSP 重新发送一次。缺省情况下，LSP 在点到点链路上的重传间隔是 5 秒。

在广播链路上发送的 LSP 报文不需要应答。

- 配置接口发送 LSP 的最小间隔时间
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **interface** *interface-type interface-number*，进入接口视图。
  3. 执行命令 **isis timer lsp-throttle** *throttle-interval* [**count** *count*]，设置发送 LSP 的最小间隔时间。

     count：用来指定在 *throttle-interval* 时间间隔内发送 LSP 报文的最大包数。整数形式，取值范围是 1 ～ 1000。

  可以设置 IS-IS 在接口上发送 LSP 报文的最小间隔时间，即两个连续的 LSP 之间的时延。该时间也是一个 CSNP 报文的多个分片之间的发送间隔。

----结束

7.4.3 配置 LSP 的参数

通过配置 LSP 生成定时器，调整 IS-IS 网络生成 LSP 报文的时间；通过配置 IS-IS 产生和接收 LSP 的大小，影响 LSP 报文的接收。

背景信息

请在运行 IS-IS 协议的路由器上进行以下配置。

操作步骤

- 配置 LSP 刷新周期
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **isis** [*process-id*]，进入 IS-IS 视图。
  3. 执行命令 **timer lsp-refresh** *refresh-time*，设置 LSP 刷新周期。

  为了保证整个区域中的 LSP 能够保持同步，IS-IS 周期性发送当前全部 LSP。

  LSP 刷新周期的缺省值为 900 秒，最大有效时间的缺省值为 1200 秒。配置时请注意，必须保证刷新周期比 LSP 的最大有效时间少 300 秒以上，使得原有的 LSP 过期之前，新的 LSP 可以到达区域内所有路由器。

  说明

  网络规模越大，LSP 刷新周期与 LSP 最大有效时间之间的差值也越大。

- 配置 LSP 最大有效时间
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **isis** [*process-id*]，进入 IS-IS 视图。
  3. 执行命令 **timer lsp-max-age** *age-time*，设置 LSP 最大有效时间。

  路由器生成系统 LSP 时，会在 LSP 中填写此 LSP 的最大有效时间。当此 LSP 被其它路由器接收后，它的有效时间会随着时间的变化不断减小。如果路由器一直没有收到更新的 LSP，而此 LSP 的有效时间已减少到 0，LSP 再保持 60 秒，若还未收到新的 LSP，那么此 LSP 将被从 LSDB 中删除。

- 配置 LSP 生成所使用的智能定时器
  1. 执行命令 **system-view**，进入系统视图。

2. 执行命令 **isis** [*process-id*]，进入 IS-IS 视图。
3. 执行命令 **timer lsp-generation** *max-interval* [*init-interval* [*incr-interval*]] [**level-1** | **level-2**]，设置 LSP 生成所使用的智能定时器。

如果没有指定 Level，则认为同时设置 Level-1 和 Level-2。

初次产生同一 LSP（或者 LSP 分片）的延迟时间为 *init-interval*，第二次产生同一 LSP（或者 LSP 分片）的延迟时间为 *incr-interval*。随后，每变化一次，延迟时间都增大为前一次的两倍，直到 *max-interval*。稳定在 *max-interval* 三次或者 IS-IS 进程被重启，延迟时间又降回到 *init-interval*。

在不使用 *incr-interval* 的情况下，初次产生同一 LSP（或者 LSP 分片）仍然使用 *init-interval* 作为延迟时间，随后都是使用 *max-interval* 作为延迟时间。同样，稳定在 *max-interval* 三次或者 IS-IS 进程被重启，延迟时间又降回到 *init-interval*。

在只使用 *max-interval* 的情况下，智能定时器退化为一般的一次性触发定时器。

- 配置 LSP 的大小
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **isis** [*process-id*]，进入 IS-IS 视图。
  3. 执行命令 **lsp-length originate** *max-size*，设置生成 LSP 的大小。
  4. 执行命令 **lsp-length receive** *max-size*，设置接收 LSP 的大小。

  说明

  设置 *max-size* 参数时请注意，生成 LSP 的 *max-size* 必须小于等于接收 LSP 的 *max-size*。

  使用 **lsp-length** 命令设置的 *max-size* 值必须满足以下要求，否则接口的 MTU 状态会被认为是 Down。
  - 以太网接口的 MTU 值大于等于 *max-size*+3。
  - P2P 接口的 MTU 值大于等于 *max-size*。
- 配置接口加入 Mesh-Group
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **interface** *interface-type interface-number*，进入接口视图。
  3. 执行命令 **isis mesh-group** { *mesh-group-number* | **mesh-blocked** }，设置接口加入 Mesh Group。

  在 NBMA 网络上，路由器的一个接口收到一个新的 LSP，会将该 LSP 扩散（Flooding）到路由器的其它接口。在连通程度比较高的、有多条点到点链路的网络中，这种处理方式会造成 LSP 重复扩散，导致带宽的浪费。

  为了避免这种情况的发生，可以将一些接口组成 Mesh Group，一个 Mesh Group 组中的路由器不把从本组接口接收的 LSP 扩散到同组中的其它接口，而只扩散到其它组的接口以及没有配置 Mesh Group 的接口。

  当对接口设置了 **mesh-blocked** 参数后，接口被阻塞，不再向外扩散 LSP。所有加入到 Mesh Group 中的接口，通过 CSNP 和 PSNP 机制来保证整个网段的 LSDB 的同步。

  说明

  在 ATM 或 FR 网络中，IS-IS 是通过虚电路（VCs）进行连接的，因此这里所指的接口是点对点的逻辑子接口。
- 配置 LSP 分片扩展

1. 执行命令 **system-view**，进入系统视图。
2. 执行命令 **isis** [*process-id*]，进入 IS-IS 视图。
3. 执行命令 **lsp-fragments-extend** [[**level-1** | **level-2** | **level-1-2**] | [**mode-1** | **mode-2**]] *，使能 IS-IS 进程的 LSP 分片扩展。
4. 执行命令 **virtual-system** *virtual-system-id*，配置一个虚拟系统。

为了使路由器生成扩展 LSP 分片，应至少配置一个虚拟 System ID。这个虚拟 System ID 在整个路由域中必须唯一。

一个 IS-IS 进程最多可配置 50 个虚拟 System ID。

配置 LSP 分片扩展时，如果不指定 mode 和 level 级别，则默认为 mode-1 和 level-1-2。

----结束

7.4.4 检查配置结果

配置好影响 IS-IS 邻居关系的各种特性后，可以查看接口发送 IS-IS 报文的各种参数和统计信息。

前提条件

已经完成建立或维持 IS-IS 邻居或邻接关系的所有配置。

操作步骤

- 使用 **display isis interface** [**verbose**] [*process-id* | **vpn-instance** *vpn-instance-name*]命令查看使能了 IS-IS 的接口信息。
- 使用以下命令查看 IS-IS 进程的统计信息：
  - **display isis statistics** [**level-1** | **level-2** | **level-1-2**] [*process-id* | **vpn-instance** *vpn-instance-name*]
  - **display isis statistics packet** [**interface** *interface-type interface-number*]
  - **display isis** *process-id* **statistics** [**updated-lsp** [**history**]] [**level-1** | **level-2** | **level-1-2** | **packet**]

----结束

7.5 调整 IS-IS 的选路(IPv4)

通过调整 IS-IS 选路，可以实现对路由选择的精确控制。

7.5.1 建立配置任务

在调整 IS-IS 的选路(IPv4)前了解此特性的应用环境、配置这些特性的前置任务和数据准备，可以帮助您快速、准确地完成配置任务。

应用环境

配置 IS-IS 的基本功能(IPv4)后，可以形成 IS-IS 路由表，实现了网络中各节点的互通。

但是，如果网络中存在多条冗余链路，那么此时 IS-IS 路由表中的路由可能不是期望的最优路由，不能满足网络规划和流量管理的需要。为了达到优化 IS-IS 网络和便于流量管理的目的，需要对网络中的路由进行更加精确的控制。

调整 IS-IS 选路的方式主要有如下几种：

- **配置 IS-IS 接口的开销(IPv4)**。

  说明

  虽然通过修改 IS-IS 接口的开销，可以在一定程度上达到控制选路的目的。但是，该方式有较大的局限性。因为在网络调整时（尤其是大规模网络），为了达到让某条路由优选而修改了接口开销，将会影响到该接口涉及的所有路由的重新计算和收敛，配置后的结果可能会与期望的不一致。

  因此，接口开销的调整尽量在配置 IS-IS 的基本功能时完成。
- 配置 IS-IS 路由渗透(IPv4)。
- 配置 IS-IS 对等价路由的处理(IPv4)。
- 控制将 IS-IS 路由下发到 IP 路由表(IPv4)。
- 配置 IS-IS 设备进入过载状态(IPv4)。

前置任务

在调整 IS-IS 的选路(IPv4)之前，需完成以下任务：

- 配置接口的网络层地址，使相邻节点网络层可达。
- **配置 IS-IS 的基本功能(IPv4)**。

数据准备

在调整 IS-IS 的选路(IPv4)之前，需要准备以下数据。

序号	数据
1	路由过滤时所采用的 ACL、IP 前缀列表或者路由策略
2	负载分担时采用的最大等价路由条数
3	下一跳优先级
4	IS-IS 设备进入过载状态的时间

7.5.2 配置 IS-IS 路由渗透(IPv4)

在双 Level 组网中，配置 IS-IS 路由渗透可以改变 Level 之间的路由渗透方式，实现对 IS-IS 路由的控制。

背景信息

如果在一个 Level-1 区域中有多台 Level-1-2 设备与 Level-2 区域相连，每台 Level-1-2 设备都会在 Level-1 LSP 中设置 ATT 标志位，则该区域中就有到达 Level-2 区域和其他 Level-1 区域的多条出口路由。

缺省情况下，Level-1 区域的路由会渗透到 Level-2 区域中，因此 Level-1-2 设备和 Level-2 设备了解整个网络的拓扑信息。由于 Level-1 区域的设备只维护本地 Level-1 区域的 LSDB

数据库，不知道整个网络的拓扑信息，所以只能选择将流量转发到最近的 Level-1-2 设备，再由 Level-1-2 设备将流量转发到 Level-2 区域。然而，该路由可能不是到达目的地的最优路由。

为了帮助 Level-1 区域内的设备选择到达其他区域的最优路由，可以配置 IPv4 IS-IS 路由渗透，将 Level-2 区域的某些路由渗透到本地 Level-1 区域。

另外，考虑到网络中部署的某些业务可能只在本地 Level-1 区域内运行，则无需将这些路由渗透到 Level-2 区域中，可以通过配置策略仅将部分 Level-1 区域的路由渗透到 Level-2 区域。

操作步骤

- 配置 Level-2 区域的路由渗透到 Level-1 区域。
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **isis** [*process-id*]，进入 IS-IS 视图。
  3. 执行命令 **import-route isis level-2 into level-1** [**tag** *tag* | **filter-policy** { *acl-number* | **acl-name** *acl-name* | **ip-prefix** *ip-prefix-name* | **route-policy** *route-policy-name* }] *，将 Level-2 区域和其他 Level-1 区域的某些路由渗透到本地 Level-1 区域。

     说明

     该命令配置在与外部区域相连的 Level-1-2 设备上。

     缺省情况下，Level-2 区域的路由信息不渗透到 Level-1 区域。配置该命令后，通过过滤策略的路由将渗透到 Level-1 区域中。
- 配置 Level-1 区域的路由渗透到 Level-2 区域。
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **isis** [*process-id*]，进入 IS-IS 视图。
  3. 执行命令 **import-route isis level-1 into level-2** [**tag** *tag* | **filter-policy** { *acl-number* | **acl-name** *acl-name* | **ip-prefix** *ip-prefix-name* | **route-policy** *route-policy-name* }] *，将 Level-1 区域的某些路由渗透到本地 Level-2 区域。

     说明

     该命令配置在与外部区域相连的 Level-1-2 设备上。

     缺省情况下，Level-1 区域的路由信息全部渗透到 Level-2 区域。配置该命令后，只有通过过滤策略的路由才能渗透到 Level-2 区域中。

----结束

7.5.3 配置 IS-IS 对等价路由的处理方式(IPv4)

当 IS-IS 网络中有多条等价路由时，既可以通过配置负载分担以提高每条链路的利用率，也可以通过配置等价路由优先级明确指定下一跳以便于业务流量的管理。

背景信息

当 IS-IS 网络中有多条冗余链路时，可能会出现多条等价路由，此时可以采取两种方式：

- 配置负载分担。流量会被均匀的分配到每条链路上。

  该方式可以提高网络中链路的利用率及减少某些链路负担过重造成阻塞发生的情况。但是由于对流量转发具有一定的随机性，因此可能不利于对业务流量的管理。

- 配置等价路由优先级。针对等价路由中的每一条路由，明确指定其优先级，优先级高的路由将被优选，优先级低的路由可以作为备用链路。

  该方式可以在不修改原有配置的基础上，指定某条路由被优选，便于业务的管理，同时提高网络的可靠性。

操作步骤

- 配置 IS-IS 路由负载分担
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **isis** [*process-id*]，进入 IS-IS 视图。
  3. 执行命令 **maximum load-balancing** *number*，配置在负载分担方式下的等价路由的最大数量。

     说明

     如果命令中指定的 *number* 小于网络中存在的等价路由数量时，则 IS-IS 会从所有等价路由中选取 *number* 条路由进行负载分担。
- 配置 IS-IS 等价路由的优先级
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **isis** [*process-id*]，进入 IS-IS 视图。
  3. 执行命令 **nexthop** *ip-address* **weight** *value*，配置等价路由的优先级。

     说明

     value 值越小，表示优先级越高。

----结束

7.5.4 控制将 IS-IS 路由下发到 IP 路由表(IPv4)

当不希望某些 IS-IS 路由被优选时，可以通过策略阻止将部分 IS-IS 路由下发到 IP 路由表来实现。

背景信息

IP 报文是根据 IP 路由表来进行转发的。IS-IS 路由表中的路由条目需要被成功下发到 IP 路由表中，该路由条目才生效。

因此，可以通过配置基本 ACL、IP-Prefix、路由策略等方式，只允许匹配的 IS-IS 路由下发到 IP 路由表中。不匹配的 IS-IS 路由将会被阻止进入 IP 路由表，更不会被优选。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **isis** [*process-id*]，进入 IS-IS 视图。

步骤 3 执行命令 **filter-policy** { *acl-number* | **acl-name** *acl-name* | **ip-prefix** *ip-prefix-name* | **route-policy** *route-policy-name* } **import**，控制将部分 IS-IS 路由下发到 IP 路由表。

----结束

7.5.5 配置 IS-IS 设备进入过载状态(IPv4)

配置 IS-IS 设备进入过载状态可以使某台 IS-IS 设备暂时从网络中隔离，从而避免造成路由黑洞。

背景信息

当网络中的某些 IS-IS 设备需要升级或维护时，需要暂时将该设备从网络中隔离。配置 IS-IS 设备进入过载状态，可以避免其他设备通过该节点来转发流量。

此外，在部署了 IS-IS 和 BGP 协议的网络中，由于 IS-IS 的收敛速度快于 BGP，因此通过手动配置 IS-IS 设备在启动或重启时进入过载状态，等待一段时间后再取消该标志位（即等待 BGP 也完成收敛），避免造成路由黑洞。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **isis** [*process-id*]，进入 IS-IS 视图。

步骤 3 执行命令 **set-overload** [**on-startup** [*timeout1* | **start-from-nbr** *system-id* [*timeout1* [*timeout2*]] | **wait-for-bgp** [*timeout1*]]] [**allow** { **interlevel** | **external** } *]，设置过载标志位。

----结束

7.5.6 检查配置结果

完成调整 IS-IS 的选路(IPv4)后，可以查看 IS-IS 的路由表信息。

操作步骤

- 使用 **display isis route** [*process-id* | [**vpn-instance** *vpn-instance-name*]] [**ipv4**] [**verbose** | [**level-1** | **level-2**] | *ip-address* [*mask* | *mask-length*]] * [| **count**]命令查看 IS-IS 的路由信息。
- 使用 **display isis lsdb** [{ **level-1** | **level-2** } | **verbose** | { **local** | *lsp-id* | **is-name** *symbolic-name* }] * [*process-id* | **vpn-instance** *vpn-instance-name*]命令查看 IS-IS 的链路状态数据库信息。

----结束

7.6 配置 IS-IS 路由聚合(IPv4)

当大规模部署 IS-IS 网络时，为了避免 IS-IS 路由表中条目过多而降低路由查找速度的现象以及降低管理的复杂度，可以配置路由聚合，减小路由表的规模。

背景信息

路由聚合是指将多条具有相同 IP 前缀的路由聚合成一条路由。

当 IS-IS 网络规模较大时，配置路由聚合，可以有效减少路由表中的条目，减小对系统资源的占用，方便管理。

此外，如果被聚合的 IP 地址范围内的某条链路频繁 Up 和 Down，该变化并不会通告到被聚合的 IP 地址范围外的设备。因此，可以避免网络中的路由振荡，在一定程度上提高了网络的稳定性。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **isis** [*process-id*]，进入 IS-IS 视图。

步骤 3 执行命令 **summary** *ip-address mask* [**avoid-feedback** | **generate_null0_route** | **tag** *tag* | [**level-1** | **level-1-2** | **level-2**]] *，设置 IS-IS 生成聚合路由。

说明

在配置路由聚合后，本地 IS-IS 设备的路由表保持不变。

但是其他 IS-IS 设备的路由表中将只有一条聚合路由，没有具体路由。直到网络中被聚合的路由都出现故障而消失时，该聚合路由才会消失。

----结束

检查配置结果

完成配置后，可以使用以下方式查看聚合路由。

- 使用 **display isis route** 命令查看 IS-IS 路由表中的聚合路由。
- 使用 **display ip routing-table** [**verbose**]命令查看 IP 路由表中的聚合路由。

7.7 配置 IS-IS 与其他路由协议交互(IPv4)

在网络中同时部署了 IS-IS 和其他路由协议时，需要配置 IS-IS 与其他路由协议的路由交互，才能使不同协议的网络正常通信。

7.7.1 建立配置任务

在配置 IS-IS 与其他路由协议交互(IPv4)前了解此特性的应用环境、配置这些特性的前置任务和数据准备，可以帮助您快速、准确地完成配置任务。

应用环境

在网络中同时部署了 IS-IS 和其他路由协议时，会涉及到如下问题：

- IS-IS 路由的优先级问题

  多种路由协议同时发现某一条路由时，如果其他协议的优先级比 IS-IS 的优先级高（例如 OSPF），那么该 IS-IS 路由将不会被优选。

  该问题可以通过配置 IS-IS 路由的优先级来解决。

- IS-IS 域与其他路由域的互通问题

  在网络中同时部署了 IS-IS 和其他路由协议时，IS-IS 路由域与其他协议路由域彼此隔离，不能互通。

  说明

  在同一台设备上运行的多个 IS-IS 进程的 LSDB 是彼此独立的，因此对于该设备上的某个 IS-IS 进程而言，其他 IS-IS 进程的路由也属于外部路由。

为了使 IS-IS 路由域的流量能够被正确转发到其他路由域，需要在有外部路由的设备上（尤其是 IS-IS 路由域的边界设备）进行相关的配置。主要的解决方式有：

- 配置 IS-IS 发布缺省路由。

  该方式配置较为简单，无需 IS-IS 域内的设备了解外部路由。IS-IS 域内去往外部其他路由域的流量都会通过发布缺省路由的设备进行转发。

- 配置 IS-IS 引入外部路由。

  该方式可以使 IS-IS 域内的设备明确获悉到外部路由，可以实现对流量转发的更准确控制。

同样，为了使其他路由域外的流量能够被正确转发到 IS-IS 域内，也必须在其他路由域内执行类似操作，才能使多个路由域能够互通。

前置任务

在调整 IS-IS 的选路(IPv4)之前，需完成以下任务：

- 配置接口的链路层协议。
- 配置接口的网络层地址，使相邻节点网络层可达。
- **配置 IS-IS 的基本功能(IPv4)**。
- 配置其他协议的基本功能。

数据准备

在调整 IS-IS 的选路(IPv4)之前，需要准备以下数据。

序号	数据
1	路由过滤时所采用的 ACL、IP 前缀列表或者路由策略
2	IS-IS 的协议优先级

7.7.2 配置 IS-IS 协议的优先级(IPv4)

当到达同一目的地址有多种协议的路由时，配置 IS-IS 协议的优先级，可以使 IS-IS 路由优选。

背景信息

一台设备同时运行多个路由协议时，可以发现到达同一目的地的多条路由，其中协议优先级高的路由将被优选。

例如，当网络中运行了 OSPF 和 IS-IS 协议，如果两种协议同时发现了到达某一目的网段的路由，那么 OSPF 路由将被优选，因为 OSPF 的协议优先级比 IS-IS 高。

通过配置 IS-IS 协议的优先级，可以将 IS-IS 路由的优先级提高，使 IS-IS 的路由被优选。并且结合路由策略的使用，可以灵活的仅将期望的部分 IS-IS 路由的优先级提高，而不影响其他的路由选择。

操作步骤

- 配置 IS-IS 协议的优先级。
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **isis** [*process-id*]，进入 IS-IS 视图。
  3. 执行命令 **preference** *preference*，配置 IS-IS 协议的优先级。

     说明

     该命令用来设置 IS-IS 协议的优先级。配置值 *preference* 越小，优先级越高。

     缺省情况下，IS-IS 协议的优先级为 15。
- 配置 IS-IS 特定路由的优先级。
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **isis** [*process-id*]，进入 IS-IS 视图。
  3. 执行命令 **preference** *preference* **route-policy** *route-policy-name*，配置 IS-IS 特定路由的优先级。

     说明

     只有通过路由策略的 IS-IS 路由将采用配置值 *preference*。

----结束

7.7.3 配置 IS-IS 发布缺省路由(IPv4)

在具有外部路由的边界 Level-1-2 设备上配置 IS-IS 发布缺省路由，IS-IS 域内去往外部其他路由域的流量都会通过发布缺省路由的设备进行转发。

背景信息

该方式使 Level-1-2 设备在 IS-IS 路由域内发布一条 0.0.0.0/0 的缺省路由。因此，IS-IS 域内的其他设备在转发流量时，将所有去往外部路由域的流量首先转发到该 Level-1-2 设备，然后通过该设备去往外部路由域。

因此，必须保证该设备上有正确的外部路由信息。

说明

配置静态缺省路由也可以实现该功能，但是配置静态缺省路由时需要在大量设备上进行配置，不利于管理。

此外，采用 IS-IS 发布缺省路由的方式更加灵活。例如，如果存在多个边界设备，那么可以通过配置路由策略，使某台边界设备在满足条件时才发布缺省路由，从而避免造成路由黑洞。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **isis** [*process-id*]，进入 IS-IS 视图。

步骤 3 执行命令 **default-route-advertise** [**always** | **match default** | **route-policy** *route-policy-name*] [**cost** *cost* | **tag** *tag* | [**level-1** | **level-1-2** | **level-2**]] * [**avoid-learning**]，配置 IS-IS 发布缺省路由。

----结束

7.7.4 配置 IS-IS 引入外部路由(IPv4)

在边界设备上配置 IS-IS 引入外部路由，可以使 IS-IS 域内的设备明确获悉外部路由，从而指导流量的转发。

背景信息

在边界设备上配置 IS-IS 发布缺省路由，可以将去往域外的流量吸收到该设备来处理。但是由于 IS-IS 域内的其他设备上没有外部路由，因此大量的流量都会被转发到该边界设备，造成该设备的过重负担。

此外，在有多个边界设备时，同样存在去往其他路由域的最优路由的选择问题，那么就必须要求 IS-IS 域内的其他设备获悉全部或部分外部路由。

无论在引入外部路由或将引入的路由发布给其他 IS-IS 设备时，都可通过配置路由策略来控制只引入部分路由或只发布部分引入的路由给其他 IS-IS 设备。

操作步骤

- 配置 IS-IS 引入外部路由。
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **isis** [*process-id*]，进入 IS-IS 视图。
  3. 配置 IS-IS 引入外部路由。
     - 当需要对引入路由的开销进行设置时，执行命令 **import-route** *protocol* [*process-id*] [**cost-type** { **external** | **internal** } | **cost** *cost* | **tag** *tag* | **route-policy** *route-policy-name* | [**level-1** | **level-2** | **level-1-2**]] *，配置 IS-IS 引入外部路由。
     - 当需要保留引入路由的原有开销时，执行命令 **import-route** { { **rip** | **isis** | **ospf** } [*process-id*] | **direct** | **unr** | **bgp** } **inherit-cost** [**tag** *tag* | **route-policy** *route-policy-name* | [**level-1** | **level-2** | **level-1-2**]] *，配置 IS-IS 引入外部路由。此时，引入的源路由协议不能是 **static**。

     说明

     配置引入外部路由后，IS-IS 设备将把引入的外部路由全部发布到 IS-IS 路由域。

     如果不希望将所有引入的外部路由都发布到 IS-IS 路由域，可以使用 **filter-policy export** 来指定发布部分路由到 IS-IS 路由域。
- （可选）配置发布部分外部路由到 IS-IS 路由域。
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **isis** [*process-id*]，进入 IS-IS 视图。
  3. 执行命令 **filter-policy** { *acl-number* | **acl-name** *acl-name* | **ip-prefix** *ip-prefix-name* | **route-policy** *route-policy-name* } **export** [*protocol* [*process-id*]]，配置发布部分外部路由到 IS-IS 路由域。

     说明

     只有通过路由策略的外部路由才能发布到 IS-IS 路由域。

----结束

7.7.5 检查配置结果

配置好 IS-IS 与其他路由协议交互后，可以查看 IS-IS 的路由表和 IP 路由表信息。

操作步骤

- 使用命令 **display isis lsdb** [{ **level-1** | **level-2** } | **verbose** | { **local** | *lsp-id* | **is-name** *symbolic-name* }] * [*process-id* | **vpn-instance** *vpn-instance-name*]查看 IS-IS 的链路状态数据库信息。
- 使用 **display isis route** [*process-id* | [**vpn-instance** *vpn-instance-name*]] [**ipv4**] [**verbose** | [**level-1** | **level-2**] | *ip-address* [*mask* | *mask-length*]] * [| **count**]命令查看 IS-IS 的路由信息。
- 使用 **display ip routing-table ip-prefix** *ip-prefix-name* [**verbose**]命令查看 IP 路由表信息。

----结束

7.8 调整 IS-IS 路由的收敛速度(IPv4)

提高对 IS-IS 网络中故障的响应速度，加快出现网络故障时的路由收敛速度，可以提高 IS-IS 网络的可靠性。

7.8.1 建立配置任务

在调整 IS-IS 路由的收敛速度(IPv4)前了解此特性的应用环境、配置这些特性的前置任务和数据准备，可以帮助您快速、准确地完成配置任务。

应用环境

IS-IS 协议运行的过程包括：

- 邻接关系的建立。通过 Hello 报文建立正确的邻接关系，为 LSP 报文的扩散做准备。
- LSP 报文的扩散。IS-IS 网络中同一区域的所有设备上的 LSDB 达到一致。
- SPF 计算。根据 LSDB 进行 SPF 计算，建立 IS-IS 路由表。

因此，如果需要加快 IS-IS 路由的收敛速度，就需要针对以上三方面来配置：

- 调整邻接故障的检测时间。
- 调整 CSNP 报文和 LSP 报文的扩散。
- 调整 SPF 的计算时间。

此外，IS-IS 还支持 IPv4 路由按优先级收敛，通过调整某些关键路由的收敛优先级，可以使得这些关键路由优先收敛，不影响关键业务的正常运行。

前置任务

调整 IS-IS 路由的收敛速度(IPv4)前，需完成以下任务：

- 配置接口的链路层协议。
- 配置接口的网络层地址，使相邻节点网络层可达。
- **配置 IS-IS 的基本功能(IPv4)**。

数据准备

在调整 IS-IS 的选路(IPv4)之前，需要准备以下数据。

序号	数据
1	Hello 报文的发送时间间隔和邻居保持时间
2	CSNP 报文和 LSP 报文的扩散时间
3	SPF 的计算时间
4	路由收敛优先级

7.8.2 调整邻接故障的检测时间

通过调整 IS-IS 对邻接故障的检测，可以更快的感知到网络中的故障。

背景信息

IS-IS 通过发送 Hello 报文来发现邻居并建立邻接关系，之后会周期性的发送 Hello 报文来维持该邻接关系，并以此方式来检测网络故障。即如果在一定时间（邻居保持时间）内没有收到对方的 Hello 报文，则认为邻居已经 Down，触发 LSP 报文的扩散和 SPF 计算，从而达到 IS-IS 路由的重新收敛。

因此，调整邻接故障的检测时间，可以更快的检测到网络中的故障。主要方式包括：

- 配置 Hello 报文发送间隔。
- 配置邻居保持时间。
- **配置动态 IPv4 BFD for IS-IS**。

  说明

  为了更快速的检测邻接故障，推荐采用配置 IPv4 BFD 的方式。

操作步骤

- 配置 Hello 报文发送间隔
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **interface** *interface-type interface-number*，进入接口视图。
  3. 执行命令 **isis timer hello** *hello-interval* [**level-1** | **level-2**]，设置接口上 Hello 报文发送间隔。

     说明

     在广播网链路上存在 Level-1 和 Level-2 两种 Hello 报文，不同类型的报文可以设置不同的值。如果不指定级别，则默认为 Level-1 和 Level-2 同时配置。

     在点到点链路上，只有一种 Hello 报文，不需要使用参数 Level-1 和 Level-2。
- 配置邻居保持时间
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **interface** *interface-type interface-number*，进入接口视图。
  3. 执行命令 **isis timer holding-multiplier** *number* [**level-1** | **level-2**]，配置邻居保持时间。

说明

在广播网链路上存在 Level-1 和 Level-2 两种 Hello 报文，不同类型的报文可以设置不同的值。如果不指定级别，则默认为 Level-1 和 Level-2 同时配置。

在点到点链路上，只有一种 Hello 报文，不需要使用参数 Level-1 和 Level-2。

----结束

7.8.3 调整 SNP 报文和 LSP 报文的扩散

通过调整 IS-IS 中 SNP 报文和 LSP 报文的扩散，可以加速网络中所有设备的 LSDB 同步。

背景信息

SNP 包括 CSNP（Complete SNP，全序列号报文）和 PSNP（Partial SNP，部分序列号报文）。CSNP 包括 LSDB 中所有 LSP 的摘要信息，从而可以在相邻路由器间保持 LSDB 的同步。在广播网链路和点到点链路中，运行机制略有不同：

- 在广播网链路上，CSNP 由 DIS 设备周期性的发送。当邻居发现 LSDB 不同步时，发送 PSNP 报文来请求缺失的 LSP 报文。
- 在点到点链路上，CSNP 只在第一次建立邻接关系时发送，邻居发送 PSNP 报文来做应答。当邻居发现 LSDB 不同步时，同样发送 PSNP 报文来请求缺失的 LSP 报文。

AR150/200 支持修改 SNP 报文和 LSP 报文的相关参数，可以加速 LSDB 的同步，包括：

- **配置 CSNP 报文的发送间隔**
- **配置 LSP 生成的智能定时器**
- **配置 LSP 的大小**
- **配置 LSP 的刷新周期**
- **配置 LSP 的最大有效时间**
- **配置接口发送 LSP 的最小时间间隔**
- **配置 LSP 快速扩散**
- **配置点到点链路上的 LSP 重传间隔**

操作步骤

- 配置 CSNP 报文的发送间隔
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **interface** *interface-type interface-number*，进入接口视图。
  3. 执行命令 **isis timer csnp** *csnp-interval* [**level-1** | **level-2**]，设置接口上 CSNP 报文发送间隔。

     说明

     参数 **Level-1** 和 **Level-2** 仅在广播网接口时可配置。
- 配置 LSP 生成的智能定时器
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **isis** [*process-id*]，进入 IS-IS 视图。
  3. 执行命令 **timer lsp-generation** *max-interval* [*init-interval* [*incr-interval*]] [**level-1** | **level-2**]，设置 LSP 生成所使用的智能定时器。

如果没有指定 Level，则认为同时设置 Level-1 和 Level-2。

初次产生同一 LSP（或者 LSP 分片）的延迟时间为 *init-interval*；第二次产生同一 LSP（或者 LSP 分片）的延迟时间为 *incr-interval*。随后，每变化一次，延迟时间都增大为前一次的两倍，直到 *max-interval*。稳定在 *max-interval* 三次或者 IS-IS 进程被重启，延迟时间又降回到 *init-interval*。

在不使用 *incr-interval* 的情况下，初次产生同一 LSP（或者 LSP 分片）仍然使用 *init-interval* 作为延迟时间，随后都是使用 *max-interval* 作为延迟时间。同样，稳定在 *max-interval* 三次或者 IS-IS 进程被重启，延迟时间又降回到 *init-interval*。

在只使用 *max-interval* 的情况下，智能定时器退化为一般的一次性触发定时器。

- 配置 LSP 的大小
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **isis** [*process-id*]，进入 IS-IS 视图。
  3. 执行命令 **lsp-length originate** *max-size*，设置生成 LSP 的大小。
  4. 执行命令 **lsp-length receive** *max-size*，设置接收 LSP 的大小。

     说明

     设置 *max-size* 参数时请注意，生成 LSP 的 *max-size* 必须小于等于接收 LSP 的 *max-size*。

     使用 **lsp-length** 命令设置的 *max-size* 值必须满足以下要求，否则接口的 MTU 状态会被认为是 Down。
     - 以太网接口的 MTU 值大于等于 *max-size*+3。
     - P2P 接口的 MTU 值大于等于 *max-size*。
- 配置 LSP 的刷新周期
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **isis** [*process-id*]，进入 IS-IS 视图。
  3. 执行命令 **timer lsp-refresh** *refresh-time*，设置 LSP 刷新周期。

     为了保证整个区域中的 LSP 能够保持同步，IS-IS 周期性发送当前全部 LSP。

     LSP 刷新周期的缺省值为 900 秒，最大有效时间的缺省值为 1200 秒。配置时请注意，必须保证刷新周期比 LSP 的最大有效时间少三百秒以上，使得原有的 LSP 过期之前，新的 LSP 可以到达区域内所有路由器。

     说明

     网络规模越大，LSP 刷新周期与 LSP 最大有效时间之间的差值也越大。
- 配置 LSP 的最大有效时间
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **isis** [*process-id*]，进入 IS-IS 视图。
  3. 执行命令 **timer lsp-max-age** *age-time*，设置 LSP 最大有效时间。

     路由器生成系统 LSP 时，会在 LSP 中填写此 LSP 的最大有效时间。当此 LSP 被其它路由器接收后，它的有效时间会随着时间的变化不断减小。如果路由器一直没有收到更新的 LSP，而此 LSP 的有效时间已减少到 0，LSP 再保持 60 秒，若还未收到新的 LSP，那么此 LSP 将被从 LSDB 中删除。
- 配置接口发送 LSP 的最小时间间隔
  1. 执行命令 **system-view**，进入系统视图。

2. 执行命令 **interface** *interface-type interface-number*，进入接口视图。
3. 执行命令 **isis timer lsp-throttle** *throttle-interval* [**count** *count*]，设置发送 LSP 的最小间隔时间。

   count：用来指定在 *throttle-interval* 时间间隔内发送 LSP 报文的最大包数。整数形式，取值范围是 1 ～ 1000。

- 配置 LSP 快速扩散
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **isis** [*process-id*]，进入 IS-IS 视图。
  3. 执行命令 **flash-flood** [*lsp-count* | **max-timer-interval** *interval* | [**level-1** | **level-2**]] *，使能 LSP 快速扩散。

     使用 **flash-flood** 命令可以加速 LSP 的扩散（Flooding）速度。用户可以通过参数 *lsp-count* 指定每次扩散的 LSP 数量，这个数量是针对所有接口的。如果需要发送的 LSP 的数量大于 *lsp-count*，则就发送 *lsp-count* 个 LSP。如果配置了定时器，在路由计算之前如果这个定时器未超时，则立即扩散；否则在该定时器超时的时候发送。

     配置 LSP 快速扩散时，如果不指定 Level-1 或 Level-2，则默认为 Level-1 和 Level-2 都配置快速扩散。
- 配置点到点链路上的 LSP 重传间隔
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **interface** *interface-type interface-number*，进入接口视图。
  3. （可选）执行命令 **isis circuit-type p2p**，将广播接口模拟为 P2P 接口。
  4. 执行命令 **isis timer lsp-retransmit** *retransmit-interval*，设置 LSP 在点到点链路上的重传间隔。

----结束

7.8.4 调整 SPF 的计算时间

通过调整 SPF 的计算时间，既可以保证 IS-IS 对网络变化的及时响应，又可以减少 SPF 计算对系统资源的过多占用。

背景信息

当网络变化比较频繁时，IS-IS 会频繁的进行 SPF 计算。频繁的 SPF 计算会消耗系统大量的 CPU 资源，从而影响正常业务的运行。

配置智能定时器的优势在于当刚开始进行 SPF 计算时，两次计算的间隔时间较小，保证 IS-IS 路由的收敛速度。之后随着整个 IS-IS 网络的拓扑趋于稳定时，则应该适当延长两次 SPF 计算的间隔时间，从而减少不必要的计算。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **isis** [*process-id*]，进入 IS-IS 视图。

步骤 3 执行命令 **timer spf** *max-interval* [*init-interval* [*incr-interval*]]，设置 SPF 智能定时器。

智能定时器的变化规律如下：

- 初次进行 SPF 计算的延迟时间为 *init-interval*；第二次进行 SPF 计算的延迟时间为 *incr-interval*。随后，每变化一次，SPF 计算的延迟时间增大为前一次的两倍，直到 *max-interval*。稳定在 *max-interval* 三次或者 IS-IS 进程被重启，延迟时间又降回到 *init-interval*。
- 在不使用 *incr-interval* 的情况下，初次进行 SPF 计算用 *init-interval* 作为延迟时间，随后都是使用 *max-interval* 作为延迟时间。稳定在 *max-interval* 三次或者 IS-IS 进程被重启，延迟时间又降回到 *init-interval*。
- 在只使用 *max-interval* 的情况下，智能定时器退化为一般的一次性触发定时器。

----结束

7.8.5 配置 IS-IS 路由按优先级收敛(IPv4)

将 IS-IS 网络中的关键路由配置为较高的收敛优先级，保证网络拓扑变化时关键路由的优先收敛，减小对重要业务的影响。

背景信息

缺省情况下，IS-IS 32 位主机路由的收敛优先级为 **medium**，其他 IS-IS 路由的收敛优先级为 **low**。

AR150/200 支持通过配置 IS-IS 路由的收敛优先级，使某些重要路由在网络拓扑发生变化时优先收敛。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **isis** [*process-id*]，进入 IS-IS 视图。

步骤 3 执行命令 **prefix-priority** [**level-1** | **level-2**] { **critical** | **high** | **medium** } { **ip-prefix** *prefix-name* | **tag** *tag-value* }，配置 IS-IS 路由的收敛优先级。

IS-IS 路由收敛优先级的应用规律。

- 对于已存在的 IS-IS 路由，收敛优先级将依据 **prefix-priority** 命令重新进行设置。
- 对新增加的 IS-IS 路由，收敛优先级将依据 **prefix-priority** 命令的过滤结果进行设置。
- 如果一条路由符合多个收敛优先级的匹配规则，则这些收敛优先级中最高者当选为路由的收敛优先级。
- Level-1 IS-IS 路由的收敛优先级高于 Level-2 IS-IS 路由的收敛优先级。
- 若不指定 Level，IS-IS 会对 Level-1 和 Level-2 的 IS-IS 路由都进行配置。

说明

prefix-priority 命令仅在公网生效。

如果用 **prefix-priority** 命令对 IS-IS 路由（除了 IS-IS 32 位主机路由）的收敛优先级进行配置后，IS-IS 32 位主机路由的缺省收敛优先级将从 medium 变为 low，其他 IS-IS 路由的收敛优先级依据 **prefix-priority** 命令的配置而变化。

步骤 4 （可选）执行命令 **quit** 返回系统视图。

步骤 5 （可选）执行命令 **ip route prefix-priority-scheduler** *critical-weight high-weight medium-weight low-weight*，配置 IPv4 路由按优先级调度的比例。

缺省情况下，IPv4 路由按优先级调度的比例为 8:4:2:1。

----结束

7.8.6 检查配置结果

配置好各种影响 IS-IS 路由收敛速度的参数后，可以查看接口发送 IS-IS 报文的各种参数。

操作步骤

- 使用 **display isis interface** [**verbose**] [*process-id* | **vpn-instance** *vpn-instance-name*]命令查看 IS-IS 接口发送的 IS-IS 报文的信息。
- 使用 **display isis route** [*process-id* | **vpn-instance** *vpn-instance-name*] [**ipv4**] [**verbose** | [**level-1** | **level-2**] | *ip-address* [*mask* | *mask-length*]] * [| **count**]命令查看 IS-IS 路由的优先级信息。

----结束

7.9 配置静态 IPv4 BFD for IS-IS

BFD 能够提供轻负荷、快速（毫秒级）的通道故障检测，配置静态 IPv4 BFD for IS-IS 是实现 BFD 检测功能的一种方式。

背景信息

配置静态 BFD 会话需要通过命令行手工配置 BFD 单跳检测，包括配置本地标识符和远端标识符等，然后手工下发 BFD 会话建立请求。

静态 BFD 的缺点在于建立和删除 BFD 会话时都需要手工触发，缺乏灵活性。而且有可能造成人为的配置错误。例如，如果配置了错误的本地标识符或者远端标识符时，BFD 会话将不能正常工作。

说明

目前，BFD 会话不会感知路由切换。如果绑定的对端 IP 地址改变引起路由切换到其他链路上，除非原链路转发不通，否则，BFD 不会重新协商。

前置任务

配置静态 IPv4 BFD for IS-IS 的前置任务:

- 配置接口的网络层地址，使相邻节点网络层可达。
- **配置 IS-IS 的基本功能(IPv4)**

数据准备

为完成此配置举例，需准备如下的数据：

序号	数据
1	启用 BFD 特性的接口的类型和编号

操作步骤

- 使能全局 BFD
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **bfd**，使能全局 BFD 能力。
  3. 执行命令 **quit**，返回系统视图。
- 配置 BFD 单跳检测
  1. 执行命令 **bfd** *cfg-name* **bind peer-ip** *ip-address* [**interface** *interface-type interface-number*]，创建 BFD 绑定。

     指定了对端 IP 和本端接口，表示检测单跳链路，即检测以该接口为出接口、以 **peer-ip** 为下一跳地址的一条固定路由。
  2. 配置标识符：
     - 执行命令 **discriminator local** *discr-value*，配置本地标识符。
     - 执行命令 **discriminator remote** *discr-value*，配置远端标识符。

     BFD 会话两端设备的本地标识符和远端标识符需要分别对应，否则会话无法正确建立。并且，本地标识符和远端标识符配置成功后不可修改。

     说明

     本地标识符 **local** *discr-value* 对应对端设备的远端标识符 **remote** *discr-value*，本地的远端标识符 **remote** *discr-value* 对应对端设备的本地标识符 **local** *discr-value*。
  3. 执行命令 **commit**，提交配置。
  4. 执行命令 **quit**，返回系统视图。
- 使能接口静态 IPv4 BFD
  1. 执行命令 **interface** *interface-type interface-number*，进入指定接口的接口视图。
  2. 执行命令 **isis bfd static**，使能接口静态 IPv4 BFD。

----结束

检查配置结果

只有配置完 BFD 会话参数并成功建立会话后，才能查看到 BFD 会话信息。

执行命令 **display isis interface verbose** 可以看到，IS-IS 进程 1 的静态 BFD 的状态为 Yes。

7.10 配置动态 IPv4 BFD for IS-IS

如果对数据传输有较高要求，需要提高链路状态变化时 IS-IS 的收敛速度，可以在运行 IS-IS 的链路上配置动态 IPv4 BFD。

背景信息

通常情况下，IS-IS 通过周期性的发送和接收 Hello 报文来保持邻居关系，而 Hello 报文的发送时间间隔最小为 3s，并且至少需要在 3 个周期内未收到对端的 Hello 报文的情况下，才宣告邻居 Down。因此 IS-IS 通过 Hello 报文感知邻居 Down 的时间为秒级，如果网络中部署了高速数据业务，在此期间将导致数据大量丢失。

BFD 能够提供毫秒级别的故障监测时间，及时检测到被保护的链路或节点故障，并上报给 IS-IS 协议，从而实现 IS-IS 路由的快速收敛。

动态 IPv4 BFD for IS-IS 由 IS-IS 协议动态触发建立 BFD 会话，即 IS-IS 在建立邻居关系时，将邻居的参数及检测参数（包括目的地址、源地址等）通告给 BFD，BFD 根据收到的参数建立起会话。动态 BFD 比静态 BFD 更具有灵活性。

说明

目前，BFD 会话不会感知路由切换。如果绑定的对端 IP 地址改变引起路由切换到其他链路上，除非原链路转发不通，否则，BFD 不会重新协商。

前置任务

配置动态 IPv4 BFD for IS-IS 的前置任务：

- 配置接口的网络层地址，使相邻节点网络层可达。
- 配置 IS-IS 的基本功能。

数据准备

为完成此配置举例，需准备如下数据：

序号	数据
1	启用 BFD 特性的 IS-IS 进程号
2	启用 BFD 特性的接口的类型和编号
3	BFD 会话的参数值

使能动态 IPv4 BFD for IS-IS 有两种方式：

- **使能指定 IS-IS 进程下动态 IPv4 BFD**，当该节点大部分 IS-IS 接口需要使能 IPv4 BFD for IS-IS 时，建议选择此方式。
- **使能指定接口下动态 IPv4 BFD**，当该节点只有小部分 IS-IS 接口需要使能 IPv4 BFD for IS-IS 时，建议选择此方式。

操作步骤

- 使能指定 IS-IS 进程下动态 IPv4 BFD
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **bfd**，使能全局 BFD 能力。
  3. 执行命令 **quit**，返回系统视图。
  4. 执行命令 **isis** *process-id*，进入 IS-IS 视图。
  5. 执行命令 **bfd all-interfaces enable**，打开 IS-IS 进程 BFD 特性的开关，建立 BFD 会话。

     当配置了全局 BFD 特性，且邻居状态为 Up 时，IS-IS 为该进程下所有满足上述条件的接口使用缺省的 BFD 参数值建立 BFD 会话。

6. （可选）执行命令 **bfd all-interfaces** { **min-rx-interval** *receive-interval* | **min-tx-interval** *transmit-interval* | **detect-multiplier** *multiplier-value* } *，配置 BFD 参数，指定用于建立 BFD 会话的各个参数值。

   执行该命令后，所有 IS-IS 接口建立 BFD 会话的参数都会改变。

7. 执行命令 **quit**，返回系统视图。

   如果需要阻止某些接口的 BFD 功能，则需要在指定的接口视图下执行命令 **isis bfd block**，阻止接口动态创建 BFD 会话。

- 使能指定接口下动态 IPv4 BFD
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **bfd**，使能全局 BFD 能力。
  3. 执行命令 **quit**，返回系统视图。
  4. 执行命令 **interface** *interface-type interface-number*，进入接口视图。
  5. 执行命令 **isis bfd enable**，打开接口 BFD 特性的开关，建立 BFD 会话。

     当配置了全局 BFD 特性，且邻居状态为 Up（广播网中 DIS Up）时，则使用缺省的 BFD 参数值建立 BFD 会话。

  6. （可选）如果需要单独配置 BFD 参数请执行命令 **isis bfd** { **min-rx-interval** *receive-interval* | **min-tx-interval** *transmit-interval* | **detect-multiplier** *multiplier-value* } *，配置 BFD 参数，指定用于建立 BFD 会话的各个参数值。

     说明

     接口上配置 BFD 特性的优先级高于进程中配置 BFD 特性的优先级。即打开接口 BFD 特性的开关，建立接口上 BFD 会话的参数以接口上的配置为准。

----结束

检查配置结果

当链路两端均使能 BFD 特性后，执行命令 **display isis** [*process-id* | **vpn-instance** *vpn-instance-name*] **bfd session** { **all** | **peer** *ip-address* | **interface** *interface-type interface-number* }，可以看到使能了 BFD 的状态为 Up。

7.11 配置 IS-IS 的基本功能(IPv6)

配置 IS-IS 的基本功能(IPv6)可以实现基于 IPv6 地址族的 IS-IS 网络中各节点的互通。配置步骤主要包括配置 IS-IS 进程和配置 IS-IS 接口。

7.11.1 建立配置任务

在配置 IS-IS 基本功能(IPv6)前了解此特性的应用环境、配置此特性的前置任务和数据准备，可以帮助您快速、准确地完成配置任务。

应用环境

在 IPv6 网络中部署 IS-IS 协议时，首先需要配置 IS-IS 的基本功能，实现网络中节点的路由互通。

只有完成基本功能的配置，才能配置其他 IS-IS 特性。

配置 IS-IS 的基本功能(IPv6)的步骤主要包括：

1. 创建 IS-IS 进程(IPv6)。
2. 使能 IS-IS 接口(IPv6)。

前置任务

在配置 IS-IS 的基本功能(IPv6)之前，需完成以下任务：

- 配置链路层协议。
- 配置接口的 IPv6 地址，使相邻节点网络层可达。
- 在系统视图下使能路由器的 IPv6 转发能力。

数据准备

在配置 IS-IS 的基本功能(IPv6)之前，需要准备以下数据。

序号	数据
1	IS-IS 进程号
2	网络实体名称
3	设备及接口的 Level 级别

7.11.2 创建 IS-IS 进程(IPv6)

配置 IS-IS 的基本功能（IPv6）首先需要创建 IS-IS 进程（IPv6），然后才能使能 IS-IS 接口（IPv6）。

背景信息

IS-IS 进程的配置包括：

- **创建 IS-IS 进程、配置设备的 NET 及使能 IS-IS 进程的 IPv6 能力**
- **（可选）配置设备的 Level 级别**

  缺省情况下，设备的 Level 级别为 **level-1-2**。

  建议根据网络规划的需要，配置设备的 Level 级别。否则，IS-IS 会为 Level-1 和 Level-2 分别建立邻居，维护两份相同的 LSDB，造成对设备资源的过多占用。
- **（可选）配置 IS-IS 主机名映射**

  配置 IS-IS 主机名映射后，使用显示命令查看 IS-IS 的相关信息时，会以配置的动态名称代替设备的 System ID，从而提高 IS-IS 网络的可维护性。
- **（可选）打开 IS-IS 的邻接状态开关**

  在本地 terminal monitor 开关已开启的情况下，当打开邻接状态输出开关后，IS-IS 邻接状态的变化会输出到配置终端上，直至邻接状态输出开关被关闭。

操作步骤

- 创建 IS-IS 进程，并配置设备的 NET，使能 IS-IS 进程的 IPv6 能力。

1. 执行命令 **system-view**，进入系统视图。
2. 执行命令 **isis** [*process-id*]，创建 IS-IS 进程，进入 IS-IS 视图。

   参数 *process-id* 用来指定一个 IS-IS 进程。如果不指定参数 *process-id*，则系统默认的进程为 1。
3. 执行命令 **network-entity** *net*，设置网络实体名称。

   注意

   建议将 Loopback 接口的地址转化为 NET，保证 NET 在网络中的唯一性。如果网络中的 NET 不唯一，容易引发路由振荡，因此要做好前期网络规划。

   IS-IS 在建立 Level-2 邻居时，不检查区域地址是否相同，而在建立 Level-1 邻居时，区域地址必须相同，否则无法建立邻居。
4. 执行命令 **ipv6 enable**，使能 IS-IS 进程的 IPv6 能力。

- （可选）配置设备的 Level 级别
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **isis** [*process-id*]，创建 IS-IS 进程，进入 IS-IS 视图。
  3. 执行命令 **is-level** { **level-1** | **level-1-2** | **level-2** }，设置路由器的 Level 级别。
- （可选）配置 IS-IS 主机名映射
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **isis** [*process-id*]，创建 IS-IS 进程，进入 IS-IS 视图。
  3. 执行命令 **is-name** *symbolic-name*，配置 IS-IS 动态主机名映射，为本地设备配置主机名称。

     该配置属于动态配置，即配置的主机名称 *symbolic-name* 以 LSP 报文的形式发布给区域中的其它 IS-IS 设备。

     在其他设备上使用 IS-IS 相关显示命令查看 IS-IS 信息时，系统 ID 将被 *symbolic-name* 代替。
  4. 执行命令 **is-name map** *system-id symbolic-name*，配置 IS-IS 静态主机名映射，为远端 IS-IS 设备配置主机名称。

     该配置属于静态配置，即只在本地设备生效，配置的主机名称 *symbolic-name* 不会通过 LSP 报文发送出去。

     因此，如果网络中的对应的 IS-IS 设备配置了动态主机名映射，那么该映射关系将覆盖本地路由器的静态映射。
- （可选）打开 IS-IS 的邻接状态开关
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **isis** [*process-id*]，创建 IS-IS 进程，进入 IS-IS 视图。
  3. 执行命令 **log-peer-change**，打开邻接状态输出开关。

----结束

7.11.3 使能 IS-IS 接口(IPv6)

只有在相应的接口下使能 IS-IS，IS-IS 才能通过该接口发送 Hello 报文建立邻居、扩散 LSP 报文。

背景信息

IS-IS 设备的 Level 级别和接口的 Level 级别共同决定了建立邻居关系的 Level 级别。两台 Level-1-2 设备建立邻居关系时，缺省情况下，会分别建立 Level-1 和 Level-2 邻居关系。如果只希望建立 Level-1 或者 Level-2 的邻居关系，可以通过修改接口的 Level 级别实现。

接口下使能 IS-IS 后，该接口会主动发送 Hello 报文尝试与对端建立邻居。如果对端不是 IS-IS 设备，或者只是希望将该接口所在网段的路由发布出去，并不希望通过该接口建立邻居，可以配置抑制该接口。配置后，该接口所在网段的路由仍然可以被发布出去，且并不发送 Hello 报文，减少对链路带宽的占用。

操作步骤

- 使能 IS-IS 接口
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **interface** *interface-type interface-number*，进入接口视图。
  3. 执行命令 **ipv6 enable**，使能指定接口的 IPv6 能力。
  4. 执行命令 **isis ipv6 enable** [*process-id*]，使能指定接口 IS-IS 的 IPv6 能力。

     配置该命令后，IS-IS 将通过该接口建立邻居、扩散 LSP 报文。

     说明

     由于 Loopback 接口不需要建立邻居，因此如果在 Loopback 接口下使能 IS-IS，只会将该接口所在的网段路由通过其他 IS-IS 接口发布出去。
- （可选）配置 IS-IS 接口的 Level 级别
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **interface** *interface-type interface-number*，进入接口视图。
  3. 执行命令 **isis circuit-level** [**level-1** | **level-1-2** | **level-2**]，设置接口的 Level 级别。

     缺省情况下，接口的 Level 级别为 **level-1-2**。

     说明

     只有当 IS-IS 设备的 Level 级别为 Level-1-2 时，改变接口的 Level 级别才有意义，否则将由 IS-IS 设备的 Level 级别决定所能建立的邻接关系层次。
- （可选）配置 IS-IS 接口为抑制状态
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **interface** *interface-type interface-number*，进入接口视图。
  3. 执行命令 **isis silent**，配置 IS-IS 接口为抑制状态。

     IS-IS 接口为抑制状态时，此接口不再接收或发送 IS-IS 报文，但接口所在网段的路由仍可以被发布到域内的其他 IS-IS 设备。

----结束

7.11.4 (可选)配置 IS-IS 接口的开销(IPv6)

配置 IS-IS 的接口开销可以控制 IS-IS 的路由选择，请根据网络规划适当配置接口的开销。

背景信息

IS-IS 有三种方式来确定接口的开销，按照优先级由高到低分别是：

- 接口开销：为单个接口设置开销。
- 全局开销：为所有接口设置开销。
- 自动计算开销：根据接口带宽自动计算开销。

如果没有显式的配置任何命令，则 IS-IS 接口的默认开销为 10，开销类型是 narrow。

操作步骤

- 配置 IS-IS 开销的类型
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **isis** [*process-id*]，进入 IS-IS 视图。
  3. 执行命令 **cost-style** { **narrow** | **wide** | **wide-compatible** | { { **narrow-compatible** | **compatible** } [**relax-spf-limit**] } }，设置 IS-IS 开销的类型。

  对于不同的开销类型，其接口开销的取值范围有所不同，接收到的路由开销取值范围也有所不同。

  - narrow 类型：接口开销取值范围为 1 ～ 63。接收到的路由开销值最大为 1023。
  - narrow-compatible 和 compatible 类型：接口开销取值范围为 1 ～ 63。接收到的路由开销值和参数 **relax-spf-limit** 有关。
    - 不设置 **relax-spf-limit** 参数：

      如果路由开销值小于等于 1023，且该路由经过的所有接口的开销值都小于等于 63：这条路由的开销值按照实际值接收。

      如果路由开销值小于等于 1023，但该路由经过的所有接口中有的接口链路开销值大于 63：IS-IS 设备只能学到该接口所在网段的路由和接口所引入的路由，这条路由的开销值按照实际值接收，之后通过此接口转发的路由将被丢弃。

      如果路由开销值大于 1023：IS-IS 设备只能学到路由开销值第一次超过 1023 的那个接口（该接口之前的所有接口的链路开销值小于等于 63）的路由。该接口所在网段的路由和引入的路由均可以被学习到，路由的开销值按照 1023 接收，之后通过此接口转发的路由将被丢弃。
    - 设置 **relax-spf-limit** 参数：

      对接口的链路开销值和路由开销值均没有限制，按照实际的路由开销值正常接收该路由。
  - wide 和 wide-compatible 类型：接口开销取值范围是 1 ～ 16777215。配置为最大值 16777215 时，该链路上生成的邻居 TLV（cost 为 16777215）不能用于路由计算，仅用于传递 TE 相关信息。接收到的路由开销值最大为 0xFFFFFFFF。
- 配置 IS-IS 接口的开销
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **interface** *interface-type interface-number*，进入接口视图。

3. 执行命令 **isis ipv6 cost** *cost* [**level-1** | **level-2**]，设置 IS-IS 接口的开销。

   使用此命令可以单独为某接口配置开销值。

- 配置 IS-IS 的全局开销
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **isis** [*process-id*]，进入 IS-IS 视图。
  3. 执行命令 **ipv6 circuit-cost** *cost* [**level-1** | **level-2**]，设置 IS-IS 全局开销。

     使用此命令可以一次性改变所有接口的开销值。
- 使能 IS-IS 自动计算接口的开销
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **isis** [*process-id*]，进入 IS-IS 视图。
  3. 执行命令 **ipv6 bandwidth-reference** *value*，配置计算带宽的参考值。缺省情况下，带宽参考值为 100，单位是 Mbit/s。
  4. 执行命令 **ipv6 auto-cost enable**，使能自动计算接口的开销值。

  只有当开销类型为 wide 或 wide-compatible 时，步骤 3 配置的带宽参考值才是有效的，此时各接口的开销值=(bandwidth-reference/接口带宽值)×10。

  当开销类型为 narrow、narrow-compatible 或 compatible 时，各个接口的开销值根据表 7-2 来确定。

表 7-2 IS-IS 接口开销和带宽对应关系表

开销值	接口带宽范围
60	接口带宽≤10Mbit/s
50	10Mbit/s<接口带宽≤100Mbit/s
40	100Mbit/s<接口带宽≤155Mbit/s
30	155Mbit/s<接口带宽≤622Mbit/s
20	622Mbit/s<接口带宽≤2.5Gbit/s
10	2.5Gbit/s<接口带宽

说明

要改变 Loopback 接口的开销，只能在接口视图下使用 **isis ipv6 cost** 命令配置。

----结束

7.11.5 (可选)配置不同网络类型接口的 IS-IS 属性(IPv6)

针对不同网络类型的接口，可以配置不同的 IS-IS 属性。

背景信息

由于 IS-IS 在广播网中和 P2P 网络中建立邻居的方式不同，因此，针对不同类型的接口，可以配置不同的 IS-IS 属性。

在广播网中，IS-IS 需要选择 DIS，因此通过配置 IS-IS 接口的 DIS 优先级，可以使拥有接口优先级最高的设备优选为 DIS。

链路两端的 IS-IS 接口的网络类型必须一致，否则双方不可以建立起邻居关系。如果对端设备的接口类型为 P2P 接口，可以将本地设备的广播网接口改为 P2P 接口，以满足与对端建立邻居的需要。

在 P2P 网络中，IS-IS 不需要选择 DIS，因此无需配置接口的 DIS 优先级。但是为了保证 P2P 链路的可靠性，可以配置 IS-IS 使用 P2P 接口在建立邻居时采用 3-way 模式，以检测单向链路故障。

操作步骤

- 配置接口的 DIS 优先级
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **interface** *interface-type interface-number*，进入接口视图。
  3. 执行命令 **isis dis-priority** *priority* [**level-1** | **level-2**]，设置用来选举 DIS 的优先级，数值越大优先级越高。
- 配置 IS-IS 接口的网络类型
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **interface** *interface-type interface-number*，进入接口视图。
  3. 执行命令 **isis circuit-type p2p**，设置接口的网络类型为 P2P。

     缺省情况下，接口网络类型根据物理接口决定。

     在使能 IS-IS 的接口上，当接口网络类型发生改变时，相关配置发生改变，具体如下。
     - 使用 **isis circuit-type p2p** 命令将广播网接口模拟成 P2P 接口时，接口发送 Hello 报文的间隔时间、宣告邻居失效前 IS-IS 没有收到的邻居 Hello 报文数目、点到点链路上 LSP 报文的重传间隔时间以及 IS-IS 各种认证均恢复为缺省配置，而 DIS 优先级、DIS 名称、广播网络上发送 CSNP 报文的间隔时间等配置均失效。
     - 使用 **undo isis circuit-type** 命令恢复接口的网络类型时，接口发送 Hello 报文的间隔时间、宣告邻居失效前 IS-IS 没有收到的邻居 Hello 报文数目、点到点链路上 LSP 报文的重传间隔时间、IS-IS 各种认证、DIS 优先级和广播网络上发送 CSNP 报文的间隔时间均恢复为缺省配置。
- 配置 P2P 链路中的邻居建立协商模型
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **interface** *interface-type interface-number*，进入接口视图。
  3. 执行命令 **isis ppp-negotiation** { **2-way** | **3-way** [**only**] }，指定接口使用的协商模型。

     缺省情况下，使用 **3-way** 协商模式。

     此命令只适用于 P2P 链路上建立邻居。对于广播链路，可以通过命令 **isis circuit-type p2p** 更改链路类型为 P2P，然后使用此命令配置邻居建立方式。
- 配置 PPP 链路协议接口的 OSICP 协商检查
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **interface** *interface-type interface-number*，进入接口视图。

3. 执行命令 **isis ppp-osicp-check**，指定 PPP 链路接口进行 OSICP 状态检查。

   缺省情况下，PPP 链路协议的 OSICP 状态不影响 IS-IS 接口的状态。

   此命令只适用于 PPP 链路协议的接口，对于运行其他链路协议的点对点接口，配置命令无效。

   配置此命令后，PPP 链路协议的 OSI 网络协商状态会影响 IS-IS 接口状态。当 PPP 协议感知 OSI 网络不通时，IS-IS 接口的链路状态将会被设为 Down，到接口网段的路由就不会在 LSP 中发布。

- 配置 IS-IS 对接收的 Hello 报文不作 IP 地址检查
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **interface** *interface-type interface-number*，进入接口视图。
  3. 执行命令 **isis peer-ip-ignore**，配置对接收的 Hello 报文不作 IP 地址检查。

----结束

7.11.6 检查配置结果

配置好 IS-IS 的基本功能(IPv6)后，可以查看 IS-IS 的邻居、接口和路由信息。

前提条件

已经完成 IS-IS 的基本功能(IPv6)的所有配置。

操作步骤

步骤 1 使用 **display isis name-table** [*process-id*]命令查看本地设备名称到系统 ID 的映射关系表。

步骤 2 使用 **display isis peer** [**verbose**] [*process-id*]命令查看 IS-IS 的邻居信息。

步骤 3 使用 **display isis interface** [**verbose**] [*process-id*]命令查看使能了 IS-IS 的接口信息。

步骤 4 使用 **display isis route** [*process-id* | **vpn-instance** *vpn-instance-name*] **ipv6** [**verbose** | [**level-1** | **level-2**] | *ipv6-address* [*prefix-length*]] * [| **count**]命令查看 IS-IS 的路由信息。

----结束

7.12 调整 IS-IS 的选路(IPv6)

通过调整 IS-IS 选路，可以实现对路由选择的精确控制。

7.12.1 建立配置任务

在调整 IS-IS 的选路(IPv6)前了解此特性的应用环境、配置这些特性的前置任务和数据准备，可以帮助您快速、准确地完成配置任务。

应用环境

配置 IS-IS 的基本功能(IPv6)后，可以得到 IS-IS 路由表，实现了网络中各节点的互通。

但是，如果网络中存在多条冗余链路，那么此时 IS-IS 路由表中的路由可能不是期望的最优路由，不能满足网络规划和流量管理的需要。为了达到优化 IS-IS 网络和便于流量管理的目的，需要对网络中的路由进行更加精确的控制。

调整 IS-IS 选路的方式主要有如下几种：

- **配置 IS-IS 接口的开销(IPv6)**。

  说明

  虽然通过修改 IS-IS 接口的开销，可以在一定程度上达到控制选路的目的。但是，该方式有较大的局限性。因为在网络调整时（尤其是大规模网络），为了达到让某条路由优选而修改了接口开销，将会影响到该接口涉及的所有路由的重新计算和收敛，配置后的结果可能会与期望的不一致。

  因此，接口开销的调整尽量在配置 IS-IS 的基本功能时完成。
- 配置 IS-IS 路由渗透(IPv6)。
- 配置 IS-IS 对等价路由的处理(IPv6)。
- 控制将 IS-IS 路由下发到 IP 路由表(IPv6)。
- 配置 IS-IS 设备进入过载状态(IPv6)。

前置任务

调整 IS-IS 的选路(IPv6)前，需完成以下任务：

- 配置接口的链路层协议。
- 配置接口的网络层地址，使相邻节点网络层可达。
- **配置 IS-IS 的基本功能(IPv6)**。

数据准备

在调整 IS-IS 的选路(IPv6)之前，需要准备以下数据。

序号	数据
1	路由过滤时所采用的 ACL6、IPv6 前缀列表或者路由策略
2	负载分担时采用的最大等价路由条数
3	IS-IS 设备进入过载状态的时间

7.12.2 配置 IS-IS 路由渗透(IPv6)

在双 Level 组网中，配置 IS-IS 路由渗透可以改变 Level 之间的路由渗透方式，实现对 IS-IS 路由的控制。

背景信息

如果在一个 Level-1 区域中有多台 Level-1-2 设备与 Level-2 区域相连，每台 Level-1-2 设备都会在 Level-1 LSP 中设置 ATT 标志位，则该区域中就有到达 Level-2 区域和其他 Level-1 区域的多条出口路由。

缺省情况下，Level-1 区域的路由会渗透到 Level-2 区域中，因此 Level-1-2 设备和 Level-2 设备了解整个网络的拓扑信息。由于 Level-1 区域的设备只维护本地 Level-1 区域的 LSDB

数据库，不知道整个网络的拓扑信息，所以只能选择将流量转发到最近的 Level-1-2 设备，再由 Level-1-2 设备将流量转发到 Level-2 区域。然而，该路由可能不是到达目的地的最优路由。

为了帮助 Level-1 区域内的设备选择到达其他区域的最优路由，可以配置 IPv4 IS-IS 路由渗透，将 Level-2 区域的某些路由渗透到本地 Level-1 区域。

另外，考虑到网络中部署的某些业务可能只在本地 Level-1 区域内运行，则无需将这些路由渗透到 Level-2 区域中，可以通过配置策略仅将部分 Level-1 区域的路由渗透到 Level-2 区域。

操作步骤

- 配置 Level-2 区域的路由渗透到 Level-1 区域。
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **isis** [*process-id*]，进入 IS-IS 视图。
  3. 执行命令 **ipv6 import-route isis level-2 into level-1** [**tag** *tag* | **filter-policy** { **ipv6-prefix** *ipv6-prefix-name* | **route-policy** *route-policy-name* }] *，将 Level-2 区域和其他 Level-1 区域的某些路由渗透到本地 Level-1 区域。

     说明

     该命令配置在与外部区域相连的 Level-1-2 设备上。

     缺省情况下，Level-2 区域的路由信息不渗透到 Level-1 区域。配置该命令后，通过过滤策略的路由将渗透到 Level-1 区域中。
- 配置 Level-1 区域的路由渗透到 Level-2 区域。
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **isis** [*process-id*]，进入 IS-IS 视图。
  3. 执行命令 **ipv6 import-route isis level-1 into level-2** [**tag** *tag* | **filter-policy** { **ipv6-prefix** *ipv6-prefix-name* | **route-policy** *route-policy-name* }] *，将 Level-1 区域的某些路由渗透到本地 Level-2 区域。

     说明

     该命令配置在与外部区域相连的 Level-1-2 设备上。

     缺省情况下，Level-1 区域的路由信息全部渗透到 Level-2 区域。配置该命令后，只有通过过滤策略的路由才能渗透到 Level-2 区域中。

----结束

7.12.3 配置 IS-IS 对等价路由的处理方式(IPv6)

当 IS-IS 网络中有多条 IPv6 等价路由时，可以通过配置负载分担以提高每条链路的利用率。

背景信息

当 IS-IS 网络中有多条 IPv6 冗余链路时，可能会出现多条 IPv6 等价路由，此时配置负载分担。流量会被均匀的分配到每条链路上。

该方式可以提高网络中链路的利用率及减少某些链路负担过重造成阻塞发生的情况。但是由于对流量转发具有一定的随机性，因此可能不利于对业务流量的管理。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **isis** [*process-id*]，进入 IS-IS 视图。

步骤 3 执行命令 **ipv6 maximum load-balancing** *number*，配置在负载分担方式下的等价路由的最大数量。

说明

如果命令中指定的 *number* 小于网络中存在的等价路由数量时，则 IS-IS 会从所有等价路由中选取 *number* 条路由进行负载分担。

----结束

7.12.4 控制将 IS-IS 路由下发到 IP 路由表(IPv6)

当不希望某些 IS-IS 路由被优选时，可以通过策略阻止将部分 IS-IS 路由下发到 IP 路由表来实现。

背景信息

IP 报文是根据 IP 路由表来进行转发的。IS-IS 路由表中的路由条目需要被成功下发到 IP 路由表中，该路由条目才生效。

因此，可以通过配置 IPv6-Prefix、路由策略等方式，只允许匹配的 IS-IS 路由下发到 IP 路由表中。不匹配的 IS-IS 路由将会被阻止进入 IP 路由表，更不会被优选。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **isis** [*process-id*]，进入 IS-IS 视图。

步骤 3 执行命令 **ipv6 filter-policy** { **ipv6-prefix** *ipv6-prefix-name* | **route-policy** *route-policy-name* } **import**，控制将部分 IS-IS 路由下发到 IP 路由表。

----结束

7.12.5 配置 IS-IS 设备进入过载状态(IPv6)

配置 IS-IS 设备进入过载状态可以使某台 IS-IS 设备暂时从网络中隔离，从而避免造成路由黑洞。

背景信息

当网络中的某些 IS-IS 设备需要升级或维护时，需要暂时将该设备从网络中隔离。配置 IS-IS 设备进入过载状态，可以避免其他设备通过该节点来转发流量。

此外，在部署了 IS-IS 和 BGP 协议的网络中，由于 IS-IS 的收敛速度快于 BGP，因此通过手动配置 IS-IS 设备在启动或重启时进入过载状态，等待一段时间后再取消该标志位（即等待 BGP 也完成收敛），避免造成路由黑洞。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **isis** [*process-id*]，进入 IS-IS 视图。

步骤 3 执行命令 **set-overload** [**on-startup** [*timeout1* | **start-from-nbr** *system-id* [*timeout1* [*timeout2*]] | **wait-for-bgp** [*timeout1*]]] [**allow** { **interlevel** | **external** } *]，设置过载标志位。

----结束

7.12.6 检查配置结果

完成调整 IS-IS 的选路(IPv6)后，可以查看 IS-IS 的路由表信息。

操作步骤

- 使用 **display isis route** [*process-id* | **vpn-instance** *vpn-instance-name*] [**ipv6**] [**verbose** | [**level-1** | **level-2**] | *ipv6-address* [*prefix-length*]] * [| **count**]命令查看 IS-IS 的路由信息。
- 使用 **display isis lsdb** [{ **level-1** | **level-2** } | **verbose** | { **local** | *lsp-id* | **is-name** *symbolic-name* }] * [*process-id* | **vpn-instance** *vpn-instance-name*]命令查看 IS-IS 的链路状态数据库信息。

----结束

7.13 配置 IS-IS 路由聚合(IPv6)

当大规模部署 IS-IS 网络时，为了避免 IS-IS 路由表中条目过多而降低路由查找速度的现象以及降低管理的复杂度，可以配置路由聚合，减小路由表的规模。

背景信息

路由聚合是指将多条具有相同 IP 前缀的路由聚合成一条路由。

当 IS-IS 网络规模较大时，配置路由聚合，可以有效减少路由表中的条目，减小对系统资源的占用，方便管理。

此外，如果被聚合的 IP 地址范围内的某条链路频繁 Up 和 Down，该变化并不会通告到被聚合的 IP 地址范围外的设备。因此，可以避免网络中的路由振荡，在一定程度上提高了网络的稳定性。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **isis** [*process-id*]，进入 IS-IS 视图。

步骤 3 执行命令 **ipv6 summary** *ipv6-address prefix-length* [**avoid-feedback** | **generate_null0_route** | **tag** *tag* | [**level-1** | **level-1-2** | **level-2**]] *，设置 IS-IS 生成聚合路由。

说明

在配置路由聚合后，本地 IS-IS 设备的路由表保持不变。

但是其他 IS-IS 设备的路由表中将只有一条聚合路由，没有具体路由。直到网络中被聚合的路由都出现故障而消失时，该聚合路由才会消失。

----结束

检查配置结果

完成配置后，可以使用以下方式查看聚合路由。

- 使用 **display isis route** 命令查看 IS-IS 路由表中的聚合路由。
- 使用 **display ipv6 routing-table** [**verbose**]命令查看 IP 路由表中的聚合路由。

7.14 配置 IS-IS 与其他路由协议交互(IPv6)

在网络中同时部署了 IS-IS 和其他路由协议时，需要配置 IS-IS 与其他路由协议的路由交互，才能使不同协议的网络正常通信。

7.14.1 建立配置任务

在配置 IS-IS 与其他路由协议交互(IPv6)前了解此特性的应用环境、配置这些特性的前置任务和数据准备，可以帮助您快速、准确地完成配置任务。

应用环境

在网络中同时部署了 IS-IS 和其他路由协议时，会涉及到如下问题：

- IS-IS 路由的优先级问题

  多种路由协议同时发现某一条路由时，如果其他协议的优先级比 IS-IS 的优先级高（例如 OSPFv3），那么该 IS-IS 路由将不会被优选。

  该问题可以通过配置 IS-IS 路由的优先级来解决。

- IS-IS 域与其他路由域的互通问题

  在网络中同时部署了 IS-IS 和其他路由协议时，IS-IS 路由域与其他协议路由域彼此隔离，不能互通。

  说明

  在同一台设备上运行的多个 IS-IS 进程的 LSDB 是彼此独立的，因此对于该设备上的某个 IS-IS 进程而言，其他 IS-IS 进程的路由也属于外部路由。

  为了使 IS-IS 路由域的流量能够被正确转发到其他路由域，需要在有外部路由的设备上（尤其是 IS-IS 路由域的边界设备）进行相关的配置。主要的解决方式有：

  - 配置 IS-IS 发布缺省路由。

    该方式配置较为简单，无需 IS-IS 域内的设备了解外部路由。IS-IS 域内去往外部其他路由域的流量都会通过发布缺省路由的设备进行转发。

  - 配置 IS-IS 引入外部路由。

    该方式可以使 IS-IS 域内的设备明确获悉到外部路由，可以实现对流量转发的更准确控制。

  同样，为了使其他路由域外的流量能够被正确转发到 IS-IS 域内，也必须在其他路由域内执行类似操作，才能使多个路由域能够互通。

前置任务

配置 IS-IS 与其他路由协议交互前，需完成以下任务：

- 配置接口的链路层协议。
- 配置接口的网络层地址，使相邻节点网络层可达。
- **配置 IS-IS 的基本功能(IPv6)**。
- 配置其他协议的基本功能。

数据准备

在调整 IS-IS 的选路(IPv4)之前，需要准备以下数据。

序号	数据
1	路由过滤时所采用的 IPv6 前缀列表或者路由策略
2	IS-IS 的协议优先级

7.14.2 配置 IS-IS 协议优先级(IPv6)

当到达同一目的地址有多种协议的路由时，配置 IS-IS 协议的优先级，可以使 IS-IS 路由优选。

背景信息

一台设备同时运行多个路由协议时，可以发现到达同一目的地的多条路由，其中协议优先级高的路由将被优选。

例如，当网络中运行了 OSPFv3 和 IS-IS 协议，如果两种协议同时发现了到达某一目的网段的路由，那么 OSPFv3 路由将被优选，因为 OSPFv3 的协议优先级比 IS-IS 高。

通过配置 IS-IS 协议的优先级，可以将 IS-IS 路由的优先级提高，使 IS-IS 的路由被优选。并且，结合路由策略的使用，可以灵活的仅将期望的部分 IS-IS 路由的优先级提高，而不影响其他的路由选择。

操作步骤

- 配置 IS-IS 协议的优先级。
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **isis** [*process-id*]，进入 IS-IS 视图。
  3. 执行命令 **ipv6 preference** *preference*，配置 IS-IS 协议的优先级。

     说明

     该命令用来设置 IS-IS 协议的优先级。配置值 *preference* 越小，优先级越高。

     缺省情况下，IS-IS 协议的优先级为 15。
- 配置 IS-IS 特定路由的优先级。
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **isis** [*process-id*]，进入 IS-IS 视图。

3. 执行命令 **ipv6 preference route-policy** *route-policy-name preference*，配置 IS-IS 特定路由的优先级。

   说明

   只有通过路由策略的 IS-IS 路由将采用配置值 *preference*。

----结束

7.14.3 配置 IS-IS 发布缺省路由(IPv6)

在具有外部路由的边界设备上配置 IS-IS 发布缺省路由，IS-IS 域内去往外部其他路由域的流量都会通过发布缺省路由的设备进行转发。

背景信息

该方式使该边界设备在 IS-IS 路由域内发布一条::/0 的缺省路由。因此，IS-IS 域内的其他设备在转发流量时，将所有去往外部路由域的流量首先转发到该边界设备，然后通过该设备去往外部路由域。

因此，必须保证该设备上有正确的外部路由信息。

说明

配置静态缺省路由也可以实现该功能，但是配置静态缺省路由时需要在大量设备上进行配置，不利于管理。

此外，采用 IS-IS 发布缺省路由的方式更加灵活。例如，如果存在多个边界设备，那么可以通过配置路由策略，使某台边界设备在满足条件时才发布缺省路由，从而避免造成路由黑洞。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **isis** [*process-id*]，进入 IS-IS 视图。

步骤 3 执行命令 **ipv6 default-route-advertise** [**always** | **match default** | **route-policy** *route-policy-name*] [**cost** *cost* | **tag** *tag* | [**level-1** | **level-1-2** | **level-2**]] * [**avoid-learning**]，配置 IS-IS 发布缺省路由。

----结束

7.14.4 配置 IS-IS 引入外部路由(IPv6)

在边界设备上配置 IS-IS 引入外部路由，可以使 IS-IS 域内的设备明确获悉外部路由，从而指导流量的转发。

背景信息

在边界设备上配置 IS-IS 发布缺省路由，可以将去往域外的流量吸收到该设备来处理。但是由于 IS-IS 域内的其他设备上没有外部路由，因此大量的流量都会被转发到该边界设备，造成该设备的过重负担。

此外，在有多个边界设备时，同样存在去往其他路由域的最优路由的选择问题，那么就必须要求 IS-IS 域内的其他设备获悉全部或部分外部路由。

无论在引入外部路由或将引入的路由发布给其他 IS-IS 设备时，都可通过配置路由策略来控制只引入部分路由或只发布部分引入的路由给其他 IS-IS 设备。

操作步骤

- 配置 IS-IS 引入外部路由。
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **isis** [*process-id*]，进入 IS-IS 视图。
  3. 执行命令 **ipv6 import-route**，配置 IS-IS 引入外部路由。

     说明

     配置引入外部路由后，IS-IS 设备将把引入的外部路由全部发布到 IS-IS 路由域。

     如果不希望将所有引入的外部路由都发布到 IS-IS 路由域，可以使用 **ipv6 filter-policy export** 来指定发布部分路由到 IS-IS 路由域。
- （可选）配置发布部分外部路由到 IS-IS 路由域。
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **isis** [*process-id*]，进入 IS-IS 视图。
  3. 执行命令 **ipv6 filter-policy** { **ipv6-prefix** *ipv6-prefix-name* | **route-policy** *route-policy-name* } **export** [*protocol* [*process-id*]]，配置发布部分外部路由到 IS-IS 路由域。

     说明

     只有通过路由策略的外部路由才能发布到 IS-IS 路由域。

----结束

7.14.5 检查配置结果

配置好 IS-IS 与其他路由协议交互后，可以查看 IS-IS 的路由表和 IP 路由表信息。

操作步骤

- 使用命令 **display isis lsdb** [{ **level-1** | **level-2** } | **verbose** | { **local** | *lsp-id* | **is-name** *symbolic-name* }] * [*process-id* | **vpn-instance** *vpn-instance-name*]查看 IS-IS 的链路状态数据库信息。
- 使用 **display isis route** [*process-id* | **vpn-instance** *vpn-instance-name*] [**ipv6**] [**verbose** | [**level-1** | **level-2**] | *ipv6-address* [*prefix-length*]] * [| **count**]命令查看 IS-IS 的路由信息。
- 使用 **display ipv6 routing-table ipv6-prefix** *ipv6-prefix-name* [**verbose**]命令查看 IP 路由表信息。

----结束

7.15 调整 IS-IS 路由的收敛速度(IPv6)

提高对 IS-IS 网络中故障的响应速度，加快出现网络故障时的路由收敛速度，可以提高 IS-IS 网络的可靠性。

7.15.1 建立配置任务

在调整 IS-IS 路由的收敛速度(IPv6)前了解此特性的应用环境、配置这些特性的前置任务和数据准备，可以帮助您快速、准确地完成配置任务。

应用环境

IS-IS 协议运行的过程包括：

- 邻接关系的建立。通过 Hello 报文建立正确的邻接关系，为 LSP 报文的扩散做准备。
- LSP 报文的扩散。IS-IS 网络中同一区域的所有设备上的 LSDB 达到一致。
- SPF 计算。根据 LSDB 进行 SPF 计算，建立 IS-IS 路由表。

因此，如果需要加快 IS-IS 路由的收敛速度，就需要针对以上三方面来配置：

- 调整邻接故障的检测时间。
- 调整 CSNP 报文和 LSP 报文的扩散。
- 调整 SPF 的计算时间。

此外，IS-IS 还支持 IPv4 路由按优先级收敛，通过调整某些关键路由的收敛优先级，可以使得这些关键路由优先收敛，不影响关键业务的正常运行。

前置任务

调整 IS-IS 路由的收敛速度(IPv6)前，需完成以下任务：

- 配置接口的链路层协议。
- 配置接口的网络层地址，使相邻节点网络层可达。
- **配置 IS-IS 的基本功能(IPv6)**。

数据准备

在调整 IS-IS 的选路(IPv6)之前，需要准备以下数据。

序号	数据
1	Hello 报文的发送时间间隔和邻居保持时间
2	CSNP 报文和 LSP 报文的扩散时间
3	SPF 的计算时间
4	路由收敛优先级

7.15.2 调整邻接故障的检测时间

通过调整 IS-IS 对邻接故障的检测，可以更快的感知到网络中的故障。

背景信息

IS-IS 通过发送 Hello 报文来发现邻居并建立邻接关系，之后会周期性的发送 Hello 报文来维持该邻接关系，并以此方式来检测网络故障。即如果在一定时间（邻居保持时间）内没有收到对方的 Hello 报文，则认为邻居已经 Down，触发 LSP 报文的扩散和 SPF 计算，从而达到 IS-IS 路由的重新收敛。

因此，调整邻接故障的检测时间，可以更快的检测到网络中的故障。主要方式包括：

- 配置 Hello 报文发送间隔。
- 配置邻居保持时间。

操作步骤

- 配置 Hello 报文发送间隔
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **interface** *interface-type interface-number*，进入接口视图。
  3. 执行命令 **isis timer hello** *hello-interval* [**level-1** | **level-2**]，设置接口上 Hello 报文发送间隔。

     说明

     在广播网链路上存在 Level-1 和 Level-2 两种 Hello 报文，不同类型的报文可以设置不同的值。如果不指定级别，则默认为 Level-1 和 Level-2 同时配置。

     在点到点链路上，只有一种 Hello 报文，不需要使用参数 Level-1 和 Level-2。
- 配置邻居保持时间
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **interface** *interface-type interface-number*，进入接口视图。
  3. 执行命令 **isis timer holding-multiplier** *number* [**level-1** | **level-2**]，配置邻居保持时间。

----结束

7.15.3 调整 SNP 报文和 LSP 报文的扩散

通过调整 IS-IS 中 SNP 报文和 LSP 报文的扩散，可以加速网络中所有设备的 LSDB 同步。

背景信息

SNP 包括 CSNP（Complete SNP，全序列号报文）和 PSNP（Partial SNP，部分序列号报文）。CSNP 包括 LSDB 中所有 LSP 的摘要信息，从而可以在相邻路由器间保持 LSDB 的同步。在广播网链路和点到点链路中，运行机制略有不同：

- 在广播网链路上，CSNP 由 DIS 设备周期性的发送。当邻居发现 LSDB 不同步时，发送 PSNP 报文来请求缺失的 LSP 报文。
- 在点到点链路上，CSNP 只在第一次建立邻接关系时发送，邻居发送 PSNP 报文来做应答。当邻居发现 LSDB 不同步时，同样发送 PSNP 报文来请求缺失的 LSP 报文。

AR150/200 支持修改 SNP 报文和 LSP 报文的相关参数，可以加速 LSDB 的同步，包括：

- **配置 CSNP 报文的发送间隔**
- **配置 LSP 生成的智能定时器**
- **配置 LSP 的大小**
- **配置 LSP 的刷新周期**
- **配置 LSP 的最大有效时间**
- **配置接口发送 LSP 的最小时间间隔**
- **配置 LSP 快速扩散**

- 配置点到点链路上的 LSP 重传间隔

操作步骤

- 配置 CSNP 报文的发送间隔
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **interface** *interface-type interface-number*，进入接口视图。
  3. 执行命令 **isis timer csnp** *csnp-interval* [**level-1** | **level-2**]，设置接口上 CSNP 报文发送间隔。

     说明

     参数 **Level-1** 和 **Level-2** 仅在广播网接口时可配置。
- 配置 LSP 生成的智能定时器
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **isis** [*process-id*]，进入 IS-IS 视图。
  3. 执行命令 **timer lsp-generation** *max-interval* [*init-interval* [*incr-interval*]] [**level-1** | **level-2**]，设置 LSP 生成所使用的智能定时器。

     如果没有指定 Level，则认为同时设置 Level-1 和 Level-2。

     初次产生同一 LSP（或者 LSP 分片）的延迟时间为 *init-interval*；第二次产生同一 LSP（或者 LSP 分片）的延迟时间为 *incr-interval*。随后，每变化一次，延迟时间都增大为前一次的两倍，直到 *max-interval*。稳定在 *max-interval* 三次或者 IS-IS 进程被重启，延迟时间又降回到 *init-interval*。

     在不使用 *incr-interval* 的情况下，初次产生同一 LSP（或者 LSP 分片）仍然使用 *init-interval* 作为延迟时间，随后都是使用 *max-interval* 作为延迟时间。同样，稳定在 *max-interval* 三次或者 IS-IS 进程被重启，延迟时间又降回到 *init-interval*。

     在只使用 *max-interval* 的情况下，智能定时器退化为一般的一次性触发定时器。
- 配置 LSP 的大小
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **isis** [*process-id*]，进入 IS-IS 视图。
  3. 执行命令 **lsp-length originate** *max-size*，设置生成 LSP 的大小。
  4. 执行命令 **lsp-length receive** *max-size*，设置接收 LSP 的大小。

     说明

     设置 *max-size* 参数时请注意，生成 LSP 的 *max-size* 必须小于等于接收 LSP 的 *max-size*。

     使用 **lsp-length** 命令设置的 *max-size* 值必须满足以下要求，否则接口的 MTU 状态会被认为是 Down。
     - 以太网接口的 MTU 值大于等于 *max-size*+3。
     - P2P 接口的 MTU 值大于等于 *max-size*。
- 配置 LSP 的刷新周期
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **isis** [*process-id*]，进入 IS-IS 视图。
  3. 执行命令 **timer lsp-refresh** *refresh-time*，设置 LSP 刷新周期。

     为了保证整个区域中的 LSP 能够保持同步，IS-IS 周期性发送当前全部 LSP。

LSP 刷新周期的缺省值为 900 秒，最大有效时间的缺省值为 1200 秒。配置时请注意，必须保证刷新周期比 LSP 的最大有效时间少三百秒以上，使得原有的LSP过期之前，新的 LSP 可以到达区域内所有路由器。

说明

网络规模越大，LSP 刷新周期与 LSP 最大有效时间之间的差值也越大。

- 配置 LSP 的最大有效时间
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **isis** [*process-id*]，进入 IS-IS 视图。
  3. 执行命令 **timer lsp-max-age** *age-time*，设置 LSP 最大有效时间。

     路由器生成系统 LSP 时，会在 LSP 中填写此 LSP 的最大有效时间。当此 LSP 被其它路由器接收后，它的有效时间会随着时间的变化不断减小。如果路由器一直没有收到更新的 LSP，而此 LSP 的有效时间已减少到 0，LSP 再保持 60 秒，若还未收到新的 LSP，那么此 LSP 将被从 LSDB 中删除。
- 配置接口发送 LSP 的最小时间间隔
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **interface** *interface-type interface-number*，进入接口视图。
  3. 执行命令 **isis timer lsp-throttle** *throttle-interval* [**count** *count*]，设置发送 LSP 的最小间隔时间。

     count：用来指定在 *throttle-interval* 时间间隔内发送 LSP 报文的最大包数。整数形式，取值范围是 1 ～ 1000。
- 配置 LSP 快速扩散
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **isis** [*process-id*]，进入 IS-IS 视图。
  3. 执行命令 **flash-flood** [*lsp-count* | **max-timer-interval** *interval* | [**level-1** | **level-2**]] *，使能 LSP 快速扩散。

     使用 **flash-flood** 命令可以加速 LSP 的扩散（Flooding）速度。用户可以通过参数 *lsp-count* 指定每次扩散的 LSP 数量，这个数量是针对所有接口的。如果需要发送的 LSP 的数量大于 *lsp-count*，则就发送 *lsp-count* 个 LSP。如果配置了定时器，在路由计算之前如果这个定时器未超时，则立即扩散；否则在该定时器超时的时候发送。

     配置 LSP 快速扩散时，如果不指定 Level-1 或 Level-2，则默认为 Level-1 和 Level-2 都配置快速扩散。
- 配置点到点链路上的 LSP 重传间隔
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **interface** *interface-type interface-number*，进入接口视图。
  3. 执行命令 **isis timer lsp-retransmit** *retransmit-interval*，设置 LSP 在点到点链路上的重传间隔。

----结束

7.15.4 调整 SPF 的计算时间

通过调整 SPF 的计算时间，既可以保证 IS-IS 对网络变化的及时响应，又可以减少 SPF 计算对系统资源的过多占用。

背景信息

当网络变化比较频繁时，IS-IS 会频繁的进行 SPF 计算。频繁的 SPF 计算会消耗系统大量的 CPU 资源，从而影响正常业务的运行。

配置智能定时器的优势在于当刚开始进行 SPF 计算时，两次计算的间隔时间较小，保证 IS-IS 路由的收敛速度。之后随着整个 IS-IS 网络的拓扑趋于稳定时，则应该适当延长两次 SPF 计算的间隔时间，从而减少不必要的计算。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **isis** [*process-id*]，进入 IS-IS 视图。

步骤 3 执行命令 **timer spf** *max-interval* [*init-interval* [*incr-interval*]]，设置 SPF 智能定时器。

智能定时器的变化规律如下：

- 初次进行 SPF 计算的延迟时间为 *init-interval*；第二次进行 SPF 计算的延迟时间为 *incr-interval*。随后，每变化一次，SPF 计算的延迟时间增大为前一次的两倍，直到 *max-interval*。稳定在 *max-interval* 三次或者 IS-IS 进程被重启，延迟时间又降回到 *init-interval*。
- 在不使用 *incr-interval* 的情况下，初次进行 SPF 计算用 *init-interval* 作为延迟时间，随后都是使用 *max-interval* 作为延迟时间。稳定在 *max-interval* 三次或者 IS-IS 进程被重启，延迟时间又降回到 *init-interval*。
- 在只使用 *max-interval* 的情况下，智能定时器退化为一般的一次性触发定时器。

----结束

7.15.5 配置 IS-IS 路由按优先级收敛(IPv6)

将 IS-IS 网络中的关键路由配置为较高的收敛优先级，保证网络拓扑变化时关键路由的优先收敛，减小对重要业务的影响。

背景信息

缺省情况下，IS-IS 128 位主机路由的收敛优先级为 **medium**，其他 IS-IS 路由的收敛优先级为 **low**。

AR150/200 支持通过配置 IS-IS 路由的收敛优先级，使某些重要路由在网络拓扑发生变化时优先收敛。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **isis** [*process-id*]，进入 IS-IS 视图。

步骤 3 执行命令 **ipv6 prefix-priority** [**level-1** | **level-2**] { **critical** | **high** | **medium** } { **ipv6-prefix** *prefix-name* | **tag** *tag-value* }，配置 IS-IS 路由的收敛优先级。

IS-IS 路由收敛优先级的应用规律。

- 对于已存在的 IS-IS 路由，收敛优先级将依据 **ipv6 prefix-priority** 命令重新进行设置。

- 对新增加的 IS-IS 路由，收敛优先级将依据 **ipv6 prefix-priority** 命令的过滤结果进行设置。
- 如果一条路由符合多个收敛优先级的匹配规则，则这些收敛优先级中最高者当选为路由的收敛优先级。
- Level-1 IS-IS 路由的收敛优先级高于 Level-2 IS-IS 路由的收敛优先级。
- 若不指定 Level，IS-IS 会对 Level-1 和 Level-2 的 IS-IS 路由都进行配置。

说明

ipv6 prefix-priority 命令仅在公网生效。

如果用 **ipv6 prefix-priority** 命令对 IS-IS 路由（除了 IS-IS 32 位主机路由）的收敛优先级进行配置后，IS-IS 32 位主机路由的缺省收敛优先级将从 medium 变为 low，其他 IS-IS 路由的收敛优先级依据 **ipv6 prefix-priority** 命令的配置而变化。

----结束

7.15.6 检查配置结果

配置好各种影响 IS-IS 路由收敛速度的参数后，可以查看接口发送 IS-IS 报文的各种参数。

操作步骤

- 使用 **display isis interface** [**verbose**] [*process-id* | **vpn-instance** *vpn-instance-name*]命令查看 IS-IS 接口发送的 IS-IS 报文的信息。
- 使用 **display isis route** [*process-id* | **vpn-instance** *vpn-instance-name*] **ipv6** [**verbose** | [**level-1** | **level-2**] | *ipv6-address* [*prefix-length*]] * [| **count**]命令查看 IS-IS 路由的优先级信息。

----结束

7.16 配置 IS-IS GR

通过配置 IS-IS GR，可以使路由器平滑重启，避免出现暂时的“黑洞”。

7.16.1 建立配置任务

在配置 IS-IS GR 前了解这些特性的应用环境、配置此特性的前置任务和数据准备，可以帮助您快速、准确地完成配置任务。

应用环境

当 IS-IS 协议重启时会引起网络暂时中断，这是因为该路由器与其它邻居的邻接关系被拆除，该路由器的 LSP 报文被删除，使得路由计算不准确，造成报文丢失。

IS-IS GR 特性可以解决这个问题。该特性使路由器将其重启状态通知给邻居，允许邻居重新建立邻接关系而保持转发不终止。

IS-IS GR 特性有以下优点：

- 在重启 IS-IS 协议时，路由器可重新向邻居发送连接请求，而不会终止邻接关系。
- 生成 LSP 报文前，GR 最大限度地减轻因等待数据库同步而导致的对网络的干扰。

- 对于第一次启动的路由器，在 LSP 报文中设置过载标记位直到数据库同步，可以保证网络不产生路由黑洞。

说明

AR150/200 只能作为 Helper 路由器，不能作为 Restarter 路由器。

前置任务

在配置 IS-IS GR 之前，需完成以下任务：

- 配置接口的网络层地址，使相邻节点网络层可达。
- **配置 IS-IS 的基本功能(IPv4)**。

数据准备

在配置 IS-IS GR 之前，需准备以下数据。

序号	数据
1	IS-IS 进程号
2	重建 GR Session 连接会话的间隔时间
3	GR Restarter 重启时是否抑制发布邻接关系

7.16.2 使能 IS-IS 协议的 GR 能力

使能 IS-IS 协议的 GR 能力是配置 IS-IS GR 的必选步骤。

背景信息

请在运行 IS-IS 协议的路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **isis** [*process-id*]，进入 IS-IS 视图。

步骤 3 执行命令 **graceful-restart**，使能 IS-IS 协议的 GR 能力。

缺省情况下，IS-IS 协议的 GR 能力被禁止。

----结束

7.16.3 配置 IS-IS 协议的 GR 会话参数

通过配置 IS-IS GR 的参数，可以避免网络中出现暂时“黑洞”的现象。

背景信息

第一次启动（不包括重启后）的路由器不会对转发状态进行维护。如果该路由器重启，则它前一次运行时生成的 LSP 可能还存在于网络中其它路由器的 LSP 数据库中。

由于路由器启动时 LSP 分片的序列号也被重新初始化，网络中其它路由器保存的该路由器之前发布的 LSP 可能比该路由器重启后新产生的 LSP 看上去更“新”。这将导致网络中出现暂时的“黑洞”（black hole），并一直持续到正常的更新过程结束，该路由器重新生成 LSP 并以最高序列号将它们发布出去。

如果该路由器的邻居在它启动过程中抑制发布邻接关系到此路由器，直到该路由器将更新的 LSP 发布出去，上述情况也可以避免。

请在运行 IS-IS 协议的路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **isis** [*process-id*]，进入 IS-IS 视图。

步骤 3 执行命令 **graceful-restart interval** *interval-value*，配置 IS-IS 重建 GR Session 连接会话的间隔时间。

GR 间隔时间在 IS-IS 的 Hello PDU 中设置为保持时间，这样在该路由器 GR 的时间内邻居不会断掉与其的邻接关系。缺省情况下，Restart 时间为 300 秒。

步骤 4 （可选）执行命令 **graceful-restart suppress-sa**，配置 GR Restarter 来抑制重启 TLV 的 SA（Suppress-Advertisement）位。

如果管理员不想对路由器进行主备倒换时在 Hello PDU 中抑制 SA 位，可执行 **undo graceful-restart suppress-sa** 命令。

缺省情况下，不对 SA 位进行抑制。

----结束

7.16.4 检查配置结果

配置完 IS-IS GR 后，可以参看 IS-IS GR 的重启状态和参数信息。

前提条件

已经完成 IS-IS GR 的所有配置。

操作步骤

步骤 1 使用 **display isis graceful-restart status** [**level-1** | **level-2**] [*process-id* | **vpn-instance** *vpn-instance-name*]命令查看 IS-IS GR 的重启状态。

----结束

7.17 维护 IS-IS 配置

维护 IS-IS，包括复位和清除 IS-IS。

7.17.1 复位 IS-IS 数据结构

通过重启 IS-IS，达到复位的目的。可以选择以 GR 的方式复位 IS-IS。

背景信息

注意

复位 IS-IS 数据结构后，以前所有的结构信息、邻接关系将全部重新建立，务必仔细确认。

在确认需要复位 IS-IS 的数据结构后，请在用户视图下执行以下命令。

操作步骤

步骤 1 使用 **reset isis all** [[*process-id* | **vpn-instance** *vpn-instance-name*] | **graceful-restart**] *命令复位 IS-IS 的数据结构。

缺省情况下，不复位 IS-IS 的数据结构。

----结束

7.17.2 复位 IS-IS 特定邻居

通过重启 IS-IS 邻居，可以复位 IS-IS 邻居关系，达到使新的配置生效的目的。

背景信息

注意

复位 IS-IS 特定邻居（执行 **reset isis peer** 命令）会导致路由器之间的 IS-IS 特定邻居关系中断。务必仔细确认是否必须执行复位 IS-IS 特定邻居的操作。

当 IS-IS 路由策略或协议发生变化后，需要通过复位 IS-IS 特定邻居使新的配置生效。如果需要复位 IS-IS 特定邻居，可在用户视图下选择执行以下命令。

操作步骤

步骤 1 使用 **reset isis peer** *system-id* [*process-id* | **vpn-instance** *vpn-instance-name*]命令复位 IS-IS 的特定邻居。

----结束

7.18 配置举例

介绍 IS-IS 配置举例。请结合配置流程图了解配置过程。配置示例中包括组网需求、配置注意事项、配置思路等。

7.18.1 配置 IS-IS 基本功能示例

举例说明通过 IS-IS 协议实现 IPv4 网络互连的基本配置方法。

组网需求

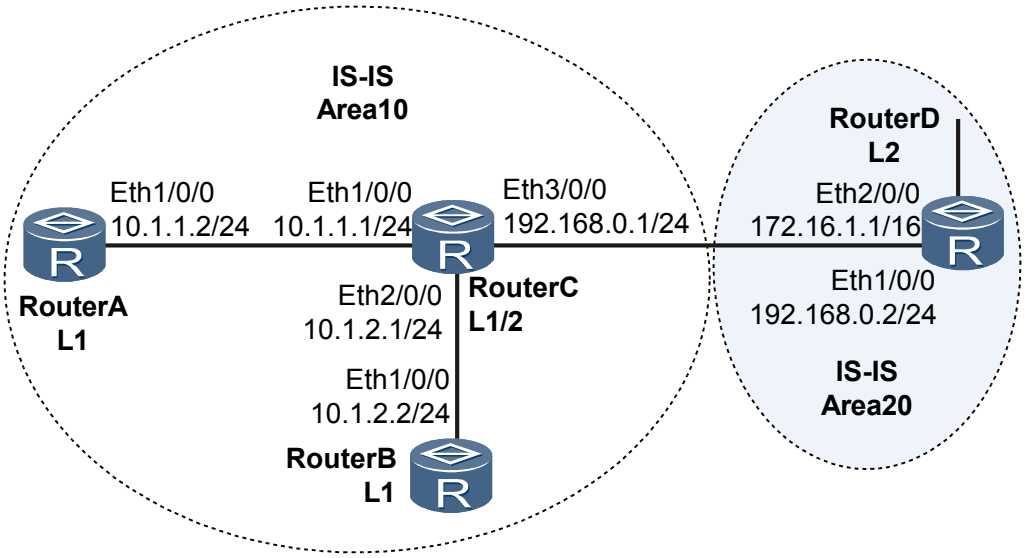
如图 7-3 所示：

- 路由器 RouterA、RouterB、RouterC 和 RouterD 属于同一自治系统，要求他们之间通过 IS-IS 协议达到 IP 网络互连的目的。
- RouterA、RouterB 和 RouterC 属于区域 10，RouterD 属于区域 20。
- RouterA 和 RouterB 是 Level-1 路由器，RouterC 是 Level-1-2 路由器，RouterD 是 Level-2 路由器。

说明

图中路由器仅 RouterA 和 RouterB 为 AR150/200。

图 7-3 配置 IS-IS 基本功能组网图



配置思路

采用如下的思路配置 IS-IS 的基本功能：

1. 在各路由器上使能 IS-IS，配置 level 级别，指定网络实体。
2. 配置 RouterA 和 RouterC 以指定的方式和密码验证 Hello 报文。
3. 查看各路由器的 IS-IS 数据库信息及路由表信息。

数据准备

为完成此配置例，需准备如下的数据：

- RouterA、RouterB、RouterC 和 RouterD 的区域地址。
- RouterA、RouterB、RouterC 和 RouterD 的级别。

操作步骤

步骤 1 配置各接口的 IP 地址（略）

步骤 2 配置 IS-IS 基本功能

配置 RouterA。

```
[RouterA] isis 1
[RouterA-isis-1] is-level level-1
[RouterA-isis-1] network-entity 10.0000.0000.0001.00
[RouterA-isis-1] quit
[RouterA] interface ethernet 1/0/0
[RouterA-Ethernet1/0/0] isis enable 1
[RouterA-Ethernet1/0/0] quit
```

配置 RouterB。

```
[RouterB] isis 1
[RouterB-isis-1] is-level level-1
[RouterB-isis-1] network-entity 10.0000.0000.0002.00
[RouterB-isis-1] quit
[RouterB] interface ethernet 1/0/0
[RouterB-Ethernet1/0/0] isis enable 1
[RouterB-Ethernet1/0/0] quit
```

配置 RouterC。

```
[RouterC] isis 1
[RouterC-isis-1] network-entity 10.0000.0000.0003.00
[RouterC-isis-1] quit
[RouterC] interface ethernet 1/0/0
[RouterC-Ethernet1/0/0] isis enable 1
[RouterC-Ethernet1/0/0] quit
[RouterC] interface ethernet 2/0/0
[RouterC-Ethernet2/0/0] isis enable 1
[RouterC-Ethernet2/0/0] quit
[RouterC] interface ethernet 3/0/0
[RouterC-Ethernet3/0/0] isis enable 1
[RouterC-Ethernet3/0/0] quit
```

配置 RouterD。

```
[RouterD] isis 1
[RouterD-isis-1] is-level level-2
[RouterD-isis-1] network-entity 20.0000.0000.0004.00
[RouterD-isis-1] quit
[RouterD] interface ethernet 2/0/0
[RouterD-Ethernet2/0/0] isis enable 1
[RouterD-Ethernet2/0/0] quit
[RouterD] interface ethernet 1/0/0
[RouterD-Ethernet1/0/0] isis enable 1
[RouterD-Ethernet1/0/0] quit
```

步骤 3 配置 RouterA 和 RouterC 验证 Hello 报文的认证模式和密码。

配置 RouterA。

```
[RouterA] interface ethernet 1/0/0
[RouterA-Ethernet1/0/0] isis authentication-mode md5 huawei
```

配置 RouterC。

```
[RouterC] interface ethernet 1/0/0
[RouterC-Ethernet1/0/0] isis authentication-mode md5 huawei
```

步骤 4 验证配置结果

显示各路由器的 IS-IS LSDB 信息。

```
[RouterA] display isis lsdb
                      Database information for ISIS(1)
                      --------------------------------
                        Level-1 Link State Database
LSPID                 Seq Num      Checksum      Holdtime      Length  ATT/P/OL
-------------------------------------------------------------------------------
0000.0000.0001.00-00* 0x00000006   0xbf7d        649           68      0/0/0
0000.0000.0001.01-00* 0x00000002   0xcfbb        1157          55      0/0/0
0000.0000.0002.00-00  0x00000003   0xef4d        545           68      0/0/0
0000.0000.0003.00-00  0x00000008   0x3340        582           111     1/0/0
Total LSP(s): 4
*(In TLV)-Leaking Route, *(By LSPID)-Self LSP, +-Self LSP(Extended),
          ATT-Attached, P-Partition, OL-Overload
[RouterB] display isis lsdb
                      Database information for ISIS(1)
                      --------------------------------
                        Level-1 Link State Database
LSPID                 Seq Num      Checksum      Holdtime      Length  ATT/P/OL
-------------------------------------------------------------------------------
0000.0000.0001.00-00  0x00000006   0xbf7d        642           68      0/0/0
0000.0000.0002.00-00* 0x00000003   0xef4d        538           68      0/0/0
0000.0000.0002.01-00* 0x00000003   0xef4b        538           68      0/0/0
0000.0000.0003.00-00  0x00000008   0x3340        574           111     1/0/0
Total LSP(s): 4
*(In TLV)-Leaking Route, *(By LSPID)-Self LSP, +-Self LSP(Extended),
          ATT-Attached, P-Partition, OL-Overload
[RouterC] display isis lsdb
                      Database information for ISIS(1)
                      --------------------------------
                        Level-1 Link State Database
LSPID                 Seq Num      Checksum      Holdtime      Length  ATT/P/OL
-------------------------------------------------------------------------------
0000.0000.0001.00-00  0x00000006   0xbf7d        638           68      0/0/0
0000.0000.0001.01-00  0x00000002   0xcfbb        871           55      0/0/0
0000.0000.0002.00-00  0x00000003   0xef4d        533           68      0/0/0
0000.0000.0003.00-00* 0x00000008   0x3340        569           111     1/0/0
Total LSP(s): 4
*(In TLV)-Leaking Route, *(By LSPID)-Self LSP, +-Self LSP(Extended),
          ATT-Attached, P-Partition, OL-Overload
                        Level-2 Link State Database
LSPID                 Seq Num      Checksum      Holdtime      Length  ATT/P/OL
-------------------------------------------------------------------------------
0000.0000.0003.00-00* 0x00000008   0x55bb        650           100     0/0/0
0000.0000.0004.00-00  0x00000005   0x6510        629           84      0/0/0
0000.0000.0004.01-00  0x00000001   0xee95        803           55      0/0/0
Total LSP(s): 3
*(In TLV)-Leaking Route, *(By LSPID)-Self LSP, +-Self LSP(Extended),
          ATT-Attached, P-Partition, OL-Overload
[RouterD] display isis lsdb
                      Database information for ISIS(1)
                      --------------------------------
                        Level-2 Link State Database
LSPID                 Seq Num      Checksum      Holdtime      Length  ATT/P/OL
-------------------------------------------------------------------------------
0000.0000.0003.00-00  0x00000008   0x55bb        644           100     0/0/0
0000.0000.0004.00-00* 0x00000005   0x6510        624           84      0/0/0
0000.0000.0004.01-00* 0x00000001   0xee95        700           55      0/0/0
Total LSP(s): 3
*(In TLV)-Leaking Route, *(By LSPID)-Self LSP, +-Self LSP(Extended),
          ATT-Attached, P-Partition, OL-Overload
```

显示各路由器的 IS-IS 路由信息。Level-1 路由器的路由表中应该有一条缺省路由，且下一跳为 Level-1-2 路由器，Level-2 路由器应该有所有 Level-1 和 Level-2 的路由。

```
[RouterA] display isis route
                       Route information for ISIS(1)
                       ------------------------------
                      ISIS(1) Level-1 Forwarding Table
                      --------------------------------
 IPV4 Destination   IntCost   ExtCost ExitInterface   NextHop         Flags
```

```
------------------------------------------------------------------------------
 10.1.1.0/24         10        NULL    Eth1/0/0          Direct        D/-/L/-
 10.1.2.0/24         20        NULL    Eth1/0/0          10.1.1.1      A/-/-/-
 192.168.0.0/24      20        NULL    Eth1/0/0          10.1.1.1      A/-/-/-
 0.0.0.0/0           10        NULL    Eth1/0/0          10.1.1.1      A/-/-/-
     Flags: D-Direct, A-Added to URT, L-Advertised in LSPs, S-IGP Shortcut,
                                U-Up/Down Bit Set
[RouterC] display isis route
                        Route information for ISIS(1)
                        -----------------------------
                       ISIS(1) Level-1 Forwarding Table
                       --------------------------------
 IPV4 Destination   IntCost   ExtCost ExitInterface   NextHop         Flags
------------------------------------------------------------------------------
 10.1.1.0/24         10        NULL    Eth1/0/0          Direct        D/-/L/-
 10.1.2.0/24         10        NULL    Eth2/0/0          Direct        D/-/L/-
 192.168.0.0/24      10        NULL    Eth3/0/0          Direct        D/-/L/-
     Flags: D-Direct, A-Added to URT, L-Advertised in LSPs, S-IGP Shortcut,
                                U-Up/Down Bit Set
                       ISIS(1) Level-2 Forwarding Table
                       --------------------------------
 IPV4 Destination   IntCost   ExtCost ExitInterface   NextHop         Flags
------------------------------------------------------------------------------
 10.1.1.0/24         10        NULL    Eth1/0/0          Direct        D/-/L/-
 10.1.2.0/24         10        NULL    Eth2/0/0          Direct        D/-/L/-
 192.168.0.0/24      10        NULL    Eth3/0/0          Direct        D/-/L/-
 172.16.0.0/16       20        NULL    Eth3/0/0          192.168.0.2   A/-/-/-
     Flags: D-Direct, A-Added to URT, L-Advertised in LSPs, S-IGP Shortcut,
                                U-Up/Down Bit Set
[RouterD] display isis route
                        Route information for ISIS(1)
                        -----------------------------
                       ISIS(1) Level-2 Forwarding Table
                       --------------------------------
 IPV4 Destination   IntCost   ExtCost ExitInterface   NextHop          Flags
------------------------------------------------------------------------------
 192.168.0.0/24      10        NULL    Eth3/0/0          Direct         D/-/L/-
 10.1.1.0/24         20        NULL    Eth3/0/0          192.168.0.1    A/-/-/-
 10.1.2.0/24         20        NULL    Eth3/0/0          192.168.0.1    A/-/-/-
 172.16.0.0/16       10        NULL    Eth2/0/0         Direct          D/-/L/-
     Flags: D-Direct, A-Added to URT, L-Advertised in LSPs, S-IGP Shortcut,
                                U-Up/Down Bit Set
```

----结束

配置文件

- RouterA 的配置文件

```
#
 sysname RouterA
#
isis 1
 is-level level-1
 network-entity 10.0000.0000.0001.00
#
interface Ethernet1/0/0
 ip address 10.1.1.2 255.255.255.0
 isis enable 1
 isis authentication-mode md5 N`C55QK<`=/Q=^Q`MAF4<1!!
#
return
```

- RouterB 的配置文件

```
#
 sysname RouterB
#
isis 1
 is-level level-1
```

```
 network-entity 10.0000.0000.0002.00
#
interface Ethernet1/0/0
 ip address 10.1.2.2 255.255.255.0
 isis enable 1
#
return
```

- RouterC 的配置文件

```
#
 sysname RouterC
#
isis 1
 network-entity 10.0000.0000.0003.00
#
interface Ethernet1/0/0
 ip address 10.1.1.1 255.255.255.0
 isis enable 1
 isis authentication-mode md5 N`C55QK<`=/Q=^Q`MAF4<1!!
#
interface Ethernet2/0/0
 ip address 10.1.2.1 255.255.255.0
 isis enable 1
#
interface Ethernet3/0/0
 ip address 192.168.0.1 255.255.255.0
 isis enable 1
#
return
```

- RouterD 的配置文件

```
#
 sysname RouterD
#
isis 1
 is-level level-2
 network-entity 20.0000.0000.0004.00
#
interface Ethernet1/0/0
 ip address 192.168.0.2 255.255.255.0
 isis enable 1
#
interface Ethernet2/0/0
 ip address 172.16.1.1 255.255.0.0
 isis enable 1
#
return
```

7.18.2 配置 IS-IS 的 DIS 选择示例

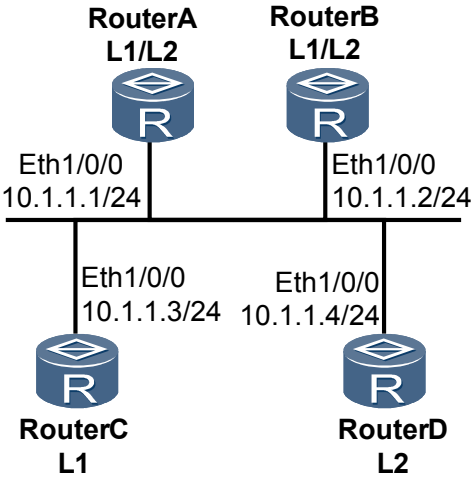
举例说明在广播类型的网络中指定 DIS 的配置方法。

组网需求

如图 7-4 所示：

- 路由器 RouterA、RouterB、RouterC 和 RouterD 都运行 IS-IS 路由协议以实现互连。
- 路由器 RouterA、RouterB、RouterC 和 RouterD 属于同一区域 10，网络类型为广播网（以太网）。
- RouterA 和 RouterB 是 Level-1-2 路由器，RouterC 是 Level-1 路由器，RouterD 是 Level-2 路由器。
- RouterA 的 DIS 优先级为 100。
- 要求通过改变接口的 DIS 优先级，将 RouterA 配置为 Level-1-2 的 DIS 路由器。

图 7-4 配置 IS-IS 的 DIS 选择组网图



配置思路

采用如下的思路配置 IS-IS 的 DIS 选择：

1. 在各路由器上使能 IS-IS，指定网络实体，实现互连。
2. 缺省优先级情况下，查看各路由器 IS-IS 接口的信息。
3. 配置路由器的 DIS 优先级。

数据准备

为完成此配置例，需准备如下的数据：

- 四台路由器的区域地址。
- 四台路由器的级别。
- RouterA 的 DIS 优先级。

操作步骤

步骤 1 配置各接口的 IPv4 地址（略）

步骤 2 查看各路由器上 Eth 接口的 MAC 地址

查看 RouterA 上接口 Ethernet1/0/0 的 MAC 地址。

```
[RouterA] display arp interface ethernet 1/0/0
IP ADDRESS      MAC ADDRESS  EXPIRE(M) TYPE        INTERFACE      VPN-INSTANCE
                                       VLAN/CEVLAN PVC
--------------------------------------------------------------------------------
10.1.1.1        00e0-fc10-afec         I -         Eth1/0/0
--------------------------------------------------------------------------------
Total:1        Dynamic:0       Static:0     Interface:1
```

查看 RouterB 上接口 Ethernet1/0/0 的 MAC 地址。

```
[RouterB] display arp interface ethernet 1/0/0
IP ADDRESS      MAC ADDRESS  EXPIRE(M) TYPE        INTERFACE      VPN-INSTANCE
                                       VLAN/CEVLAN PVC
```

```
--------------------------------------------------------------------------------
10.1.1.2        00e0-fccd-acdf          I -         Eth1/0/0
--------------------------------------------------------------------------------
Total:1         Dynamic:0       Static:0    Interface:1
```

查看 RouterC 上接口 Ethernet1/0/0 的 MAC 地址。

```
[RouterC] display arp interface ethernet 1/0/0
IP ADDRESS      MAC ADDRESS  EXPIRE(M) TYPE        INTERFACE      VPN-INSTANCE
                                       VLAN/CEVLAN PVC
--------------------------------------------------------------------------------
10.1.1.3        00e0-fc50-25fe          I -         Eth1/0/0
--------------------------------------------------------------------------------
Total:1         Dynamic:0       Static:0    Interface:1
```

查看 RouterD 上接口 Ethernet1/0/0 的 MAC 地址。

```
[RouterD] display arp interface ethernet 1/0/0
IP ADDRESS      MAC ADDRESS  EXPIRE(M) TYPE        INTERFACE      VPN-INSTANCE
                                       VLAN/CEVLAN PVC
--------------------------------------------------------------------------------
10.1.1.4        00e0-fcfd-305c          I -         Eth1/0/0
--------------------------------------------------------------------------------
Total:1         Dynamic:0       Static:0    Interface:1
```

步骤 3 启动 IS-IS

配置 RouterA。

```
[RouterA] isis 1
[RouterA-isis-1] network-entity 10.0000.0000.0001.00
[RouterA-isis-1] quit
[RouterA] interface ethernet 1/0/0
[RouterA-Ethernet1/0/0] isis enable 1
[RouterA-Ethernet1/0/0] quit
```

配置 RouterB。

```
[RouterB] isis 1
[RouterB-isis-1] network-entity 10.0000.0000.0002.00
[RouterB-isis-1] quit
[RouterB] interface ethernet 1/0/0
[RouterB-Ethernet1/0/0] isis enable 1
[RouterB-Ethernet1/0/0] quit
```

配置 RouterC。

```
[RouterC] isis 1
[RouterC-isis-1] network-entity 10.0000.0000.0003.00
[RouterC-isis-1] is-level level-1
[RouterC-isis-1] quit
[RouterC] interface ethernet 1/0/0
[RouterC-Ethernet1/0/0] isis enable 1
[RouterC-Ethernet1/0/0] quit
```

配置 RouterD。

```
[RouterD] isis 1
[RouterD-isis-1] network-entity 10.0000.0000.0004.00
[RouterD-isis-1] is-level level-2
[RouterD-isis-1] quit
[RouterD] interface ethernet 1/0/0
[RouterD-Ethernet1/0/0] isis enable 1
[RouterD-Ethernet1/0/0] quit
```

查看 RouterA 的 IS-IS 邻居信息。

```
[RouterA] display isis peer
                      Peer information for ISIS(1)
                      ----------------------------
```

```
  System Id    Interface         Circuit Id        State  HoldTime Type     PRI
 ------------------------------------------------------------------------------
0000.0000.0002 Eth1/0/0           0000.0000.0002.01  Up    9s       L1(L1L2) 64
0000.0000.0003 Eth1/0/0           0000.0000.0002.01  Up    27s      L1       64
0000.0000.0002 Eth1/0/0           0000.0000.0004.01  Up    28s      L2(L1L2) 64
0000.0000.0004 Eth1/0/0           0000.0000.0004.01  Up    8s       L2       64

Total Peer(s): 4
```

显示 RouterA 的 IS-IS 接口信息。

```
[RouterA] display isis interface
                    Interface information for ISIS(1)
                    ---------------------------------

 Interface       Id      IPV4.State          IPV6.State      MTU  Type  DIS
 Eth1/0/0        001         Up                  Down         1497 L1/L2 No/No
```

显示 RouterB 的 IS-IS 接口信息。

```
[RouterB] display isis interface
                    Interface information for ISIS(1)
                    ---------------------------------

 Interface       Id      IPV4.State          IPV6.State      MTU  Type  DIS
 Eth1/0/0        001         Up                  Down         1497 L1/L2 Yes/No
```

显示 RouterD 的 IS-IS 接口信息。

```
[RouterD] display isis interface
                    Interface information for ISIS(1)
                    ---------------------------------

 Interface       Id      IPV4.State          IPV6.State      MTU  Type  DIS
 Eth1/0/0        001         Up                  Down         1497 L1/L2 No/Yes
```

说明

从接口信息中可以看到，在使用缺省 DIS 优先级的情况下，在 Level-1 级别的路由器中，RouterB 上接口的 MAC 地址最大，因此 RouterB 为 Level-1 的 DIS；在 Level-2 级别的路由器中，RouterD 上接口的 MAC 地址最大，因此 RouterD 为 Level-2 的 DIS。Level-1 和 Level-2 的伪节点分别是 0000.0000.0002.01 和 0000.0000.0004.01。

步骤 4 配置 RouterA 的 DIS 优先级

```
[RouterA] interface ethernet 1/0/0
[RouterA-Ethernet1/0/0] isis dis-priority 100
```

查看 RouterA 的 IS-IS 邻居信息。

```
[RouterA] display isis peer
                      Peer information for ISIS(1)
                      ----------------------------

  System Id    Interface         Circuit Id        State  HoldTime Type     PRI
 ------------------------------------------------------------------------------
0000.0000.0002 Eth1/0/0           0000.0000.0001.01  Up    21s      L1(L1L2) 64
0000.0000.0003 Eth1/0/0           0000.0000.0001.01  Up    27s      L1       64
0000.0000.0002 Eth1/0/0           0000.0000.0001.01  Up    28s      L2(L1L2) 64
0000.0000.0004 Eth1/0/0           0000.0000.0001.01  Up    30s      L2       64

Total Peer(s): 4
```

步骤 5 验证配置结果

查看 RouterA 的 IS-IS 接口信息。

```
[RouterA] display isis interface
                    Interface information for ISIS(1)
                    ---------------------------------

 Interface       Id      IPV4.State          IPV6.State      MTU  Type  DIS
 Eth1/0/0        001         Up                  Down         1497 L1/L2 Yes/Yes
```

说明

从上述信息中可以看到，在改变 IS-IS 接口的 DIS 优先级后，RouterA 立即成为 Level-1-2 的 DIS（DR），且伪节点是 0000.0000.0001.01。

显示 RouterB 的 IS-IS 邻居和接口信息

```
[RouterB] display isis peer
                      Peer information for ISIS(1)
                      ----------------------------

  System Id    Interface          Circuit Id       State  HoldTime Type     PRI
  ------------------------------------------------------------------------------
0000.0000.0001 Eth1/0/0            0000.0000.0001.01 Up    7s       L1(L1L2) 100
0000.0000.0003 Eth1/0/0            0000.0000.0001.01 Up    25s      L1       64
0000.0000.0001 Eth1/0/0            0000.0000.0001.01 Up    7s       L2(L1L2) 100
0000.0000.0004 Eth1/0/0            0000.0000.0001.01 Up    25s      L2       64

Total Peer(s): 4
 [RouterB] display isis interface
                     Interface information for ISIS(1)
                     ---------------------------------

 Interface       Id      IPV4.State          IPV6.State      MTU  Type  DIS
 Eth1/0/0         001         Up                 Down         1497 L1/L2 No/No
```

显示 RouterD 的 IS-IS 邻居和接口信息

```
[RouterD] display isis peer
                      Peer information for ISIS(1)
                      ----------------------------

  System Id    Interface          Circuit Id       State  HoldTime Type     PRI
  ------------------------------------------------------------------------------
0000.0000.0001 Eth1/0/0            0000.0000.0001.01 Up    9s       L2       100
0000.0000.0002 Eth1/0/0            0000.0000.0001.01 Up    28s      L2       64

Total Peer(s): 2
[RouterD] display isis interface
                     Interface information for ISIS(1)
                     ---------------------------------

 Interface       Id      IPV4.State          IPV6.State      MTU  Type  DIS
 Eth1/0/0         001         Up                 Down         1497 L1/L2 No/No
```

----结束

配置文件

- RouterA 的配置文件

```
#
 sysname RouterA
#
isis 1
 network-entity 10.0000.0000.0001.00
#
interface Ethernet1/0/0
 ip address 10.1.1.1 255.255.255.0
 isis enable 1
 isis dis-priority 100
#
return
```

- RouterB 的配置文件

```
#
 sysname RouterB
#
isis 1
 network-entity 10.0000.0000.0002.00
#
interface Ethernet1/0/0
 ip address 10.1.1.2 255.255.255.0
```

```
 isis enable 1
#
return
```

- RouterC 的配置文件

```
#
 sysname RouterC
#
isis 1
 is-level level-1
 network-entity 10.0000.0000.0003.00
#
interface Ethernet1/0/0
 ip address 10.1.1.3 255.255.255.0
 isis enable 1
#
return
```

- RouterD 的配置文件

```
#
 sysname RouterD
#
isis 1
 is-level level-2
 network-entity 10.0000.0000.0004.00
#
interface Ethernet1/0/0
 ip address 10.1.1.4 255.255.255.0
 isis enable 1
#
return
```

7.18.3 配置 IS-IS IPv6 的基本功能示例

举例说明通过 IS-IS 协议实现 IPv6 网络互连的基本配置方法。

组网需求

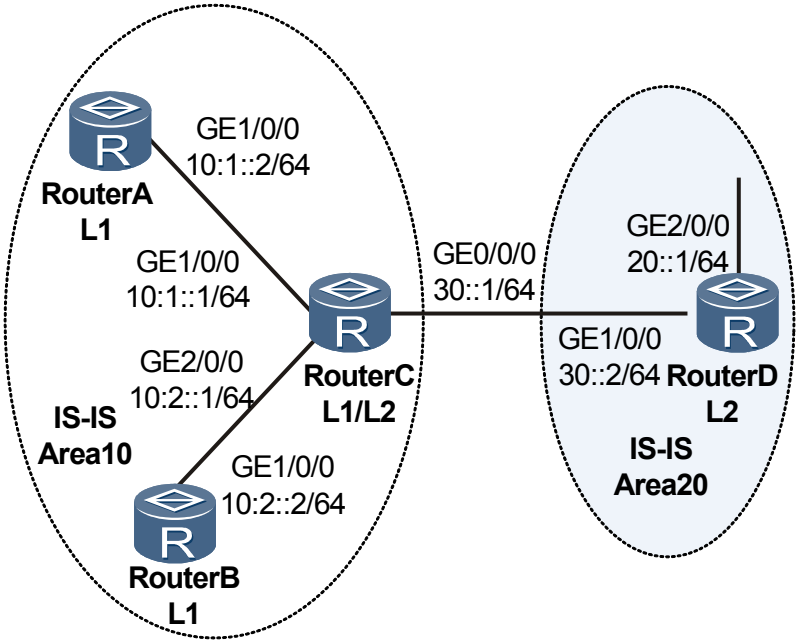
如图 7-5 所示：

- 路由器 RouterA、RouterB、RouterC 和 RouterD 属于同一自治系统，要求它们之间通过 IS-IS 协议达到 IPv6 网络互连的目的。
- RouterA、RouterB 和 RouterC 属于区域 10，RouterD 属于区域 20。
- RouterA 和 RouterB 是 Level-1 路由器，RouterC 是 Level-1-2 路由器，RouterD 是 Level-2 路由器。

说明

AR150/200 仅可作为 RouterA 或 RouterB。

图 7-5 配置 IS-IS 的 IPv6 特性基本功能组网图



配置思路

采用如下的思路配置 IS-IS IPv6 的基本功能：

1. 使能各路由器的 IPv6 转发能力，配置各接口的 IPv6 地址。
2. 在各路由器上使能 IS-IS，配置 Level 级别，指定网络实体。

数据准备

为完成此配置例，需准备如下的数据：

- RouterA、RouterB、RouterC 和 RouterD 各接口的 IPv6 地址。
- RouterA、RouterB、RouterC 和 RouterD 的区域号。
- RouterA、RouterB、RouterC 和 RouterD 的级别。

操作步骤

步骤 1 使能 IPv6 转发能力，配置各接口的 IPv6 地址，以 RouterA 为例，其他路由器的配置过程相同，不再赘述

```
<Huawei> system-view
[Huawei] sysname RouterA
[RouterA] ipv6
[RouterA] interface ethernet 1/0/0
[RouterA-Ethernet1/0/0] ipv6 enable
[RouterA-Ethernet1/0/0] ipv6 address 10:1::2/64
```

步骤 2 配置 IS-IS

配置 RouterA。

```
[RouterA] isis 1
```

```
[RouterA-isis-1] is-level level-1
[RouterA-isis-1] network-entity 10.0000.0000.0001.00
[RouterA-isis-1] ipv6 enable
[RouterA-isis-1] quit
[RouterA] interface ethernet 1/0/0
[RouterA-Ethernet1/0/0] isis ipv6 enable 1
[RouterA-Ethernet1/0/0] quit
```

配置 RouterB。

```
[RouterB] isis 1
[RouterB-isis-1] is-level level-1
[RouterB-isis-1] network-entity 10.0000.0000.0002.00
[RouterB-isis-1] ipv6 enable
[RouterB-isis-1] quit
[RouterB] interface ethernet 1/0/0
[RouterB-Ethernet1/0/0] isis ipv6 enable 1
[RouterB-Ethernet1/0/0] quit
```

配置 RouterC。

```
[RouterC] isis 1
[RouterC-isis-1] network-entity 10.0000.0000.0003.00
[RouterC-isis-1] ipv6 enable
[RouterC-isis-1] quit
[RouterC] interface ethernet 1/0/0
[RouterC-Ethernet1/0/0] isis ipv6 enable 1
[RouterC-Ethernet1/0/0] quit
[RouterC] interface ethernet 2/0/0
[RouterC-Ethernet2/0/0] isis ipv6 enable 1
[RouterC-Ethernet2/0/0] quit
[RouterC] interface ethernet 0/0/0
[RouterC-Ethernet0/0/0] isis ipv6 enable 1
[RouterC-Ethernet0/0/0] isis circuit-level level-2
[RouterC-Ethernet0/0/0] quit
```

配置 RouterD。

```
[RouterD] isis 1
[RouterD-isis-1] is-level level-2
[RouterD-isis-1] network-entity 20.0000.0000.0004.00
[RouterD-isis-1] ipv6 enable
[RouterD-isis-1] quit
[RouterD] interface ethernet 1/0/0
[RouterD-Ethernet1/0/0] isis ipv6 enable 1
[RouterD-Ethernet1/0/0] quit
[RouterD] interface ethernet 2/0/0
[RouterD-Ethernet2/0/0] isis ipv6 enable 1
[RouterD-Ethernet2/0/0] quit
```

步骤 3 验证配置结果

显示 RouterA 的 IS-IS 路由表。

```
[RouterA] display isis route
                      Route information for ISIS(1)
                      -----------------------------
                     ISIS(1) Level-1 Forwarding Table
                     --------------------------------
IPV4 Destination     IntCost    ExtCost ExitInterface   NextHop         Flags
-------------------------------------------------------------------------------
0.0.0.0/0            10         NULL
IPV6 Dest.      ExitInterface     NextHop                 Cost         Flags
-------------------------------------------------------------------------------
 ::/0           Eth1/0/0          FE80::A83E:0:3ED2:1     10           A/-/-
 10:1::/64      Eth1/0/0          Direct                  10           D/L/-
 10:2::/64      Eth1/0/0          FE80::A83E:0:3ED2:1     20           A/-/-
      Flags: D-Direct, A-Added to URT, L-Advertised in LSPs, S-IGP Shortcut,
                                U-Up/Down Bit Set
```

显示 RouterC 的 IS-IS 邻居的详细信息。

```
[RouterC] display isis peer verbose
                        Peer information for ISIS(1)
                        ----------------------------
  System Id    Interface          Circuit Id       State HoldTime Type     PRI
0000.0000.0001 Eth1/0/0           0000000001        Up   24s      L1       --
  MT IDs supported     : 0(UP)
  Local MT IDs         : 0
  Area Address(es)     : 10
  Peer IPv6 Address(es): FE80::996B:0:9419:1
  Uptime               : 00:44:43
  Adj Protocol         : IPV6
  Restart Capable      : YES,
  Suppressed Adj       : NO
0000.0000.0002 Eth2/0/0           0000000001        Up   28s      L1       --
  MT IDs supported     : 0(UP)
  Local MT IDs         : 0
  Area Address(es)     : 10
  Peer IPv6 Address(es): FE80::DC40:0:47A9:1
  Uptime               : 00:46:13
  Adj Protocol         : IPV6
  Restart Capable      : YES,
  Suppressed Adj       : NO
0000.0000.0004 Eth0/0/0           0000000001        Up   24s      L2       --
  MT IDs supported     : 0(UP)
  Local MT IDs         : 0
  Area Address(es)     : 20
  Peer IPv6 Address(es): FE80::F81D:0:1E24:2
  Uptime               : 00:53:18
  Adj Protocol         : IPV6
  Restart Capable      : YES,
  Suppressed Adj       : NO
Total Peer(s): 3
```

显示 RouterC 的 IS-IS LSDB 的详细信息。

```
[RouterC] display isis lsdb verbose
                       Database information for ISIS(1)
                       --------------------------------

                         Level-1 Link State Database
LSPID                 Seq Num      Checksum      Holdtime      Length  ATT/P/OL
-------------------------------------------------------------------------------
0000.0000.0001.00-00  0x0000000c   0x4e06        1117          113     0/0/0
 SOURCE       0000.0000.0001.00
 NLPID        IPV6
 AREA ADDR    10
 INTF ADDR V6 10:1::2
 Topology     Standard
 NBR  ID      0000.0000.0003.00  COST: 10
 IPV6         10:1::/64                     COST: 10
0000.0000.0002.00-00  0x00000009   0x738c        1022          83      0/0/0
 SOURCE       0000.0000.0002.00
 NLPID        IPV6
 AREA ADDR    10
 INTF ADDR V6 10:2::2
 Topology     Standard
 NBR  ID      0000.0000.0003.00  COST: 10
 IPV6         10:2::/64                     COST: 10
0000.0000.0003.00-00* 0x00000020   0x6b10        771           140     1/0/0
 SOURCE       0000.0000.0003.00
 NLPID        IPV6
 AREA ADDR    10
 INTF ADDR V6 30::1
 INTF ADDR V6 10:2::1
 INTF ADDR V6 10:1::1
 Topology     Standard
 NBR  ID      0000.0000.0002.00  COST: 10
 NBR  ID      0000.0000.0001.00  COST: 10
 IPV6         10:2::/64                     COST: 10
```

```
 IPV6         10:1::/64                       COST: 10
Total LSP (s) : 5
  *(In TLV) -Leaking Route, *(By LSPID) -Self LSP, +-Self LSP (Extended),
         ATT-Attached, P-Partition, OL-Overload
                      Level-2 Link State Database
LSPID                 Seq Num      Checksum      Holdtime      Length  ATT/P/OL
---------------------------------------------------------------------------
0000.0000.0003.00-00* 0x00000017   0x61b4        771           157     0/0/0
 SOURCE       0000.0000.0003.00
 NLPID        IPV6
 AREA ADDR    10
 INTF ADDR V6 30::1
 INTF ADDR V6 10:2::1
 INTF ADDR V6 10:1::1
 Topology     Standard
 NBR  ID      0000.0000.0004.00  COST: 10
 IPV6         30::/64                         COST: 10
 IPV6         10:2::/64                       COST: 10
 IPV6         10:1::/64                       COST: 10
0000.0000.0004.00-00  0x0000000b   0x6dfa        1024          124     0/0/0
 SOURCE       0000.0000.0004.00
 NLPID        IPV6
 AREA ADDR    20
 INTF ADDR V6 30::2
 INTF ADDR V6 20::1
 Topology     Standard
 NBR  ID      0000.0000.0003.00  COST: 10
 NBR  ID      0000.0000.0005.00  COST: 10
 IPV6         30::/64                         COST: 10
 IPV6         20::/64                         COST: 10
Total LSP (s) : 3
  *(In TLV) -Leaking Route, *(By LSPID) -Self LSP, +-Self LSP (Extended),
         ATT-Attached, P-Partition, OL-Overload
```

----结束

配置文件

- RouterA 的配置文件

```
#
 sysname RouterA
#
 ipv6
#
isis 1
 is-level level-1
 network-entity 10.0000.0000.0001.00
 #
 ipv6 enable topology standard
#
interface Ethernet1/0/0
 ipv6 enable
 ipv6 address 10:1::2/64
 isis ipv6 enable 1
#
return
```

- RouterB 的配置文件

```
#
 sysname RouterB
#
 ipv6
#
isis 1
 is-level level-1
 network-entity 10.0000.0000.0002.00
 #
 ipv6 enable topology standard
```

```
#
interface Ethernet1/0/0
 ipv6 enable
 ipv6 address 10:2::2/64
 isis ipv6 enable 1
#
return
```

- RouterC 的配置文件

```
#
 sysname RouterC
#
 ipv6
#
isis 1
 network-entity 10.0000.0000.0003.00
 #
 ipv6 enable topology standard
#
interface Ethernet0/0/0
 ipv6 enable
 ipv6 address 30::1/64
 isis ipv6 enable 1
 isis circuit-level level-2
#
interface Ethernet1/0/0
 ipv6 enable
 ipv6 address 10:1::1/64
 isis ipv6 enable 1
#
interface Ethernet2/0/0
 ipv6 enable
 ipv6 address 10:2::1/64
 isis ipv6 enable 1
#
return
```

- RouterD 的配置文件

```
#
 sysname RouterD
#
 ipv6
#
isis 1
 is-level level-2
 network-entity 20.0000.0000.0004.00
 #
 ipv6 enable topology standard
#
interface Ethernet2/0/0
 ipv6 enable
 ipv6 address 20::1/64
 isis ipv6 enable 1
#
interface Ethernet1/0/0
 ipv6 enable
 ipv6 address 30::2/64
 isis ipv6 enable 1
#
return
```

7.18.4 配置 IS-IS 快速收敛示例

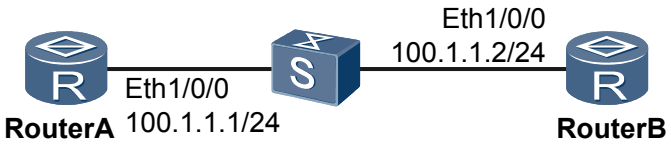
举例说明通过调整 IS-IS 定时器实现路由快速收敛的配置方法。

组网需求

如图 7-6 所示，

- 路由器 RouterA 和 RouterB 之间运行 IS-IS 协议。
- 路由器 RouterA 和 RouterB 同属于区域 10，都是 Level-2 路由器。
- RouterA 和 RouterB 之间有一台二层交换机，不需要配置。

图 7-6 配置 IS-IS 快速收敛组网图



配置思路

采用如下思路配置 IS-IS 快速收敛特性：

1. 在各路由器上使能 IS-IS 基本功能。
2. 在 RouterA 和 RouterB 上使能 BFD 检测机制。
3. 在 RouterA 和 RouterB 上配置快速收敛时间参数。

数据准备

为完成此配置例，需准备如下的数据：

- 两台路由器的级别和所属区域地址。
- 快速收敛时间参数值。

操作步骤

步骤 1 配置各路由器接口的 IP 地址（略）

步骤 2 配置 IS-IS 基本功能

配置 RouterA。

```
[RouterA] isis 1
[RouterA-isis-1] is-level level-2
[RouterA-isis-1] network-entity 10.0000.0000.0001.00
[RouterA-isis-1] quit
[RouterA] interface ethernet 1/0/0
[RouterA-Ethernet1/0/0] isis enable 1
[RouterA-Ethernet1/0/0] quit
```

配置 RouterB。

```
[RouterB] isis 1
[RouterB-isis-1] is-level level-2
[RouterB-isis-1] network-entity 10.0000.0000.0002.00
[RouterB-isis-1] quit
[RouterB] interface ethernet 1/0/0
[RouterB-Ethernet1/0/0] isis enable 1
[RouterB-Ethernet1/0/0] quit
```

步骤 3 配置 BFD 特性

配置 RouterA。

```
[RouterA] bfd
[RouterA-bfd] quit
[RouterA] bfd atob bind peer-ip 100.1.1.2 interface ethernet 1/0/0
[RouterA-bfd-session-atob] discriminator local 1
[RouterA-bfd-session-atob] discriminator remote 2
[RouterA-bfd-session-atob] commit
[RouterA-bfd-session-atob] quit
[RouterA] interface ethernet 1/0/0
[RouterA-Ethernet1/0/0] isis bfd static
[RouterA-Ethernet1/0/0] quit
```

配置 RouterB。

```
[RouterB] bfd
[RouterB-bfd] quit
[RouterB] bfd btoa bind peer-ip 100.1.1.1 interface ethernet 1/0/0
[RouterB-bfd-session-btoa] discriminator local 2
[RouterB-bfd-session-btoa] discriminator remote 1
[RouterB-bfd-session-btoa] commit
[RouterB-bfd-session-btoa] quit
[RouterB] interface ethernet 1/0/0
[RouterB-Ethernet1/0/0] isis bfd static
[RouterB-Ethernet1/0/0] quit
```

步骤 4 配置快速收敛时间参数

配置 RouterA。

```
[RouterA] isis 1
[RouterA-isis-1] flash-flood
[RouterA-isis-1] timer spf 1 20 100
[RouterA-isis-1] timer lsp-generation 1 1 120
[RouterA-isis-1] quit
```

配置 RouterB。

```
[RouterB] isis 1
[RouterB-isis-1] timer spf 1 20 100
[RouterB-isis-1] timer lsp-generation 1 1 120
[RouterB-isis-1] quit
```

说明

- IS-IS 协议中，当 LSDB 发生变化时需要进行路由计算、产生新的 LSP 来通告这种变化。频繁的路由计算会占用大量的系统资源，导致系统性能下降。延迟 SPF 计算和产生新的 LSP 时间，及快速 LSP 扩散，在一定程度上可以提高路由计算的效率，减少系统资源的消耗。
- **flash-flood** 命令用来使能 LSP 快速扩散特性，以便加快 IS-IS 网络的收敛速度。
- **timer spf** 命令用来设置 SPF 计算的间隔时间，缺省值是 5 秒。
- **timer lsp-generation** 命令用来设置产生 LSP 的延迟时间，缺省值是 2 秒。

步骤 5 检查配置结果

在 RouterB 的 Eth1/0/0 端口上执行命令 **shutdown**，模拟链路 Down 掉。

```
[RouterB] interface ethernet 1/0/0
[RouterB-Ethernet1/0/0] shutdown
```

在 RouterA 上查看邻居信息。

```
<RouterA> display isis peer
```

此时 RouterA 的邻居信息已经不存在。

当 BFD 感知链路 Down 掉时，上报给路由管理模块。IS-IS 立刻删除邻居，触发路由计算，使网络迅速收敛。

----结束

配置文件

- RouterA 的配置文件

```
#
 sysname RouterA
#
 bfd
#
isis 1
 is-level level-2
 timer lsp-generation 1 1 120 level-1
 timer lsp-generation 1 1 120 level-2
 flash-flood level-1
 flash-flood level-2
 network-entity 10.0000.0000.0001.00
 timer spf 1 20 100
#
interface Ethernet1/0/0
 ip address 100.1.1.1 255.255.255.0
 isis enable 1
 isis bfd static
#
bfd atob bind peer-ip 100.1.1.2 interface Ethernet1/0/0
 discriminator local 1
 discriminator remote 2
 commit
#
return
```

- RouterB 的配置文件

```
#
 sysname RouterB
#
 bfd
#
isis 1
 is-level level-2
 timer lsp-generation 1 1 120 level-1
 timer lsp-generation 1 1 120 level-2
 flash-flood level-1
 flash-flood level-2
 network-entity 10.0000.0000.0002.00
 timer spf 1 20 100
#
interface Ethernet1/0/0
 ip address 100.1.1.2 255.255.255.0
 isis enable 1
 isis bfd static
#
bfd btoa bind peer-ip 100.1.1.1 interface Ethernet1/0/0
 discriminator local 2
 discriminator remote 1
 commit
#
return
```

8 BGP 配置

关于本章

BGP 协议通常应用于大型和复杂的网络，用于在 AS 之间传递路由信息。

8.1 BGP 概述
BGP 协议主要用于控制路由的传播和选择最佳路由。

8.2 AR150/200 中支持的 BGP 特性
系统支持的 BGP 特性包括：路由聚合、对等体组、路由反射器、联盟、团体、MP-BGP、BGP ORF、BGP Tracking、路由衰减、负载分担、BGP 路径 MTU 自动发现、BGP 下一跳延时响应、BFD for BGP、BGP GR 和 BGP 安全。

8.3 配置 BGP 的基本功能
配置 BGP 基本功能是组建 BGP 网络的基础。

8.4 配置 BGP 的路由属性
BGP 具有很多路由属性，通过配置这些属性可以改变 BGP 的选路结果。

8.5 配置 BGP 发布路由
BGP 用来传递路由。在 BGP 发布路由时，可以灵活的对路由进行过滤或使用路由策略，只发布符合自己要求的路由，并且修改路由的属性，达到引导网络流量的目的。

8.6 配置 BGP 接收路由
BGP 用来传递路由。在 BGP 接收路由时，可以灵活的对路由进行过滤或使用路由策略，只接收符合自己需要的路由，并且修改路由的属性，达到引导网络流量的目的。

8.7 配置 BGP 路由聚合
配置路由聚合，可以减小对等体路由表中的路由数量。

8.8 配置 BGP 对等体组
通过配置 BGP 对等体组，可以简化 BGP 网络配置，提高路由的发布效率。

8.9 配置 BGP 路由反射器
通过配置 BGP 路由反射器，可以解决多个 IBGP 对等体建立全连接的问题，简化网络配置，提高路由发布效率。

8.10 配置 BGP 联盟
大型 BGP 网络中，配置联盟不但可以减少 IBGP 连接的数量，还可以简化路由策略的管理，提高路由的发布效率。

8.11 配置 BGP 团体属性
团体属性可以简化路由策略的管理。

8.12 配置基于前缀的 BGP ORF
通过配置基于前缀的 BGP ORF，可以将本端基于前缀的入口策略发送给对端，使对端在发送路由时应用该入口策略对路由进行过滤，实现路由按需发送。

8.13 调整 BGP 网络的收敛速度
通过调整 BGP 对等体间的连接参数，可以对 BGP 网络的收敛速度进行调整和优化，从而适应大型网络中网络状况不断变化的情况。

8.14 配置 BGP 路由衰减
通过配置 BGP 路由衰减，可以抑制不稳定的 BGP 路由。

8.15 配置向对等体发送缺省路由
配置向对等体发送缺省路由功能后，无论本地路由表中是否存在缺省路由，都将向指定对等体发布一条下一跳地址为本地地址的缺省路由。通过向对等体发送缺省路由，可以减少网络中的路由数量。

8.16 配置 BGP 负载分担
通过配置 BGP 负载分担，可以合理利用网络资源，减少网络拥塞。

8.17 配置路径 MTU 自动发现功能
通过配置路径 MTU 自动发现功能，可以发现从源端到目的端的路径上最小 MTU 值，使 BGP 消息按照路径 MTU 传输，提高传输效率，增强 BGP 性能。

8.18 配置 BGP 下一跳延时响应
通过配置 BGP 下一跳延时响应，可以减少路由变化时的流量丢失。

8.19 配置 BFD for BGP
通过配置 BFD for BGP 功能，为 BGP 提供更为快速的故障检测机制，提高网络收敛速度。

8.20 配置 BGP GR
通过配置 BGP GR 功能，能够避免因为协议重启而导致流量中断。

8.21 配置 BGP 安全性
为提高 BGP 的安全性，可以在建立 TCP 连接时进行认证。

8.22 BGP 维护
BGP 维护包括复位 BGP 连接和清除 BGP 的统计信息。

8.23 配置举例
BGP 配置举例包括组网需求、组网图、配置注意事项、配置思路和配置步骤。

8.1 BGP 概述

BGP 协议主要用于控制路由的传播和选择最佳路由。

BGP（Border Gateway Protocol）是一种用于自治系统 AS（Autonomous System）之间的动态路由协议。早期发布的三个版本分别是 BGP-1（RFC1105）、BGP-2（RFC1163）和 BGP-3（RFC1267），当前使用的版本是 BGP-4（RFC4271）。

BGP-4 作为事实上的 Internet 外部路由协议标准，被广泛应用于 ISP（Internet Service Provider）之间。

说明

下文中若不做特殊说明，所指的 BGP 均为 BGP-4。

BGP 特性描述如下：

- BGP 是一种外部网关协议（EGP），与 OSPF、RIP 等内部网关协议（IGP）不同，其着眼点不在于发现和计算路由，而在于控制路由的传播和选择最佳路由。
- BGP 使用 TCP 作为其传输层协议（端口号 179），提高了协议的可靠性。
- BGP 支持无类别域间路由 CIDR（Classless Inter-Domain Routing）。
- 路由更新时，BGP 只发送更新的路由，大大减少了 BGP 传播路由所占用的带宽，适用于在 Internet 上传播大量的路由信息。
- BGP 路由通过携带 AS 路径信息彻底解决路由环路问题。
- BGP 提供了丰富的路由策略，能够对路由实现灵活的过滤和选择。
- BGP 易于扩展，能够适应网络新的发展。

BGP 在路由器上以下列两种方式运行：

- IBGP（Internal BGP）
- EBGP（External BGP）

当 BGP 运行于同一自治系统内部时，被称为 IBGP；当 BGP 运行于不同自治系统之间时，称为 EBGP。

8.2 AR150/200 中支持的 BGP 特性

系统支持的 BGP 特性包括：路由聚合、对等体组、路由反射器、联盟、团体、MP-BGP、BGP ORF、BGP Tracking、路由衰减、负载分担、BGP 路径 MTU 自动发现、BGP 下一跳延时响应、BFD for BGP、BGP GR 和 BGP 安全。

几种主要的路由属性

- 源（Origin）属性
- AS 路径（AS_Path）属性
- 下一跳（Next_Hop）属性
- MED（Multi-Exit-Discriminator）属性
- 本地优先级（Local_Pref）属性
- 团体（Community）属性

BGP 选择路由的策略

在 AR150/200 的实现中，当到达同一目的地存在多条活跃路由时，BGP 采取如下策略进行路由选择：

1. 优选协议首选值（PrefVal）最高的路由。

   协议首选值（PrefVal）是华为设备的特有属性，该属性仅在本地有效。

2. 优选本地优先级（Local_Pref）最高的路由。

   如果路由没有本地优先级，BGP 选路时将该路由按缺省的本地优先级 100 来处理。通过执行 **default local-preference** 命令可以修改 BGP 路由的缺省本地优先级。

3. 优选本地生成的路由（本地生成的路由优先级高于从邻居学来的路由）。

   本地生成的路由包括通过 **network** 命令或 **import-route** 命令引入的路由、手动聚合路由和自动聚合路由。

   a. 优选聚合路由（聚合路由优先级高于非聚合路由）。

   b. 通过 **aggregate** 命令生成的手动聚合路由的优先级高于通过 **summary automatic** 命令生成的自动聚合路由。

   c. 通过 **network** 命令引入的路由的优先级高于通过 **import-route** 命令引入的路由。

4. 优选 AS 路径（AS_Path）最短的路由（AS_Path 中 AS 号的个数最少）。

   - AS_Path 的长度不包括 AS_CONFED_SEQUENCE 和 AS_CONFED_SET。
   - AS_SET 的长度为 1，无论 AS_SET 中包括多少 AS 号，都将算作一个 AS 号。
   - 执行 **bestroute as-path-ignore** 命令后，BGP 选路时，忽略 AS_Path 的比较。

5. 比较 Origin 属性，依次优选 Origin 类型为 IGP、EGP、Incomplete 的路由。

6. 优选 MED（Multi Exit Discriminator）值最低的路由。

   - BGP 只比较来自同一个 AS（不包括联盟的子 AS）的路由的 MED 值。即，只有两条路由的 AS_SEQUENCE（不包括 AS_CONFED_SEQUENCE）属性的第一个 AS 号相同时，BGP 才会比较二者的 MED 值。
   - 如果路由没有 MED 属性，BGP 选路时将该路由的 MED 值按缺省值 0 来处理；执行 **bestroute med-none-as-maximum** 命令后，BGP 选路时将该路由的 MED 值按最大值 4294967295 来处理。
   - 执行 **compare-different-as-med** 命令后，BGP 将强制比较来自不同自治系统中的邻居的路由的 MED 值。除非能够确认不同的自治系统采用了同样的 IGP 和路由选择方式，否则不要使用 **compare-different-as-med** 命令（可能产生环路）。
   - 执行 **bestroute med-confederation** 命令后，只有当 AS_Path 中不包含外部 AS 号（不属于联盟的子 AS），且 AS_CONFED_SEQUENCE 的第一个 AS 号相同时，才能比较 MED 值的大小。
   - 执行 **deterministic-med** 命令后，将消除路由接收顺序对选路结果的影响。

7. 优选从 EBGP 邻居学来的路由（EBGP 路由优先级高于 IBGP 路由）。

   依次优选 EBGP 路由、IBGP 路由、本地交叉（LocalCross）、远端交叉（RemoteCross）路由。

   本地 PE 从其他 PE 学习到的 VPNv4 路由，根据其属性中的 ERT 值，逐个与本地的 VPN 实例下的 IRT 进行匹配。如果能够匹配则将这条路由复制一份到该 VPN 实例下，称为远端交叉（RemoteCross）。 在同一个 PE 上，PE 将某一个 VPN 实例下的路由根据 ERT、IRT 的匹配规则（和远端交叉的规则一样）复制到其他 VPN 实例下，称为本地交叉（LocalCross）。

8. 优选到 BGP 下一跳 IGP Metric 较小的路由。

   说明

   如果配置了负载分担，当上述所有规则相同，且存在多条 As_Path 完全相同的外部路由，则根据配置的路由条数选择多条路由进行负载分担。

9. 优选 Cluster_List 最短的路由。
10. 优选 Router ID 最小的路由器发布的路由。

    说明

    如果路由携带 Originator_ID 属性，选路过程中将比较 Originator_ID 的大小（不再比较 Router ID），并优选 Originator_ID 最小的路由。

11. 比较对等体的 IP Address，优选从具有较小 IP Address 的对等体学来的路由。

BGP 发布路由的策略

在 AR150/200 的实现中，BGP 发布路由时采用如下策略：

- 存在多条活跃路由时，BGP 发言者（BGP Speaker）只将最优路由发布给对等体；
- BGP 发言者只把自己优选的路由发布给对等体；
- BGP 发言者从 EBGP 获得的路由会向它所有 BGP 对等体发布，但不会向通告该路由的对等体发布（包括 EBGP 对等体和 IBGP 对等体）；
- BGP 发言者从 IBGP 获得的路由不向它的 IBGP 对等体发布；
- BGP 发言者从 IBGP 获得的路由发布给它的 EBGP 对等体；
- 连接一旦建立，BGP 发言者将把自己所有 BGP 优选的路由发布给新对等体。

应用 BGP 负载分担时的选路策略

在 BGP 中，由于协议本身的特殊性，它产生的路由的下一跳地址可能不是当前路由器直接相连的邻居。常见的一个场景是，IBGP 之间发布路由信息时不改变下一跳。这种情况下，为了能够将报文正确转发出去，路由器必须先找到一个直接可达的地址，通过这个地址到达路由表中指示的下一跳。在上述过程中，去往直接可达地址的路由被称为依赖路由，BGP 路由依赖于这些路由指导报文转发。根据下一跳地址找到依赖路由的过程就是路由迭代（iteration）。

AR150/200 支持基于迭代的 BGP 负载分担，即，如果依赖路由本身是负载分担的（假设有三个下一跳地址），则 BGP 也会生成相同数量的下一跳地址来指导报文转发。需要说明的是，基于迭代的 BGP 负载分担并不需要命令配置，这一特性在 AR150/200 上始终启用。

BGP 的负载分担与 IGP 的负载分担在实现方法上有所不同：

- 在 IGP 中，对到达同一目的地址的不同路由，IGP 根据本身的路由算法计算路由的度量值（metric），在度量值相等的路由间进行负载分担。
- BGP 由于本身并没有路由算法，不能根据一个明确的度量值决定是否对路由进行负载分担。但 BGP 有很多路由属性，这些属性在 BGP 选路策略中的优先顺序是不同的。对 BGP 负载分担的处理则是加入到这些选路策略中的，即在所有高优先级路由属性相同的情况下，BGP 根据所配置的最大负载分担的路由条数进行负载分担。

说明

- 缺省情况下，BGP 只对 AS_Path 属性完全相同的路由进行负载分担。可以使用 **bestroute as-path-ignore** 命令配置 BGP 在进行负载分担时不比较路由的 AS_Path 属性。
- BGP 负载分担特性同样适用于联盟内部的自治系统之间。

路由聚合

在大规模的网络中，BGP 路由表十分庞大，使用路由聚合（Route Aggregation）可以大大减小路由表的规模。

路由聚合实际上是将多条路由合并的过程。这样 BGP 在向对等体通告路由时，可以只通告聚合后的路由，而不是将所有的具体路由都通告出去。

AR150/200 支持自动聚合和手动聚合方式。使用后者还可以控制聚合路由的属性，以及决定是否发布具体路由。

IBGP 和 IGP 同步

同步是指 IBGP 和 IGP 之间的同步，其目的是为了避免出现误导外部 AS 路由器的现象。

如果设置了同步特性，在 IBGP 路由加入路由表并发布给 EBGP 对等体之前，会先检查 IGP 路由表。只有在 IGP 也知道这条 IBGP 路由时，它才会被加入到路由表，并发布给 EBGP 对等体。

在下面的情况中，可以安全地关闭同步特性。

- 本 AS 不是过渡 AS
- 本 AS 内所有路由器建立 IBGP 全连接

说明

缺省情况下，AR150/200 的同步功能是关闭的。

对等体组

对等体组（Peer Group）是一些具有某些相同策略的对等体的集合。当一个对等体加入对等体组中时，此对等体将获得与所在对等体组相同的配置。当对等体组的配置改变时，组内成员的配置也相应改变。

在大型 BGP 网络中，对等体的数量会很多，其中很多对等体具有相同的策略，在配置时会重复使用一些命令，利用对等体组在很多情况下可以简化配置。

另外，将多个对等体加入同一对等体组中还可以提高路由发布效率。

路由反射器

为保证 IBGP 对等体之间的路由同步，需要在 IBGP 对等体之间建立全连接关系。假设在一个 AS 内部有 n 台路由器，那么应该建立的 IBGP 连接数就为 n(n-1)/2。当 IBGP 对等体数目很多时，对网络资源和 CPU 资源的消耗都很大。

利用路由反射可以解决这一问题。在一个 AS 内，其中一台路由器作为路由反射器 RR（Route Reflector），其它路由器作为客户机（Client）与路由反射器之间建立 IBGP 连接。路由反射器在客户机之间传递（反射）路由信息，各个客户机之间不需要建立 BGP 连接。

既不是反射器也不是客户机的 BGP 路由器被称为非客户机（Non-Client）。非客户机与路由反射器之间，以及所有的非客户机之间仍然必须建立全连接关系。

联盟

联盟（Confederation）是处理 AS 内部的 IBGP 网络连接激增的另一种方法，它将一个自治系统划分为若干个子自治系统，每个子自治系统内部的 IBGP 对等体建立全连接关系，子自治系统之间建立 EBGP 连接关系。

在不属于联盟的 BGP 发言者看来，属于同一个联盟的多个子自治系统是一个整体，外界不需要了解内部的子自治系统情况，联盟 ID 就是标识联盟这一整体的自治系统号。

联盟的缺陷是：从非联盟向联盟方案转变时，要求路由器重新进行配置，逻辑拓扑也要改变。

在大型 BGP 网络中，路由反射器和联盟可以被同时使用。

团体

对等体组可以使一组对等体共享相同的策略，而利用团体属性可以使多个 AS 中的一组 BGP 路由器共享相同的策略。团体是一个路由属性，在 BGP 对等体之间传播，它并不受到 AS 范围的限制。

BGP 路由器在将带有团体属性的路由发布给其它对等体之前，可以改变此路由原有的团体属性。

除了使用公认的团体属性外，用户还可以使用团体属性过滤器过滤自定义扩展团体属性，以便更为灵活的控制路由策略。

MP-BGP 概述

传统的 BGP-4 只能管理 IPv4 单播路由信息，对于使用其它网络层协议（如 IPv6 等）的应用，在跨自治系统传播时就受到一定限制。

为了提供对多种网络层协议的支持，IETF 对 BGP-4 进行了扩展，形成 MP-BGP，目前的 MP-BGP 标准是 RFC2858（Multiprotocol Extensions for BGP-4，BGP-4 的多协议扩展）。

MP-BGP 前向兼容，即支持 BGP 扩展的路由器与不支持 BGP 扩展的路由器可以互通。

MP-BGP 的扩展属性

BGP-4 使用的报文中，与 IPv4 相关的三种路由属性都由 Update 报文携带，这三种路由属性分别是：NLRI（Network Layer Reachability Information）、路径属性中的 Next_Hop、路径属性中的 Aggregator（该属性中包含形成聚合路由的 BGP 发言者的 IP 地址）。

为实现对多种网络层协议的支持，BGP-4 需要将网络层协议的信息反映到 NLRI 及 Next_Hop。MP-BGP 中引入了两个新的路径属性：

- MP_REACH_NLRI：Multiprotocol Reachable NLRI，多协议可达 NLRI。用于发布可达路由及下一跳信息。
- MP_UNREACH_NLRI：Multiprotocol Unreachable NLRI，多协议不可达 NLRI。用于撤销不可达路由。

这两种属性都是可选非过渡（Optional non-transitive）的，因此，不提供多协议能力的 BGP 发言者将忽略这两个属性，不把它们传递给其它邻居。

地址族

BGP 采用地址族（Address Family）来区分不同的网络层协议，关于地址族的一些取值可以参考 RFC1700（Assigned Numbers）。AR150/200 实现多种 MP-BGP 扩展应用，例如对 VPN 的扩展、对 IPv6 的扩展等，不同的扩展应在各自的地址族视图下配置。

说明

本章不对 MP-BGP 地址族视图下的、与特定应用相关的命令作详细介绍。

BGP IPv6 地址族下的配置请参见“BGP4+配置”，MP-BGP 在组播中的应用请参见《Huawei AR150&200 系列企业路由器 配置指南-IP 组播》中“MBGP 配置”。

BGP VPNv4 地址族、BGP VPN 实例地址族和 BGP L2VPN 地址族下的配置请参见《Huawei AR150&200 系列企业路由器 配置指南-VPN》。

BGP ORF

基于地址前缀列表的 BGP ORF（Outbound Route Filters）用于实现 BGP 路由的按需发布。设备按照出口策略（目前仅支持地址前缀列表 IP-Prefix List）在 BGP 发布路由时对路由进行过滤，只发布对端需要的路由，减少了不必要路由的发布。另外，这个出口策略由对端设备（路由接收者）提供，本端设备无须为每一个 BGP 邻居都维护一个出口策略，这样也就大大减轻了本端设备的负担，同时减少了很多的配置工作量。

BGP Tracking

BGP tracking 功能，可以通过调整从发现邻居不可达到中断连接的时间间隔，来调整 BGP 网络的收敛速度，而且该功能部署简单，扩展性好。

BGP 衰减

路由衰减（Route Dampening）用来解决路由不稳定的问题。路由不稳定的主要表现形式是路由振荡（Route Flapping），即路由表中的某条路由反复消失和重现。

发生路由振荡时，路由协议就会向邻居发布路由更新，收到更新报文的路由器需要重新计算路由并修改路由表。所以频繁的路由振荡会消耗大量的带宽资源和 CPU 资源，严重时会影响到网络的正常工作。

在多数情况下，BGP 协议都应用于复杂的网络环境中，路由变化比较频繁。为了防止持续的路由振荡带来的不利影响，BGP 使用衰减来抑制不稳定的路由。

BGP 路径 MTU 自动发现

BGP 路径 MTU 自动发现功能可以发现从源端到目的端的路径上最小 MTU 值（简称为路径 MTU），从而使 TCP 在传输 BGP 消息时按照路径 MTU 值进行传输，可以提高 BGP 消息的传输效率。

BGP 下一跳延时响应

连接到 RR 的 PE 设备在上游出现路由路径变化时，BGP 下一跳延时响应可以加快 BGP 收敛速度，减少流量的丢失。

BFD for BGP

AR150/200 支持使用 BFD（Bidirectional Forwarding Detection）为 BGP 邻居关系提供更快速的链路故障检测。

通过 BFD 检测 BGP 对等体间的链路故障，并报告给 BGP 协议，可以实现 BGP 路由的快速收敛。

BGP GR

当 BGP 协议重启时会导致对等体关系重新建立和转发中断，使能平滑重启 GR（Graceful Restart）功能后可以避免流量中断。

BGP 安全

- 通过 MD5 和 Key-Chain 对 BGP 邻居的合法性进行验证，防止报文假冒和非法篡改。
- GTSM（Generalized TTL Security Mechanism）机制通过 TTL 的检测来达到防止攻击的目的。GTSM 通过检测 IP 报文头中的 TTL 值是否在一个预先定义好的特定范围内，对 IP 层以上业务进行保护，增强系统的安全性。
- 限制从对等体接收的路由数量，防止资源耗尽性攻击。请参考：**8.6.3 配置控制 BGP 路由信息的接收**。
- AS_Path 长度保护。通过在入口和出口两个方向对 AS_Path 的长度进行限定，直接丢弃 AS_Path 超限的报文。请参考：**8.4.7 配置 AS_Path 属性**。

8.3 配置 BGP 的基本功能

配置 BGP 基本功能是组建 BGP 网络的基础。

8.3.1 建立配置任务

在配置 BGP 的基本功能前了解此特性的应用环境、配置此特性的前置任务和数据准备，有助于快速、准确地完成配置任务。

应用环境

为了实现网络中 AS 间的通信，可以在网络中配置 BGP 协议。本节讲述最基本的 BGP 网络配置过程。

由于 BGP 使用 TCP 连接，所以在配置 BGP 时需要指定对等体的 IP 地址。BGP 对等体不一定是相邻的路由器，利用逻辑链路也可以建立 BGP 对等体关系。为了增强 BGP 连接的稳定性，推荐使用 Loopback 接口地址建立连接。

BGP 的基本功能包括三个主要部分，即创建 BGP 进程，建立 BGP 对等体关系和引入路由：

- 创建 BGP 进程：创建 BGP 进程是配置所有 BGP 特性的首要步骤。
- 建立 BGP 对等体关系：只有对等体关系建立成功后，设备之间才可以交换 BGP 路由信息。
- 引入路由：BGP 协议本身不会发现路由，只有引入其他协议的路由，才能产生 BGP 路由，实现 AS 间通信。

说明

在本节中，不对 BGP 和 MP-BGP 进行严格的区分，命令的适用情况请参考所在的视图。

为方便配置，BGP-IPv4 单播地址族视图下的命令可以在 BGP 视图下执行，但在配置文件中这些命令仍位于 BGP-IPv4 单播地址族视图下。

前置任务

在配置 BGP 基本功能之前，需完成以下任务：

- 配置接口的链路层协议参数（和 IP 地址），使接口的链路协议状态为 Up

数据准备

在配置 BGP 的基本功能之前，需要准备以下数据。

序号	数据
1	本地 AS 号和 Router ID
2	对等体的 IPv4 地址和 AS 号
3	更新报文的源接口

8.3.2 启动 BGP 进程

启动 BGP 进程是配置所有 BGP 特性的首要步骤。启动 BGP 进程时需指定设备所属的 AS 号。

背景信息

请在需要建立 BGP 连接的路由器上进行下列配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **bgp** *as-number*，启动 BGP（指定本地 AS 编号），进入 BGP 视图。

步骤 3 （可选）执行命令 **router-id** *ipv4-address*，配置 BGP 的 Router ID。

配置或改变 BGP 的 Router ID 会导致路由器之间的 BGP Peer 关系重置。

窍门

为了增加网络的可靠性，建议将 Router ID 手工配置为 Loopback 接口的地址。如果没有配置，则 BGP 会自动选取系统视图下的 Router ID 作为 BGP 协议的 Router ID。系统视图下的 Router ID 选择规则，请参见命令 **router-id** 中的描述。

----结束

8.3.3 配置 BGP 对等体

配置 BGP 对等体且对等体建立成功后，设备之间才可以交换 BGP 路由信息。

背景信息

由于 BGP 使用 TCP 连接，所以在配置 BGP 时需要指定对等体的 IP 地址。BGP 对等体不一定是相邻的路由器，利用逻辑链路也可以建立 BGP 对等体关系。为了增强 BGP 连接的稳定性，推荐使用 Loopback 接口地址建立连接。

属于同一 AS 的设备之间配置 IBGP 对等体，属于不同 AS 的设备之间配置 EBGP 对等体。

操作步骤

- 配置 IBGP 对等体
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **bgp** *as-number*，进入 BGP 视图。
  3. 执行命令 **peer** *ipv4-address* **as-number** *as-number*，指定对等体的 IP 地址及其所属的 AS 编号。

     指定对等体所属的 AS 编号应该和本地 AS 号相同。

     指定的对等体的 IP 地址可以是以下三种：

     - 直连对等体的接口 IP 地址。
     - 路由可达的对等体的 Loopback 接口地址。
     - 直连对等体的子接口的 IP 地址。
  4. （可选）执行命令 **peer** *ipv4-address* **connect-interface** *interface-type interface-number* [*ipv4-source-address*]，指定 BGP 连接所使用的建立 TCP 连接会话的源接口和源地址。

     缺省情况下，BGP 使用与邻居直连的物理接口作为 TCP 连接的本地接口。

     说明

     当使用 Loopback 接口或子接口 IP 地址建立 BGP 连接时，建议对等体两端同时配置命令 **peer connect-interface**，以保证两端连接的正确性。如果仅有一端配置命令，可能会导致 BGP 连接建立失败。
  5. （可选）执行命令 **peer** *ipv4-address* **description** *description-text*，配置对等体的描述信息。

     通过配置描述信息可以方便网络管理。
- 配置 EBGP 对等体
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **bgp** *as-number*，进入 BGP 视图。
  3. 执行命令 **peer** *ipv4-address* **as-number** *as-number*，指定对等体的 IP 地址及其所属的 AS 编号。

     指定对等体所属的 AS 编号应该和本地 AS 号不同。

     指定的对等体的 IP 地址可以是以下三种：

     - 直连对等体的接口 IP 地址；
     - 路由可达的对等体的 Loopback 接口地址；
     - 直连对等体的子接口的 IP 地址。
  4. （可选）执行命令 **peer** *ipv4-address* **connect-interface** *interface-type interface-number* [*ipv4-source-address*]，指定 BGP 连接所使用的建立 TCP 连接会话的源接口和源地址。

     缺省情况下，BGP 使用与邻居直连的物理接口作为 TCP 连接的本地接口。

     说明

     当使用 Loopback 接口或子接口 IP 地址建立 BGP 连接时，建议对等体两端同时配置命令 **peer connect-interface**，以保证两端连接的正确性。如果仅有一端配置命令，可能会导致 BGP 连接建立失败。

5. （可选）执行命令 **peer** *ipv4-address* **ebgp-max-hop** [*hop-count*]，配置 EBGP 连接的最大跳数。

   参数 *hop-count* 的缺省值为 255。

   通常情况下，EBGP 对等体之间必须具有直连的物理链路，如果不满足这一要求，则必须使用 **peer ebgp-max-hop** 命令允许它们之间经过多跳建立 TCP 连接。

   说明

   BGP 使用 Loopback 口建立 EBGP 邻居时，必须配置命令 **peer ebgp-max-hop**（其中 *hop-count*≥2），否则邻居无法建立。

6. （可选）执行命令 **peer** *ipv4-address* **description** *description-text*，配置对等体的描述信息。

   通过配置描述信息可以方便网络管理。

----结束

8.3.4 配置 BGP 引入路由

BGP 可以引入其它协议的路由。当引入动态路由协议时，需要指定协议进程号。

背景信息

BGP 协议自身不能发现路由，所以需要将其他协议的路由（如 IGP 或者静态路由等）引入到 BGP 路由表中，从而将这些路由在 AS 之内和 AS 之间传播。

BGP 引入路由时支持 Import 和 Network 两种方式：

- Import 方式是按协议类型，将 RIP 路由、OSPF 路由、ISIS 路由、静态路由和直连路由等协议的路由注入到 BGP 路由表中。
- Network 方式比 Import 方式更精确，将指定前缀和掩码的一条路由注入到 BGP 路由表中。

操作步骤

- Import 方式
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **bgp** *as-number*，进入 BGP 视图。
  3. （可选）执行命令 **ipv4-family unicast**，进入 BGP-IPv4 单播地址族视图。

     缺省情况下，系统默认配置在 IPv4 单播地址族视图下。
  4. 执行命令 **import-route** *protocol* [*process-id*] [**med** *med* | **route-policy** *route-policy-name*] *，配置 BGP 引入其他协议的路由。

     通过配置 *med* 参数，可以指定引入路由的 MED 度量值。EBGP 对等体在判断流量进入 AS 选路时将选择 MED 最小的路由。

     通过配置 **route-policy** *route-policy-name* 参数，可以对从其他协议引入的路由进行过滤。

     说明

     引入 IS-IS、OSPF 或 RIP 路由时，需要指定协议进程号。

5. （可选）执行命令 **default-route imported**，允许 BGP 引入缺省路由。

   default-route imported 命令需要与 **import-route** 命令配合使用，才能引入缺省路由。因为单独使用 **import-route** 命令无法引入缺省路由，且 **default-route imported** 命令只用于引入本地 IP 路由表中已经存在的缺省路由。

- Network 方式
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **bgp** *as-number*，进入 BGP 视图。
  3. （可选）执行命令 **ipv4-family unicast**，进入 BGP-IPv4 单播地址族视图。

     缺省情况下，系统默认配置在 IPv4 单播地址族视图下。
  4. 执行命令 **network** *ipv4-address* [*mask* | *mask-length*] [**route-policy** *route-policy-name*]，配置 BGP 引入本地路由。

     如果没有指定掩码或掩码长度，则按有类地址处理。

     要引入的本地路由必须存在于本地的 IP 路由表中。

     使用路由策略可以更为灵活的控制所引入的路由。

     说明

     - **network** 命令中指定的目的地址和掩码必须与本地 IP 路由表中对应的表项完全一致，否则不能引入指定路由。
     - 使用 **undo network** 命令删除已有的配置时，需要注意指定正确的掩码。

----结束

8.3.5 检查配置结果

BGP 的基本功能配置成功后，可以查看 BGP 对等体信息与 BGP 路由信息。

前提条件

已经完成 BGP 的基本功能的所有配置。

操作步骤

- 使用 **display bgp peer** [**verbose**]命令查看所有 BGP 对等体的信息。
- 使用 **display bgp peer** *ipv4-address* { **log-info** | **verbose** }命令查看指定 BGP 对等体的相关信息。
- 使用 **display bgp routing-table** [*ipv4-address* [*mask* | *mask-length*]]令查看 BGP 路由信息。

----结束

8.4 配置 BGP 的路由属性

BGP 具有很多路由属性，通过配置这些属性可以改变 BGP 的选路结果。

8.4.1 建立配置任务

在配置 BGP 的路由属性前了解此特性的应用环境、配置此特性的前置任务和数据准备，有助于快速、准确地完成配置任务。

应用环境

BGP 具有很多路由属性，通过配置这些属性可以改变 BGP 的选路结果。

- BGP 协议优先级

  通过配置 BGP 协议优先级，可以影响 RM 对 BGP 和其他路由协议之间进行路由选路。
- BGP 路由信息的首选值

  通过配置路由信息首选值，当 BGP 路由表中存在到相同目的地址的路由时，优先选择首选值高的路由。
- Local_Pref 属性

  通过配置 Local_Pref 属性值，作用同路由信息首选值，但优先级比它低。
- MED 属性

  通过配置 MED 属性，用于判断流量进入 AS 时的最佳路由，在其它条件相同的情况下，将优先选择 MED 值较小者作为最佳路由。
- Next_Hop 属性

  利用 Next_Hop 属性的变化，可以灵活控制 BGP 的路由选择。
- AS_Path 属性

  AS_Path 属性用于防止路由环路和控制路由选择。

前置任务

在配置 BGP 的选路策略之前，需完成以下任务：

- 配置接口的网络层地址，使相邻节点的网络层可达
- **配置 BGP 的基本功能**

数据准备

在配置 BGP 选路策略之前，需要准备以下数据。

序号	数据
1	AS 号
2	BGP 协议优先级的值
3	Local_Pref 值
4	MED 值

8.4.2 配置 BGP 协议优先级

通过配置 BGP 协议优先级，可以影响 BGP 和其他路由协议间的路由选择。

背景信息

由于路由器上可能同时运行多个动态路由协议，就存在各个路由协议之间路由信息共享和选择的问题。系统为每一种路由协议设置一个缺省优先级。在不同协议发现同一条路由时，优先级高的路由将被优选。

请在运行 BGP 协议的路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **bgp** *as-number*，进入 BGP 视图。

步骤 3 执行命令 **ipv4-family unicast**，进入 IPv4 单播地址族视图。

步骤 4 执行命令 **preference** { *external internal local* | **route-policy** *route-policy-name* }，设定 BGP 协议的优先级。

配置优先级的值越小，优先级越高。

BGP 有三种路由：

- 从外部对等体学到的路由（EBGP）
- 从内部对等体学到的路由（IBGP）
- 本地产生的路由（Local Origined），是指通过聚合命令（**summary automatic** 自动聚合和 **aggregate** 手动聚合）所聚合的路由。

可以为这三种路由设定不同的优先级。

另外，还可以通过应用路由策略，为符合匹配条件的特定路由配置优先级。对于不符合匹配条件的路由，则使用缺省优先级。

说明

目前不支持通过 **peer route-policy** 命令在对等体上应用路由策略来设置 BGP 协议的优先级。

----结束

8.4.3 配置 BGP 路由信息的首选值

通过配置路由信息首选值，当 BGP 路由表中存在到相同目的地址的路由时，优先选择首选值高的路由。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **bgp** *as-number*，进入 BGP 视图。

步骤 3 执行命令 **peer** { *group-name* | *ipv4-address* } **preferred-value** *value*，为对等体配置首选值。

缺省情况下，从邻居学来的路由的初始首选值为 0。

当到达同一地址前缀有多条路由时，优先选择首选值大的路由。

----结束

8.4.4 配置本机的缺省 Local_Pref 属性值

通过配置 Local_Pref 属性值，判断流量离开 AS 时的最佳路由。

背景信息

Local_Pref 属性用于判断流量离开 AS 时的最佳路由。当 BGP 的设备通过不同的 IBGP 对等体得到目的地址相同但下一跳不同的多条路由时，将优先选择 Local_Pref 属性值较高的路由。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **bgp** *as-number*，进入 BGP 视图。

步骤 3 执行命令 **ipv4-family unicast**，进入 IPv4 单播地址族视图。

步骤 4 执行命令 **default local-preference** *preference*，配置本机的缺省 Local_Pref 属性值。

----结束

8.4.5 配置 MED 属性

MED 属性相当于 IGP 使用的度量值。通过配置 MED 属性，用于判断流量进入 AS 时的最佳路由，在其它条件相同的情况下，将优先选择 MED 值较小者作为最佳路由。

背景信息

MED 属性相当于 IGP 使用的度量值（Metrics），它用于判断流量进入 AS 时的最佳路由。当一个运行 BGP 的路由器通过不同的 EBGP 对等体得到目的地址相同但下一跳不同的多条路由时，在其它条件相同的情况下，将优先选择 MED 值较小者作为最佳路由。

操作步骤

- 配置本地设备的缺省 MED 值

  请在运行 BGP 协议的路由器上进行以下配置。

  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **bgp** *as-number*，进入 BGP 视图。
  3. 执行命令 **ipv4-family unicast**，进入 IPv4 单播地址族视图。
  4. 执行命令 **default med** *med*，配置缺省 MED 值。

     说明

     default med 命令只对本路由器上用 **import-route** 命令引入的路由和 BGP 的聚合路由生效。

- 比较来自不同 AS 的路由的 MED 值

  请在运行 BGP 协议的路由器上进行以下配置。

  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **bgp** *as-number*，进入 BGP 视图。
  3. 执行命令 **ipv4-family unicast**，进入 IPv4 单播地址族视图。

4. 执行命令 **compare-different-as-med**，比较来自不同 AS 的 MED 值。

   一般情况下，BGP 路由器只比较来自同一 AS（不同对等体）的路由的 MED 属性值。可以通过配置命令来允许 BGP 比较来自不同 AS 的路由的 MED 属性值。

- 配置 Deterministic-MED 功能

  请在运行 BGP 协议的路由器上进行以下配置。

  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **bgp** *as-number*，进入 BGP 视图。
  3. 执行命令 **ipv4-family unicast**，进入 IPv4 单播地址族视图。
  4. 执行命令 **deterministic-med**，使能 Deterministic-MED 的功能。

     未配置此命令时，在对从多个不同 AS 接收到的相同前缀的路由进行选路时，选路的结果和路由收来的顺序相关。配置了该命令后，在对从多个不同 AS 收来的相同前缀的路由进行选路时，会按路由 AS_Path 中的最左 AS 进行分组。在相同最左 AS 的组内进行比较后，再用组中的最优路由和其他组内的最优路由进行比较，从而消除了选路的结果和路由接收顺序的相关性。

- 配置 MED 值丢失时的处理方式

  请在运行 BGP 协议的路由器上进行以下配置。

  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **bgp** *as-number*，进入 BGP 视图。
  3. 执行命令 **ipv4-family unicast**，进入 IPv4 单播地址族视图。
  4. 执行命令 **bestroute med-none-as-maximum**，设置当路由没有 MED 值时将其作为最大值处理。

     当路由属性中没有 MED 值时，如果配置了该命令，则 BGP 在选路时将 MED 值作为最大值处理，否则将 MED 当作 0 处理。

- 比较联盟内路由的 MED 值

  请在运行 BGP 协议的路由器上进行以下配置。

  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **bgp** *as-number*，进入 BGP 视图。
  3. 执行命令 **ipv4-family unicast**，进入 IPv4 单播地址族视图。
  4. 执行命令 **bestroute med-confederation**，比较联盟内路由的 MED 值。

----结束

8.4.6 配置 Next_Hop 属性

利用 Next_Hop 属性的变化，可以灵活控制 BGP 的路由选择。

操作步骤

- 向 IBGP 对等体发布路由时，修改下一跳地址

  当设备通过 EBGP 邻居学到路由再转发给其他 IBGP 邻居时，默认不修改下一跳，但其 EBGP 邻居发来的路由的下一跳都是其 EBGP 邻居的 Peer 地址，本端对等体所属 AS 域内的 IBGP 邻居收到这样的路由后，由于下一跳不可达导致路由无法活

跃。因此，需要在 ASBR 上修改 EBGP 邻居发来的路由的 Next_Hop，改变下一跳地址，使得发给 IBGP 邻居的路由的下一跳是其自身的地址，IBGP 邻居收到这样的路由后（由于域内都配置了 IGP）发现下一跳可达，路由即为活跃路由。

1. 执行命令 **system-view**，进入系统视图。
2. 执行命令 **bgp** *as-number*，进入 BGP 视图。
3. 执行命令 **ipv4-family unicast**，进入 IPv4 单播地址族视图。
4. 执行命令 **peer** { *ipv4-address* | *group-name* } **next-hop-local**，配置发布路由时将自身地址作为下一跳。

   缺省情况下，向 IBGP 对等体发布路由时，不修改下一跳地址。

   说明

   如果配置了 BGP 负载分担，则不论是否配置了 **peer next-hop-local** 命令，本地路由器向 IBGP 对等体组发布路由时都先将下一跳地址改变为自身地址。

- 向 IBGP 对等体发布从 IGP 学到的路由时，不修改下一跳地址

  请在运行 BGP 协议且引入 IGP 路由的路由器上进行下列配置。

  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **bgp** *as-number*，进入 BGP 视图。
  3. 执行命令 **ipv4-family unicast**，进入 IPv4 单播地址族视图。
  4. 执行命令 **peer** { *ipv4-address* | *group-name* } **next-hop-invariable**，配置发布引入的 IGP 路由时不改变该 IGP 路由的下一跳地址。

     缺省情况下，对等体在发布所引入的 IGP 路由时会将下一跳地址改为本地与对端连接的接口地址。

- ASBR 向 EBGP 对等体发布路由时，不修改下一跳地址

  请在运行 BGP 协议的路由器上进行以下配置。

  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **bgp** *as-number*，进入 BGP 视图。
  3. 执行命令 **ipv4-family vpnv4** [**unicast**]，进入 BGP-VPNv4 子地址族图。
  4. 执行命令 **peer** { *group-name* | *ipv4-address* } **next-hop-invariable**，配置向 EBGP 对等体发布路由时不改变下一跳。

     缺省情况下，不同 AS 域的 PE 间建立的是 EBGP 对等体，发布路由时会改变下一跳地址。

     在采用 RR 的跨域 VPN OptionC 方式组网中，需要在 RR 上执行 **peer next-hop-invariable** 命令，配置向 EBGP 对等体发布路由时不改变下一跳，保证对端 PE 可以在流量传输时可以迭代到通往本端 PE 的 BGP LSP。

- 按策略进行下一跳迭代

  请在运行 BGP 协议的路由器上进行以下配置。

  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **bgp** *as-number*，进入 BGP 视图。
  3. 执行命令 **ipv4-family unicast**，进入 IPv4 单播地址族视图。
  4. 执行命令 **nexthop recursive-lookup route-policy** *route-policy-name*，配置路由按策略来迭代下一跳。

缺省情况下，没有配置下一跳迭代路由策略。

配置下一跳的迭代路由策略，可以有选择地进行路由迭代，按一定的条件来限制迭代的结果路由。如果路由不能通过策略，则该路由不能被迭代。

----结束

8.4.7 配置 AS_Path 属性

AS_Path 属性用于防止路由环路和控制路由选择。

操作步骤

- 允许本地 AS 编号重复出现

  通常情况下，BGP 通过 AS 号检测路由环路。但在 Hub and Spoke 组网方式下，如果在 Hub 节点的 PE 和 CE 之间运行 EBGP，当 Hub-PE 将路由信息通告给 Hub-CE 时带上本自治系统的 AS 号。当 Hub-CE 将携带有 Hub-PE 所在自治系统 AS 号的路由更新信息重新发布给 Hub-PE 时，Hub-PE 就会拒绝接收这条路由信息。

  为保证 Hub and Spoke 组网方式中路由能够正确传递，从 Hub-CE 发布私网路由到 Spoke-CE 途中经过的相关 BGP 对等体需要配置允许 AS_Path 中 AS 号重复 1 次的路由通过。

  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **bgp** *as-number*，进入 BGP 视图。
  3. 执行命令 **ipv4-family unicast**，进入 IPv4 单播地址族视图。
  4. 执行命令 **peer** { *ipv4-address* | *group-name* } **allow-as-loop** [*number*]，允许本地 AS 编号重复出现。

     通常情况下，BGP 会检查对等体发来的路由的 AS_Path 属性，如果其中已存在本地 AS 编号，则 BGP 会忽略此路由，以免形成路由环路。

     但在某些特殊应用中，使用此命令可以允许对等体发来的路由的 AS_Path 属性中已存在本地 AS 编号，同时还可以设置允许本地 AS 编号重复出现的次数。

- 配置不将 AS_Path 属性作为选路条件

  请在运行 BGP 协议的路由器上进行以下配置。

  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **bgp** *as-number*，进入 BGP 视图。
  3. 执行命令 **ipv4-family unicast**，进入 IPv4 单播地址族视图。
  4. 执行命令 **bestroute as-path-ignore**，不将 AS_Path 属性作为选路条件。

- 配置伪 AS 编号

  常规情况下，一个路由器只支持一个 BGP 进程，即只支持一个 AS 号。但是在某些特殊情况下，例如网络迁移更换 AS 号的时候，可以通过配置 **peer fake-as** 命令，为指定对等体设置一个伪 AS 号来为了保证网络切换的顺利。

  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **bgp** *as-number*，进入 BGP 视图。
  3. 执行命令 **peer** { *ipv4-address* | *group-name* } **fake-as** *fake-as-number*，配置伪 AS 编号。

使用此命令可以将本地真实的 AS 编号隐藏，位于其他 AS 内的 EBGP 对等体只能看到这个伪 AS 编号，即其他 AS 内的对等体在指定本端对等体所在的 AS 编号时，应该设置成这个伪 AS 编号。

说明

本命令只能应用于 EBGP 对等体。

- 替换 AS_Path 属性中的 AS 编号

在 PE 上使能了 BGP 的 AS 号替换功能后，当 PE 向指定对等体中的 CE 发布路由时，如果路由的 AS_Path 中有与 CE 相同的 AS 号，将被替换成 PE 的 AS 号后再发布。

注意

请谨慎配置命令 **peer substitute-as**，如果配置不当会引起路由环路。

1. 执行命令 **system-view**，进入系统视图。
2. 执行命令 **bgp** *as-number*，进入 BGP 视图。
3. 执行命令 **ipv4-family vpn-instance** *vpn-instance-name*，进入 BGP-VPN 实例 IPv4 地址族视图。
4. 执行命令 **peer** { *ipv4-address* | *group-name* } **substitute-as**，替换 AS_Path 属性中的 AS 编号。

- 配置 AS_Path 属性中仅携带公有 AS 编号

通常情况下，BGP 在向对等体发布路由时携带 AS 编号（可能是公有的 AS 编号，也可能是私有的 AS 编号）。公有 AS 编号可以直接在 Internet 上使用，由因特网地址分配组织 IANA（Internet Assigned Number Authority）管理和分配；而私有 AS 编号不能直接发布到 Internet 上，否则可能造成网络环路，这时可以通过配置使 AS_Path 属性中仅携带公有 AS 编号。

1. 执行命令 **system-view**，进入系统视图。
2. 执行命令 **bgp** *as-number*，进入 BGP 视图。
3. 执行命令 **ipv4-family unicast**，进入 IPv4 单播地址族视图。
4. 执行命令 **peer** { *ipv4-address* | *group-name* } **public-as-only**，配置 AS_Path 属性中仅携带公有 AS 编号。

   通常情况下，AS 编号的取值范围是 1 ～ 4294967295，其中公有 AS 编号范围是 1 ～ 64511，65536（x.y 形式为：1.0）～ 4294967295（x.y 形式为：65535.65535），私有 AS 编号范围是 64512 ～ 65534，65535 则作为保留 AS 编号在特殊应用中使用。

   该命令只能应用于 EBGP 对等体。

- 配置 AS_Path 属性中 AS 号的最大个数

请在运行 BGP 协议的路由器上进行以下配置。

1. 执行命令 **system-view**，进入系统视图。
2. 执行命令 **bgp** *as-number*，进入 BGP 视图。
3. 执行命令 **as-path-limit** *as-path-limit-num*，配置 AS_Path 属性中 AS 号的最大个数。

缺省情况下，AS_Path 属性中 AS 号的最大个数是 255。

配置 **as-path-limit** 命令后，接收路由时会检查 AS_Path 属性中的 AS 号是否超限。如果超限则丢弃路由，因此，AS_Path 属性中 AS 号的最大个数被限制得过小，会造成路由的丢失。

- 配置取消检查 EBGP 对等体发来的更新消息中 AS_Path 属性的第一个 AS 号

  请在运行 BGP 协议的路由器上进行以下配置。

  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **bgp** *as-number*，进入 BGP 视图。
  3. 执行命令 **undo check-first-as**，配置取消检查 EBGP 对等体发来的更新消息中 AS_Path 属性的第一个 AS 号。

     缺省情况下，BGP 检查 EBGP 对等体发来的更新消息中 AS_Path 列表的第一个 AS 号，确认第一个 AS 号必须是该 EBGP 对等体所在的 AS。否则，该更新信息被拒绝，EBGP 连接中断。

     注意

     配置 **undo check-first-as** 命令后产生环路的可能性增大，请慎重使用。

     配置更改后，如果希望对已接收的路由重新进行检查，可以执行 **refresh bgp** 命令。

----结束

8.4.8 检查配置结果

BGP 的路由属性配置成功后，可以查看路由的各属性相关信息。

前提条件

已经完成 BGP 路由属性的所有配置。

操作步骤

- 使用 **display bgp paths** [*as-regular-expression*]命令查看 AS 路径信息。
- 使用 **display bgp routing-table different-origin-as** 命令查看源 AS 不一致（目的地址相同）的路由。
- 使用 **display bgp routing-table regular-expression** *as-regular-expression* 命令查看匹配 AS 正则表达式的路由信息。
- 使用 **display bgp routing-table** [*network* [{ *mask* | *mask-length* } [**longer-prefixes**]]] 命令查看 BGP 路由表中的信息。

----结束

8.5 配置 BGP 发布路由

BGP 用来传递路由。在 BGP 发布路由时，可以灵活的对路由进行过滤或使用路由策略，只发布符合自己要求的路由，并且修改路由的属性，达到引导网络流量的目的。

8.5.1 建立配置任务

在配置 BGP 发布路由前了解此特性的应用环境、配置此特性的前置任务和数据准备，有助于快速、准确地完成配置任务。

应用环境

BGP 应用于 AS 之间传递路由信息，路由的发布直接影响流量的转发。

BGP 路由表路由数量通常比较大，传递大量的路由对设备来说是一个很大的负担，为了减小路由发送规模，需要对发布的路由进行控制，只发送自己想要发布的路由或者只发布对等体需要的路由。

另外，到达同一个目的地址，可能存在多条路由，这些路由分别需要穿越不同的 AS，为了把业务流量引导向某些特定的 AS，也需要对发布的路由进行筛选。

使用过滤器可以对 BGP 将要发布的路由进行过滤。BGP 支持针对某个对等体（组）将要发布的路由信息进行过滤。

前置任务

在配置 BGP 发布路由之前，需完成以下任务：

- **配置 BGP 的基本功能**

数据准备

在配置 BGP 发布路由之前，需要准备以下数据。

序号	数据
1	访问控制列表 ACL 的编号或名称
2	地址前缀列表的名称、序号和匹配模式
3	AS 路径过滤器编号或名称
4	团体属性过滤器的编号或名称、匹配模式
5	扩展团体属性过滤器的编号或名称、匹配模式
6	Route-Policy 的名字、匹配模式、节点号

8.5.2 配置 BGP 过滤器

充分利用 BGP 过滤器，可以灵活地对发布的路由进行过滤。

背景信息

目前提供以下六种过滤器供 BGP 使用：

- **访问控制列表 ACL（Access Control List）**
- **地址前缀列表（IP-Prefix List）**

- **AS 路径过滤器（AS-Path-Filter）**
- **团体属性过滤器（Community-Filter）**
- **扩展团体属性过滤器（Extcommunity-Filter）**
- **Route-Policy**

操作步骤

- 配置访问控制列表 ACL

  访问控制列表 ACL 是由 **permit** 和 **deny** 语句组成的一系列有顺序的规则，这些规则根据数据包的源地址、目的地址、端口号等来描述。ACL 通过这些规则对数据包进行分类，这些规则应用到路由器接口上，路由器根据这些规则判断哪些数据包可以接收，哪些数据包需要拒绝。

  ACL 的有关配置请参见《Huawei AR150&200 系列企业路由器 配置指南-IP 业务》中的描述。

  访问控制列表 ACL 可以做为 Route-policy 的一个匹配条件，也可以在 **filter-policy** { *acl-number* | **acl-name** *acl-name* } **export** [*protocol* [*process-id*]]或 **peer** { *group-name* | *ipv4-address* } **filter-policy** { *acl-number* | **acl-name** *acl-name* } **export** 命令中直接使用。

- 配置地址前缀列表

  地址前缀列表是一种针对路由目的地址信息做过滤的工具，它使用名字作为地址前缀列表的标识。地址前缀列表比较灵活，可以实现精确过滤，比如，可以对某一条路由或某一网段的路由进行过滤。但是当需要过滤的路由数量较大，且没有相同的前缀时，配置地址前缀列表会比较繁琐。

  地址前缀列表可以做为 Route-policy 的一个匹配条件，也可以在 **filter-policy ip-prefix** *ip-prefix-name* **export** [*protocol* [*process-id*]]或 **peer** { *group-name* | *ipv4-address* } **ip-prefix** *ip-prefix-name* **export** 命令中直接使用。

  请在运行 BGP 协议的路由器上进行下列配置。

  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **ip ip-prefix** *ip-prefix-name* [**index** *index-number*] { **permit** | **deny** } *ip-address mask-length* [**greater-equal** *greater-equal-value*] [**less-equal** *less-equal-value*]，配置 IPv4 地址前缀列表。

     掩码长度范围可以表示为 *mask-length* <= *greater-equal-value* <= *less-equal-value* <= 32。如果只指定了 **greater-equal**，前缀范围为[*greater-equal-value*，32]；如果只指定了 **less-equal**，前缀范围为[*mask-length*，*less-equal-value*]。

     IPv4 地址前缀列表由列表名标识，每个前缀列表可以包含多个表项。各表项可以独立指定一个网络前缀形式的匹配范围，并使用索引号标识。比如下面这个名称为 **abcd** 的 IPv4 地址前缀列表：

     ```
     #
     ip ip-prefix abcd index 10 permit 1.0.0.0 8
     ip ip-prefix abcd index 20 permit 2.0.0.0 8
     ```

     在匹配过程中，系统按索引号升序依次检查各个表项，只要有一个表项满足条件，就认为通过该过滤列表，不再去匹配其他表项。

     AR150/200 默认所有未匹配的路由将被拒绝通过过滤列表。如果所有表项都配置成 **deny** 模式，则任何路由都不能通过该过滤列表。因此，需要在多条 **deny**

模式的表项后定义一条 **permit 0.0.0.0 0 less-equal 32** 表项，允许其它所有 IPv4 路由信息通过。

说明

如果定义了多于一个的前缀列表表项，则至少应该有一个表项的匹配模式为 **permit** 模式。

- 配置 AS 路径过滤器

AS 路径过滤器是利用 BGP 路由携带的 AS-Path 列表对路由进行过滤，在不希望流量从某些 AS 穿过，可以利用 AS 路径过滤器对携带这些 AS 号的路由进行过滤。另外，利用 ACL 或者地址前缀列表过滤 BGP 路由，一方面有可能配置比较繁琐（需要定义多个 ACL 或者前缀列表），另一方面有新的路由加入不好维护，这时也可以使用 AS 路径过滤器。

说明

路由聚合后，如果路由的 AS 路径信息丢失，AS 路径过滤器就不能对这些聚合路由进行过滤，但是对于 AS 路径信息没有丢失的源路由仍旧可以过滤。

AS 路径过滤器可以做为 Route-policy 的一个匹配条件，也可以在 **peer as-path-filter** 命令中直接使用。

请在运行 BGP 协议的路由器上进行下列配置。

1. 执行命令 **system-view**，进入系统视图。
2. 执行命令 **ip as-path-filter** { *as-path-filter-number* | *as-path-filter-name* } { **permit** | **deny** } *regular-expression*，配置 AS 路径过滤器。

   AS 路径过滤器使用正则表达式来定义匹配规则。正则表达式由元字符和数值两部分组成：

   - 元字符定义了匹配的规则
   - 数值定义了匹配的对象

表 8-1 元字符描述

元字符	含义
\	转义字符
.	匹配除“\n”之外任何单个字符，包括空格
*	之前的字符在目标对象中出现 0 次或连续多次
+	之前的字符在目标对象中出现 1 次或连续多次
|	竖线左边和右边的字符为“或”的关系
^	之后的字符必须出现在目标对象的开始
$	之前的字符必须出现在目标对象的结束
[xyz]	匹配方括号内列出的任意字符
[^xyz]	匹配除了方括号内列出的字符外的任意字符（^号在字符前）
[a-z]	匹配指定范围内的任意字符

元字符	含义
[^a-z]	匹配不在指定范围内的任意字符
{n}	n 是一个非负整数，匹配连续出现的确定 n 次
{n,}	n 是一个非负整数，匹配连续出现的至少 n 次
{n,m}	m 和 n 均为非负整数，n<=m。匹配连续出现的次数为 n ～ m 次。使用时注意，逗号与 n 和 m 之间不能有空格

例如，^10 表示只匹配第一个值为 10 的 AS_Path 属性。其中符号^表示匹配一个字符串的开始。

在同一个过滤器编号下，可以定义多条过滤规则（permit 或 deny）。在匹配过程中，这些规则之间是“或”的关系，即只要路由信息通过其中一项规则，就认为通过由该过滤器编号标识的这组 AS 路径过滤器。

说明

正则表达式的使用，请参见《Huawei AR150&200 系列企业路由器 配置指南-基础配置》的“命令行介绍”。

- 配置团体属性过滤器

  BGP 的团体属性是用来标识一组具有共同性质的路由。利用团体属性可以人为的对路由进行分类，方便对路由进行管理。

  实际应用中，部分 AS 内路由可能不需要发布到其他的外部 AS，而 AS 外路由需要发布到其他的外部 AS，这些路由前缀不同（不便于使用地址前缀列表），可能来自不同 AS（不便于使用 AS 路径过滤器），这时可以在 AS 边缘给这些 AS 内路由设置相同的团体属性值，给 AS 外路由设置另外一个团体属性值，这样就可以利用团体属性值去控制和过滤路由。

  请在运行 BGP 协议的路由器上进行下列配置。

  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **ip community-filter**，配置团体属性过滤器。
     - 配置标准团体属性过滤器：执行命令 **ip community-filter** { **basic** *comm-filter-name* { **permit** | **deny** } [*community-number* | *aa:nn*] * &<1-9> | *basic-comm-filter-num* { **permit** | **deny** } [*community-number* | *aa:nn*] * &<1-16> } [**internet** | **no-export-subconfed** | **no-advertise** | **no-export**] *
     - 配置高级团体属性过滤器：执行命令 **ip community-filter** { **advanced** *comm-filter-name* | *adv-comm-filter-num* } { **permit** | **deny** } *regular-expression*

- 配置扩展团体属性过滤器

  BGP 的扩展团体属性过滤器类似于团体属性过滤器，主要用于对私网路由的过滤。

  请在运行 BGP 协议的路由器上进行下列配置。

  1. 执行命令 **system-view**，进入系统视图。
  2. 选择执行如下命令，配置扩展团体属性过滤器。

- 配置基本扩展团体属性过滤器：执行命令 **ip extcommunity-filter**{ *basic-extcomm-filter-num* | **basic** *basic-extcomm-filter-name* } { **deny** | **permit** } { **rt** { *as-number*:*nn* | *ipv4-address*:*nn* } } &<1-16>。
- 配置高级扩展团体属性过滤器：执行命令 **ip extcommunity-filter** { *adv-extcomm-filter-num* | **advanced** *adv-extcomm-filter-name* } { **deny** | **permit** } *regular-expression*。

对于相同的扩展团体属性过滤器号，用户可以定义多个表项。在匹配过程中，各表项之间是“或”的关系，即只要路由信息通过这组过滤器中的一条，就认为通过由该过滤器号标识的扩展团体属性过滤器。

- 配置 Route-Policy

Route-Policy 用来匹配给定的路由信息或者路由信息的某些属性，并在条件满足时改变这些路由信息的属性。匹配条件可以使用上面几种过滤器，所以 Route-Policy 的使用非常灵活，功能也非常强大。

请在运行 BGP 协议的路由器上进行下列配置。

1. 执行命令 **system-view**，进入系统视图。
2. 执行命令 **route-policy** *route-policy-name* { **permit** | **deny** } **node** *node*，创建路由策略的节点，并进入路由策略视图。

   一个 Route-Policy 由多个节点构成，例如，**route-policy route-policy-example permit node 10** 和 **route-policy route-policy-example deny node 20**，它们定义了两个节点 10 和 20，但是它们都属于一个 Route-Policy，即 **route-policy-example**。Route-Policy 节点间的过滤关系是“或”，有以下两种情况：

   - 如果一条路由通过了一个节点的过滤，就可通过该 Route-Policy，不再使用其他节点进行过滤。例如，**route-policy route-policy-example permit node 10** 和 **route-policy route-policy-example deny node 20**，如果路由通过了 **route-policy route-policy-example permit node 10** 的过滤，就不再匹配 **route-policy route-policy-example deny node 20**。
   - 如果一条路由没有通过任何一个节点的过滤，路由信息将无法通过该 Route-Policy。

   当引用该 Route-Policy 进行路由信息过滤时，*node* 的值小的节点先进行测试。例如，**route-policy route-policy-example permit node 10** 和 **route-policy route-policy-example deny node 20**，**route-policy route-policy-example permit node 10** 因为节点值较小，所以先进行测试。

   说明

   AR150/200 默认所有未匹配的路由将被拒绝通过 Route-Policy。如果 Route-Policy 中定义了一个以上的节点，则各节点中至少应该有一个节点的匹配模式是 **permit**。
3. （可选）执行下列命令，配置当前节点的路由策略中的 **if-match** 子句。

   执行 **if-match** 子句对路由进行过滤。如不指定 **if-match** 子句，则所有路由信息都会通过该节点的过滤。

   - 匹配访问控制列表 ACL：**if-match acl** { *acl-number* | *acl-name* }
   - 匹配地址前缀列表：**if-match ip-prefix** *ip-prefix-name*

     说明

     对于同一个 Route-Policy 节点，命令 **if-match acl** 和命令 **if-match ip-prefix** 不能同时配置，后配置的命令会覆盖先配置的命令。

- 匹配 BGP 路由信息的 AS 路径信息：**if-match as-path-filter** { *as-path-filter-number* | *as-path-filter-name* } &<1-16>
- 匹配 BGP 路由信息的团体属性：
  - **if-match community-filter** { *basic-comm-filter-num* [**whole-match**] | *adv-comm-filter-num* } * &<1-16>
  - **if-match community-filter** *comm-filter-name* [**whole-match**]
- 匹配 BGP 路由信息的扩展团体属性：**if-match extcommunity-filter** { { *basic-extcomm-filter-num* | *adv-extcomm-filter-num* } &<1-16> | *basic-extcomm-filter-name* | *advanced-extcomm-filter-name* }

步骤 3 各命令之间没有顺序关系。在一个节点中，可以没有 **if-match** 子句，也可以有多个 **if-match** 子句。

说明

对于同一个 Route-policy 节点，在匹配的过程中，各个 **if-match** 子句间是“与”的关系，即路由信息必须同时满足所有匹配条件，才可以执行 **apply** 子句的动作。例如，**route-policy route-policy-example permit node 10** 定义了两个 **if-match** 子句，分别是 **if-match acl 2003** 和 **if-match as-path-filter 100**，则只有路由同时匹配这两个条件，才算通过节点 10 的过滤。

4. （可选）执行下列命令，配置当前节点中路由策略的 **apply** 子句。

执行 **apply** 子句可以为通过 **if-match** 子句过滤的路由设置路由属性。如果不执行该步骤，则不会修改通过 **if-match** 子句过滤的路由的属性。

- 在 BGP 的 AS_Path 属性中替换或加入指定的 AS 号：**apply as-path** *as-number*
- 删除指定的 BGP 团体属性：**apply comm-filter** *comm-filter-number* **delete**

  窍门

  apply comm-filter delete 命令用来根据团体属性过滤器中指定的值删除团体属性，所引用的 **ip community-filter** 命令每条只能包含一个团体属性，如果要删除多个团体属性，则可通过配置多条命令来解决。如果在同一个列表号下配置了多个团体属性，则这几个属性都无法删除。举例请参见《Huawei AR150&200 系列企业路由器 命令参考》。
- 删除 BGP 路由信息的团体属性：**apply community none**
- 设置 BGP 路由信息的团体属性：**apply community** { { *community-number* | *aa:nn* } &<1-32> | **internet** | **no-advertise** | **no-export** | **no-export-subconfed** } * [**additive**]
- 设置 BGP 扩展团体属性（Route-Target）：**apply extcommunity** { **rt** { *as-number:nn* | *4as-number:nn* | *ipv4-address:nn* } } &<1-16> [**additive**]
- 设置 BGP 路由信息的本地优先级：**apply local-preference** *preference*
- 设置 BGP 路由信息的 Origin 属性：**apply origin** { **igp** | **egp** *as-number* | **incomplete** }
- 设置 BGP 路由信息的首选值：**apply preferred-value** *preferred-value*
- 设置 EBGP 路由的衰减参数：**apply dampening** *half-life-reach reuse suppress ceiling*

步骤 4 各命令之间没有顺序关系，在一个节点中，可以没有 **apply** 子句，也可以有多个 **apply** 子句。

----结束

8.5.3 配置控制 BGP 路由信息的发布

配置路由的发布策略后，只有符合条件的路由信息才会被发布给 BGP 对等体。

操作步骤

- 配置 BGP 向全局发布路由

  在发布路由时，可以对路由信息进行过滤。请在运行 BGP 协议的路由器上进行下列配置。

  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **bgp** *as-number*，进入 BGP 视图。
  3. 执行命令 **ipv4-family unicast**，进入 IPv4 单播地址族视图。
  4. 选择执行下列命令，配置 BGP 对全局发布的路由信息进行过滤。
     - 基于访问控制列表 ACL：**filter-policy** { *acl-number* | **acl-name** *acl-name* } **export** [*protocol* [*process-id*]]
     - 基于地址前缀列表：**filter-policy ip-prefix** *ip-prefix-name* **export** [*protocol* [*process-id*]]

     指定 *protocol* 参数可以只对特定路由协议的信息进行过滤；如果没有指定此参数，则对所有要发布的 BGP 路由信息进行过滤，包括通过 **import-route (BGP)** 命令引入的路由和使用 **network (BGP)**命令引入的本地路由。

     说明

     在 **filter-policy** 命令中，如果使用了 ACL 且 ACL 过滤规则中没有指定某个 VPN 实例，则 BGP 是对所有地址族下的路由信息进行过滤，包括来自公网和私网的路由信息。如果 ACL 过滤规则中指定了 VPN 实例，则仅对来自该 VPN 的数据流量进行过滤，而不是对路由信息进行过滤。

- 配置 BGP 向特定对等体（组）发布路由

  在发布路由时，可以对路由信息进行过滤。请在运行 BGP 协议的路由器上进行下列配置。

  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **bgp** *as-number*，进入 BGP 视图。
  3. 执行命令 **ipv4-family unicast**，进入 IPv4 单播地址族视图。
  4. 选择执行如下命令，配置对特定对等体（组）发布的路由信息进行过滤。
     - 基于访问控制列表 ACL：**peer** { *ipv4-address* | *group-name* } **filter-policy** { *acl-number* | **acl-name** *acl-name* } **export**
     - 基于前缀列表：**peer** { *ipv4-address* | *group-name* } **ip-prefix** *ip-prefix-name* **export**
     - 基于 AS 路径过滤器：**peer** { *ipv4-address* | *group-name* } **as-path-filter** { *as-path-filter-number* | *as-path-filter-name* } **export**
     - 基于 Route-Policy：**peer** { *ipv4-address* | *group-name* } **route-policy** *route-policy-name* **export**

     对等体组的成员可以与所在的组使用不同的出方向路由策略，即对外发布路由时，各对等体可以选择自己的策略。

----结束

8.5.4 配置 BGP 软复位

当 BGP 的入口策略改变后，系统可以在不中断 BGP 连接的情况下，对 BGP 路由表进行动态刷新。

背景信息

BGP 的入口策略改变后，为了使新的策略生效，必须复位 BGP 连接，但这样会造成短暂的 BGP 连接中断。BGP 支持路由刷新（Route-refresh）能力，当策略改变后，系统可以在不中断 BGP 连接的情况下，对 BGP 路由表进行动态刷新。

- 对于支持 Route-refresh 能力的 BGP 对等体，可以配置 **refresh bgp** 命令手工对 BGP 连接进行软复位，完成对路由表的刷新。
- 对于不支持 Route-refresh 能力的 BGP 对等体，可以配置 **peer keep-all-routes** 命令，保留该对等体的所有原始路由，不需要复位 BGP 连接即可完成路由表的刷新。

请在运行 BGP 协议的路由器上进行下列配置。

操作步骤

- 对于支持路由刷新能力的 BGP 对等体
  1. （可选）使能路由刷新能力
     a. 执行命令 **system-view**，进入系统视图。
     b. 执行命令 **bgp** *as-number*，进入 BGP 视图。
     c. 执行命令 **peer** { *ipv4-address* | *group-name* } **capability-advertise route-refresh**，使能 Route-refresh 能力。

     缺省情况下，使能 Route-refresh 能力。

     在所有 BGP 路由器使能 Route-refresh 能力的情况下，如果 BGP 的入方向路由策略发生了变化，本地路由器会向对等体（组）发布 Route-refresh 消息，收到此消息的对等体（组）会将其路由信息重新发给本地 BGP 路由器。这样，在不中断 BGP 连接的情况下，就可以对 BGP 路由表进行动态更新，并应用新的策略。
  2. 配置 BGP 软复位
     a. 在用户视图下，执行命令 **refresh bgp** [**vpn-instance** *vpn-instance-name* **ipv4-family**] { **all** | *ipv4-address* | **group** *group-name* | **external** | **internal** } { **export** | **import** }，可以立即软复位 BGP 连接。

     参数 **external** 和 **internal** 分别表示软复位 EBGP 连接和 IBGP 连接。

     参数 **export** 和 **import** 分别表示触发出方向和入方向的 BGP 软复位。
- 对于不支持路由刷新能力的 BGP 对等体
  - 保留对等体（组）的所有路由更新信息
    1. 执行命令 **system-view**，进入系统视图。
    2. 执行命令 **bgp** *as-number*，进入 BGP 视图。
    3. 执行命令 **ipv4-family unicast**，进入 IPv4 单播地址族视图。
    4. 执行命令 **peer** { *ipv4-address* | *group-name* } **keep-all-routes**，保留对等体（组）的所有路由更新信息。

    缺省情况下，只保存来自对等体（组）的通过入口策略的路由更新信息。

配置此命令后，不论是否使用了入方向过滤策略，都将保存指定对等体（组）发来的所有路由更新信息。当本地路由策略改变时，这些信息可以用来重新生成 BGP 路由。

说明

该命令需要在本地和对等体上均配置。第一次配置 **peer keep-all-routes** 命令后会导致与对等体会话重新连接。

对于支持 Route-refresh 能力的路由器，不需要配置 **peer keep-all-routes** 命令。如果配置该命令，不会导致与对等体的会话重新连接，但路由器通过执行 **refresh bgp** 命令刷新路由表功能将不会生效。

----结束

8.5.5 检查配置结果

BGP 发布路由配置成功后，可以查看配置的过滤器信息、与指定过滤器匹配的路由信息和 BGP 向对等体发布的路由信息。

前提条件

已经完成配置 BGP 发布路由的所有配置。

操作步骤

- 使用 **display ip as-path-filter** [*as-path-filter-number* | *as-path-filter-name*]命令查看已配置的 AS 路径过滤器信息。
- 使用 **display ip community-filter** [*basic-comm-filter-num* | *adv-comm-filter-num* | *comm-filter-name*]命令查看已配置的团体属性过滤器信息。
- 使用 **display ip extcommunity-filter** [*extcomm-filter-number* | *extcomm-filter-name*]命令查看已配置的扩展团体属性过滤器信息。
- 使用 **display bgp routing-table as-path-filter** { *as-path-filter-number* | *as-path-filter-name* }命令查看与指定 AS 路径过滤器匹配的路由信息。
- 使用 **display bgp routing-table community-filter** { { *community-filter-name* | *basic-community-filter-number* } [**whole-match**] | *advanced-community-filter-number* }命令查看匹配指定 BGP 团体属性过滤器的路由。
- 使用 **display bgp routing-table peer** *ipv4-address* **advertised-routes** [**statistics**]命令查看 BGP 向对等体发布的路由信息。

----结束

8.6 配置 BGP 接收路由

BGP 用来传递路由。在 BGP 接收路由时，可以灵活的对路由进行过滤或使用路由策略，只接收符合自己需要的路由，并且修改路由的属性，达到引导网络流量的目的。

8.6.1 建立配置任务

在配置 BGP 接收路由前了解此特性的应用环境、配置此特性的前置任务和数据准备，有助于快速、准确地完成配置任务。

应用环境

BGP 应用于 AS 之间传递路由信息，路由的接收直接影响流量的转发。

BGP 路由器可能收到不同对等体发来的到达同一目的网络的路由，为了控制网络流量的转发路径，就需要对 BGP 接收的路由进行筛选。

另外，由于可能会受到服务攻击，BGP 从对等体接收到任意数量的路由，大量消耗路由器的资源。无论过量 BGP 路由是因为恶意攻击还是因为某错误配置导致，管理员都必须根据网络规划和路由器容量，对运行时所使用的资源进行限制。

使用过滤器可以对 BGP 将要接收的路由进行过滤。BGP 可以对接收的全局路由信息或者只对某个对等体（组）发来的路由信息进行过滤或使用路由策略。

前置任务

在配置 BGP 接收路由之前，需完成以下任务：

- **配置 BGP 的基本功能**

数据准备

在配置 BGP 接收路由之前，需要准备以下数据。

序号	数据
1	访问控制列表 ACL 的编号或名称
2	地址前缀列表的名称、序号和匹配模式
3	AS 路径过滤器编号或名称
4	团体属性过滤器的编号或名称、匹配模式
5	扩展团体属性过滤器的编号或名称、匹配模式
6	Route-Policy 的名字、匹配模式、节点号

8.6.2 配置 BGP 过滤器

充分利用 BGP 过滤器，可以灵活地对收到的路由进行过滤。

背景信息

为了灵活的控制路由的接收，需要使用过滤器对路由进行过滤，目前提供以下六种过滤器供 BGP 使用：

- **访问控制列表 ACL（Access Control List）**
- **地址前缀列表（IP-Prefix List）**
- **AS 路径过滤器（AS-Path-Filter）**
- **团体属性过滤器（Community-Filter）**
- **扩展团体属性过滤器（Extcommunity-Filter）**

- **Route-Policy**

操作步骤

- 配置访问控制列表 ACL

  访问控制列表 ACL 是由 **permit** 和 **deny** 语句组成的一系列有顺序的规则，这些规则根据数据包的源地址、目的地址、端口号等来描述。ACL 通过这些规则对数据包进行分类，这些规则应用到路由器接口上，路由器根据这些规则判断哪些数据包可以接收，哪些数据包需要拒绝。

  ACL 的有关配置请参见《Huawei AR150&200 系列企业路由器 配置指南-IP 业务》中的描述。

  访问控制列表 ACL 可以做为 Route-policy 的一个匹配条件，也可以在 **filter-policy** { *acl-number* | **acl-name** *acl-name* } **import** 或 **peer** { *group-name* | *ipv4-address* } **filter-policy** { *acl-number* | **acl-name** *acl-name* } **import** 命令中直接使用。

- 配置地址前缀列表

  地址前缀列表是一种针对路由目的地址信息做过滤的工具，它使用名字作为地址前缀列表的标识。地址前缀列表比较灵活，可以实现精确过滤，比如，可以对某一条路由或某一网段的路由进行过滤。但是当需要过滤的路由数量较大，且没有相同的前缀时，配置地址前缀列表会比较繁琐。

  地址前缀列表可以做为 Route-policy 的一个匹配条件，也可以在 **filter-policy ip-prefix** *ip-prefix-name* **import** 或 **peer** { *group-name* | *ipv4-address* } **ip-prefix** *ip-prefix-name* **import** 命令中直接使用。

  请在运行 BGP 协议的路由器上进行下列配置。

  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **ip ip-prefix** *ip-prefix-name* [**index** *index-number*] { **permit** | **deny** } *ip-address mask-length* [**greater-equal** *greater-equal-value*] [**less-equal** *less-equal-value*]，配置 IPv4 地址前缀列表。

     掩码长度范围可以表示为 *mask-length* <= *greater-equal-value* <= *less-equal-value* <= 32。如果只指定了 **greater-equal**，前缀范围为[*greater-equal-value*，32]；如果只指定了 **less-equal**，前缀范围为[*mask-length*，*less-equal-value*]。

     IPv4 地址前缀列表由列表名标识，每个前缀列表可以包含多个表项。各表项可以独立指定一个网络前缀形式的匹配范围，并使用索引号标识。比如下面这个名称为 **abcd** 的 IPv4 地址前缀列表：

     ```
     #
     ip ip-prefix abcd index 10 permit 1.0.0.0 8
     ip ip-prefix abcd index 20 permit 2.0.0.0 8
     ```

     在匹配过程中，系统按索引号升序依次检查各个表项，只要有一个表项满足条件，就认为通过该过滤列表，不再去匹配其他表项。

     AR150/200 默认所有未匹配的路由将被拒绝通过过滤列表。如果所有表项都配置成 **deny** 模式，则任何路由都不能通过该过滤列表。因此，需要在多条 **deny** 模式的表项后定义一条 **permit 0.0.0.0 0 less-equal 32** 表项，允许其它所有 IPv4 路由信息通过。

说明

如果定义了多于一个的前缀列表表项，则至少应该有一个表项的匹配模式为 **permit** 模式。

- 配置 AS 路径过滤器

AS 路径过滤器是利用 BGP 路由携带的 AS-Path 列表对路由进行过滤，在不希望流量从某些 AS 穿过，可以利用 AS 路径过滤器对携带这些 AS 号的路由进行过滤。另外，利用 ACL 或者地址前缀列表过滤 BGP 路由，一方面有可能配置比较繁琐（需要定义多个 ACL 或者前缀列表），另一方面有新的路由加入不好维护，这时也可以使用 AS 路径过滤器。

说明

路由聚合后，如果路由的 AS 路径信息丢失，AS 路径过滤器就不能对这些聚合路由进行过滤，但是对于 AS 路径信息没有丢失的源路由仍旧可以过滤。

AS 路径过滤器可以做为 Route-policy 的一个匹配条件，也可以在 **peer as-path-filter** 命令中直接使用。

请在运行 BGP 协议的路由器上进行下列配置。

1. 执行命令 **system-view**，进入系统视图。
2. 执行命令 **ip as-path-filter** { *as-path-filter-number* | *as-path-filter-name* } { **permit** | **deny** } *regular-expression*，配置 AS 路径过滤器。

   AS 路径过滤器使用正则表达式来定义匹配规则。正则表达式由元字符和数值两部分组成：

   - 元字符定义了匹配的规则
   - 数值定义了匹配的对象

表 8-2 元字符描述

元字符	含义
\	转义字符
.	匹配除“\n”之外任何单个字符，包括空格
*	之前的字符在目标对象中出现 0 次或连续多次
+	之前的字符在目标对象中出现 1 次或连续多次
|	竖线左边和右边的字符为“或”的关系
^	之后的字符必须出现在目标对象的开始
$	之前的字符必须出现在目标对象的结束
[xyz]	匹配方括号内列出的任意字符
[^xyz]	匹配除了方括号内列出的字符外的任意字符（^号在字符前）
[a-z]	匹配指定范围内的任意字符
[^a-z]	匹配不在指定范围内的任意字符

元字符	含义
{n}	n 是一个非负整数，匹配连续出现的确定 n 次
{n,}	n 是一个非负整数，匹配连续出现的至少 n 次
{n,m}	m 和 n 均为非负整数，n<=m。匹配连续出现的次数为 n ～ m 次。使用时注意，逗号与 n 和 m 之间不能有空格

例如，^10 表示只匹配第一个值为 10 的 AS_Path 属性。其中符号^表示匹配一个字符串的开始。

在同一个过滤器编号下，可以定义多条过滤规则（permit 或 deny）。在匹配过程中，这些规则之间是“或”的关系，即只要路由信息通过其中一项规则，就认为通过由该过滤器编号标识的这组 AS 路径过滤器。

说明

正则表达式的使用，请参见《Huawei AR150&200 系列企业路由器 配置指南-基础配置》的“命令行介绍”。

- 配置团体属性过滤器

  BGP 的团体属性是用来标识一组具有共同性质的路由。利用团体属性可以人为的对路由进行分类，方便对路由进行管理。

  实际应用中，部分 AS 内路由可能不需要发布到其他的外部 AS，而 AS 外路由需要发布到其他的外部 AS，这些路由前缀不同（不便于使用地址前缀列表），可能来自不同 AS（不便于使用 AS 路径过滤器），这时可以在 AS 边缘给这些 AS 内路由设置相同的团体属性值，给 AS 外路由设置另外一个团体属性值，这样就可以利用团体属性值去控制和过滤路由。

  请在运行 BGP 协议的路由器上进行下列配置。

  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **ip community-filter**，配置团体属性过滤器。
     - 配置标准团体属性过滤器：执行命令 **ip community-filter** { **basic** *comm-filter-name* { **permit** | **deny** } [*community-number* | *aa:nn*] * &<1-9> | *basic-comm-filter-num* { **permit** | **deny** } [*community-number* | *aa:nn*] * &<1-16> } [**internet** | **no-export-subconfed** | **no-advertise** | **no-export**] *
     - 配置高级团体属性过滤器：执行命令 **ip community-filter** { **advanced** *comm-filter-name* | *adv-comm-filter-num* } { **permit** | **deny** } *regular-expression*

- 配置扩展团体属性过滤器

  BGP 的扩展团体属性过滤器类似于团体属性过滤器，主要用于对私网路由的过滤。

  请在运行 BGP 协议的路由器上进行下列配置。

  1. 执行命令 **system-view**，进入系统视图。
  2. 选择执行如下命令，配置扩展团体属性过滤器。
     - 配置基本扩展团体属性过滤器：执行命令 **ip extcommunity-filter**{ *basic-extcomm-filter-num* | **basic** *basic-extcomm-filter-name* } { **deny** | **permit** } { **rt** { *as-number:nn* | *ipv4-address:nn* } } &<1-16>。

- 配置高级扩展团体属性过滤器：执行命令 **ip extcommunity-filter** { *adv-extcomm-filter-num* | **advanced** *adv-extcomm-filter-name* } { **deny** | **permit** } *regular-expression*。

对于相同的扩展团体属性过滤器号，用户可以定义多个表项。在匹配过程中，各表项之间是“或”的关系，即只要路由信息通过这组过滤器中的一条，就认为通过由该过滤器号标识的扩展团体属性过滤器。

- 配置 Route-Policy

Route-Policy 用来匹配给定的路由信息或者路由信息的某些属性，并在条件满足时改变这些路由信息的属性。匹配条件可以使用上面几种过滤器，所以 Route-Policy 的使用非常灵活，功能也非常强大。

请在运行 BGP 协议的路由器上进行下列配置。

1. 执行命令 **system-view**，进入系统视图。
2. 执行命令 **route-policy** *route-policy-name* { **permit** | **deny** } **node** *node*，创建路由策略的节点，并进入路由策略视图。

   一个 Route-Policy 由多个节点构成，例如，**route-policy route-policy-example permit node 10** 和 **route-policy route-policy-example deny node 20**，它们定义了两个节点 10 和 20，但是它们都属于一个 Route-Policy，即 **route-policy-example**。Route-Policy 节点间的过滤关系是“或”，有以下两种情况：

   - 如果一条路由通过了一个节点的过滤，就可通过该 Route-Policy，不再使用其他节点进行过滤。例如，**route-policy route-policy-example permit node 10** 和 **route-policy route-policy-example deny node 20**，如果路由通过了 **route-policy route-policy-example permit node 10** 的过滤，就不再匹配 **route-policy route-policy-example deny node 20**。
   - 如果一条路由没有通过任何一个节点的过滤，路由信息将无法通过该 Route-Policy。

   当引用该 Route-Policy 进行路由信息过滤时，*node* 的值小的节点先进行测试。例如，**route-policy route-policy-example permit node 10** 和 **route-policy route-policy-example deny node 20**，**route-policy route-policy-example permit node 10** 因为节点值较小，所以先进行测试。

   说明

   AR150/200 默认所有未匹配的路由将被拒绝通过 Route-Policy。如果 Route-Policy 中定义了一个以上的节点，则各节点中至少应该有一个节点的匹配模式是 **permit**。
3. （可选）执行下列命令，配置当前节点的路由策略中的 **if-match** 子句。

   执行 **if-match** 子句对路由进行过滤。如不指定 **if-match** 子句，则所有路由信息都会通过该节点的过滤。

   - 匹配访问控制列表 ACL：**if-match acl** { *acl-number* | *acl-name* }
   - 匹配地址前缀列表：**if-match ip-prefix** *ip-prefix-name*

     说明

     对于同一个 Route-Policy 节点，命令 **if-match acl** 和命令 **if-match ip-prefix** 不能同时配置，后配置的命令会覆盖先配置的命令。
   - 匹配 BGP 路由信息的 AS 路径信息：**if-match as-path-filter** { *as-path-filter-number* | *as-path-filter-name* } &<1-16>
   - 匹配 BGP 路由信息的团体属性：

- **if-match community-filter** { *basic-comm-filter-num* [**whole-match**] | *adv-comm-filter-num* } * &<1-16>
- **if-match community-filter** *comm-filter-name* [**whole-match**]

- 匹配 BGP 路由信息的扩展团体属性：**if-match extcommunity-filter** { { *basic-extcomm-filter-num* | *adv-extcomm-filter-num* } &<1-16> | *basic-extcomm-filter-name* | *advanced-extcomm-filter-name* }

步骤 3 各命令之间没有顺序关系。在一个节点中，可以没有 **if-match** 子句，也可以有多个 **if-match** 子句。

说明

对于同一个 Route-policy 节点，在匹配的过程中，各个 **if-match** 子句间是“与”的关系，即路由信息必须同时满足所有匹配条件，才可以执行 **apply** 子句的动作。例如，**route-policy route-policy-example permit node 10** 定义了两个 **if-match** 子句，分别是 **if-match acl 2003** 和 **if-match as-path-filter 100**，则只有路由同时匹配这两个条件，才算通过节点 10 的过滤。

4. （可选）执行下列命令，配置当前节点中路由策略的 **apply** 子句。

执行 **apply** 子句可以为通过 **if-match** 子句过滤的路由设置路由属性。如果不执行该步骤，则不会修改通过 **if-match** 子句过滤的路由的属性。

- 在 BGP 的 AS_Path 属性中替换或加入指定的 AS 号：**apply as-path** *as-number*
- 删除指定的 BGP 团体属性：**apply comm-filter** *comm-filter-number* **delete**

窍门

apply comm-filter delete 命令用来根据团体属性过滤器中指定的值删除团体属性，所引用的 **ip community-filter** 命令每条只能包含一个团体属性，如果要删除多个团体属性，则可通过配置多条命令来解决。如果在同一个列表号下配置了多个团体属性，则这几个属性都无法删除。举例请参见《Huawei AR150&200 系列企业路由器 命令参考》。

- 删除 BGP 路由信息的团体属性：**apply community none**
- 设置 BGP 路由信息的团体属性：**apply community** { { *community-number* | *aa:nn* } &<1-32> | **internet** | **no-advertise** | **no-export** | **no-export-subconfed** } * [**additive**]
- 设置 BGP 扩展团体属性（Route-Target）：**apply extcommunity** { **rt** { *as-number:nn* | *4as-number:nn* | *ipv4-address:nn* } } &<1-16> [**additive**]
- 设置 BGP 路由信息的本地优先级：**apply local-preference** *preference*
- 设置 BGP 路由信息的 Origin 属性：**apply origin** { **igp** | **egp** *as-number* | **incomplete** }
- 设置 BGP 路由信息的首选值：**apply preferred-value** *preferred-value*
- 设置 EBGP 路由的衰减参数：**apply dampening** *half-life-reach reuse suppress ceiling*

步骤 4 各命令之间没有顺序关系，在一个节点中，可以没有 **apply** 子句，也可以有多个 **apply** 子句。

----结束

8.6.3 配置控制 BGP 路由信息的接收

配置路由的接收策略后，只有符合入口策略的路由信息才会被接收。

操作步骤

- 配置 BGP 从全局接收路由

  在接收路由时，可以对路由信息进行过滤。请在运行 BGP 协议的路由器上进行下列配置。

  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **bgp** *as-number*，进入 BGP 视图。
  3. 执行命令 **ipv4-family unicast**，进入 IPv4 单播地址族视图。
  4. 选择执行下列命令，配置 BGP 对从全局接收的路由信息进行过滤。
     - 基于访问控制列表 ACL：**filter-policy** { *acl-number* | **acl-name** *acl-name* } **import**
     - 基于地址前缀列表：**filter-policy ip-prefix** *ip-prefix-name* **import**

     说明

     在 **filter-policy** 命令中，如果使用了 ACL 且 ACL 过滤规则中没有指定某个 VPN 实例，则 BGP 是对所有地址族下的路由信息进行过滤，包括来自公网和私网的路由信息。如果 ACL 过滤规则中指定了 VPN 实例，则仅对来自该 VPN 的数据流量进行过滤，而不是对路由信息进行过滤。

- 配置 BGP 从特定对等体（组）接收路由

  在接收路由时，可以对路由信息进行过滤。请在运行 BGP 协议的路由器上进行下列配置。

  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **bgp** *as-number*，进入 BGP 视图。
  3. 执行命令 **ipv4-family unicast**，进入 IPv4 单播地址族视图。
  4. 选择执行如下命令，配置对特定对等体（组）接收的路由信息进行过滤。
     - 基于访问控制列表 ACL：**peer** { *ipv4-address* | *group-name* } **filter-policy** { *acl-number* | **acl-name** *acl-name* } **import**
     - 基于前缀列表：**peer** { *ipv4-address* | *group-name* } **ip-prefix** *ip-prefix-name* **import**
     - 基于 AS 路径过滤器：**peer** { *ipv4-address* | *group-name* } **as-path-filter** { *as-path-filter-number* | *as-path-filter-name* } **import**
     - 基于 Route-Policy：**peer** { *ipv4-address* | *group-name* } **route-policy** *route-policy-name* **import**

     对等体组的成员可以与所在的组使用不同的入方向路由策略，即接收路由时，各对等体可以选择自己的策略。

- 限制从对等体（组）接收的路由数量

  当设备遭到恶意攻击或者网络中出现错误配置时，会导致 BGP 从邻居接收到大量的路由，从而消耗大量路由器的资源。因此管理员必须根据网络规划和路由器容量，对运行时所使用的资源进行限制。BGP 提供了基于对等体的路由控制，限定邻居发来的路由数量，这样可以避免上述问题。

  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **bgp** *as-number*，进入 BGP 视图。
  3. 执行命令 **ipv4-family unicast**，进入 IPv4 单播地址族视图。

4. 执行命令 **peer** { *group-name* | *ipv4-address* } **route-limit** *limit* [*percentage*] [**alert-only** | **idle-forever** | **idle-timeout** *times*]，设置允许从对等体（组）收到的路由数量。

   该命令提供基于单个对等体（组）的控制，并可根据实际需求选择配置具体参数，来控制对等体（组）路由数超限后的行为：

   - 设置 **alert-only** 参数：邻居不中断连接也不再接收超限后的路由，产生告警并记入日志。
   - 设置 **idle-forever** 参数：邻居中断连接，不自动重新尝试建连，产生告警并记入日志。此时通过 **display bgp peer** [**verbose**]查看，可见 peer 的状态为 Idle。如果需要恢复 BGP 连接，可执行命令 **reset bgp**。
   - 设置 **idle-timeout** 参数：邻居中断连接，定时器到时后重新尝试建连，产生告警并记入日志。此时通过 **display bgp peer** [**verbose**]查看，可见 peer 的状态为 Idle。如果需要在定时器到时前恢复 BGP 连接，可执行命令 **reset bgp**。
   - 如果不设置以上三个参数：邻居断连，30 秒后重新尝试建连，产生告警并记入日志。

   说明

   如果路由器收到的路由数量超出了设定的最大限额，且第一次配置 **peer route-limit** 命令，无论是否还配置了 **alert-only** 参数，本地路由器都将与对等体重新建立邻居关系。

----结束

8.6.4 配置 BGP 软复位

当 BGP 的入口策略改变后，系统可以在不中断 BGP 连接的情况下，对 BGP 路由表进行动态刷新。

背景信息

BGP 的入口策略改变后，为了使新的策略生效，必须复位 BGP 连接，但这样会造成短暂的 BGP 连接中断。BGP 支持路由刷新（Route-refresh）能力，当策略改变后，系统可以在不中断 BGP 连接的情况下，对 BGP 路由表进行动态刷新。

- 对于支持 Route-refresh 能力的 BGP 对等体，可以配置 **refresh bgp** 命令手工对 BGP 连接进行软复位，完成对路由表的刷新。
- 对于不支持 Route-refresh 能力的 BGP 对等体，可以配置 **peer keep-all-routes** 命令，保留该对等体的所有原始路由，不需要复位 BGP 连接即可完成路由表的刷新。

请在运行 BGP 协议的路由器上进行下列配置。

操作步骤

- 对于支持路由刷新能力的 BGP 对等体
  1. （可选）使能路由刷新能力
     a. 执行命令 **system-view**，进入系统视图。
     b. 执行命令 **bgp** *as-number*，进入 BGP 视图。
     c. 执行命令 **peer** { *ipv4-address* | *group-name* } **capability-advertise route-refresh**，使能 Route-refresh 能力。

     缺省情况下，使能 Route-refresh 能力。

在所有 BGP 路由器使能 Route-refresh 能力的情况下，如果 BGP 的入方向路由策略发生了变化，本地路由器会向对等体（组）发布 Route-refresh 消息，收到此消息的对等体（组）会将其路由信息重新发给本地 BGP 路由器。这样，在不中断 BGP 连接的情况下，就可以对 BGP 路由表进行动态更新，并应用新的策略。

2. 配置 BGP 软复位

   a. 在用户视图下，执行命令 **refresh bgp** [**vpn-instance** *vpn-instance-name* **ipv4-family**] { **all** | *ipv4-address* | **group** *group-name* | **external** | **internal** } { **export** | **import** }，可以立即软复位 BGP 连接。

   参数 **external** 和 **internal** 分别表示软复位 EBGP 连接和 IBGP 连接。

   参数 **export** 和 **import** 分别表示触发出方向和入方向的 BGP 软复位。

- 对于不支持路由刷新能力的 BGP 对等体
  - 保留对等体（组）的所有路由更新信息

    1. 执行命令 **system-view**，进入系统视图。
    2. 执行命令 **bgp** *as-number*，进入 BGP 视图。
    3. 执行命令 **ipv4-family unicast**，进入 IPv4 单播地址族视图。
    4. 执行命令 **peer** { *ipv4-address* | *group-name* } **keep-all-routes**，保留对等体（组）的所有路由更新信息。

    缺省情况下，只保存来自对等体（组）的通过入口策略的路由更新信息。

    配置此命令后，不论是否使用了入方向过滤策略，都将保存指定对等体（组）发来的所有路由更新信息。当本地路由策略改变时，这些信息可以用来重新生成 BGP 路由。

    说明

    该命令需要在本地和对等体上均配置。第一次配置 **peer keep-all-routes** 命令后会导致与对等体会话重新连接。

    对于支持 Route-refresh 能力的路由器，不需要配置 **peer keep-all-routes** 命令。如果配置该命令，不会导致与对等体的会话重新连接，但路由器通过执行 **refresh bgp** 命令刷新路由表功能将不会生效。

----结束

8.6.5 检查配置结果

BGP 接收路由配置成功后，可以查看与指定过滤器匹配的引入路由信息。

前提条件

已经完成配置 BGP 接收路由的所有配置。

操作步骤

- 使用 **display ip as-path-filter** [*as-path-filter-number* | *as-path-filter-name*]命令查看已配置的 AS 路径过滤器信息。
- 使用 **display ip community-filter** [*basic-comm-filter-num* | *adv-comm-filter-num* | *comm-filter-name*]命令查看已配置的团体属性过滤器信息。
- 使用 **display ip extcommunity-filter** [*extcomm-filter-number*]命令查看已配置的扩展团体属性过滤器信息。

- 使用 **display bgp routing-table as-path-filter** { *as-path-filter-number* | *as-path-filter-name* }命令查看与指定 AS 路径过滤器匹配的路由信息。
- 使用 **display bgp routing-table community-filter** { { *community-filter-name* | *basic-community-filter-number* } [**whole-match**] | *advanced-community-filter-number* }命令查看匹配指定 BGP 团体属性过滤器的路由。
- 使用 **display bgp routing-table peer** *ipv4-address* **received-routes** [**active**] [**statistics**]命令查看 BGP 从对等体收到的路由信息。
- 使用 **display bgp routing-table peer** *ipv4-address* **accepted-routes** 命令查看 BGP 从指定邻居收到的通过策略过滤的路由信息。

----结束

8.7 配置 BGP 路由聚合

配置路由聚合，可以减小对等体路由表中的路由数量。

应用环境

在中型或大型 BGP 网络中，BGP 路由表会变得十分庞大，存储路由表占用大量的路由器内存资源，传输和处理路由信息需要占用大量的网络资源。使用路由聚合（Routes Aggregation）可以大大减小路由表的规模；另外通过对路由进行聚合，隐藏一些具体的路由，可以减少路由震荡对网络带来的影响。BGP 路由聚合结合灵活的路由策略，使 BGP 更有效的传递和控制路由。

BGP 支持两种聚合方式：自动聚合和手动聚合。自动聚合的路由优先级低于手动聚合的路由优先级。

前置任务

在配置 BGP 路由聚合之前，需完成以下任务：

- **配置 BGP 的基本功能**

操作步骤

- 配置自动聚合
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **bgp** *as-number*，进入 BGP 视图。
  3. 执行命令 **ipv4-family unicast**，进入 IPv4 单播地址族视图。
  4. 执行命令 **summary automatic**，配置对本地引入的路由自动聚合。

     该命令对 BGP 引入的路由进行聚合，引入的路由可以是直连路由、静态路由、RIP 路由、OSPF 路由、IS-IS 路由。配置该命令后，BGP 将按照自然网段聚合路由。该命令对 **network** 命令引入的路由无效。
- 配置手动聚合
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **bgp** *as-number*，进入 BGP 视图。
  3. 执行命令 **ipv4-family unicast**，进入 IPv4 单播地址族视图。

4. 执行命令 **aggregate** *ipv4-address* { *mask* | *mask-length* } [**as-set** | **attribute-policy** *route-policy-name1* | **detail-suppressed** | **origin-policy** *route-policy-name2* | **suppress-policy** *route-policy-name3*] *，配置手动路由聚合。

   通过设置关键字 **as-set**，可创建一条聚合路由，该路由的自治系统 AS（Autonomous System）路径包含了具体路由的 AS 路径信息。若需聚合较多路由时，请慎用此关键字，因为当具体路由的变化较频繁时，会导致路由振荡。

   关键字 **detail-suppressed** 抑制该聚合路由所包含的所有具体路由，只发布该聚合路由。生成的聚合路由带 Atomic-aggregate 属性，并且不能携带原具体路由的团体属性。

   关键字 **suppress-policy** 能产生聚合路由，但抑制指定路由的通告。可以用 **route-policy** 的 **if-match** 子句有选择地抑制一些具体路由，即匹配该策略的路由将被抑制，但其它未通过策略的具体路由仍被通告。也可以通过 **peer route-policy** 命令，配置不希望发布给对等体的策略达到相同效果。

   使用关键字 **origin-policy** 仅在匹配 **route-policy** 时才生成聚合路由。

   关键字 **attribute-policy** 可设置聚合路由的属性。如果在策略中使用命令 **apply as-path** 配置了 AS_Path 属性，且 **aggregate** 命令设置了关键字 **as-set**，那么策略中的 **apply as-path** 命令将不会生效。通过 **peer route-policy** 命令也可以完成同样的工作。

   手动聚合对 BGP 本地路由表中已经存在的路由有效，例如 BGP 路由表中不存在路由 10.1.1.1/24，即使配置了命令 **aggregate 10.1.1.1 16** 对其进行聚合，BGP 也不会生成聚合路由。

   使用手动聚合时可以应用多种策略，并可以设置路由的属性。

----结束

检查配置结果

完成配置后，可以按以下指导来检查配置结果。

- 使用 **display bgp routing-table** [*network* [*mask* | *mask-length*]]命令查看 BGP 聚合路由信息。

8.8 配置 BGP 对等体组

通过配置 BGP 对等体组，可以简化 BGP 网络配置，提高路由的发布效率。

8.8.1 建立配置任务

在配置 BGP 对等体组前了解此特性的应用环境、配置此特性的前置任务和数据准备，有助于快速、准确地完成配置任务。

应用环境

一个 BGP 对等组是具有相同的更新策略和配置要求的一系列 BGP 邻居。

在大型 BGP 网路中，对等体的数目众多，配置和维护极为不便。对于存在相同配置的 BGP 对等体，可以把它们创建为一个 BGP 对等体组。使用对等体组进行批量配置，可以简化管理的难度，还可以提高路由发布效率。

根据对等体所在的 AS 是否相同，对等体组可分为以下三类：

- IBGP 对等体组：所包括的对等体属于同一个内部 AS。
- 纯 EBGP 对等体组：所包括的对等体属于同一个外部 AS。
- 混合 EBGP 对等体组：所包括的对等体属于不同的外部 AS。

对单个对等体和对等体组同时进行配置了某个功能时，对单个对等体的配置优先生效。创建对等体组之后，可以向对等体组内加入新的对等体，新加入的对等体如果没有单独的配置，那么将继承对等体组的配置。加入对等体组之后，如果某个 BGP 对等体有特殊的配置要求，还可以进行单独的配置，单个对等体的配置将覆盖从对等体组继承的配置。

前置任务

在配置 BGP 对等体组之前，需完成以下任务：

- **配置 BGP 的基本功能**

数据准备

在配置 BGP 对等体组之前，需要准备以下数据。

序号	数据
1	对等体组的类型、名称、所包含的对等体

8.8.2 创建 IBGP 对等体组

BGP 有多个 IBGP 对等体时，创建 IBGP 对等体组可以简化 BGP 网络的配置和管理。创建 IBGP 对等体组不需要指定自治系统号。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **bgp** *as-number*，进入 BGP 视图。

步骤 3 执行命令 **group** *group-name* **internal**，创建 IBGP 对等体组。

步骤 4 执行命令 **peer** *ipv4-address* **group** *group-name*，向对等体组中加入对等体。

说明

重复步骤 4，可向组中加入多个对等体。如果该对等体还未创建，系统会自动在 BGP 视图下创建该对等体，并设置其 AS 编号为对等体组的 AS 编号。

创建 IBGP 对等体组不需要指定自治系统号。

对等体组建立以后，可以为对等体组批量配置 BGP 的功能。默认情况下，对等体组内的对等体将继承对等体组的配置。如果为对等体直接进行单独的配置，那么单独的配置将代替从对等体组继承的配置。

----结束

8.8.3 创建纯 EBGP 对等体组

BGP 有属于同一 AS 的多个 EBGP 对等体时，创建 EBGP 对等体组可以简化 BGP 网络的配置和管理。一个纯 EBGP 对等体组的所有对等体的 AS 号必须相同。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **bgp** *as-number*，进入 BGP 视图。

步骤 3 执行命令 **group** *group-name* **external**，创建 EBGP 对等体组。

步骤 4 执行命令 **peer** *group-name* **as-number** *as-number*，设置该对等体组所属的 AS 编号。

如果对等体组中已经存在对等体，则不能改变该对等体组的自治系统号，也不能使用 **undo** 命令删除已指定的自治系统号。

步骤 5 执行命令 **peer** *ipv4-address* **group** *group-name*，向对等体组中加入对等体。

说明

重复步骤 5，可向组中加入多个对等体。如果该对等体还未创建，系统会自动在 BGP 视图下创建该对等体，并设置其 AS 编号为对等体组的 AS 编号。

对等体组建立以后，可以为对等体组批量配置 BGP 的功能。默认情况下，对等体组内的对等体将继承对等体组的配置。如果为对等体直接进行单独的配置，那么单独的配置将代替从对等体组继承的配置。

----结束

8.8.4 创建混合 EBGP 对等体组

BGP 有属于不同 AS 的多个 EBGP 对等体时，创建混合 EBGP 对等体组，可以简化 BGP 网络的配置和管理。创建混合 EBGP 对等体组时，需要单独指定各对等体的自治系统号。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **bgp** *as-number*，进入 BGP 视图。

步骤 3 执行命令 **group** *group-name* **external**，创建 EBGP 对等体组。

步骤 4 执行命令 **peer** *ipv4-address* **as-number** *as-number*，创建对等体，设置对等体的 AS 编号。

步骤 5 执行命令 **peer** *ipv4-address* **group** *group-name*，向对等体组中加入对等体。

说明

重复步骤 4 和步骤 5，可向组中加入多个对等体。

在混合 EBGP 对等体组中，需要单独指定各对等体的自治系统号。

对等体组建立以后，可以为对等体组批量配置 BGP 的功能。默认情况下，对等体组内的对等体将继承对等体组的配置。如果为对等体直接进行单独的配置，那么单独的配置将代替从对等体组继承的配置。

----结束

8.8.5 检查配置结果

BGP 对等体组配置成功后，可以查看 BGP 对等体的详细信息和对等体组信息。

前提条件

已经完成 BGP 对等体组的所有配置。

操作步骤

- 使用 **display bgp peer** [*ipv4-address*] **verbose** 命令查看对等体详细信息。
- 使用 **display bgp group** [*group-name*]命令查看对等体组信息。

  说明

  该命令只有在创建 BGP 对等体组的设备上执行才能查看该对等体组的信息。

  如果指定了对等体组则显示该对等体组的详细信息；如果没有指定对等体组，将显示所有 BGP 对等体组信息。

----结束

8.9 配置 BGP 路由反射器

通过配置 BGP 路由反射器，可以解决多个 IBGP 对等体建立全连接的问题，简化网络配置，提高路由发布效率。

8.9.1 建立配置任务

在配置 BGP 路由反射器前了解此特性的应用环境、配置此特性的前置任务和数据准备，有助于快速、准确地完成配置任务。

应用环境

BGP 协议使用 AS-Path 属性来防止环路，但是当路由在 AS 内的 IBGP 对等体之间传递时，BGP 不会修改路由的 AS-Path 属性，这就有可能形成环路。为了防止这种情况的发生，BGP 协议规定从 IBGP 对等体收到的路由不能再向其他的 IBGP 对等体发布。这样，为保证 IBGP 对等体之间的连通性，IBGP 对等体之间就必须建立逻辑全连接（Full-mesh）关系。当 IBGP 对等体数目很多时，建立逻辑全连接的路由器开销和配置工作量都很大，而且网络不便于维护。

使用联盟或者路由反射器（Route-Reflector，通常简写为 RR），可以解决这个问题。联盟是把一个大的 AS 划分成一些小的子 AS，这样只需要在子 AS 内保持 IBGP 对等体逻辑全连接关系即可。而路由反射器只需要对作为反射器的路由器进行配置即可，无须改变其他设备的配置，所以相对于联盟来说，路由反射器配置更简单，部署更灵活。

前置任务

在配置 BGP 路由反射器之前，需完成以下任务：

- **配置 BGP 的基本功能**

数据准备

在配置 BGP 路由反射器之前，需要准备以下数据。

序号	数据
1	确定各路由器的角色（RR、客户机、非客户机）
2	（可选）RR 的集群 ID

8.9.2 配置路由反射器及指定客户机

在特定的地址族下配置路由反射器和客户机，可以解决 IBGP 必须逻辑全连接的问题，减少网络配置和维护工作量，优化网络性能。

背景信息

在一个 AS 内，其中一台路由器作为路由反射器（Route Reflector，通常简写为 RR），其它路由器作为客户机（Client）。客户机与 RR 之间建立 IBGP 连接。RR 在客户机之间反射路由信息，客户机之间不需要建立 BGP 连接。既不是 RR 也不是客户机的 BGP 设备被称为非客户机（Non-Client）。非客户机与 RR 之间，以及所有的非客户机之间仍然必须建立全连接关系。

当 RR 收到对等体发来的路由，首先使用 BGP 选路策略来选择最佳路由。在向客户机和非客户机发布学习到的路由信息时，RR 按照如下规则发布路由：

- 从非客户机 IBGP 对等体学到的路由，发布给此 RR 的所有客户机。
- 从客户机学到的路由，发布给此 RR 的所有非客户机和客户机。

另外，对于从 EBGP 对等体学到的路由，RR 会发布给所有的非客户机和客户机。

RR 的配置简单方便，只需要对作为反射器的路由器进行配置，客户机并不需要知道自己是客户机。

请在运行 BGP 协议，并且需要被指定为 RR 的路由器上进行下列配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **bgp** *as-number*，进入 BGP 视图。

步骤 3 执行命令 **ipv4-family unicast**，进入 IPv4 单播地址族视图。

步骤 4 执行命令 **peer** { *ipv4-address* | *group-name* } **reflect-client**，配置 RR 及其客户机。

重复执行此步骤可以为该 RR 添加多个客户机。

在某个地址族下配置的 **reflect-client** 信息只能在该地址族有效，不能被其它地址族继承。因此建议用户在需要的特定的地址族下配置 **reflect-client** 信息。

----结束

8.9.3 （可选）禁止客户机之间的路由反射

当路由反射器的客户机已经是全连接时，禁止客户机之间的路由反射，可以减少链路开销。

背景信息

通常情况下，RR 从客户机学到的路由，会发布给此自己的所有非客户机和客户机。如果 RR 的所有客户机之间已经建立逻辑全连接关系，那么客户机之间就可以相互传递路由，不再需要 RR 为客户机反射路由。在这种情况下，可以禁止客户机之间的路由反射，减少 RR 的负担。

请在运行 BGP 协议的 RR 上进行下列配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **bgp** *as-number*，进入 BGP 视图。

步骤 3 执行命令 **ipv4-family unicast**，进入 IPv4 单播地址族视图。

步骤 4 执行命令 **undo reflect between-clients**，禁止客户机之间的路由反射。

如果 RR 的客户机已经是全连接的，可以使用 **undo reflect between-clients** 命令禁止客户间的反射，以便减少开销。缺省情况下，使能客户机之间的路由反射。

此命令只能在 RR 上配置。

----结束

8.9.4 （可选）配置路由反射器的集群 ID

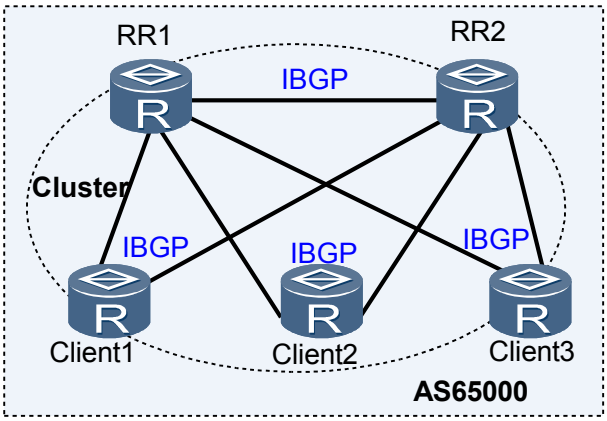
一个集群里有多个 RR 时，给所有位于同一集群内的 RR 配置相同的集群 ID，可以避免路由环路。

背景信息

在 AS 内，为了增加网络的可靠性，避免 RR 单点故障导致其客户机和非客户机无法收到路由信息，通常会设置备份 RR。

如图 8-1 所示，在 AS65000 内，存在两个路由反射器 RR1 和 RR2，其中 RR1 和 RR2 互为备份，为网络提供保障，Client1、Client2 和 Client3 是 RR1 和 RR2 的共同客户机。RR1 和 RR2 之间配置 IBGP 连接，即两个 RR 互为非客户机。

图 8-1 路由反射器集群



在这种组网中，容易出现路由循环。例如，当客户机 Client1 从外部对等体接收到一条更新路由后，它通过 IBGP 向 RR1 和 RR2 通告这条路由。然后会同时出现下面两种情况：

- RR1 会把这条路由向他的客户机和非客户机（RR2）通告。
- RR2 会把这条路由向他的客户机和非客户机（RR1）通告。

这样在两个反射器 RR1 和 RR2 就形成了路由循环。

为了防止路由循环，可以把图 8-1 中所有的路由器配置到一个集群（Cluster）中，并为他们配置一个集群 ID（Cluster-id）。这样，当客户机 Client1 从外部对等体接收到一条更新路由后，它通过 IBGP 向 RR1 和 RR2 通告这条路由，

- RR1 接收到该更新路由后，它向其他的客户机和非客户机反射，同时将本地 Cluster_ID 添加到 Cluster_List 前面。
- RR2 接收到该反射路由后，检查 Cluster_List，发现自己的 Cluster_ID 已经包含在 Cluster_List 中，因此，它丢弃该更新路由。

说明

Cluster_List 的应用保证了同一 AS 内的不同 RR 之间不出现路由循环。

请在运行 BGP 协议的路由器上进行下列配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **bgp** *as-number*，进入 BGP 视图。

步骤 3 执行命令 **ipv4-family unicast**，进入 IPv4 单播地址族视图。

步骤 4 执行命令 **reflector cluster-id** *cluster-id*，配置 RR 的集群 ID。

当一个集群里有多个 RR 时，需要使用此命令给所有位于同一个集群内的 RR 配置相同的 *cluster-id*，以避免路由环路。

说明

为了保证客户机可以学习到反射器发来的路由，反射器上配置的 Cluster ID 不能和客户机的 Cluster ID 相同（缺省情况下，客户机使用自己的 Router ID 做为 Cluster ID），如果相同，客户机会将收到的路由丢弃。

----结束

8.9.5 （可选）禁止 BGP 路由下发到 IP 路由表

在 RR 上禁止 BGP 路由下发到 IP 路由表，可以有效的避免流量从该 RR 转发，提高路由传递效率。

背景信息

通常情况下，BGP 路由下发到 IP 路由表，用于指导流量转发。如果不需要该路由器承担转发任务，就可以禁止 BGP 路由下发到 IP 路由表。

禁止 BGP 路由下发到 IP 路由表主要用在存在 RR 的场景。在一个 AS 内，RR 主要有两个作用，一个是用来传递路由，另外一个是用来转发流量。但是如果 RR 连接了很多客户机和非客户机，路由传递任务较重，RR 的 CPU 资源消耗很大，无力再承担转发任务，这时可以在该 RR 上禁止 BGP 路由下发到 IP 路由表，这样 RR 将主要用来传递路由，提高了路由传递效率。

请在运行 BGP 协议的路由器上进行下列配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **bgp** *as-number*，进入 BGP 视图。

步骤 3 执行命令 **ipv4-family unicast**，进入 IPv4 单播地址族视图。

步骤 4 执行命令 **bgp-rib-only** [**route-policy** *route-policy-name*]，禁止 BGP 路由下发到 IP 路由表。

缺省情况下，BGP 优选的路由下发到 IP 路由表。

bgp-rib-only 命令中配置参数 **route-policy** *route-policy-name* 时，通过策略的路由不下发 IP 路由表，没有通过策略的则正常下发，不修改路由属性。

说明

命令 **bgp-rib-only** 与命令 **active-route-advertise** 互斥。

----结束

8.9.6 检查配置结果

BGP 路由反射器配置成功后，可以查看 BGP 路由反射器的配置信息和传递的 BGP 路由信息。

前提条件

已经完成 BGP 路由反射器的所有配置。

操作步骤

- 使用 **display bgp** [**vpnv4** [**vpn-instance** *vpn-instance-name* | **all**]] **peer** [*ipv4-address*] **verbose** 命令查看对等体详细信息。
- 使用 **display bgp routing-table** [*network* [{ *mask* | *mask-length* } [**longer-prefixes**]]] 命令查看 BGP 路由表中的信息。

----结束

8.10 配置 BGP 联盟

大型 BGP 网络中，配置联盟不但可以减少 IBGP 连接的数量，还可以简化路由策略的管理，提高路由的发布效率。

应用环境

联盟是处理 AS 内部的 IBGP 网络连接激增的一种方法，它将一个自治系统划分为若干个子自治系统，每个子自治系统内部的 IBGP 对等体建立全连接关系或者配置反射器，子自治系统之间建立 EBGP 连接关系。

相比较于反射器，联盟更便于实现 IGP 扩展。

前置任务

在配置 BGP 联盟之前，需完成以下任务：

- 配置接口的链路层协议参数（和 IP 地址），使接口的链路协议状态为 Up
- **配置 BGP 的基本功能**

操作步骤

- BGP 联盟的基本配置

  请在运行 BGP 协议的路由器上进行以下配置。

  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **bgp** *as-number*，进入 BGP 视图。
  3. 执行命令 **confederation id** *as-number*，配置联盟 ID。
  4. 执行命令 **confederation peer-as** *as-number* &<1-32>，指定与本地 AS 连接的其他 EBGP 对等体所属的子自治系统号。

     配置属于联盟的子自治系统时使用的 *as-number* 在联盟内部有效。

     属于同一联盟的所有 EBGP 对等体都必须配置 **confederation id** 和 **confederation peer-as** 命令，且指定相同的联盟 ID。

     说明

     同一联盟内不能同时配置 2 字节 AS 号的 Old Speaker 和 4 字节 AS 号的 New Speaker。因为 AS4_Path 不支持联盟，这种配置可能会引起环路。

- 配置联盟的兼容性

  如果其他路由器的联盟实现机制不同于 RFC 标准，可以配置此命令，以便和非标准的设备兼容。请在运行 BGP 协议的路由器上进行下列配置。

1. 执行命令 **system-view**，进入系统视图。
2. 执行命令 **bgp** *as-number*，进入 BGP 视图。
3. 执行命令 **confederation nonstandard**，配置与非标准的 AS 联盟兼容。

   缺省情况下，配置的联盟与 RFC3065 一致。

----结束

检查配置结果

完成配置后，可以按以下指导来检查配置结果。

- 使用 **display bgp peer** [*ipv4-address*] **verbose** 命令查看对等体详细信息。
- 使用 **display bgp routing-table** [*network* [{ *mask* | *mask-length* } [**longer-prefixes**]]] 命令查看 BGP 路由表中的路由信息。

8.11 配置 BGP 团体属性

团体属性可以简化路由策略的管理。

8.11.1 建立配置任务

在配置 BGP 团体属性前了解此特性的应用环境、配置此特性的前置任务和数据准备，有助于快速、准确地完成配置任务。

应用环境

团体属性用来简化路由策略的应用和降低维护管理的难度，利用团体可以使多个 AS 中的一组 BGP 路由器共享相同的策略。BGP 路由器在将带有团体属性的路由发布给其它对等体之前，可以先改变此路由原有的团体属性。团体属性是一种路由属性，在 BGP 对等体之间传播，且不受 AS 的限制。

前置任务

在配置 BGP 团体属性之前，需完成以下任务：

- **配置 BGP 的基本功能**

数据准备

在控制 BGP 路由信息的发布与接收之前，需要准备以下数据。

序号	数据
1	团体属性值
2	路由策略的名称、节点序号、匹配条件
3	过滤方向（发布/接收）和所使用的路由策略名称

8.11.2 配置团体属性相关路由策略

为路由信息配置团体属性，需要先配置应用了团体属性的相关策略。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **route-policy** *route-policy-name* { **deny** | **permit** } **node** *node*，创建路由策略的节点，并进入路由策略视图。

步骤 3 （可选）配置路由策略过滤条件，即配置 If-match 子句。只有通过过滤条件的路由信息才能被添加或修改团体属性。

具体配置可以参见**（可选）配置 If-match 子句**

步骤 4 配置 BGP 路由信息的团体属性或扩展团体属性。

- 执行命令 **apply community** { { *community-number* | *aa:nn* } &<1-32> | **internet** | **no-advertise** | **no-export** | **no-export-subconfed** } * [**additive**]，配置 BGP 路由信息的团体属性。

  说明

  一条命令中最多可以配置 32 个团体属性。

- 执行命令 **apply extcommunity** { **rt** { *as-number:nn* | *ipv4-address:nn* } } &<1-16> [**additive**]，配置 BGP 扩展团体属性（Route-Target）。

----结束

8.11.3 配置发布团体属性

只有配置发布团体属性，在路由策略中定义的团体属性才能生效。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **bgp** *as-number*，进入 BGP 视图。

步骤 3 执行命令 **ipv4-family unicast**，进入 IPv4 单播地址族视图。

步骤 4 执行命令 **peer** { *ipv4-address* | *group-name* } **route-policy** *route-policy-name* **export**，配置出方向的路由策略。

说明

配置 BGP 团体时，必须使用路由策略来定义具体的团体属性，然后在发布路由信息时应用此路由策略。

关于路由策略的配置，请参考“路由策略配置”。

步骤 5 根据实际组网选择执行如下命令，将团体属性传给对等体/对等体组。

- 将标准团体属性传给对等体/对等体组：执行命令 **peer** { *ipv4-address* | *group-name* } **advertise-community**

  缺省情况下，不将团体属性发布给任何对等体（组）。

- 将扩展团体属性传给对等体/对等体组：执行命令 **peer** { *ipv4-address* | *group-name* } **advertise-ext-community**

缺省情况下，不将扩展团体属性发布给任何对等体（组）。

----结束

8.11.4 检查配置结果

BGP 团体属性配置成功后，可以查看 BGP 团体属性的相关信息。

前提条件

已经完成 BGP 团体属性的所有配置。

操作步骤

- 使用 **display bgp routing-table** *network* [*mask* | *mask-length*]命令查看指定 BGP 路由的详细信息。
- 使用 **display bgp routing-table community** [*community-number* | *aa:nn*] &<1-29> [**internet** | **no-advertise** | **no-export** | **no-export-subconfed**] * [**whole-match**]命令查看指定 BGP 团体的路由信息。

----结束

8.12 配置基于前缀的 BGP ORF

通过配置基于前缀的 BGP ORF，可以将本端基于前缀的入口策略发送给对端，使对端在发送路由时应用该入口策略对路由进行过滤，实现路由按需发送。

应用环境

BGP 传递路由时可以在路由接收设备和发送设备上应用路由策略对路由进行过滤。

- 在路由接收端对路由进行过滤，则路由发送设备需要发送大量路由，路由接收设备需要处理大量不需要的路由，而且这些不需要的路由会占用网络带宽。
- 在路由发送端对路由进行过滤，则当路由发送端有很多 BGP 邻居时，需要在路由发送端配置大量的出口策略，不便于网络规划和维护，而且这些策略会占用路由发送设备大量的内存资源。

基于前缀的 BGP ORF（Outbound Route Filters）用于实现 BGP 路由的按需发布。BGP 在发布路由时对路由按照出口策略进行过滤，而这个出口策略由对端设备（路由接收者）提供，本端设备无须为每一个 BGP 邻居都配置一个出口策略。这样，减轻了两端设备的负担，节省了网络带宽，减少了配置的工作量。

说明

目前仅支持基于地址前缀列表（IP-Prefix List）的出口策略。

前置任务

在配置基于前缀的 BGP ORF 之前，需完成以下任务：

- 配置 BGP 的基本功能
- **配置 IPv4 地址前缀列表**

数据准备

在配置基于前缀的 BGP ORF 之前，需要准备以下数据。

序号	数据
1	对等体地址或对等体组名称
2	地址前缀列表名称

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **bgp** *as-number*，进入 BGP 视图。

步骤 3 执行命令 **ipv4-family unicast**，进入 IPv4 单播地址族视图。

步骤 4 执行命令 **peer** { *group-name* | *ipv4-address* } **capability-advertise orf** [**cisco-compatible**] **ip-prefix** { **both** | **receive** | **send** }，配置 BGP 对等体（组）使能基于地址前缀列表的 ORF 功能。

缺省情况下，未使能 BGP 对等体（组）基于地址前缀列表的 ORF 功能。

说明

该步骤需要在两端设备上都进行配置。

由于其他厂商设备的 ORF 能力码与 RFC 规定的能力码可能不同，因此为了与其他厂商设备互通，需要确认两端使能了相同的模式（都是 **cisco-compatible** 模式或者都是 **rfc-compatible** 模式），缺省情况下，是 **rfc-compatible** 模式。

BGP ORF 有三种模式：**send**，**receive**，**both**。**send** 模式说明可以发送 ORF 信息；**receive** 模式说明可以接收 ORF 信息；**both** 模式说明既可以发送也可以接收 ORF 信息。如果要让本端设备（路由发送端）能够接收 ORF IP-Prefix 前缀信息，则本端设备必须配置 **both** 或者 **receive**，对端设备（路由接收端）必须配置 **both** 或者 **send**。

步骤 5 执行命令 **peer** { *group-name* | *ipv4-address* } **ip-prefix** *ip-prefix-name* **import**，配置对等体（组）基于地址前缀列表的入口路由过滤策略。

说明

该步骤只需在路由接收端配置。名为 *ip-prefix-name* 的地址前缀列表必须已经配置，否则无法对路由进行过滤。IPv4 地址前缀列表的配置请参考**配置 IPv4 地址前缀列表**。

----结束

检查配置结果

完成基于前缀的 BGP ORF 的所有配置后，可以按以下指导来检查配置结果。

- 执行 **display bgp peer** [*ipv4-address*] **verbose** 命令查看基于前缀的 BGP ORF 协商信息。
- 执行 **display bgp peer** *ipv4-address* **orf ip-prefix** 命令查看从指定对等体收到的基于地址前缀的 BGP ORF 信息。

  说明

  display bgp peer *ipv4-address* **orf ip-prefix** 命令只有在路由发送端执行才会有显示结果。

8.13 调整 BGP 网络的收敛速度

通过调整 BGP 对等体间的连接参数，可以对 BGP 网络的收敛速度进行调整和优化，从而适应大型网络中网络状况不断变化的情况。

8.13.1 建立配置任务

在配置调整 BGP 网络的收敛速度前了解此特性的应用环境、配置此特性的前置任务和数据准备，有助于快速、准确地完成配置任务。

应用环境

BGP 用于大型网络中传递路由。但是大型网络中网络状况变化频繁，这会影响到 BGP 邻居关系的建立和维持，进而会影响到 BGP 网络的收敛速度。

BGP 协议自身的增量更新和路由衰减能够在一定程度上抑制路由频繁变化的情况，但是不能从本质上减弱网络震荡对 BGP 网络连接的影响。

通过配置 BGP 定时器、去使能 EBGP 连接快速复位、使能 BGP tracking 等功能可以抑制 BGP 网络震荡，提高 BGP 网络的收敛速度。

- BGP 连接重传定时器

  连接重传定时器用来设置 BGP 两次发起 TCP 连接的时间间隔。当 BGP 发起 TCP 连接后，如果成功建立起 TCP 连接，则关闭连接重传定时器。如果 TCP 连接建立不成功，则会在连接重传定时器超时后重新尝试建立连接。

  通过改变 BGP 连接重传定时器的大小可以加速 BGP 邻居的建立或者减缓 BGP 邻居的建立。比如，减小 BGP 连接重传定时器，将减小 BGP 再次发起 TCP 连接的等待时间，加快重新建立的速度。而在有些特殊场合，某个 BGP 邻居反复震荡，这时可以增大 BGP 连接重传定时器，抑制由于邻居震荡导致的路由震荡，便于 BGP 网络快速收敛。

- BGP 存活定时器和保持定时器

  BGP 的 Keepalive 消息用于维护 BGP 邻居关系，并且检测连接的有效性。

  当对等体间建立了 BGP 连接后，它们定时向对端发送 Keepalive 消息，以防止路由器认为 BGP 连接已中断。若路由器在设定的连接保持时间（Hold time）内未收到对端的 Keepalive 消息或任何其它类型的报文，则认为此 BGP 连接已中断，从而退出此 BGP 连接。

- BGP 更新报文定时器

  BGP 协议不会定期更新路由表，当 BGP 路由发生变化时，才会通过 Update 消息增量地更新路由表。但如果同一路由频繁变化时，为避免每次变化路由器都要发送 Update 消息给对等体，可以配置发送该同一路由的 Update 消息的时间间隔。

- EBGP 连接快速复位

  EBGP 连接快速复位功能，缺省情况下是使能的，目的是为了使 EBGP 快速感知 EBGP 连接所使用的接口的状态。但是在有些场合，接口的状态变化频繁，这时可以去使能 EBGP 连接快速复位功能，EBGP 直连会话不会随着接口不断 Up 和 Down 而重建与删除，便于网络快速收敛。

- BGP tracking

使能 BGP tracking 功能，可以通过调整从发现邻居不可达到中断连接的时间间隔，来调整 BGP 网络的收敛速度。而且该功能部署简单，扩展性好。

前置任务

在配置调整 BGP 网络的收敛速度之前，需完成以下任务：

- **配置 BGP 的基本功能**

数据准备

在配置调整 BGP 网络的收敛速度之前，需要准备以下数据。

序号	数据
1	BGP 连接重传定时器的值
2	BGP 存活定时器和保持定时器的值
3	BGP 更新报文定时器的值
4	BGP 发现邻居不可达到中断连接的时间间隔

8.13.2 配置 BGP 连接重传定时器

通过改变 BGP 连接重传定时器值的大小可以加快或者减缓 BGP 邻居建立的速度。

背景信息

BGP 发起 TCP 连接后，如果成功建立起 TCP 连接，则关闭连接重传定时器。如果 TCP 连接建立不成功，则会在连接重传定时器超时后重新尝试建立连接。

- 设置较小的连接重传定时器，可以减少等待下次连接建立的时间，加快连接失败后重新建立的速度。
- 设置较大的连接重传定时器，可以减小由于邻居反复震荡引起的路由振荡。

BGP 支持在全局或者单个对等体（组）配置连接重传定时器。定时器生效的优先级是单个对等体高于对等体组，对等体组高于全局。

操作步骤

- 配置全局连接重传定时器

  请在运行 BGP 协议的路由器上进行下列配置。

  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **bgp** *as-number*，进入 BGP 视图。
  3. 执行命令 **timer connect-retry** *connect-retry-time*，配置 BGP 全局连接重传定时器。

     缺省情况下，连接重传定时器是 32 秒。
- 配置对等体或对等体组的连接重传定时器

  请在运行 BGP 协议的路由器上进行下列配置。

1. 执行命令 **system-view**，进入系统视图。
2. 执行命令 **bgp** *as-number*，进入 BGP 视图。
3. 执行命令 **peer** { *group-name* | *ipv4-address* } **timer connect-retry** *connect-retry-time*，配置对等体或对等体组的连接重传定时器。

   缺省情况下，连接重传定时器是 32 秒。

   对等体（组）连接重传定时器的优先级高于全局连接重传定时器的优先级。

----结束

8.13.3 配置 BGP 存活时间和保持时间定时器

BGP 存活时间定时器和保持时间定时器的大小决定了 BGP 感知网络故障的速度，按照实际网络状况调整定时器的值可以增强网络性能。

背景信息

BGP 的 Keepalive 消息用于维持 BGP 连接关系。一对 BGP 对等体之间依靠互相发送 Keepalive 消息来告诉邻居自己的状态，如果 BGP 在保持时间定时器超时后，仍旧未收到邻居发来的 Keepalive 消息，则会认为连接已经中断。

- 减小存活时间和保持时间，BGP 可以更快速的检测到链路的故障，有利于 BGP 网络快速收敛，但是网络中的 Keepalive 消息会增多，会增加路由器的负担，并且会占用一定的网络带宽。
- 增大存活时间和保持时间，网络中的 Keepalive 消息减少，这样会减少路由器的负担和网络带宽的占用，但是过长的保持时间使得 BGP 不能及时检测到链路状态的变化，不利于 BGP 网络快速收敛，可能会造成较多的流量损失。

注意

改变定时器的值（执行 **timer** 或 **peer timer** 命令）会导致路由器之间的 BGP Peer 关系中断。务必仔细确认是否必须改变定时器的值。

BGP 支持在全局或者单个对等体（组）配置存活时间和保持时间定时器。定时器生效的优先级单个对等体高于对等体组，对等体组高于全局。

操作步骤

- 配置全局定时器

  请在运行 BGP 协议的路由器上进行下列配置。

  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **bgp** *as-number*，进入 BGP 视图。
  3. 执行命令 **timer keepalive** *keepalive-time* **hold** *hold-time*，配置 BGP 定时器。

     合理的最大 Keepalive 消息发送间隔为保持时间的三分之一，且该发送间隔不能小于 1 秒，因此，保持时间如果不为 0，则最小为 3 秒。缺省情况下，存活时间为 60 秒，保持时间为 180 秒。

说明

建议配置的保持时间大于 20 秒。如果保持时间小于 20 秒，可能会造成邻居会话的中断。

以下两种定时器取值配置需要尽量避免：

- *keepalive-time* 值和 *hold-time* 值同时取 0，这种配置将导致 BGP 定时器无效，即 BGP 不会发送 Keepalive 消息。
- *hold-time* 值远大于 *keepalive-time* 值，例如，不能将 *keepalive-time* 值设置为 1 而将 *hold-time* 值设置为 65535。因为如果保持时间过长，BGP 将不能及时检测到连接的有效性。

当对等体建立连接之后，实际的 *keepalive-time* 值和 *hold-time* 值是通过双方协商来确定的。其中，取对等体双方的 Open 报文中的 *hold-time* 的较小值为最终的 *hold-time* 值；取（协商的 *hold-time* 值÷3）和本地配置的 *keepalive-time* 值中较小的作为最终的 *keepalive-time* 值。

- 配置对等体的定时器

  请在运行 BGP 协议的路由器上进行下列配置。

  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **bgp** *as-number*，进入 BGP 视图。
  3. 执行命令 **peer** { *ipv4-address* | *group-name* } **timer keepalive** *keepalive-time* **hold** *hold-time*，配置对等体或对等体组的 keepalive 发送间隔和保持时间。

     存活时间和保持时间的取值关系同**配置全局定时器**。

     说明

     建议配置的保持时间大于 20 秒。如果保持时间小于 20 秒，可能会造成邻居会话的中断。

     对等体定时器的优先级高于全局定时器。

----结束

8.13.4 配置更新报文定时器

设置合理的更新报文定时器，可以抑制路由频繁变化对 BGP 的影响，有利于 BGP 网络稳定。

背景信息

BGP 的 Update 消息用于在对等体之间交换路由信息。Update 消息可以发布多条属性相同的可达路由信息，也可以撤销多条不可达路由信息。

BGP 协议不会定期更新路由表，当 BGP 路由发生变化时，才会通过 Update 消息增量地更新路由表。但如果同一路由频繁变化时，为避免每次变化路由器都要发送 Update 消息给对等体，可以配置发送该同一路由的 Update 消息的时间间隔。

请在运行 BGP 协议的路由器上进行下列配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **bgp** *as-number*，进入 BGP 视图。

步骤 3 执行命令 **peer** { *ipv4-address* | *group-name* } **route-update-interval** *interval*，配置更新报文定时器。

缺省情况下，IBGP 对等体的路由更新定时器为 15 秒，EBGP 对等体的路由更新定时器为 30 秒。

ipv4-address 和 *group-name* 参数分别表示对单个对等体和对等体组进行配置。路由更新定时器生效的优先级关系是个体优先，即单个对等体优先于对等体组。

----结束

8.13.5 去使能 EBGP 连接快速复位

通过去使能 EBGP 连接快速复位功能，可以防止路由振荡带来的 EBGP 会话的反复重建与删除，有利于 BGP 网络快速收敛。

背景信息

EBGP 连接快速复位功能缺省情况下是使能的，目的是为了使 BGP 协议不必等待保持时间定时器超时，而立即快速响应接口故障，删除接口上的 EBGP 直连会话，便于 BGP 快速收敛。

说明

EBGP 连接快速复位功能只能快速响应接口故障，而不能快速响应接口故障恢复。接口故障恢复后，BGP 协议依靠自身状态机机制来恢复会话。

但是如果 EBGP 连接所使用的接口状态反复变化，EBGP 会话就会反复重建与删除，造成网络震荡。这时，可以去使能 EBGP 连接快速复位功能，BGP 协议会等待保持时间定时器超时，才会删除接口上的 EBGP 直连会话，这样就在一定程度上抑制了 BGP 网络震荡，有利于 BGP 网络快速收敛。同时，在一定程度上节约了网络带宽。

请在运行 BGP 协议的路由器上进行下列配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **bgp** *as-number*，进入 BGP 视图。

步骤 3 执行命令 **undo ebgp-interface-sensitive**，去使能 EBGP 连接快速复位。

说明

该命令适用于 EBGP 连接所使用的接口状态不断变化的场合。如果接口状态恢复稳定，建议立即执行 **ebgp-interface-sensitive** 命令恢复缺省配置，也即，使能 EBGP 连接快速复位功能，从而便于 BGP 快速收敛。

----结束

8.13.6 使能 BGP Tracking

通过部署 BGP Tracking 功能，调整从发现邻居不可达到中断连接的时间间隔，可以抑制路由震荡引发的 BGP 邻居关系震荡，提高 BGP 网络的稳定性。

背景信息

为了实现 BGP 快速收敛，可以通过配置 BFD 来探测邻居状态变化，但 BFD 需要全网部署，扩展性较差。在无法部署 BFD 检测邻居状态时，可以本地配置 BGP tracking 功能，快速感知链路不可达或者邻居不可达，实现网络的快速收敛。

请在 BGP 路由器上进行如下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **bgp** *as-number*，进入 BGP 视图。

步骤 3 执行命令 **peer** { *group-name* | *ipv4-address* } **tracking** [**delay** *delay-time*]，使能对于指定对等体的 BGP Tracking 功能。

缺省情况下，不使能 BGP Tracking。

group-name 和 *ipv4-address* 参数分别代表对对等体组和单个对等体进行配置。BGP Tracking 定时器生效的优先级关系是个体优先，即单个对等体优先于对等体组。

如果不指定参数 *delay-time* 的值，则表示使用缺省值 0 秒，即，BGP 发现邻居不可达后立即断开连接。

根据实际组网，配置合适的 *delay-time* 可以保证网络的稳定性。

- 如果 IBGP 邻居的建立依赖于 IGP 路由，建议 IBGP 邻居配置的 *delay-time* 大于 IGP 路由收敛时间。否则，如果网络中的闪断导致 IGP 路由震荡那么在 IGP 路由震荡恢复之前，BGP 邻居关系就已经断连。

  说明

  IGP 配置了 GR（Graceful-Restart）能力，且 BGP 邻居建立依赖于 IGP 路由，当网络中的节点发生故障并主备倒换时，IGP 不删除从该节点收到的路由，依赖此路由建立起来的 BGP 邻居也不感知节点发生了故障，此时 BGP Tracking 功能不生效。

- 如果 BGP 邻居 GR 能力协商成功，BGP 邻居主备倒换，建议配置的 *delay-time* 大于 GR 收敛时间。否则，在 GR 收敛之前，BGP 邻居已经中断连接，导致 GR 失效。

----结束

8.13.7 检查配置结果

调整 BGP 网络收敛速度配置成功后，可以查看 BGP 对等体和对等体组信息。

前提条件

已经完成调整 BGP 网络收敛速度的所有配置。

操作步骤

- 使用 **display bgp peer** [**verbose**]命令查看 BGP 对等体信息。
- 使用 **display bgp group** [*group-name*]命令查看 BGP 对等体组信息。

----结束

8.14 配置 BGP 路由衰减

通过配置 BGP 路由衰减，可以抑制不稳定的 BGP 路由。

应用环境

路由不稳定的主要表现形式是路由振荡（Route flapping），即路由表中的某条路由反复消失和重现。一般情况下，BGP 都应用于复杂的网络环境中，路由变化十分频繁。而频繁的路由振荡会消耗大量的带宽资源和 CPU 资源，严重时会影响到网络的正常工作。

BGP 路由衰减（Route Dampening）可以用来解决路由振荡的问题，它使用惩罚值（Penalty Value）来衡量一条路由的稳定性。当一条路由出现振荡，就给他分配一个惩罚值。振荡越多，惩罚值越高。如果惩罚值超出预设的门限，该路由就不再对外发布。直到一段时间后惩罚值降低到可重新使用的门限值。

路由衰减只适用于 EBGP 路由。对于从 IBGP 收来的路由不能进行衰减，因为 IBGP 路由经常含有本 AS 的路由，内部网络路由要求转发表尽可能一致，IGP 快速收敛就是为了达到信息同步，转发一致。如果衰减对 IBGP 路由起作用，不同路由器的衰减参数不一致时，会导致转发表不一致。

前置任务

在配置 BGP 振荡抑制之前，需完成以下任务：

- 配置 BGP 的基本功能

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **bgp** *as-number*，进入 BGP 视图。

步骤 3 执行命令 **ipv4-family unicast**，进入 IPv4 单播地址族视图。

步骤 4 执行命令 **dampening** [*half-life-reach reuse suppress ceiling* | **route-policy** *route-policy-name*] *，配置 BGP 路由衰减参数。

说明

dampening 命令只对 EBGP 路由生效。

配置 BGP 路由衰减时，所指定的 *reuse*、*suppress*、*ceiling* 三个阈值是依次增大的，即必须满足：*reuse*<*suppress*<*ceiling*。

通过按策略区分路由，当 **dampening** 命令引用路由策略时，BGP 可以对不同的路由采用不同的 Dampening 参数进行抑制处理。

----结束

检查配置结果

完成配置后，可以按以下指导来检查配置结果。

- 使用 **display bgp routing-table flap-info** [**regular-expression** *as-regular-expression* | **as-path-filter** *as-path-filter-number* | *network-address* [{ *mask* | *mask-length* } [**longer-match**]]]命令查看路由振荡统计信息。
- 使用 **display bgp routing-table dampened** 命令查看 BGP 衰减的路由。
- 使用 **display bgp routing-table dampening parameter** 命令查看 BGP 衰减的配置参数。

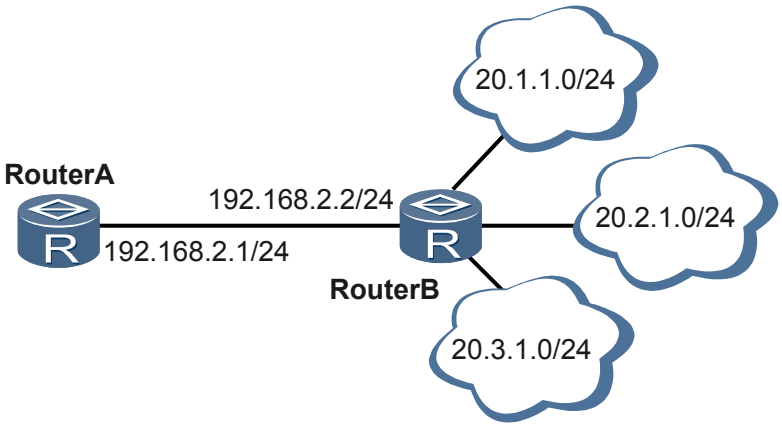
8.15 配置向对等体发送缺省路由

配置向对等体发送缺省路由功能后，无论本地路由表中是否存在缺省路由，都将向指定对等体发布一条下一跳地址为本地地址的缺省路由。通过向对等体发送缺省路由，可以减少网络中的路由数量。

应用环境

在中型或大型 BGP 网络中，BGP 路由表会变得十分庞大，存储路由表占用大量的路由器内存资源，传输和处理路由信息需要占用大量的网络资源。当对等体的 BGP 路由表中的多条路由都只是由本端发送时，可以在本端配置向对等体发送缺省路由功能。无论本端的路由表中是否存在缺省路由，都向对等体发布一条下一跳地址为本地地址的缺省路由，这可以很大程度地减少网络路由数量，节省对等体的内存资源与网络资源。

图 8-2 向对等体发送缺省路由典型组网图



如图 8-2 所示，RouterA 与 RouterB 之间建立 BGP 对等体关系，RouterB 通过引入路由的方式将到达 20.1.1.0/24、20.2.1.0/24 和 20.3.1.0/24 三个网段的路由加入到 BGP 路由表中，并且 RouterA 通过 RouterB 学习到这 3 条路由。这样 RouterA 上就会保留 3 条 BGP 路由。如果希望节省 RouterA 上的存储资源与 RouterB 向 RouterA 的传输占用的带宽资源，可以通过在 RouterB 上配置向对等体发送缺省路由功能，并使用路由策略禁止 20.1.1.0/24、20.2.1.0/24 和 20.3.1.0/24 三个网段的路由发往 RouterA，使 RouterA 上只保留一条缺省路由，而流量依然可以到达那三个网段。

前置任务

在配置向对等体发送缺省路由之前，需完成以下任务：

- **配置 BGP 的基本功能**

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **bgp** *as-number*，进入 BGP 视图。

步骤 3 执行命令 **ipv4-family unicast**，进入 IPv4 单播地址族视图。

步骤 4 执行命令 **peer** { *group-name* | *ipv4-address* } **default-route-advertise** [**route-policy** *route-policy-name*] [**conditional-route-match-all** { *ipv4-address1* { *mask1* | *mask-length1* } } &<1-4> | **conditional-route-match-any** { *ipv4-address2* { *mask2* | *mask-length2* } } &<1-4>]，向对等体或对等体组发送缺省路由。

配置 **route-policy** *route-policy-name* 参数，可以修改发 BGP 发布的缺省路由的属性。

当配置 **conditional-route-match-all** { *ipv4-address1* { *mask1* | *mask-length1* } } &<1-4>参数后，只有参数值中指定的路由都包含于本端 IP 路由表中时，本端才向对等体发送缺省路由。

当配置 **conditional-route-match-any** { *ipv4-address2* { *mask2* | *mask-length2* } } &<1-4>参数后，只要本端 IP 路由表中含有参数值中指定的任意一条路由，本端就会向对等体发送缺省路由。

说明

执行 **peer default-route-advertise** 命令后，不论本地路由表中是否存在缺省路由，都将向指定对等体发布一条下一跳地址为本地地址的缺省路由。

----结束

检查配置结果

完成配置后，可以按以下指导来检查配置结果。

- 在对等体上使用 **display bgp routing-table** [*ipv4-address* [*mask* | *mask-length*]]看收到的 BGP 缺省路由。

8.16 配置 BGP 负载分担

通过配置 BGP 负载分担，可以合理利用网络资源，减少网络拥塞。

应用环境

在大型网路中，到达同一目的地通常会存在多条有效路由，但是 BGP 只将最优路由发布给对等体，这一特点往往会造成很多流量负载不均衡的情况。

有两种方法解决流量负载不均衡的问题，

- 通过 BGP 强大的策略控制流量的负载均衡。例如通过路由策略修改 BGP 路由的本地优先级（Local_Pref）、AS 路径（AS_Path）、Origin 和 MED（Multi Exit Discriminator）等属性来引导网络流量走不同的路径，实现负载均衡。修改 BGP 路由的属性的配置请参考**配置 BGP 的路由属性**。
- 通过多路径选路实现负载分担，达到负载均衡的目的。这种负载分担的特点是需要存在等价路由，通过配置等价路由负载分担的路由条数，可以实现多路径负载分担。

说明

只有路由属性中，**AR150/200 中支持的 BGP 特性**中“BGP 选择路由的策略”所描述的前 8 个属性完全相同，且 AS-Path 属性也相同时，才能成为 BGP 等价路由，实现 BGP 的负载分担。

前置任务

在配置 BGP 负载分担之前，需完成以下任务：

- **配置 BGP 的基本功能**

数据准备

在配置 BGP 负载分担之前，需要准备以下数据。

序号	数据
1	配置 BGP 负载分担的路由条数
2	配置 EBGP 和 IBGP 负载分担的路由条数

操作步骤

- 配置 BGP 路由负载分担的路由条数

  请在运行 BGP 协议的路由器上进行下列配置。

  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **bgp** *as-number*，进入 BGP 视图。
  3. 执行命令 **ipv4-family unicast**，进入 IPv4 单播地址族视图。
  4. 执行命令 **maximum load-balancing** [**ebgp** | **ibgp**] *number*，配置 BGP 负载分担的路由条数。

     缺省情况下，BGP 负载分担的路由条数为 1，也就是不进行负载分担。

     - 选择 **ebgp** 参数，仅 EBGP 路由参与负载分担。
     - 选择 **ibgp** 参数，仅 IBGP 路由参与负载分担。
     - **ebgp** 和 **ibgp** 参数都不被选择，EBGP 和 IBGP 路由都参与负载分担，且参与负载分担的路由条数相同。

     说明

     如果配置了 **maximum load-balancing** *number* 命令，那么再配置 **maximum load-balancing ebgp** *number* 或 **maximum load-balancing ibgp** *number* 命令都不会生效；如果配置了 **maximum load-balancing ebgp** *number* 或 **maximum load-balancing ibgp** *number* 命令，那么再配置 **maximum load-balancing** *number* 命令也不会生效。

     在公网中到达同一目的地的路由形成负载分担时，系统会首先判断最优路由的类型。若最优路由为 IBGP 路由则只是 IBGP 路由参与负载分担，若最优路由为 EBGP 路由则只是 EBGP 路由参与负载分担，即公网中到达同一目的地的 IBGP 和 EBGP 路由不能形成负载分担。
  5. （可选）执行命令 **load-balancing as-path-ignore**，配置路由在形成负载分担时不比较路由的 AS-Path 属性。

     缺省情况下，路由在形成负载分担时比较路由的 AS-Path 属性。

说明

- 如果到达目的地址存在多条路由，但是这些路由分别经过了不同的 AS，缺省情况下，这些路由不能形成负载分担。如果用户需要这些路由参与负载分担，就可以执行 **load-balancing as-path-ignore** 命令。配置 **load-balancing as-path-ignore** 命令后会改变路由参与负载分担的条件，路由形成负载分担时不再比较 AS-Path 属性，配置时需要慎重考虑。
- **load-balancing as-path-ignore** 命令和 **bestroute as-path-ignore** 命令互斥，不能同时使能。

- 配置 EBGP 和 IBGP 路由负载分担的最大条数

  这种配置主要用于 VPN 里 CE 双归属的场景。当一台 CE 双归属两台 PE，CE 和其中一台 PE 处于相同的 AS，和另外一台 PE 处于不同的 AS，这时可以配置 EBGP 和 IBGP 路由负载分担的条数，使路由的类型（EBGP/IBGP）不再作为判断条件，从而实现私网流量在 EBGP 和 IBGP 路由之间负载分担。

  请在运行 BGP 协议的路由器上进行下列配置。

  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **bgp** *as-number*，进入 BGP 视图。
  3. 执行命令 **ipv4-family vpn-instance** *vpn-instance-name*，进入 BGP-VPN 实例视图。
  4. 执行命令 **maximum load-balancing eibgp** *number*，配置 EBGP 和 IBGP 路由负载分担的最大条数。

     缺省情况下，系统不配置 EBGP 和 IBGP 路由负载分担的最大条数。
  5. （可选）执行命令 **load-balancing as-path-ignore**，配置路由在形成负载分担时不比较路由的 AS-Path 属性。

     缺省情况下，路由在形成负载分担时比较路由的 AS-Path 属性。

     说明

     - 配置 **load-balancing as-path-ignore** 命令后会改变路由负载分担的条件，路由形成负载分担时不再比较 AS-Path 属性，配置时需要慎重考虑。
     - **load-balancing as-path-ignore** 命令和 **bestroute as-path-ignore** 命令互斥，不能同时使能。

----结束

检查配置结果

完成 BGP 负载分担的所有配置后，可以按以下指导来检查配置结果。

- 使用 **display bgp routing-table** [*network* [{ *mask* | *mask-length* } [**longer-prefixes**]]] 命令查看 BGP 路由表中的信息。
- 使用 **display ip routing-table vpn-instance** *vpn-instance-name* [**verbose**]命令查看 VPN 实例路由表的信息。

8.17 配置路径 MTU 自动发现功能

通过配置路径 MTU 自动发现功能，可以发现从源端到目的端的路径上最小 MTU 值，使 BGP 消息按照路径 MTU 传输，提高传输效率，增强 BGP 性能。

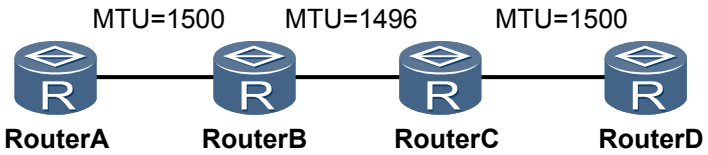
应用环境

通信路径上不同网络的链路层 MTU 不同，当主机间要通过很多网络才能通信时，对通信双方最重要的是通信路径中最小的 MTU。通信路径中最小的 MTU 被称为路径 MTU。

路径 MTU 取决于所选择的路径，而主机间的路径 MTU 值有可能会发生变化，并且在通信的两个方向上可能不一致。使能路径 MTU 自动发现功能可以发现从源端到目的端的路径 MTU 值，路径 MTU 是 TCP 在传输 BGP 消息时封装 IP 数据包的依据。

如图 8-3 所示，RouterA 和 RouterD 之间建立 BGP 邻居，BGP 消息封装在 TCP 数据包中传送，缺省最大报文段长度 MSS（Maximum Segment Size）值为 536，当 RouterA 向 RouterD 发送 BGP 消息时，只能以 MSS 等于 536 的最大报文段长度进行传输。这样大量的 BGP 消息会被分配到不同的报文中，并且与 BGP 消息对应的 ACK 报文也会比较多，这样的传输方式效率低下。此时可以在 BGP 邻居之间使用路径 MTU 发现机制，在图 8-3 中，RouterA 向 RouterD 之间的路径 MTU 是 1496，如果以 MSS 值等于 1496 进行消息传输，就可以提高 BGP 消息的传输效率，提高 BGP 的性能。

图 8-3 路径 MTU 自动发现组网示例图



前置任务

在配置路径 MTU 自动发现功能之前，需完成以下任务：

- 配置 BGP 的基本功能

数据准备

在配置路径 MTU 自动发现功能之前，需要准备以下数据：

序号	数据
1	（可选）路径 MTU 的老化时间

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **bgp** *as-number*，进入 BGP 视图。

步骤 3 执行命令 **peer** { *group-name* | *ipv4-address* } **path-mtu auto-discovery**，使能路径 MTU 自动发现功能。

缺省情况下，没有使能邻居的路径 MTU 学习功能。

配置该命令后，邻居通过学习传输路径上最大数据报文的字节数，避免了 BGP 消息在传输过程中被再次分组分片。

说明

由于两个 BGP 邻居之间消息发送和应答的传输路径可能不一致，所以建议在两端都执行该命令，这样，两个 BGP 邻居在相互发送消息时都可以按照路径 MTU 发送。

步骤 4 执行命令 **quit**，退回系统视图。

步骤 5 执行命令 **tcp timer pathmtu-age** *age-time* 配置 IPv4 路径 MTU 的老化时间。

缺省情况下，IPv4 路径 MTU 的老化时间是 0 分钟，即不进行老化。

不同的路由，路径 MTU 也可能不同。两台主机之间的路径 MTU 不一定是固定值，它取决于传输消息时所选择的路由。如果相互通信的两台主机之间存在多条路由，并且传输报文选择的路由变化频繁，这时就需要为路径 MTU 配置老化时间。配置路径 MTU 老化时间后，系统会按照老化时间间隔更新路径 MTU，从而适应网络的变化情况，提高传输效率。

----结束

检查配置结果

完成路径 MTU 自动发现功能的所有配置后，可以按以下指导来检查配置结果。

- 使用 **display bgp peer** [*ipv4-address*] **verbose** 命令查看 BGP peer 详细信息中的路径 MTU 自动发现功能是否配置成功。

8.18 配置 BGP 下一跳延时响应

通过配置 BGP 下一跳延时响应，可以减少路由变化时的流量丢失。

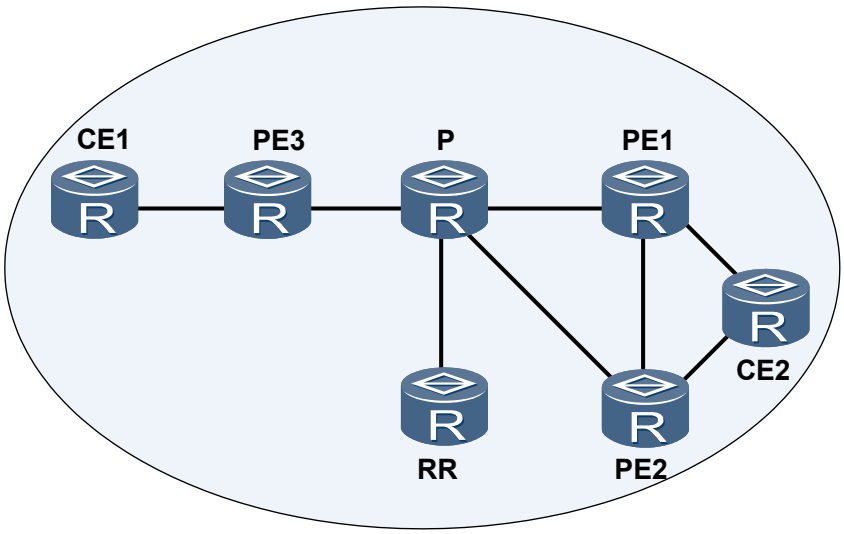
应用环境

BGP 下一跳延时响应可以加快 BGP 收敛速度，减少流量的丢失。

如图 8-4 所示，PE1、PE2 和 PE3 都是 RR 的客户机，CE2 双归属 PE1 和 PE2，PE1 和 PE2 同时向 RR 发布到 CE2 的路由，RR 优选 PE1 发布过来的路由再向 PE3 发布，PE3 上只有一条到 CE2 的路由，并且把路由向 CE1 发布，实现 CE1 和 CE2 的通信。未使能 BGP 下一跳延时响应时，如果 PE1 故障，PE3 首先感知到下一跳不可达，向 CE1 发布撤销到达 CE2 的路由，这时流量中断。之后 BGP 收敛完成，RR 优选 PE2 发布的路由，并且向 PE3 发布路由更新消息，PE3 把路由发布给 CE1，流量恢复正常，在这个过程中，BGP 收敛比较慢，流量损失很大。

如果在 PE3 上使能 BGP 下一跳延时响应，PE3 检测到 PE1 不可达时，暂时不进行选路，也不会向 CE1 发布撤销路由。在 BGP 收敛后，RR 优选 PE2 发布的路由，并且发布给 PE3，PE3 再进行选路，并向 CE1 发布路由更新，此时流量收敛完成。整个过程相比于未使能 BGP 下一跳延时响应时，PE3 上减少了撤销路由的发送和 PE3 本地路由的删除这两个步骤，所以 BGP 收敛速度加快，流量损失减少。

图 8-4 BGP 下一跳延时响应组网图



BGP 下一跳延时响应只适用于下游到达同一目的地有多个链路的场景。如果下游链路唯一，当链路故障时无法进行链路切换，那么此时配置 BGP 下一跳延时响应会造成更大流量损失。

前置任务

在配置 BGP 下一跳延时响应之前，需完成以下任务：

- **配置 BGP 的基本功能**

数据准备

在配置 BGP 下一跳延时响应之前，需要准备以下数据。

序号	数据
1	延迟响应下一跳变化的时间

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **bgp** *as-number*，进入 BGP 视图。

步骤 3 执行命令 **nexthop recursive-lookup delay** [*delay-time*]，配置 BGP 响应下一跳变化的延迟时间。

如果不指定 *delay-time*，则响应下一跳变化的延迟时间缺省为 5 秒。

说明

BGP 收敛依赖于 IGP，缺省值 5 秒只适用于 IGP 收敛较快的场景，如果 IGP 收敛速度较慢，建议配置的 *delay-time* 大于 IGP 收敛时间。

----结束

检查配置结果

完成 BGP 下一跳延时响应的所有配置后，可以执行以下命令来检查配置结果。

- 使用 **display current-configuration configuration bgp | include nexthop recursive-lookup delay** 命令查看响应下一跳变化的延迟时间。

8.19 配置 BFD for BGP

通过配置 BFD for BGP 功能，为 BGP 提供更为快速的故障检测机制，提高网络收敛速度。

应用环境

随着科技的发展，语音、视频及其它点播业务应用广泛，而这些业务对于丢包和延时非常敏感。BGP 协议通过周期性的向对等体发送 Keepalive 报文来实现邻居检测。但这种机制检测到故障所需时间比较长，超过 1 秒钟。当数据达到吉比特速率级时，这么长的检测时间将导致大量数据丢失，无法满足电信级网络高可靠性的需求。

为了解决上述问题，BGP 协议引入了 BFD for BGP 特性。BFD 检测是毫秒级，可以在 50ms 内通报 BGP 对等体间链路的故障，因此能够提高 BGP 路由的收敛速度，保障链路快速切换，减少流量损失。

说明

默认情况下，华为设备之间 IBGP 为多跳会话。华为设备和默认 IBGP 为单跳会话的其他厂商设备对接时，如果链路两端配置 BFD for IGP 会话和 BFD for IBGP 会话，会导致 BFD for IGP 会话和 BFD for IBGP 会话不能同时正常建立。建议用户此时只配置 BFD for IGP 会话或者只配置 BFD for IBGP 会话。

目前，BFD 会话不会感知路由切换。如果绑定的对端 IP 地址改变引起路由切换到其他链路上，除非原链路转发不通，否则，BFD 不会重新协商。

前置任务

在配置 BFD for BGP 特性之前，需完成以下任务：

- **配置 BGP 的基本功能**

数据准备

在配置 BFD for BGP 之前，需准备以下数据。

序号	数据
1	需要配置 BFD 功能的 BGP 对等体地址或对等体组名称
2	BFD 的相关检测参数：最小接收间隔、最大接收间隔、检测时间倍数、等待 BFD 会话恢复的时间
3	需要配置 BFD 功能的 VPN 实例名称

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **bfd**，对本节点使能全局 BFD 功能。

步骤 3 执行命令 **quit**，返回系统视图。

步骤 4 执行命令 **bgp** *as-number*，进入 BGP 视图。

步骤 5 （可选）执行命令 **ipv4-family vpn-instance** *vpn-instance-name*，进入 BGP-VPN 实例 IPv4 地址族视图。

说明

该步骤用来进入 BGP-VPN 实例 IPv4 地址族视图，配置私网 BFD for BGP 功能。如果需要配置公网 BFD for BGP 功能，请跳过该步骤。

步骤 6 执行命令 **peer** { *group-name* | *ipv4-address* } **bfd enable**，配置对等体或对等体组的 BFD 功能，以缺省参数建立 BFD 会话。

当对等体的状态是 Established 时，BFD 会话才能被创建。

如果在对等体组上配置了 BFD 特性，则属于该对等体组且没有使能 **peer bfd block** 的对等体都将创建 BFD 会话。

步骤 7 （可选）执行命令 **peer** { *group-name* | *ipv4-address* } **bfd** { **min-tx-interval** *min-tx-interval* | **min-rx-interval** *min-rx-interval* | **detect-multiplier** *multiplier* | **wtr** *wtr-value* } *，修改 BFD 会话的参数。

说明

BFD 参数生效的优先级是单个对等体优先于对等体组。如果在对等体上配置 BFD 参数，建立的 BFD 会话以对等体上的配置为准。

BFD 报文实际发送时间间隔和检测倍数一般推荐使用缺省值，即不执行该步骤。具体参数如何配置取决于网络状况以及对网络可靠性的要求，对于网络可靠性要求较高链路，可以配置减小 BFD 报文实际发送时间间隔；对于网络可靠性要求较低的链路，可以配置增大 BFD 报文实际发送时间间隔。

说明

本地 BFD 报文实际发送时间间隔 = MAX { 本地配置的发送时间间隔，对端配置的接收时间间隔 }；本地实际接收时间间隔 = MAX { 对端配置的发送时间间隔，本地配置的接收时间间隔}；本实际地检测时间 = 本地实际接收时间间隔 × 对端配置的 BFD 检测倍数。

例如，

- 本地配置的发送时间间隔为 200ms，本地配置的接收时间间隔为 300ms，本地检测倍数为 4
- 对端配置的发送时间间隔为 100ms，对端配置的接收时间间隔为 600ms，对端检测倍数为 5

则：

- 本地实际的发送时间间隔为 MAX { 200ms，600ms } = 600ms，本地实际接收时间间隔为 MAX { 100ms，300ms } = 300ms，本地实际检测时间间隔为 300ms × 5 = 1500ms
- 对端实际的发送时间间隔为 MAX { 100ms，300ms } = 300ms，对端实际接收时间间隔为 MAX { 200ms，600ms } = 600ms，对端实际检测时间间隔为 600ms × 4 = 2400ms

为了抑制链路震荡导致的 BFD 会话和 BGP 会话的频繁震荡，可以选择 **wtr** *wtr-value* 参数配置等待 BFD 会话恢复时间，在 BFD 会话状态变为 DOWN 之后，如果链路状态恢复，BFD 会话状态不会立即变为 UP，而是等待 *wtr-value* 超时之后 BFD 会话状态才变为 UP，那么当链路在 *wtr-value* 时间内再次发生故障时 BFD 也就不会立即将链路故障的消息再次通知给 BGP，BGP 会话就能保持一定的稳定性。

等待 BFD 会话恢复时间 *wtr-value* 的缺省值是 0，也就是默认不启动等待 BFD 会话恢复功能。

步骤 8 （可选）执行命令 **peer** *ipv4-address* **bfd block**，阻止对等体从对等体组中继承 BFD 功能。

当对等体加入了对等体组且这个组使能了 BFD 特性，对等体会继承这个对等体组的 BFD 特性，创建 BFD 会话。如果不希望对等体从对等体组继承 BFD 特性，可以执行该步骤阻止对等体从对等体组中继承 BFD 功能。

说明

peer bfd block 命令和 **peer bfd enable** 命令是两条互斥命令，配置 **peer bfd block** 命令后，会自动删除 BFD 会话。

----结束

检查配置结果

完成 BFD for BGP 的所有配置后，可以执行以下命令来检查配置结果。

- 使用 **display bgp bfd session** { [**vpnv4 vpn-instance** *vpn-instance-name*] **peer** *ipv4-address* | **all** }命令查看 BGP 建立的 BFD 会话信息。

8.20 配置 BGP GR

通过配置 BGP GR 功能，能够避免因为协议重启而导致流量中断。

8.20.1 建立配置任务

在配置 BGP GR 前了解此特性的应用环境、配置此特性的前置任务和数据准备，有助于快速、准确地完成配置任务。

应用环境

当 BGP 协议重启时会导致对等体关系重新建立和转发中断，使能平滑重启 GR（Graceful Restart）功能后可以避免流量中断。

从平滑重启过程中完成的任务的角度进行区分时，设备可以分为如下两类：

- GR Restarter：协议重起的设备。指由管理员触发或故障触发重启的设备，必须是有 GR 能力的设备，即路由协议使能并协商了 GR 能力。
- GR Helper：GR Restarter 的邻居，本身必须具备了 GR 能力，才能协助 GR Restarter 进行 GR。

说明

AR150/200 只能作为 Helper 路由器，不能作为 Restarter 路由器。

前置任务

在配置 BGP GR 之前，需要完成以下任务：

- **配置 BGP 的基本功能**

数据准备

在配置 BGP GR 之前，需准备以下数据。

序号	数据
1	BGP 自治系统号
2	重建 BGP 会话的最大时间
3	等待 End-Of-RIB 消息的时间

8.20.2 使能 BGP 协议的 GR 能力

使能或禁止 GR 特性的操作可能会删除并重建所有的会话与实例。

背景信息

使能 GR 能力后，设备可以与周围同样使能了 GR 能力的邻居建立 GR 会话。通过控制协议的会话协商机制，GR Restarter 和 GR Helper 可以了解彼此的 GR 能力。当 GR Helper 检查到 GR Restarter 发生重启时，将不删除和 GR Restarter 相关的路由和转发表项，而是等待重建 BGP 连接。BGP 连接重新建立后，GR Restarter 和 GR Helper 将在新连接上完成 BGP 路由更新。

请在需要使能 BGP GR 的路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **bgp** *as-number*，进入 BGP 视图。

步骤 3 执行命令 **graceful-restart**，使能 BGP 协议的 GR 能力。

缺省情况下，BGP 协议的 GR 能力被禁止。

----结束

8.20.3 配置 BGP 协议的 GR 会话参数

可以根据需要调整 BGP 会话的 GR 参数，但在通常情况下建议使用缺省值。更改 Restart 时间将重新建立 BGP 对等体关系。

背景信息

GR Time 是 GR Helper 发现 GR Restarter Down 后保持转发信息不删除的时间。当 GR Helper 发现对端的 GR Restarter 处于 Down 状态时，在 GR Time 时间内仍保留从 GR Restarter 得到的拓扑信息或路由，不删除这些信息。

请在需要使能 BGP GR 的路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **bgp** *as-number*，进入 BGP 视图。

步骤 3 执行命令 **graceful-restart timer restart** *time*，配置重建 BGP 会话的最大时间。

Restart 时间是路由器重启的最大时间，即接收侧（Receiving Speaker）发现对端重启到重新建立 BGP 会话的最大等待时间。缺省情况下，Restart 时间为 150 秒。

说明

更改 Restart 时间将重新建立 BGP 对等体关系。

步骤 4 执行命令 **graceful-restart timer wait-for-rib** *time*，配置重启侧（Restarting Speaker）和接收侧（Receiving Speaker）等待 End-of-RIB 消息的时间。

缺省情况下，等待 End-of-RIB 消息的时间为 600 秒。

说明

用户可以根据需要调整 BGP 会话的 GR 参数，但在通常情况下建议使用缺省值。

----结束

8.20.4 检查配置结果

BGP GR 配置成功后，可以查看 BGP GR 的状态。

前提条件

已经完成 BGP GR 的所有配置。

操作步骤

- 使用 **display bgp peer verbose** 命令查看 BGP GR 的状态。

----结束

8.21 配置 BGP 安全性

为提高 BGP 的安全性，可以在建立 TCP 连接时进行认证。

8.21.1 建立配置任务

在配置提高 BGP 网络的安全性前了解此特性的应用环境、配置此特性的前置任务和数据准备，有助于快速、准确地完成配置任务。

应用环境

为提高 BGP 的安全性，可以在 BGP 网络中配置 BGP MD5 认证、Keychain 认证或 GTSM 功能：

- 配置 BGP MD5 验证

  BGP 使用 TCP 作为传输协议，只要 TCP 数据包的源地址、目的地址、源端口、目的端口和 TCP 序号是正确的，BGP 就会认为这个数据包有效，但数据包的大部分

参数对于攻击者来说是不难获得的。为了保证 BGP 协议免受攻击，BGP 邻居之间使用 TCP 的 MD5 认证来降低被攻击的可能性。

为防止 BGP 对等体所设置的 MD5 密码被破解，需要周期性的更新 MD5 认证密码。

- 配置 Keychain 认证

  Keychain 由多个认证密钥组成，每个密钥包含一个 ID 和密码。密钥存在生命期，通过密钥的生命期可以在 Keychain 中滚动选择不同的认证密钥。BGP 会话两端绑定相同规则的 Keychain 后，Keychain 可以滚动选择认证密钥来增强 BGP 防攻击性。

- 配置 BGP GTSM 特性

  GTSM 机制通过 TTL 的检测来达到防止攻击的目的。如果攻击者模拟真实的 BGP 协议报文，对一台路由器不断的发送报文。路由器接口板收到这些报文后，发现是发送给本机的报文，则直接上送控制层面的 BGP 协议处理，而不加辨别其“合法性”。这样导致路由器控制层面因为处理这些“合法”报文，系统异常繁忙，CPU 占用率高。

  配置 GTSM 功能，通过检测 IP 报文头中的 TTL 值是否在一个预先定义好的特定范围内来对路由器进行保护，增强系统的安全性。

  说明

  - AR150/200 支持 BGP GTSM。
  - 因为 GTSM 只支持单播地址，因此需要在路由协议作用范围内的所有路由器上部署 GTSM。

前置任务

在配置 BGP 的安全性之前，需完成以下任务：

- **配置 BGP 的基本功能**

数据准备

在配置 BGP 安全性之前，需要准备以下数据。

序号	数据
1	各路由器的 BGP 对等体的 IP 地址或者对等体组的名称
2	MD5 验证密码
3	Key-Chain 认证的名称

8.21.2 配置 MD5 认证

BGP 的 MD5 认证只是为 TCP 连接设置 MD5 认证密码，由 TCP 完成认证。如果认证失败，则不建立 TCP 连接。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **bgp** *as-number*，进入 BGP 视图。

步骤 3 执行命令 **peer** { *ipv4-address* | *group-name* } **password** { **cipher** *cipher-password* | **simple** *simple-password* }，配置 MD5 认证密码。

在用户设置密码时就用户设置的密码在配置文件中的记录形式可以分为明文和密文两种形式选择：

- 配置 **cipher** *cipher-password* 参数表示使用密文记录密码，即记录经过特殊算法加密后的字符串。
- 配置 **simple** *simple-password* 参数表示使用明文记录密码，即直接记录用户设置的字符串。

说明

当在 BGP 视图下配置时，对 MP-BGP 的 VPNv4 扩展同样有效，因为它们使用同一个 TCP 连接。

由于符号$@$@用于升级时区分新老密码类型，MD5 密码不允许同时以$@$@开始和结束。

----结束

8.21.3 配置 Keychain 认证

BGP 对等体两端必须都配置 Keychain 认证，且配置的 Keychain 必须使用相同的加密算法和密码，才能正常建立 TCP 连接，交互 BGP 消息。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **bgp** *as-number*，进入 BGP 视图。

步骤 3 执行命令 **peer** { *ipv4-address* | *group-name* } **keychain** *keychain-name*，配置 Keychain 认证。

BGP 对等体两端必须同时配置 Keychain 认证，且配置的 Keychain 必须使用相同的加密算法和密码，才能正常建立 TCP 连接，交互 BGP 消息。

配置 BGP Keychain 认证前，必须配置 *keychain-name* 对应的 Keychain，否则 TCP 连接不能正常建立。

说明

- 当在 BGP 视图下配置时，对 MP-BGP 的 VPNv4 扩展同样有效，因为它们使用同一个 TCP 连接。
- BGP MD5 认证与 BGP Keychain 认证互斥。

----结束

8.21.4 配置 BGP GTSM 功能

配置 GTSM 功能，通过检测 IP 报文头中的 TTL 值是否在一个预先定义好的特定范围内来对路由器进行保护。

操作步骤

- 调整 GTSM

  请在对等体两端均进行以下配置。

  1. 执行命令 **system-view**，进入系统视图。

2. 执行命令 **bgp** *as-number*，进入 BGP 视图。
3. 执行命令 **peer** { *group-name* | *ipv4-address* } **valid-ttl-hops** [*hops*]，配置 BGP GTSM 功能。

   被检测报文的 TTL 值有效范围为 [255 - *hops*+1, 255]。例如，对于 EBGP 直连路由，*hops* 的取值为 1，即有效的 TTL 值设为 255。缺省情况下，参数 *hops* 取值为 255，即 TTL 有效值范围为 [1, 255]。

   说明

   - 当在 BGP 视图下配置时，对 MP-BGP 的 VPNv4 扩展同样有效，因为它们使用同一个 TCP 连接。
   - GTSM 和 EBGP-MAX-HOP 功能均会影响到发送出去的 BGP 报文的 TTL 值，存在冲突，只能对同一对等体或对等体组使能两种功能中的一种。

   使能 BGP 的 GTSM 策略后，接口板对所有 BGP 报文的 TTL 值进行检查。根据实际组网的需要，对于不符合 TTL 值范围的报文，GTSM 可以设置为通过或丢弃。配置 GTSM 缺省动作为丢弃时，可以根据网络拓扑选择合适的 TTL 有限值范围，不符合 TTL 值范围的报文会被接口板直接丢弃，这样就避免了网络攻击者模拟的“合法”BGP 报文占用 CPU。

- 设置 GTSM 缺省动作

  请在配置了 GTSM 功能的路由器上进行以下配置。

  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **gtsm default-action** { **drop** | **pass** }，设置未匹配 GTSM 策略的报文的缺省动作。

     缺省情况下，未匹配 GTSM 策略的报文可以通过过滤。

     说明

     如果仅仅配置了缺省动作，但没有配置 GTSM 策略时，GTSM 不起作用。

- 配置丢弃报文的 LOG 信息

  请在配置了 GTSM 功能的路由器上进行以下配置。

  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **gtsm log drop-packet all**，打开指定单板的 LOG 信息的开关，在单板 GTSM 丢弃报文时记录 LOG 信息。

  通过记录丢弃报文的日志信息，可以方便故障的定位。

----结束

8.21.5 检查配置结果

BGP 网络安全性配置成功后，可以查看对等体的认证信息。

前提条件

已经完成 BGP 安全性的所有配置。

操作步骤

- 使用 **display bgp peer** [*ipv4-address*] **verbose** 命令查看 BGP 对等体的 MD5 认证和 Keychain 认证信息。

- 使用 **display bgp peer verbose** 可以查看 BGP 对等体的 GTSM 功能是否开启，以及已配置的最大有效 TTL 跳数。
- 使用 **display gtsm statistics all** 命令查看各单板上的 GTSM 的统计信息，包括报文总数、通过的报文数量、丢弃的报文数量。

----结束

8.22 BGP 维护

BGP 维护包括复位 BGP 连接和清除 BGP 的统计信息。

8.22.1 复位 BGP 连接

可以选择以 GR 的方式复位 BGP。复位 BGP 连接会导致对等体关系中断。

背景信息

注意

复位 BGP 连接（执行 **reset bgp** 命令）会导致路由器之间的 BGP Peer 关系中断。务必仔细确认是否必须执行复位 BGP 连接的操作。

当 BGP 路由策略（路由器不支持 Router Refresh）发生变化后，需要通过复位 BGP 连接使新的配置生效。如果需要复位 BGP 连接，可在用户视图下选择执行以下命令。

操作步骤

- 在确认需要复位所有 BGP 连接后，请在用户视图下执行 **reset bgp all** 命令。
- 在确认需要复位与指定 AS 之间的 BGP 连接后，请在用户视图下执行 **reset bgp** *as-number* 命令。
- 在确认需要复位与指定对等体的 BGP 连接后，请在用户视图下执行 **reset bgp** *ipv4-address* 命令。
- 在确认需要复位所有 EBGP 连接后，请在用户视图下执行 **reset bgp external** 命令。
- 在确认需要复位与指定对等体组的 BGP 连接后，请在用户视图下执行 **reset bgp group** *group-name* 命令。
- 在确认需要复位所有 IBGP 连接后，请在用户视图下执行 **reset bgp internal** 命令。

----结束

8.22.2 清除 BGP 统计信息

清除 BGP 包括清除 BGP Accounting 统计信息、震荡统计信息和衰减信息。

背景信息

注意

清除 BGP 统计信息后，以前的信息将无法恢复，务必仔细确认。

操作步骤

- 在确认需要清除路由的振荡统计信息后，请在用户视图下执行 **reset bgp flap-info** [**regexp** *as-path-regexp* | **as-path-filter** { *as-path-filter-number* | *as-path-filter-name* } | *ipv4-address* [*mask* | *mask-length*]]命令。
- 在确认需要清除路由的衰减信息并释放被抑制的路由后，请在用户视图下执行 **reset bgp dampening** [*ipv4-address* [*mask* | *mask-length*]]命令。
- 在确认需要清除指定对等体的振荡统计信息后，请在用户视图下执行 **reset bgp** *ipv4-address* **flap-info** 命令。

----结束

8.23 配置举例

BGP 配置举例包括组网需求、组网图、配置注意事项、配置思路和配置步骤。

8.23.1 配置 BGP 的基本功能示例

配置 BGP 的基本功能后可以组建 BGP 网络。

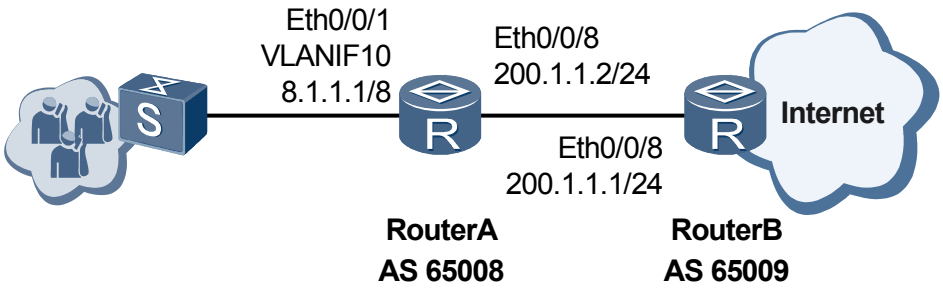
组网需求

如图 8-5 所示，有自治系统 65008 和 65009，其中 RouterA 属于自治系统 65008，RouterB 属于自治系统 65009，RouterA 和 RouterB 之间建立 EBGP 连接。

说明

AR150/200 仅可作为 RouterA。

图 8-5 配置 BGP 基本组网图



配置思路

采用如下的思路配置 BGP 的基本功能：

1. 在 RouterA 和 RouterB 之间配置 EBGP 连接。
2. 在 RouterA 上通过 **network** 命令发布路由，查看 RouterA、RouterB 的路由表信息。

数据准备

为完成此配置例，需准备如下的数据：

- RouterA 的 Router ID 1.1.1.1，所在的 AS 号 65008。
- RouterB 的 Router ID 2.2.2.2，所在的 AS 号 65009。

操作步骤

步骤 1 配置各接口的 IP 地址（略）

步骤 2 配置 EBGP

配置 RouterA。

```
[RouterA] bgp 65008
[RouterA-bgp] router-id 1.1.1.1
[RouterA-bgp] peer 200.1.1.1 as-number 65009
```

配置 RouterB。

```
[RouterB] bgp 65009
[RouterB-bgp] router-id 2.2.2.2
[RouterB-bgp] peer 200.1.1.2 as-number 65008
```

查看 BGP 对等体的连接状态。

```
[RouterB-bgp] quit
[RouterB] display bgp peer

 BGP local router ID : 2.2.2.2
 Local AS number : 65009
 Total number of peers : 1                 Peers in established state : 1

  Peer            V          AS  MsgRcvd  MsgSent  OutQ  Up/Down       State PrefRcv

  200.1.1.2       4       65008       38       38     0 00:35:56 Established       1
```

可以看出，RouterB 到 RouterA 的 BGP 连接已建立。

步骤 3 配置 RouterA 发布路由 8.0.0.0/8

配置 RouterA 发布路由。

```
[RouterA-bgp] ipv4-family unicast
[RouterA-bgp-af-ipv4] network 8.0.0.0 255.0.0.0
```

查看 RouterA 路由表信息。

```
[RouterA-bgp-af-ipv4] quit
[RouterA-bgp] quit
[RouterA] display bgp routing-table

 BGP Local router ID is 1.1.1.1
 Status codes: * - valid, > - best, d - damped,
               h - history,  i - internal, s - suppressed, S - Stale
               Origin : i - IGP, e - EGP, ? - incomplete
```

```
Total Number of Routes: 1
     Network            NextHop        MED        LocPrf    PrefVal Path/Ogn

*>   8.0.0.0            0.0.0.0         0                    0      i
```

查看 RouterB 的路由表。

```
[RouterB] display bgp routing-table

BGP Local router ID is 2.2.2.2
Status codes: * - valid, > - best, d - damped,
              h - history,  i - internal, s - suppressed, S - Stale
              Origin : i - IGP, e - EGP, ? - incomplete


Total Number of Routes: 1
     Network            NextHop        MED        LocPrf    PrefVal Path/Ogn

*>   8.0.0.0            200.1.1.2       0                    0      65008i
```

----结束

配置文件

- RouterA 的配置文件

```
#
 sysname RouterA
#
vlan batch 10
#
interface Vlanif10
 ip address 8.1.1.1 255.0.0.0
#
interface Ethernet0/0/1
 port link-type trunk
 port trunk allow-pass vlan 10
#
interface Ethernet0/0/8
 ip address 200.1.1.2 255.255.255.0
#
bgp 65008
 router-id 1.1.1.1
 peer 200.1.1.1 as-number 65009
 #
 ipv4-family unicast
  undo synchronization
  network 8.0.0.0
  peer 200.1.1.1 enable
#
return
```

- RouterB 的配置文件

```
#
 sysname RouterB
#
interface Ethernet0/0/8
 ip address 200.1.1.1 255.255.255.0
#
bgp 65009
 router-id 2.2.2.2
 peer 200.1.1.2 as-number 65008
 #
 ipv4-family unicast
  undo synchronization
  peer 200.1.1.2 enable
```

```
#
return
```

8.23.2 配置 BGP 团体示例

利用团体属性，可以灵活控制 BGP 的路由选择。

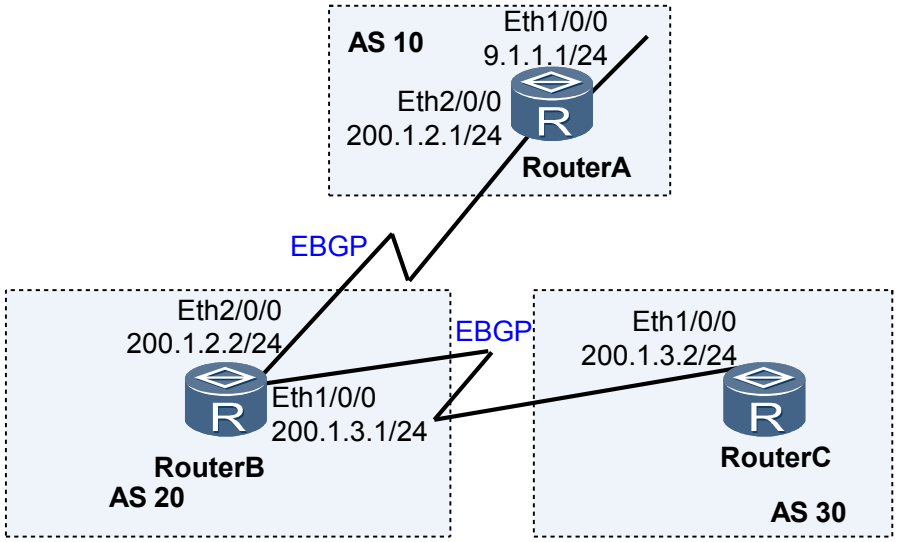
组网需求

企业 A、企业 B 和企业 C 分属于三个 AS，其中企业 B 的网络与其他两个企业通过 EBGP 相连接，实现相互通信。由于企业 A 和企业 C 之间属于竞争关系，企业 A 为了提高安全性希望自己所在 AS 发送给企业 B 的路由只在企业 B 内传播，而不传播给企业 C。为了解决此问题，可在企业 A 往企业 B 发送路由的设备上配置团体属性功能。

如图 8-6 所示，RouterB 分别与 RouterA、RouterC 之间建立 EBGP 连接。如果用户希望 RouterA 引入的路由发布到 AS20 的 RouterB 后，只在 AS20 内传播而不发送给其他 AS，则可以通过在 RouterA 上配置 No_Export 团体属性，使 AS10 发布到 AS20 中的 BGP 路由，不再被 AS20 向其他 AS 发布。

说明
AR150/200 仅可作为 RouterC。

图 8-6 配置 BGP 团体组网图



配置思路

采用如下的思路配置 BGP 团体：

1. RouterA 和 RouterB 之间，RouterB 和 RouterC 之间分别配置 EBGP 连接，使 AS 之间通过 EBGP 连接实现相互通信。
2. 在 RouterA 上配置路由策略，向 RouterB 发布 No_Export 团体属性，使 AS10 发布到 AS20 中的 BGP 路由，不再被 AS20 向其他 AS 发布。

数据准备

为完成此配置例，需准备如下的数据：

- RouterA 的 Router ID 1.1.1.1，所在 AS 号 10。
- RouterB 的 Router ID 2.2.2.2，所在 AS 号 20。
- RouterC 的 Router ID 3.3.3.3，所在 AS 号 30。

操作步骤

步骤 1 配置各接口的 IP 地址（略）

步骤 2 配置 EBGP

配置 RouterA。

```
[RouterA] bgp 10
[RouterA-bgp] router-id 1.1.1.1
[RouterA-bgp] peer 200.1.2.2 as-number 20
[RouterA-bgp] ipv4-family unicast
[RouterA-bgp-af-ipv4] network 9.1.1.0 255.255.255.0
[RouterA-bgp-af-ipv4] quit
[RouterA-bgp] quit
```

配置 RouterB。

```
[RouterB] bgp 20
[RouterB-bgp] router-id 2.2.2.2
[RouterB-bgp] peer 200.1.2.1 as-number 10
[RouterB-bgp] peer 200.1.3.2 as-number 30
[RouterB-bgp] quit
```

配置 RouterC。

```
[RouterC] bgp 30
[RouterC-bgp] router-id 3.3.3.3
[RouterC-bgp] peer 200.1.3.1 as-number 20
[RouterC-bgp] quit
```

在 RouterB 上查看路由 9.1.1.0/24 的详细信息。

```
[RouterB] display bgp routing-table 9.1.1.0

 BGP local router ID : 2.2.2.2
 Local AS number : 20
 Paths:   1 available, 1 best, 1 select
 BGP routing table entry information of 9.1.1.0/24:
 From: 200.1.2.1 (1.1.1.1)
 Route Duration: 00h00m42s
 Direct Out-interface: Ethernet2/0/0
 Original nexthop: 200.1.2.1
 Qos information : 0x0
 AS-path 10, origin igp, MED 0, pref-val 0, valid, external, best, select, active, pre 255
 Advertised to such 2 peers:
    200.1.2.1
    200.1.3.2
```

可以看到 RouterB 把收到的 BGP 路由发布给了位于 AS30 内的 RouterC。

查看 RouterC 的 BGP 路由表。

```
[RouterC] display bgp routing-table

 BGP Local router ID is 3.3.3.3
 Status codes: * - valid, > - best, d - damped,
               h - history,  i - internal, s - suppressed, S - Stale
```

```
                Origin : i - IGP, e - EGP, ? - incomplete

 Total Number of Routes: 1
      Network            NextHop        MED        LocPrf    PrefVal Path/Ogn

 *>   9.1.1.0/24         200.1.3.1                              0       20 10i
```

从路由表可以确认，RouterC 从 RouterB 那里学到了目的地址为 9.1.1.0/24 的路由。

步骤 3 配置 BGP 团体属性

在 RouterA 上配置路由策略，使 RouterA 发布给 RouterB 的 BGP 路由，不再被 RouterB 发布给其他 AS。

```
[RouterA] route-policy comm_policy permit node 10
[RouterA-route-policy] apply community no-export
[RouterA-route-policy] quit
```

应用路由策略。

```
[RouterA] bgp 10
[RouterA-bgp] ipv4-family unicast
[RouterA-bgp-af-ipv4] peer 200.1.2.2 route-policy comm_policy export
[RouterA-bgp-af-ipv4] peer 200.1.2.2 advertise-community
```

在 RouterB 上查看路由 9.1.1.0/24 的详细信息。

```
[RouterB] display bgp routing-table 9.1.1.0

 BGP local router ID : 2.2.2.2
 Local AS number : 20
 Paths:   1 available, 1 best, 1 select
 BGP routing table entry information of 9.1.1.0/24:
 From: 200.1.2.1 (1.1.1.1)
 Route Duration: 00h00m09s
 Direct Out-interface: Ethernet2/0/0
 Original nexthop: 200.1.2.1
 Qos information : 0x0
 Community:no-export
 AS-path 10, origin igp, MED 0, pref-val 0, valid, external, best, select, active, pre 255
 Not advertised to any peer yet
```

通过以上显示信息可以看到 9.1.1.0/24 这条路由携带的团体属性，并且 RouterB 没有把 9.1.1.0/24 这条路由发布给其他区域的对等体。

----结束

配置文件

- RouterA 的配置文件

```
#
 sysname RouterA
#
interface Ethernet1/0/0
 ip address 9.1.1.1 255.255.255.0
#
interface Ethernet2/0/0
 ip address 200.1.2.1 255.255.255.0
#
bgp 10
 router-id 1.1.1.1
 peer 200.1.2.2 as-number 20
 #
 ipv4-family unicast
  undo synchronization
  network 9.1.1.0 255.255.255.0
```

```
  peer 200.1.2.2 enable
  peer 200.1.2.2 route-policy comm_policy export
  peer 200.1.2.2 advertise-community
#
route-policy comm_policy permit node 10
 apply community no-export
#
return
```

- RouterB 的配置文件

```
#
 sysname RouterB
#
interface Ethernet2/0/0
 ip address 200.1.2.2 255.255.255.0
#
interface Ethernet3/0/0
 ip address 200.1.3.1 255.255.255.0
#
bgp 20
 router-id 2.2.2.2
 peer 200.1.2.1 as-number 10
 peer 200.1.3.2 as-number 30
 #
 ipv4-family unicast
  undo synchronization
  peer 200.1.2.1 enable
  peer 200.1.3.2 enable
#
return
```

- RouterC 的配置文件

```
#
 sysname RouterC
#
interface Ethernet1/0/0
 ip address 200.1.3.2 255.255.255.0
#
bgp 30
 router-id 3.3.3.3
 peer 200.1.3.1 as-number 20
 #
 ipv4-family unicast
  undo synchronization
  peer 200.1.3.1 enable
#
return
```

9 BGP4+配置

关于本章

BGP4+通常应用于大型和复杂的 IPv6 网络，用于在 AS 之间传递路由信息。

9.1 BGP4+概述
BGP4+主要用于控制路由的传播和选择最佳路由。

9.2 AR150/200 中支持的 BGP4+特性
系统支持的 BGP4+特性包括：负载分担、路由聚合、路由衰减、团体、路由反射器、联盟、BGP4+ GR 和 BGP4+ NSR。

9.3 配置 BGP4+的基本功能
只有配置了基本功能，才可以组建 BGP4+网络。

9.4 配置 BGP4+的路由属性
BGP4+具有很多路由属性，通过配置这些属性可以改变 BGP4+的选路策略。

9.5 控制路由信息的发布与接收
BGP4+可以针对某个对等体对要发布的路由信息进行过滤或使用路由策略。

9.6 配置 BGP4+对等体间连接参数
通过配置 BGP4+对等体间连接参数，可以对 BGP4+网络的性能进行调整和优化。

9.7 配置 BGP4+ Tracking
不适合 BFD 部署的网络上，通过配置 BGP4+ Tracking，可以实现 IBGP4+路由的快速收敛。

9.8 配置 BGP4+路由衰减
通过配置 BGP4+路由衰减，可以抑制不稳定的 BGP4+路由信息。

9.9 配置 BGP4+负载分担
通过配置 BGP4+负载分担，可以合理利用网路资源，减少网络拥塞。

9.10 配置 BGP4+对等体组
通过配置 BGP4+对等体组，可以简化路由策略的管理，提高路由的发布效率。

9.11 配置 BGP4+路由反射器
通过配置 BGP4+路由反射器，可以解决多个 IBGP 对等体建立全连接的问题。

9.12 配置 BGP4+联盟

大型 BGP4+网络中，配置联盟可简化路由策略的管理，提高路由的发布效率。

9.13 配置 BGP4+安全性
为提高 BGP4+的安全性，可以在建立 TCP 连接时进行认证。

9.14 BGP4+维护
BGP4+维护包括复位 BGP4+连接和清除 BGP4+的统计信息。

9.15 配置举例
BGP4+配置举例包括组网需求、组网图、配置注意事项、配置思路和配置步骤。

9.1 BGP4+概述

BGP4+主要用于控制路由的传播和选择最佳路由。

BGP4+是一种用于自治系统 AS（Autonomous System）之间的动态路由协议，它是对 BGP 的扩展。

传统的 BGP4 只能管理 IPv4 的路由信息，对于使用其它网络层协议（如 IPv6 等）的应用，在跨自治系统传播路由信息时就受到一定限制。

为了提供对多种网络层协议的支持，IETF 对 BGP4 进行了扩展，形成 BGP4+，目前的 BGP4+标准是 RFC2858（Multiprotocol Extensions for BGP-4，BGP-4 多协议扩展）。

为了实现对 IPv6 协议的支持，BGP4 需要将 IPv6 协议的信息反映到 NLRI（Network Layer Reachable Information）属性及 Next_Hop 属性中。

BGP4+中引入的两个 NLRI 属性分别是：

- MP_REACH_NLRI：Multiprotocol Reachable NLRI，多协议可达 NLRI。用于发布可达路由及下一跳信息。
- MP_UNREACH_NLRI：Multiprotocol Unreachable NLRI，多协议不可达 NLRI。用于撤销不可达路由。

BGP4+中的 Next_Hop 属性用 IPv6 地址来表示，可以是 IPv6 全球单播地址或者下一跳的链路本地地址。

BGP4+是利用 BGP 的多协议扩展属性，来达到在 IPv6 网络中应用的目的，BGP 协议原有的消息机制和路由机制并没有改变。

9.2 AR150/200 中支持的 BGP4+特性

系统支持的 BGP4+特性包括：负载分担、路由聚合、路由衰减、团体、路由反射器、联盟、BGP4+ GR 和 BGP4+ NSR。

AR150/200 支持的 BGP4+特性大部分与 BGP 特性相同。请参见“BGP 配置”的 AR150/200 中支持的 BGP 特性。

BGP4+不支持路由自动聚合和 MP-BGP。

说明

BGP4+功能使用 License 授权，缺省情况下，设备的 BGP4+功能受限无法使用。如果需要使用 BGP4+功能，请联系华为办事处申请并购买如下 License，

- AR150&200 数据业务增值包

9.3 配置 BGP4+的基本功能

只有配置了基本功能，才可以组建 BGP4+网络。

9.3.1 建立配置任务

在配置 BGP4+的基本功能前了解此特性的应用环境、配置此特性的前置任务和数据准备，有助于快速、准确地完成配置任务。

应用环境

在 IPv6 网络中配置 BGP4+。

前置任务

在配置 BGP4+基本功能之前，需完成以下任务：

- 使能 IPv6 能力
- 配置接口的链路层协议参数和 IPv6 地址，使接口的链路协议状态为 Up

数据准备

在配置 BGP4+的基本功能之前，需要准备以下数据。

序号	数据
1	本地 AS 编号和 Router ID
2	对等体的 IPv6 地址和 AS 号
3	（可选）建立 BGP4+会话的接口

9.3.2 启动 BGP 进程

启动 BGP4+进程是配置所有 BGP4+特性的首要步骤。启动 BGP4+进程时需指定设备所属的 AS 号。

背景信息

请在需要建立 BGP4+连接的路由器上进行下列配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **bgp** *as-number*，启动 BGP（指定本地 AS 号），进入 BGP 视图。

步骤 3 （可选）执行命令 **router-id** *ipv4-address*，配置 BGP 的 Router ID。

配置或改变 BGP 的 Router ID 会导致路由器之间的对等体关系重置。

窍门

- 为了增加网络的可靠性，建议将 Router ID 手工配置为 Loopback 接口的地址。如果没有配置，则 BGP 会自动选取系统视图下的 Router ID 作为 BGP 协议的 Router ID。系统视图下的 Router ID 选择规则，请参见《Huawei AR150&200 系列企业路由器 命令参考》。
- 如果路由器的所有接口都没有配置 IPv4 地址，则必须配置 Router ID。

----结束

9.3.3 配置 IPv6 对等体

配置 BGP4+对等体且对等体建立成功后，设备之间才可以交换 BGP4+路由信息。

操作步骤

- 配置 IBGP 对等体

  请在需要建立 IBGP 对等体的路由器上进行下列配置。

  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **bgp** *as-number*，进入 BGP 视图。
  3. 执行命令 **peer** { *ipv6-address* | *group-name* } **as-number** *as-number*，配置对等体的地址及所在的 AS。

     所指定对等体所属的 AS 编号应该和本地 AS 号相同。

     当所指定的对等体的 IPv6 地址为 Loopback 接口地址或子接口的 IPv6 地址时，需要再**配置 BGP4+连接所使用的本地接口**，以保证 Peer 的正确建立。
  4. （可选）执行命令 **peer** { *ipv6-address* | *group-name* } **listen-only**，配置对等体（组）仅监听连接请求，而不主动发送连接请求。

     此命令配置后会导致已经建立的 peer 关系中断，本端等待对端发起连接请求后重新建立 peer 关系。通过配置可以避免连接请求冲突的现象发生。

     说明

     该命令只可在对等体间的一端配置，若两端都配置该命令，则该对等体间的连接不能成功建立。
  5. 执行命令 **ipv6-family** [**unicast**]，进入 IPv6 单播地址族视图。
  6. 执行命令 **peer** { *ipv6-address* | *group-name* } **enable**，使能 IPv6 对等体。

     在 BGP 视图下配置 BGP4+对等体之后，还需要在 IPv6 单播地址族视图使能该对等体。

- 配置 EBGP 对等体

  请在需要建立 EBGP 对等体的路由器上进行下列配置。

  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **bgp** *as-number*，进入 BGP 视图。
  3. 执行命令 **peer** { *ipv6-address* | *group-name* } **as-number** *as-number*，配置对等体的 IPv6 地址及所在的 AS。

     所指定对等体所属的 AS 编号应该和本地 AS 号不同。

     当所指定的对等体的 IP 地址为 Loopback 接口地址或子接口的 IP 地址时，需要再**配置 BGP4+连接所使用的本地接口**，以保证 Peer 的正确建立。
  4. 执行命令 **peer** { *ipv6-address* | *group-name* } **ebgp-max-hop** [*hop-count*]，配置 EBGP 连接的最大跳数。

     通常情况下，EBGP 对等体之间必须具有直连的物理链路，如果不满足这一要求，则必须使用 **peer ebgp-max-hop** 命令允许它们之间经过多跳建立 TCP 连接。

     说明

     BGP 使用 Loopback 口建立 EBGP 邻居时，必须配置命令 **peer ebgp-max-hop**（其中 *hop-count* ≥2），否则邻居无法建立。
  5. （可选）执行命令 **peer** { *ipv6-address* | *group-name* } **listen-only**，配置对等体（组）仅监听连接请求，而不主动发送连接请求。

此命令配置后会导致已经建立的对等体关系中断，配置命令的一端等待对端发起连接请求后重新建立对等体关系。通过配置可以避免连接请求冲突的现象发生。

说明

该命令只可在对等体间的一端配置，若两端都配置该命令，则该对等体间的连接不能成功建立。

6. 执行命令 **ipv6-family** [**unicast**]，进入 IPv6 单播地址族视图。
7. 执行命令 **peer** { *ipv6-address* | *group-name* } **enable**，使能 IPv6 对等体。

在 BGP 视图下配置 BGP4+对等体之后，还需要在 IPv6 单播地址族视图使能该对等体。

----结束

9.3.4 （可选）配置 BGP4+连接所使用的本地接口

两台设备通过多链路建立 BGP4+对等体关系时，需要在设备上配置 BGP4+连接所使用的本地接口。

背景信息

请在 BGP4+路由器上进行下列配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **bgp** *as-number*，进入 BGP 视图。

步骤 3 执行命令 **peer** { *ipv6-address* | *group-name* } **connect-interface** *interface-type interface-number* [*ipv6-source-address*]，指定建立 TCP 连接的源接口和源地址。

通常情况下，BGP4+使用与邻居直连的物理接口作为 TCP 连接的会话接口。

为了提高 BGP4+连接的可靠性和稳定性，可将 BGP4+连接所使用的本地接口配置成 Loopback 接口，这样当网络中存在冗余链路时，不会因为其中某个接口或链路的故障而使 BGP4+连接中断。

说明

两台路由器通过多链路建立对等体时，建议使用 **peer connect-interface** 命令指定建立 BGP4+连接的本地接口。

----结束

9.3.5 检查配置结果

BGP4+的基本功能配置成功后，可以查看 BGP4+对等体的信息。

前提条件

已经完成 BGP4+的基本功能的所有配置。

操作步骤

- 使用 **display bgp ipv6 peer** *ipv6-address* { **log-info** | **verbose** }命令查看 BGP4+对等体信息。

----结束

9.4 配置 BGP4+的路由属性

BGP4+具有很多路由属性，通过配置这些属性可以改变 BGP4+的选路策略。

9.4.1 建立配置任务

在配置 BGP4+的路由选择前了解此特性的应用环境、配置此特性的前置任务和数据准备，有助于快速、准确地完成配置任务。

应用环境

BGP4+具有很多路由属性，通过配置这些属性可以改变 BGP4+的选路策略。

- BGP4+协议优先级

  通过配置 BGP4+协议优先级，可以影响 RM（Route Management）对 BGP4+和其他路由协议之间进行路由选路。
- BGP4+路由信息的首选值

  通过配置路由信息首选值，当 BGP4+路由表中存在到相同目的地址的路由时，优先选择首选值高的路由。
- Local_Pref 属性

  通过配置 Local_Pref 属性值，作用同路由信息首选值，但优先级比它低。
- MED 属性

  通过配置 MED 属性，用于 EBGP 对等体判断流量进入 AS 时，选择 MED 最小的路由。
- Next_Hop 属性

  一条路由的下一跳不可达，则忽略该路由。
- 团体属性

  团体属性可以简化路由策略的管理。但它比对等体组的管理范围要大得多，它是对多个 BGP 路由器的路由策略进行控制。
- AS_Path 属性

  配置 AS_Path 属性后，选择 AS 路径较短的路由。

前置任务

在配置 BGP4+的路由属性之前，需完成以下任务：

- **配置 BGP4+的基本功能**

数据准备

在配置 BGP4+路由属性之前，需要准备以下数据。

序号	数据
1	AS 号
2	协议优先级的值
3	Local_Pref 属性的值
4	MED 属性的值
5	如果使用团体，需要所应用的路由策略名称

9.4.2 配置 BGP4+协议的优先级

通过配置 BGP4+协议优先级，可以影响 BGP4+和其他路由协议间的路由选择。

背景信息

请在 BGP4+路由器上进行下列配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **bgp** *as-number*，进入 BGP 视图。

步骤 3 执行命令 **ipv6-family** [**unicast**]，进入 IPv6 单播地址族视图。

步骤 4 执行命令 **preference** { *external internal local* | **route-policy** *route-policy-name* }，设定 BGP4+协议的优先级。

说明

目前不支持通过 **peer route-policy** 命令在对等体上应用 Route-Policy 来设置 BGP 协议的优先级。

----结束

9.4.3 配置 BGP4+路由信息的首选值

通过配置路由信息首选值，当 BGP4+路由表中存在到相同目的地址的路由时，优先选择首选值高的路由。

背景信息

请在 BGP4+路由器上进行下列配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **bgp** *as-number*，进入 BGP 视图。

步骤 3 执行命令 **ipv6-family** [**unicast**]，进入 IPv6 单播地址族视图。

步骤 4 执行命令 **peer** { *group-name* | *ipv6-address* } **preferred-value** *value*，为对等体配置首选值。

缺省情况下，从邻居学来的路由的初始首选值为 0。

----结束

9.4.4 配置本机的缺省 Local_Pref 属性值

Local_Pref 属性用于判断流量离开 AS 时的最佳路由。当 BGP4+设备通过不同的 IBGP4+对等体得到目的地址相同但下一跳不同的多条路由时，将优先选择 Local_Pref 属性值较高的路由。

背景信息

请在 BGP4+路由器上进行下列配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **bgp** *as-number*，进入 BGP 视图。

步骤 3 执行命令 **ipv6-family** [**unicast**]，进入 IPv6 单播地址族视图。

步骤 4 执行命令 **default local-preference** *preference*，配置本机的缺省 Local_Pref 属性值。

----结束

9.4.5 配置 MED 属性

MED 属性相当于 IGP 使用的度量值。通过配置 MED 属性，用于 EBGP 对等体判断流量进入 AS 时的选择 MED 最小的路由。

背景信息

请在 BGP4+路由器上进行下列配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **bgp** *as-number*，进入 BGP 视图。

步骤 3 执行命令 **ipv6-family** [**unicast**]，进入 IPv6 单播地址族视图。

步骤 4 选择执行下列命令，配置 BGP4+ MED 属性。

- 配置缺省 MED 值：执行命令 **default med** *med*
- 比较来自不同 AS 的 MED 值：执行命令 **compare-different-as-med**
- 配置 Deterministic-MED 功能：执行命令 **deterministic-med**

  未配置此命令时，在对从多个不同 AS 接收到的相同前缀的路由进行选路时，选路的结果和路由收来的顺序相关。配置了该命令后，在对从多个不同 AS 收来的相同前缀的路由进行选路时，会按路由 AS_Path 中的最左 AS 进行分组。在相同最左 AS 的组

内进行比较后，再用组中的最优路由和其他组内的最优路由进行比较，从而消除了选路的结果和路由接收顺序的相关性。

- 设置当 MED 值丢失时将其按最大值处理：执行命令 **bestroute med-none-as-maximum**
- 比较本联盟内通告的路由的 MED 值：执行命令 **bestroute med-confederation**

步骤 4 各命令之间无顺序关系。

----结束

9.4.6 配置 Next_Hop 属性

利用 Next_Hop 属性的变化，可以灵活控制 BGP4+的路由选择。

操作步骤

- 向 IBGP 对等体发布路由时，修改下一跳地址

  请在 IBGP 路由器上进行下列配置。

  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **bgp** *as-number*，进入 BGP 视图。
  3. 执行命令 **ipv6-family** [**unicast**]，进入 IPv6 单播地址族视图。
  4. 执行命令 **peer** *ipv6-address* **next-hop-local**，配置发布路由时将自身地址作为下一跳。

     在某些组网环境中，为保证 IBGP 邻居能够找到正确的下一跳，可以配置在向 IBGP 对等体发布路由时，改变下一跳地址为自身地址。

     说明

     如果配置了 BGP4+负载分担，则不论是否配置了 **peer next-hop-local** 命令，本地路由器向 IBGP 对等体组发布路由时都先将下一跳地址改变为自身地址。

- 按策略进行下一跳迭代

  请在 BGP 路由器上进行下列配置。

  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **bgp** *as-number*，进入 BGP 视图。
  3. 执行命令 **ipv6-family** [**unicast**]，进入 IPv6 单播地址族视图。
  4. 执行命令 **nexthop recursive-lookup route-policy** *route-policy-name*，配置路由按策略来迭代下一跳。

     缺省情况下，没有配置下一跳迭代路由策略。

     配置下一跳的迭代路由策略，可以有选择地进行路由迭代，按一定的条件来限制迭代的结果路由。如果路由不能通过策略，则该路由不能被迭代。

----结束

9.4.7 配置 AS_Path 属性

AS_Path 属性用于防止路由环路和控制路由选择。

操作步骤

- 配置 IPv6 地址族视图下的 AS_Path 属性

  请在 BGP4+路由器上进行下列配置。

  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **bgp** *as-number*，进入 BGP 视图。
  3. 执行命令 **ipv6-family** [**unicast**]，进入 IPv6 单播地址族视图。
  4. 选择执行下列命令，配置 AS_Path 属性。
     - 允许本地 AS 编号重复出现：执行命令 **peer** { *ipv6-address* | *group-name* } **allow-as-loop** [*number*]
     - 不将 AS_Path 属性作为选路条件：执行命令 **bestroute as-path-ignore**
     - 配置 AS_Path 属性中仅携带公有 AS 编号：执行命令 **peer** { *ipv6-address* | *group-name* } **public-as-only**

     步骤 4 各命令之间无顺序关系，根据需要可选配置。

- 配置伪 AS 编号

  请在 BGP4+路由器上进行下列配置。

  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **bgp** *as-number*，进入 BGP 视图。
  3. 执行命令 **peer** { *ipv6-address* | *group-name* } **fake-as** *fake-as-number*，配置伪 AS 编号。

     使用此命令可以将本地真实的 AS 编号隐藏，位于其他 AS 内的 EBGP 对等体只能看到这个伪 AS 编号，即其他 AS 内的对等体在指定本端对等体所在的 AS 编号时，应该设置成这个伪 AS 编号。

     说明

     本命令只能应用于 EBGP 对等体。

----结束

9.4.8 配置 BGP4+团体

团体属性可以简化路由策略的管理。但它比对等体组的管理范围要大得多，它是对多个 BGP4+设备的路由策略进行控制。

操作步骤

- 配置向对等体发布团体属性

  请在 BGP4+路由器上进行下列配置。

  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **bgp** *as-number*，进入 BGP 视图。
  3. 执行命令 **ipv6-family** [**unicast**]，进入 IPv6 单播地址族视图。
  4. 选择执行如下命令，将团体属性传给对等体组。
     - 将标准团体属性传给对等体组：执行命令 **peer** { *ipv6-address* | *group-name* } **advertise-community**

- 将扩展团体属性传给对等体组：执行命令 **peer** { *ipv6-address* | *group-name* } **advertise-ext-community**

- 对发布的路由信息应用路由策略

  请在 BGP4+路由器上进行下列配置。

  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **bgp** *as-number*，进入 BGP 视图。
  3. 执行命令 **ipv6-family** [**unicast**]，进入 IPv6 单播地址族视图。
  4. 执行命令 **peer** { *ipv6-address* | *group-name* } **route-policy** *route-policy-name* **export**，配置出方向的路由策略。

  说明

  - 配置 BGP4+团体时，必须使用路由策略来定义具体的团体属性，然后在发布路由信息时应用此路由策略。
  - 关于路由策略的配置，请参考**路由策略配置**。关于团体属性的配置，请参考 **BGP 配置**。

----结束

9.4.9 检查配置结果

控制 BGP4+的路由选择配置成功后，可以查看路由的各属性相关信息。

前提条件

已经完成 BGP4+的路由属性的所有配置。

操作步骤

- 使用 **display bgp ipv6 paths** [*as-regular-expression*]命令查看 AS 路径信息。
- 使用 **display bgp ipv6 routing-table different-origin-as** 命令查看源 AS 不一致的路由。
- 使用 **display bgp ipv6 routing-table regular-expression** *as-regular-expression* 命令查看匹配 AS 正则表达式的路由信息。
- 使用 **display bgp ipv6 routing-table community** [*aa:nn* &<1-29>] [**internet** | **no-advertise** | **no-export** | **no-export-subconfed**] * [**whole-match**]命令查看指定 BGP4+团体的路由信息。
- 使用 **display bgp ipv6 routing-table community-filter** { { *community-filter-name* | *basic-community-filter-number* } [**whole-match**] | *advanced-community-filter-number* }命令查看匹配指定 BGP4+团体属性过滤器的路由。

----结束

9.5 控制路由信息的发布与接收

BGP4+可以针对某个对等体对要发布的路由信息进行过滤或使用路由策略。

9.5.1 建立配置任务

在配置控制路由信息的发布与接收前了解此特性的应用环境、配置此特性的前置任务和数据准备，有助于快速、准确地完成配置任务。

应用环境

- 控制 BGP4+的路由信息的发布和接收，包括对路由信息进行过滤，应用路由策略。
- 软复位 BGP4+连接

  在 AR150/200 的实现中，BGP4+支持 Route-refresh 能力。当策略改变后，系统可以在不中断 BGP4+连接的情况下，自动对 BGP4+路由表进行动态刷新。

  如果网络中存在有不支持 Route-refresh 的路由器，则需要配置 **peer keep-all-routes** 命令，将其所有路由更新保存在本地，并通过执行 **refresh bgp** 命令手工对 BGP4+连接进行软复位。

前置任务

在控制路由信息的发布与接收之前，需完成以下任务：

- **配置 BGP4+的基本功能**

数据准备

在控制 BGP4+路由信息的发布和接收之前，需要准备以下数据。

序号	数据
1	要引入的外部路由名称和进程号
2	路由策略中使用的过滤列表名称
3	路由衰减的各项参数：可达路由的半衰期、不可达路由的半衰期、路由解除抑制状态的阈值、路由进入抑制状态的阈值、惩罚上限值

9.5.2 配置 BGP4+发布本地 IPv6 路由

要发布的本地路由必须存在于本地的 IPv6 路由表中，使用路由策略可以更为灵活的控制所发布的路由。

背景信息

请在 BGP4+路由器上进行下列配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **bgp** *as-number*，进入 BGP 视图。

步骤 3 执行命令 **ipv6-family** [**unicast**]，进入 IPv6 单播地址族视图。

步骤 4 执行命令 **network** *ipv6-address prefix-length* [**route-policy** *route-policy-name*]，发布精确匹配的本地 IPv6 路由。

network 命令用来将 IPv6 路由信息静态注入到 BGP4+路由表中。

指定的目的地址和前缀长度必须与本地 IP 路由表中对应的表项完全一致，路由才能正确发布。如果网络掩码没有指定，此路由将被按照自然网段精确匹配。

要发布的本地路由必须存在于本地的 IPv6 路由表中，使用路由策略可以更为灵活的控制所发布的路由。

----结束

9.5.3 配置 BGP4+路由聚合

配置路由聚合，可以减小对等体路由表中的路由数量。

背景信息

请在 BGP4+路由器上进行下列配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **bgp** *as-number*，进入 BGP 视图。

步骤 3 执行命令 **ipv6-family** [**unicast**]，进入 IPv6 单播地址族视图。

步骤 4 执行命令 **aggregate** *ipv6-address prefix-length* [**as-set** | **attribute-policy** *route-policy-name1* | **detail-suppressed** | **origin-policy** *route-policy-name2* | **suppress-policy** *route-policy-name3*] *，配置手动路由聚合。

手动聚合对 BGP4+本地路由表中已经存在的路由表项有效，例如 BGP 路由表中不存在路由 9:3::1/64，即使配置了命令 **aggregate 9:3::1 64** 对其进行聚合，BGP4+也不会将这条聚合路由发布出去。

使用手动聚合时可以应用多种策略，并可以设置路由的属性。

----结束

9.5.4 配置 BGP4+引入和过滤外部路由

BGP4+对引入的路由信息进行过滤后，只有符合条件的路由信息才会进入 BGP4+本地路由表，发布给 BGP4+对等体。

背景信息

请在 BGP4+路由器上进行下列配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **bgp** *as-number*，进入 BGP 视图。

步骤 3 执行命令 **ipv6-family** [**unicast**]，进入 IPv6 单播地址族视图。

步骤 4 执行命令 **default-route imported**，允许 BGP4+引入缺省路由。

如果没有配置 **default-route imported** 命令，则使用 **import-route** 命令引入其他协议的路由时，不能引入缺省路由。

步骤 5 执行命令 **import-route** *protocol* [*process-id*] [**med** *med* | **route-policy** *route-policy-name*] *，配置 BGP4+引入其他协议的路由。

说明

引入动态路由协议时，需要指定协议号。

步骤 6 执行命令 **filter-policy** *ipv6-prefix-name* **export** [*protocol* [*process-id*]]，对引入的路由信息进行过滤。

BGP4+对引入的路由信息进行过滤后，只有符合条件的路由信息才会进入 BGP4+本地路由表，发布给 BGP4+对等体。指定 *protocol* [*process-id*]参数可以只对特定路由协议的信息进行过滤。如果没有指定此参数，则对所有要发布的本地 BGP 路由信息进行过滤，包括引入的路由和使用 **network** 命令发布的本地路由。

----结束

9.5.5 配置向对等体发送缺省路由

不论本地路由表中是否存在缺省路由，都将向指定对等体发布一条下一跳地址为本地地址的缺省路由。

背景信息

请在 BGP4+路由器上进行下列配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **bgp** *as-number*，进入 BGP 视图。

步骤 3 执行命令 **ipv6-family unicast**，进入 IPv6 单播地址族视图。

步骤 4 执行命令 **peer** { *ipv6-address* | *group-name* } **default-route-advertise** [**route-policy** *route-policy-name*]，向对等体组发送缺省路由。

说明

执行 **peer default-route-advertise** 命令后，不论本地路由表中是否存在缺省路由，都将向指定对等体发布一条以下一跳地址为本地地址的缺省路由。

----结束

9.5.6 配置路由信息的发布策略

配置路由的发布策略后，只有符合条件的路由信息才会进入 BGP4+本地路由表，发布给 BGP4+对等体。

背景信息

请在 BGP4+路由器上进行下列配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **bgp** *as-number*，进入 BGP 视图。

步骤 3 执行命令 **ipv6-family** [**unicast**]，进入 IPv6 单播地址族视图。

步骤 4 选择执行下列命令，基于不同的过滤器配置出方向路由策略。

- 基于 route-policy：执行命令 **peer** { *ipv6-address* | *group-name* } **route-policy** *route-policy-name* **export**
- 基于 AS 路径列表：执行命令 **peer** { *ipv6-address* | *group-name* } **as-path-filter** { *as-path-filter-number* | *as-path-filter-name* } **export**
- 基于前缀列表：执行命令 **peer** { *ipv6-address* | *group-name* } **ipv6-prefix** *ip-prefix-name* **export**

步骤 4 各命令之间无顺序关系。

对等体组的成员可以与所在的组使用不同的出方向路由更新策略，即对外发布路由时，各对等体组成员可以选择自己的策略。

----结束

9.5.7 配置路由信息的接收策略

只有满足接收策略的路由才能被 BGP4+对等体接收，并加到路由表中。

背景信息

请在 BGP4+路由器上进行下列配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **bgp** *as-number*，进入 BGP 视图。

步骤 3 执行命令 **ipv6-family** [**unicast**]，进入 IPv6 单播地址族视图。

步骤 4 选择执行下列命令，基于不同的策略对接收的路由信息进行过滤。

- 对接收的全局路由信息进行过滤：执行命令 **filter-policy ipv6-prefix** *ipv6-prefix-name* **import**
- 对从指定对等体接收的路由信息进行过滤：执行命令 **peer** { *ipv6-address* | *group-name* } **route-policy** *route-policy-name* **import**
- 基于 AS 路径列表过滤：执行命令 **peer** { *ipv6-address* | *group-name* } **as-path-filter** { *as-path-filter-number* | *as-path-filter-name* } **import**
- 基于地址前缀列表过滤：**peer** { *ipv6-address* | *group-name* } **ipv6-prefix** *ipv6-prefix-name* **import**

步骤 4 各命令之间无顺序关系。

对 BGP 接收的路由进行过滤，只有满足某些条件的路由才能被 BGP 接收，并加到路由表中。

对等体组的成员可以与所在的组使用不同的入方向路由策略，即接收路由时，各对等体可以选择自己的策略。

----结束

9.5.8 配置 BGP4+软复位

当策略改变后，系统可以在不中断 BGP4+连接的情况下，自动对 BGP4+路由表进行动态刷新。

操作步骤

- 使能 Route-refresh 能力

  请在 BGP4+路由器上进行下列配置。

  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **bgp** *as-number*，进入 BGP 视图。
  3. 执行命令 **peer** { *ipv6-address* | *group-name* } **capability-advertise** { **route-refresh** | **4-byte-as** }，使能 Route-refresh 能力。

     在所有 BGP4+路由器使能 Route-refresh 能力的情况下，如果 BGP4+的路由策略发生了变化，本地路由器会向对等体发布 Route-refresh 消息，收到此消息的对等体会将其路由信息重新发给本地 BGP4+路由器。这样，在不中断 BGP4+连接的情况下，就可以对 BGP4+路由表进行动态更新，并应用新的策略。

     缺省情况下，使能 Route-refresh 能力。

- 保留对等体的所有路由更新

  请在 BGP4+路由器上进行下列配置。

  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **bgp** *as-number*，进入 BGP 视图。
  3. 执行命令 **ipv6-family** [**unicast**]，进入 IPv6 单播地址族视图。
  4. 执行命令 **peer** { *ipv6-address* | *group-name* } **keep-all-routes**，保留对等体的所有路由更新。

     配置此命令后，不论是否使用了过滤策略，都将保存指定对等体发来的所有路由更新。当本地路由策略改变时，这些信息可以用来重新生成 BGP4+路由。

- 手工软复位 BGP4+连接

  请在 BGP4+路由器上进行下列配置。

  1. 执行命令 **refresh bgp ipv6** { **all** | *ipv6-address* | **group** *group-name* | **external** | **internal** } { **export** | **import** }，软复位 BGP4+连接。

     手工软复位 BGP4+连接时，请在用户视图下执行以上操作。

----结束

9.5.9 检查配置结果

控制路由信息的发布与接收配置成功后，可以查看与指定过滤器匹配的发布路由信息。

前提条件

已经完成控制路由信息的发布与接收的所有配置。

操作步骤

- 使用 **display bgp ipv6 network** 命令查看 BGP4+通过 **network** 命令发布的路由信息。
- 使用 **display bgp ipv6 routing-table as-path-filter** { *as-path-filter-number* | *as-path-filter-name* }命令查看与指定 AS 路径过滤器匹配的路由。

- 使用 **display bgp ipv6 routing-table community-filter** { { *community-filter-name* | *basic-community-filter-number* } [**whole-match**] | *advanced-community-filter-number* }命令查看匹配指定 BGP4+团体属性过滤器的路由。
- 使用 **display bgp ipv6 routing-table peer** *ipv6-address* { **advertised-routes** | **received-routes** } [**statistics**]命令查看 BGP4+对等体发布或者收到的路由信息。

----结束

9.6 配置 BGP4+对等体间连接参数

通过配置 BGP4+对等体间连接参数，可以对 BGP4+网络的性能进行调整和优化。

9.6.1 建立配置任务

在配置 BGP4+对等体间连接参数前了解此特性的应用环境、配置此特性的前置任务和数据准备，有助于快速、准确地完成配置任务。

应用环境

当对等体间建立了 BGP4+连接后，它们定时向对端发送 Keepalive 消息，以防止路由器认为 BGP4+连接已中断。若路由器在设定的连接保持时间（Hold time）内未收到对端的 Keepalive 消息或任何其它类型的报文，则认为此 BGP4+连接已中断，从而退出此 BGP4+连接。

路由器在与对等体建立 BGP4+连接时，将比较双方保持时间，以数值较小者做为协商后的保持时间。如果协商结果为 0，则不发送 Keepalive 消息，且不检测 Hold time 是否超时。

定时器取值的改变会造成短暂的 BGP 连接中断，这是因为对等体双方要重新进行协商。

前置任务

在配置 BGP4+对等体间连接参数之前，需完成以下任务：

- **配置 BGP4+的基本功能**

数据准备

在配置 BGP4+对等体间连接参数之前，需要准备以下数据。

序号	数据
1	BGP4+定时器的值
2	发送更新报文的时间间隔
3	连接重传时间间隔

9.6.2 配置对等体的定时器

合理的定时器可以增强网络性能。需要注意，改变 BGP4+定时器的值会导致对等体关系中断。

背景信息

注意

改变定时器的值（执行 **peer timer** 命令）会导致路由器之间的对等体关系中断。务必仔细确认是否必须改变定时器的值。

请在 BGP4+路由器上进行下列配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **bgp** *as-number*，进入 BGP 视图。

步骤 3 执行命令 **peer** { *ipv6-address* | *group-name* } **timer keepalive** *keepalive-time* **hold** *hold-time*，配置对等体/组的 keepalive 发送间隔和保持时间。

在实际配置时，*hold-time* 的值至少应为 *keepalive-time* 的 3 倍。缺省情况下，存活时间为 60 秒，保持时间为 180 秒。

说明

建议配置的保持时间大于 20 秒。如果保持时间小于 20 秒，可能会造成邻居会话的中断。

以下两种定时器取值配置需要尽量避免：

- *keepalive-time* 值和 *hold-time* 值同时取 0，这种配置将导致 BGP 定时器无效，即 BGP 不会根据定时器检测链路故障。
- *hold-time* 值远大于 *keepalive-time* 值，如 **timer keepalive 1 hold 65535**，过长的保持时间不能保证及时检测到链路的故障。

----结束

9.6.3 配置更新报文的发送时间间隔

路由变化时，路由器会发送 Update 报文通知对等体。但如果同一路由频繁变化时，为避免每次变化路由器都要发送 Update 报文给对等体，可以配置发送该同一路由的 Update 报文的时间间隔。

背景信息

请在 BGP4+路由器上进行下列配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **bgp** *as-number*，进入 BGP 视图。

步骤 3 执行命令 **ipv6-family** [**unicast**]，进入 IPv6 单播地址族视图。

步骤 4 执行命令 **peer** *ipv6-address* **route-update-interval** *interval*，配置更新报文的发送时间间隔。

缺省情况下，IBGP 对等体的路由更新时间间隔为 15 秒，EBGP 对等体的路由更新时间间隔为 30 秒。

----结束

9.6.4 配置 BGP4+连接重传时间间隔

通过改变 BGP4+连接重传时间间隔值的大小可以加速或者减缓 BGP4+邻居的建立，从而适应网络的变化情况。

背景信息

BGP 发起 TCP 连接后，如果成功建立起 TCP 连接，则关闭连接重传定时器。如果 TCP 连接建立不成功，则会在连接重传定时器超时后再次重新尝试建立连接。

- 如果想加快连接失败后重新建立的速度，这时可以把连接重传时间间隔值设置的小一些，减少等待下次连接建立的时间。
- 如果邻居的状态反复震荡，这时可以把连接重传时间间隔值设置的大一些，以减小由于邻居震荡引起的路由振荡，便于路由快速收敛。

请在 BGP4+路由器上进行下列配置。

操作步骤

- 配置全局连接重传时间间隔

  请在 BGP4+路由器上进行下列配置。

  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **bgp** *as-number*，进入 BGP 视图。
  3. 执行命令 **timer connect-retry** *connect-retry-time*，配置 BGP4+全局连接重传时间间隔。

     缺省情况下，连接重传时间间隔是 32 秒。

- 配置对等体或对等体组的连接重传时间间隔

  请在 BGP4+路由器上进行下列配置。

  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **bgp** *as-number*，进入 BGP 视图。
  3. 执行命令 **peer** { *group-name* | *ipv6-address* } **timer connect-retry** *connect-retry-time*，配置对等体或对等体组的连接重传时间间隔。

     缺省情况下，连接重传时间间隔是 32 秒。

     对等体连接重传时间间隔的优先级高于全局连接重传时间间隔的优先级。

----结束

9.6.5 检查配置结果

BGP4+对等体间连接参数配置成功后，可以查看 BGP4+对等体和对等体组信息。

前提条件

已经完成 BGP4+对等体间连接参数的所有配置。

操作步骤

- 使用 **display bgp ipv6 peer** *ipv6-address* { **log-info** | **verbose** }命令查看 BGP4+对等体信息。

----结束

9.7 配置 BGP4+ Tracking

不适合 BFD 部署的网络上，通过配置 BGP4+ Tracking，可以实现 IBGP4+路由的快速收敛。

9.7.1 建立配置任务

在配置 BGP4+ Tracking 前了解此特性的应用环境、配置此特性的前置任务和数据准备，有助于快速、准确地完成配置任务。

应用环境

由于 BFD 部署复杂，且扩展性较差，因此在不适合 BFD 部署的网络上，通过配置 BGP4+ Tracking 可实现 BGP 路由的快速收敛。

BGP4+ Tracking 容易部署，只需要在本端配置即可实现，不需要关注对端对等体。但是 BGP4+ Tracking 的收敛速度比 BFD 慢，不适于对收敛速度要求较高的语音业务。

前置任务

在配置 BGP4+ Tracking 之前，需要完成以下任务：

- **配置 BGP4+的基本功能**

数据准备

在配置 BGP4+ Tracking 之前，需要准备以下数据。

序号	数据
1	（可选）断开连接的延迟时间

9.7.2 使能 BGP4+ Tracking

BGP4+ Tracking 能加快网络的收敛速度，且部署简单，还可以调整从发现邻居不可达到中断连接的时间间隔。

背景信息

请在 BGP4+路由器上进行如下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **bgp** *as-number*，进入 BGP 视图。

步骤 3 执行命令 **peer** { *group-name* | *ipv6-address* } **tracking** [**delay** *delay-time*]，指定对等体的 BGP4+ Tracking 功能。

缺省情况下，不使能 BGP4+ Tracking。

根据实际组网，配置合适的 *delay-time* 可以保证网络的稳定性。

- *delay-time* 为 0 时，BGP 发现邻居不可达后立即断开连接。
- 网络中的闪断会导致 IGP 路由震荡，如果 IBGP 邻居配置 *delay-time* 为 0，则会导致邻居关系震荡。因此根据实际组网，IBGP 邻居需配置大于 IGP 路由收敛时间的 *delay-time*。
- BGP 邻居 GR 协商成功的情况下，BGP 邻居主备倒换，需配置大于 GR 收敛时间的 *delay-time*。如果 *delay-time* 小于 GR 收敛时间，BGP 邻居会中断连接，导致 GR 失效。

----结束

9.7.3 检查配置结果

BGP4+ Tracking 配置成功后，可以查看 BGP4+对等体和对等体组的详细信息。

前提条件

已经完成 BGP4+ Tracking 的所有配置。

检查配置结果

完成上述配置后，请执行下面的命令检查配置结果。

- 使用 **display bgp ipv6 peer** [[*ipv6-address*] **verbose**]命令查看 BGP4+对等体信息。
- 使用 **display bgp ipv6 group** [*group-name*]命令查看 BGP4+对等体组信息。

9.8 配置 BGP4+路由衰减

通过配置 BGP4+路由衰减，可以抑制不稳定的 BGP4+路由信息。

9.8.1 建立配置任务

在配置 BGP4+路由衰减前了解此特性的应用环境、配置此特性的前置任务和数据准备，有助于快速、准确地完成配置任务。

应用环境

通过配置 BGP4+衰减，可以抑制不稳定的路由信息，不将这类路由加入到 BGP4+路由表中，也不将这类路由向其他 BGP4+对等体发布。

前置任务

在配置 BGP4+路由衰减之前，需完成以下任务：

- **配置 BGP4+的基本功能**

数据准备

在配置 BGP4+路由衰减之前，需要准备以下数据。

序号	数据
1	衰减的各项参数：可达路由的半衰期、不可达路由的半衰期、路由解除抑制状态的阈值、路由进入抑制状态的阈值、惩罚上限值

9.8.2 使能 BGP4+路由衰减

BGP4+路由衰减可以增加网络的稳定性。配置路由衰减时可以灵活使用路由策略。

背景信息

请在 BGP4+路由器上进行下列配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **bgp** *as-number*，进入 BGP 视图。

步骤 3 执行命令 **ipv6-family** [**unicast**]，进入 IPv6 单播地址族视图。

步骤 4 执行命令 **dampening** [*half-life-reach reuse suppress ceiling* | **route-policy** *route-policy-name*] *，配置 BGP 路由衰减参数。

----结束

9.8.3 检查配置结果

BGP4+路由衰减配置成功后，可以查看 BGP4+衰减路由、配置参数及相关振荡信息。

前提条件

已经完成 BGP4+路由衰减的所有配置。

操作步骤

- 使用 **display bgp ipv6 routing-table dampened** 命令查看 BGP4+衰减的路由。

- 使用 **display bgp ipv6 routing-table dampening parameter** 命令查看 BGP4+衰减的配置参数。
- 使用 **display bgp ipv6 routing-table flap-info** [**regular-expression** *as-regular-expression* | **as-path-filter** { *as-path-filter-number* | *as-path-filter-name* } | *network-address* [*prefix-length* [**longer-match**]]]命令查看 BGP4+路由振荡统计信息。

----结束

9.9 配置 BGP4+负载分担

通过配置 BGP4+负载分担，可以合理利用网路资源，减少网络拥塞。

应用环境

在大型网路中，到达同一目的地通常会存在多条有效路由，但是 BGP4+只将最优路由发布给对等体，这一特点往往会造成很多流量负载不均衡的情况。

有两种方法解决流量负载不均衡的问题，

- 通过 BGP 强大的策略控制流量的负载均衡。例如通过路由策略修改 BGP 路由的本地优先级（Local_Pref）、AS 路径（AS_Path）、Origin 和 MED（Multi Exit Discriminator）等属性来引导网络流量走不同的路径，实现负载均衡。修改 BGP 路由的属性的配置请参考**配置 BGP4+的路由属性**。
- 通过多路径选路实现负载分担，达到负载均衡的目的。这种负载分担的特点是需要存在等价路由，通过配置等价路由负载分担的路由条数，可以实现多路径负载分担。

  说明

  只有路由属性中，**AR150/200 中支持的 BGP 特性**中“BGP 选择路由的策略”所描述的前 8 个属性完全相同，且 AS-Path 属性也相同时，才能成为 BGP4+等价路由，实现 BGP4+的负载分担。

前置任务

在配置 BGP4+负载分担之前，需完成以下任务：

- **配置 BGP4+的基本功能**

数据准备

在配置 BGP4+负载分担之前，需要准备以下数据。

序号	数据
1	配置 BGP4+负载分担的路由条数

操作步骤

- 配置 BGP4+路由负载分担的路由条数

  请在运行 BGP4+协议的路由器上进行下列配置。

1. 执行命令 **system-view**，进入系统视图。
2. 执行命令 **bgp** *as-number*，进入 BGP 视图。
3. 执行命令 **ipv6-family** [**unicast**]，进入 IPv6 单播地址族视图。
4. 执行命令 **maximum load-balancing** [**ebgp** | **ibgp**] *number*，配置 BGP4+负载分担的路由条数。

   缺省情况下，BGP4+负载分担的路由条数为 1，也就是不进行负载分担。

   - 选择 **ebgp** 参数，仅 EBGP 路由参与负载分担。
   - 选择 **ibgp** 参数，仅 IBGP 路由参与负载分担。
   - **ebgp** 和 **ibgp** 参数都不选择，EBGP 和 IBGP 路由都参与负载分担，且参与负载分担的路由条数相同。

   说明

   如果配置了 **maximum load-balancing** *number* 命令，那么再配置 **maximum load-balancing ebgp** *number* 或 **maximum load-balancing ibgp** *number* 命令都不会生效；如果配置了 **maximum load-balancing ebgp** *number* 或 **maximum load-balancing ibgp** *number* 命令，那么再配置 **maximum load-balancing** *number* 命令也不会生效。

   在公网中到达同一目的地的路由形成负载分担时，系统会首先判断最优路由的类型。若最优路由为 IBGP 路由则只是 IBGP 路由参与负载分担，若最优路由为 EBGP 路由则只是 EBGP 路由参与负载分担，即公网中到达同一目的地的 IBGP 和 EBGP 路由不能形成负载分担。
5. （可选）执行命令 **load-balancing as-path-ignore**，配置路由在形成负载分担时不比较路由的 AS-Path 属性。

   缺省情况下，路由在形成负载分担时比较路由的 AS-Path 属性。

   说明

   - 如果到达目的地址存在多条路由，但是这些路由分别经过了不同的 AS，缺省情况下，这些路由不能形成负载分担。如果用户需要这些路由参与负载分担，就可以执行 **load-balancing as-path-ignore** 命令。配置 **load-balancing as-path-ignore** 命令后会改变路由参与负载分担的条件，路由形成负载分担时不再比较 AS-Path 属性，配置时需要慎重考虑。
   - **load-balancing as-path-ignore** 命令和 **bestroute as-path-ignore** 命令互斥，不能同时使能。

----结束

检查配置结果

完成 BGP4+负载分担的所有配置后，可以按以下指导来检查配置结果。

- 使用 **display bgp ipv6 routing-table** [*ipv6-address prefix-length*]命令查看 BGP4+路由表中的信息。
- 使用 **display ipv6 routing-table** [**verbose**]命令查看 IPv6 路由表信息。

9.10 配置 BGP4+对等体组

通过配置 BGP4+对等体组，可以简化路由策略的管理，提高路由的发布效率。

9.10.1 建立配置任务

在配置 BGP4+对等体组前了解此特性的应用环境、配置此特性的前置任务和数据准备，有助于快速、准确地完成配置任务。

应用环境

在大型 BGP4+网路中，对等体的数目众多，配置和维护极为不便。使用对等体组可以简化管理的难度，还可以提高路由发布效率。根据对等体所在的 AS 是否相同，对等体组可分为 IBGP 对等体组和 EBGP 对等体组。对于 EBGP 对等体组，根据所包括的对等体是否属于同一个外部 AS，又可分为纯 EBGP 对等体组和混合 EBGP 对等体组。

前置任务

在配置 BGP4+对等体组之前，需完成以下任务：

- **配置 BGP4+的基本功能**

数据准备

在配置 BGP4+对等体组之前，需要准备以下数据。

序号	数据
1	对等体组的类型、名称、所包括的对等体

9.10.2 创建 IBGP 对等体组

BGP4+有多个 IBGP 对等体时，创建 IBGP 对等体组可以简化路由策略的管理。创建 IBGP 对等体组不需要指定自治系统号。

背景信息

请在 BGP4+路由器上进行下列配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **bgp** *as-number*，进入 BGP 视图。

步骤 3 执行命令 **group** *group-name* [**internal**]，创建对等体组。

步骤 4 执行命令 **ipv6-family** [**unicast**]，进入 IPv6 单播地址族视图。

步骤 5 执行命令 **peer** *group-name* **enable**，使能对等体组。

步骤 6 执行命令 **peer** *ipv6-address* **group** *group-name*，向对等体组中加入 IPv6 对等体。

说明

在将 IBGP 对等体加入对等体组之后，系统会自动在 BGP 视图下创建该 IPv6 对等体，并在 IPv6 地址族视图下使能该 IPv6 对等体。

----结束

9.10.3 创建纯 EBGP 对等体组

BGP4+有属于同一 AS 的多个 EBGP 对等体时，创建 EBGP 对等体可以简化路由策略的管理。一个纯 EBGP 对等体组的所有对等体的 AS 号必须相同。

背景信息

请在 BGP4+路由器上进行下列配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **bgp** *as-number*，进入 BGP 视图。

步骤 3 执行命令 **group** *group-name* **external**，创建对等体组。

步骤 4 执行命令 **peer** *group-name* **as-number** *as-number*，设置对等体组的 AS 号。

步骤 5 执行命令 **ipv6-family** [**unicast**]，进入 IPv6 单播地址族视图。

步骤 6 执行命令 **peer** *group-name* **enable**，使能对等体组。

步骤 7 执行命令 **peer** *ipv6-address* **group** *group-name*，加入 IPv6 对等体。

在将 EBGP 对等体加入对等体组之后，系统会自动在 BGP 视图下创建该 EBGP 对等体，并在 IPv6 地址族视图下使能该 EBGP 对等体。

在创建纯 EBGP 对等体时，需要指定对等体组的自治系统号。

如果对等体组中已经加入了对等体，那么不能够为该对等体组指定自治系统号。

----结束

9.10.4 创建混合 EBGP 对等体组

BGP4+有属于不同 AS 的多个 EBGP 对等体时，创建混合 EBGP 对等体组，可以简化路由策略的管理。创建混合 EBGP 对等体组时，需要单独指定各对等体的自治系统号。

背景信息

请在 BGP4+路由器上进行下列配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **bgp** *as-number*，进入 BGP 视图。

步骤 3 执行命令 **group** *group-name* **external**，创建对等体组。

步骤 4 执行命令 **peer** *ipv6-address* **as-number** *as-number*，设置 IPv6 对等体的 AS 号。

步骤 5 执行命令 **ipv6-family** [**unicast**]，进入 IPv6 单播地址族视图。

步骤 6 执行命令 **peer** *group-name* **enable**，使能对等体组。

步骤 7 执行命令 **peer** *ipv6-address* **group** *group-name*，加入已创建的 IPv6 对等体。

在将 EBGP 对等体加入对等体组之后，系统自动在 IPv6 地址族视图下使能该 EBGP 对等体。

在创建混合 EBGP 对等体组时，需要单独创建对等体，并可设置不同的自治系统号，但不能设置对等体组的自治系统号。

----结束

9.10.5 检查配置结果

BGP4+对等体组配置成功后，可以查看 BGP4+对等体的详细信息和对等体组信息。

前提条件

已经完成 BGP4+对等体组的所有配置。

操作步骤

- 使用 **display bgp ipv6 group** [*group-name*]命令查看 IPv6 对等体组信息。

----结束

9.11 配置 BGP4+路由反射器

通过配置 BGP4+路由反射器，可以解决多个 IBGP 对等体建立全连接的问题。

9.11.1 建立配置任务

在配置 BGP4+路由反射器前了解此特性的应用环境、配置此特性的前置任务和数据准备，有助于快速、准确地完成配置任务。

应用环境

在 AS 内部，为保证 IBGP 对等体之间的连通性，需要在 IBGP 对等体之间建立全连接关系。当 IBGP 对等体数目很多时，建立全连接网的开销很大。使用路由反射器或者联盟，可以解决这个问题。

前置任务

在配置 BGP4+路由反射器之前，需完成以下任务：

- **配置 BGP4+的基本功能**

数据准备

在配置 BGP4+路由反射器之前，需要准备以下数据。

序号	数据
1	确定各路由器的角色（反射器、客户机、非客户机）

9.11.2 配置路由反射器及指定客户机

需要在特定的地址族下配置路由反射器和客户机。

背景信息

请在 BGP4+路由器上进行下列配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **bgp** *as-number*，进入 BGP 视图。

步骤 3 执行命令 **ipv6-family** [**unicast**]，进入 IPv6 单播地址族视图。

步骤 4 执行命令 **peer** { *ipv6-address* | *group-name* } **reflect-client**，配置路由反射器及其客户。

配置此命令的路由器作为路由反射器，并同时指定哪些对等体作为其客户机。

----结束

9.11.3 （可选）禁止客户之间的路由反射

当路由反射器的客户机已经是全连接时，禁止客户机之间的路由反射，可以减少开销。

背景信息

请在 BGP4+路由器上进行下列配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **bgp** *as-number*，进入 BGP 视图。

步骤 3 执行命令 **ipv6-family** [**unicast**]，进入 IPv6 单播地址族视图。

步骤 4 执行命令 **undo reflect between-clients**，禁止客户机之间的路由反射。

如果路由反射器的客户机已经是全连接的，可以使用 **undo reflect between-clients** 命令禁止客户间的反射，以便减少开销。

缺省情况下，使能客户机之间的路由反射。

此命令只能在路由反射器上配置。

----结束

9.11.4 （可选）配置路由反射器的集群 ID

一个集群里有多个路由反射器时，给所有位于同一集群内的路由反射器配置相同的集群 ID，可以避免路由环路。

背景信息

请在 BGP4+路由器上进行下列配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **bgp** *as-number*，进入 BGP 视图。

步骤 3 执行命令 **ipv6-family** [**unicast**]，进入 IPv6 单播地址族视图。

步骤 4 执行命令 **reflector cluster-id** *cluster-id*，配置路由反射器的集群 ID。

窍门

当一个集群里有多个路由反射器时，需要使用此命令给所有位于同一个集群内的路由反射器配置相同的 Cluster_ID，以避免路由循环。

----结束

9.11.5 检查配置结果

BGP4+路由反射器配置成功后，可以查看 BGP4+路由信息和对等体组信息。

前提条件

已经完成 BGP4+路由反射器的所有配置。

操作步骤

- 使用 **display bgp ipv6 peer** [**verbose**]命令查看所有 BGP4+对等体信息。
- 使用 **display bgp ipv6 peer** *ipv6-address* { **log-info** | **verbose** }命令查看对等体信息。

----结束

9.12 配置 BGP4+联盟

大型 BGP4+网络中，配置联盟可简化路由策略的管理，提高路由的发布效率。

9.12.1 建立配置任务

在配置 BGP4+联盟前了解此特性的应用环境、配置此特性的前置任务和数据准备，有助于快速、准确地完成配置任务。

应用环境

联盟是处理 AS 内部的 IBGP 网络连接激增的另一种方法，它将一个自治系统划分为若干个子自治系统，每个子自治系统内部的 IBGP 对等体建立全连接关系或者配置反射器，子自治系统之间建立 EBGP 连接关系。

前置任务

在配置 BGP4+联盟之前，需完成以下任务：

- **配置 BGP4+的基本功能**

数据准备

在配置 BGP4+联盟之前，需要准备以下数据。

序号	数据
1	确定联盟 ID 和子 AS 编号

9.12.2 配置 BGP4+联盟属性

配置 BGP4+联盟可解决 AS 内部 IBGP 连接数量激增的问题。

背景信息

请在 BGP4+路由器上进行下列配置。

操作步骤

- BGP4+联盟的基本配置
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **bgp** *as-number*，进入 BGP 视图。
  3. 执行命令 **confederation id** *as-number*，配置联盟 ID。
  4. 执行命令 **confederation peer-as** *as-number* &<1-32>，指定与本地 AS 连接的其他 EBGP 对等体所属的子自治系统号。

     一个联盟最多可包括 32 个子自治系统。配置属于联盟的子自治系统时使用的 *as-number* 联盟内部有效。

     属于同一联盟的所有 EBGP 对等体都必须配置 **confederation id** 和 **confederation peer-as** 命令，且指定相同的联盟 ID。

     说明

     同一联盟内不能同时配置 2 字节 AS 号的 Old Speaker 和 4 字节 AS 号的 New Speaker。因为 AS4_Path 不支持联盟，这种配置可能会引起环路。
- 配置联盟的兼容性
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **bgp** *as-number*，进入 BGP 视图。
  3. 执行命令 **confederation nonstandard**，配置联盟的兼容性。

     如果其他路由器的联盟实现机制不同于 RFC 标准，可以配置此命令，以便和非标准的设备兼容。

----结束

9.12.3 检查配置结果

BGP4+联盟配置成功后，可以查看 BGP4+路由信息和对等体的详细信息。

前提条件

已经完成 BGP4+联盟的所有配置。

操作步骤

- 使用 **display bgp ipv6 peer [verbose]**命令查看所有 BGP4+对等体详细信息。

----结束

9.13 配置 BGP4+安全性

为提高 BGP4+的安全性，可以在建立 TCP 连接时进行认证。

9.13.1 建立配置任务

在配置提高 BGP4+网络的安全性前了解此特性的应用环境、配置此特性的前置任务和数据准备，有助于快速、准确地完成配置任务。

应用环境

- BGP4+验证

  BGP4+使用 TCP 作为传输层协议，为提高 BGP4+的安全性，可以在建立 TCP 连接时进行 MD5 认证。但 BGP4+的 MD5 认证并不能对 BGP4+报文认证，它只是为 TCP 连接设置 MD5 认证密码，由 TCP 完成认证。如果认证失败，则不建立 TCP 连接。

- 配置 BGP4+的 GTSM 特性

  GTSM 机制通过 TTL 的检测来达到防止攻击的目的。如果攻击者模拟真实的 BGP4+协议报文，对一台路由器不断的发送报文。路由器接口板收到这些报文后，发现是发送给本机的报文，则直接上送控制层面的 BGP4+协议处理，而不加辨别其“合法性”。这样导致路由器控制层面因为处理这些“合法”报文，系统异常繁忙，CPU 占用率高。

  配置 GTSM 功能，通过检测 IP 报文头中的 TTL 值是否在一个预先定义好的特定范围内来对路由器进行保护，增强系统的安全性。

说明

- AR150/200 支持 BGP4+ GTSM。
- 因为 GTSM 只支持单播地址，因此需要在路由协议作用范围内的所有路由器上部署 GTSM。

前置任务

在配置 BGP4+的安全性之前，需完成以下任务：

- **配置 BGP4+的基本功能**

数据准备

在配置 BGP4+安全性之前，需要准备以下数据。

序号	数据
1	各路由器的 BGP4+对等体的 IP 地址或者对等体组的名称
2	MD5 验证密码
3	Key-Chain 认证的名称

9.13.2 配置 MD5 验证

BGP4+的 MD5 认证只是为 TCP 连接设置 MD5 认证密码，由 TCP 完成认证。如果认证失败，则不建立 TCP 连接。

背景信息

请在 BGP4+路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **bgp** *as-number*，进入 BGP 视图。

步骤 3 执行命令 **peer** { *ipv6-address* | *group-name* } **password** { **cipher** *cipher-password* | **simple** *simple-password* }，配置 MD5 认证密码。

说明

当在 BGP 视图下配置时，对 MP-BGP 的 VPNv6 扩展同样有效，因为它们使用同一个 TCP 连接。

符号^#^#和$@$@用来识别变长密码，^#^#作为新密码的前缀和后缀，$@$@作为老密码的前缀和后缀，所以不支持以“$@$@”或“^#^#”同时作为明文密码的起始和结束字符。

BGP MD5 认证与 BGP Keychain 认证互斥。

----结束

9.13.3 配置 Keychain 认证

BGP4+对等体两端必须都配置 Keychain 认证，且配置的 Keychain 必须使用相同的加密算法和密码，才能正常建立 TCP 连接，交互 BGP4+消息。

背景信息

请在 BGP 路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **bgp** *as-number*，进入 BGP 视图。

步骤 3 执行命令 **peer** { *ipv6-address* | *group-name* } **keychain** *keychain-name*，配置 Keychain 认证。

BGP 对等体两端必须都配置 Keychain 认证，且配置的 Keychain 必须使用相同的加密算法和密码，才能正常建立 TCP 连接，交互 BGP 消息。

配置 BGP Keychain 认证前，必须配置 *keychain-name* 对应的 Keychain，否则 TCP 连接不能正常建立。

说明

- 当在 BGP 视图下配置时，对 MP-BGP 的 VPNv6 扩展同样有效，因为它们使用同一个 TCP 连接。
- BGP MD5 认证与 BGP Keychain 认证互斥。

----结束

9.13.4 配置 BGP4+ GTSM 功能

配置 GTSM 功能，通过检测 IP 报文头中的 TTL 值是否在一个预先定义好的特定范围内来对路由器进行保护。

操作步骤

- 配置 BGP4+ GTSM 基本功能

  请在对等体两端均进行以下配置。

  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **bgp** *as-number*，进入 BGP 视图。
  3. 执行命令 **peer** { *group-name* | *ipv6-address* } **valid-ttl-hops** [*hops*]，配置 BGP4+ GTSM 功能。

     被检测报文的 TTL 值有效范围为 [255 - *hops*+1, 255]。例如，对于 EBGP 直连路由，*hops* 的取值为 1，即有效的 TTL 值设为 255。缺省情况下，参数 *hops* 取值为 255，即 TTL 有效值范围为 [1, 255]。

     说明

     - 当在 BGP 视图下配置时，对 MP-BGP 的 VPNv6 扩展同样有效，因为它们使用同一个 TCP 连接。
     - GTSM 和 EBGP-MAX-HOP 功能均会影响到发送出去的 BGP4+报文的 TTL 值，存在冲突，只能对同一对等体或对等体组使能两种功能中的一种。

     使能 BGP4+的 GTSM 策略后，接口板对所有 BGP4+报文的 TTL 值进行检查。根据实际组网的需要，对于不符合 TTL 值范围的报文，GTSM 可以设置为通过或丢弃。配置 GTSM 缺省动作为丢弃时，可以根据网络拓扑选择合适的 TTL 有限值范围，不符合 TTL 值范围的报文会被接口板直接丢弃，这样就避免了网络攻击者模拟的“合法”BGP 报文占用 CPU。

- 设置 GTSM 缺省动作

  请在配置了 GTSM 功能的路由器上进行以下配置。

  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **gtsm default-action** { **drop** | **pass** }，设置未匹配 GTSM 策略的报文的缺省动作。

     缺省情况下，未匹配 GTSM 策略的报文可以通过过滤。

     说明

     如果仅仅配置了缺省动作，但没有配置 GTSM 策略时，GTSM 不起作用。

- 配置丢弃报文的 LOG 信息

  请在配置了 GTSM 功能的路由器上进行以下配置。

1. 执行命令 **system-view**，进入系统视图。
2. 执行命令 **gtsm log drop-packet all**，打开所有单板的 LOG 信息的开关，在单板 GTSM 丢弃报文时记录 LOG 信息。

   通过记录丢弃报文的日志信息，可以方便故障的定位。

----结束

9.13.5 检查配置结果

BGP4+网络安全性配置成功后，可以查看对等体的认证信息。

前提条件

已经完成 BGP4+安全性的所有配置。

操作步骤

- 使用 **display gtsm statistics all** 命令查看 GTSM 的统计信息。

  执行 **display gtsm statistics** 可以看到各单板上的 GTSM 的统计信息，包括 BGP4+和 OSPF 的报文总数、通过的报文数量、丢弃的报文数量。
- 使用 **display bgp ipv6 peer** *ipv6-address* **verbose** 命令查看 BGP4+对等体的 GTSM 信息。
- 使用 **display bgp group** [*group-name*]命令查看 BGP4+对等体组的 GTSM 信息。

  执行 **display bgp peer verbose** 和 **display bgp group** 可以查看到 BGP 对等体/组中 GTSM 功能是开启，以及已配置的最大有效 TTL 跳数。

----结束

9.14 BGP4+维护

BGP4+维护包括复位 BGP4+连接和清除 BGP4+的统计信息。

9.14.1 复位 BGP4+连接

清除 BGP4+包括清除 BGP4+ Accounting 统计信息、震荡统计信息和衰减信息。

背景信息

注意

复位 BGP4+连接（执行 **reset bgp ipv6** 命令）会导致路由器之间的对等体关系中断。务必仔细确认是否必须执行复位 BGP4+连接的操作。

当 BGP4+配置发生变化后，通过复位 BGP4+连接可以使新的配置生效。如果需要复位 BGP4+连接，可在用户视图下选择执行以下 **reset** 命令。

操作步骤

- 在 BGP4+配置发生变化的情况下，可以在用户视图下，执行 **reset bgp ipv6 all** 命令来复位所有 BGP4+连接，使新的配置生效。
- 在 BGP4+配置发生变化的情况下，可以在用户视图下，执行 **reset bgp ipv6** *as-number* 命令来复位与指定 AS 内对等体的 BGP4+连接，使新的配置生效。
- 在 BGP4+配置发生变化的情况下，可以在用户视图下，执行 **reset bgp ipv6** { *ipv6-address* | **group** *group-name* }命令来复位与指定对等体（组）建立的 BGP4+连接，使新的配置生效。
- 在 BGP4+配置发生变化的情况下，可以在用户视图下，执行 **reset bgp ipv6 external** 命令来复位 External BGP4+连接，使新的配置生效。
- 在 BGP4+配置发生变化的情况下，可以在用户视图下，执行 **reset bgp ipv6 internal** 命令来复位 Internal BGP4+连接，使新的配置生效。

----结束

9.14.2 清除 BGP4+统计信息

只要在用户视图下打开相应模块的调试开关，设备就能够产生调试信息。调试信息显示被调试模块接受或者发送数据报的信息内容。

背景信息

注意

清除 BGP4+的统计信息后，以前的统计信息将无法恢复，务必仔细确认。

操作步骤

- 在确认需要清除路由的衰减信息并释放被抑制的路由，请在用户视图下执行 **reset bgp ipv6 dampening** [*ipv6-address prefix-length*]命令。
- 在确认需要清除路由的振荡统计信息后，请在用户视图下执行 **reset bgp ipv6 flap-info** [*ipv6-address prefix-length* | **regexp** *as-path-regexp* | **as-path-filter** { *as-path-filter-number* | *as-path-filter-name* }]命令。

----结束

9.15 配置举例

BGP4+配置举例包括组网需求、组网图、配置注意事项、配置思路和配置步骤。

9.15.1 配置 BGP4+基本功能示例

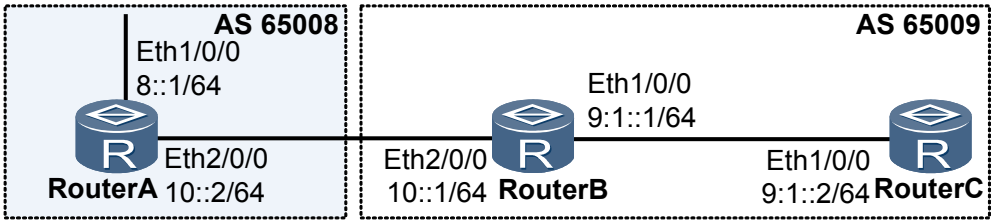
配置 BGP4+的基本功能后可以组建 BGP4+网络。

组网需求

如图 9-1 所示，有自治系统 65008 和 65009，其中 RouterA 属于自治系统 65008，RouterB 和 RouterC 属于自治系统 65009，要求使用 BGP4+协议来交换自治系统之间的路由信息。

说明
AR150/200 仅可作为 RouterC。

图 9-1 配置 BGP4+基本组网图



配置思路

采用如下的思路配置 BGP4+的基本功能：

1. 在 RouterB、RouterC 之间配置 IBGP 连接。
2. 在 RouterA 和 RouterB 之间配置 EBGP 连接。

数据准备

为完成此配置例，需准备如下的数据：

- RouterA 的 Router ID 1.1.1.1，所在的 AS 号 65008。
- RouterB、RouterC 的 Router ID 分别为 2.2.2.2、3.3.3.3，所在的 AS 号 65009。

操作步骤

步骤 1 配置各接口的 IPv6 地址（略）

步骤 2 配置 IBGP

配置 RouterB。

```
[RouterB] ipv6
[RouterB] bgp 65009
[RouterB-bgp] router-id 2.2.2.2
[RouterB-bgp] peer 9:1::2 as-number 65009
[RouterB-bgp] ipv6-family unicast
[RouterB-bgp-af-ipv6] peer 9:1::2 enable
[RouterB-bgp-af-ipv6] network 9:1:: 64
```

配置 RouterC。

```
[RouterC] ipv6
[RouterC] bgp 65009
[RouterC-bgp] router-id 3.3.3.3
[RouterC-bgp] peer 9:1::1 as-number 65009
[RouterC-bgp] ipv6-family unicast
```

```
[RouterC-bgp-af-ipv6] peer 9:1::1 enable
[RouterC-bgp-af-ipv6] network 9:1:: 64
```

步骤 3 配置 EBGP

配置 RouterA。

```
[RouterA] ipv6
[RouterA] bgp 65008
[RouterA-bgp] router-id 1.1.1.1
[RouterA-bgp] peer 10::1 as-number 65009
[RouterA-bgp] ipv6-family unicast
[RouterA-bgp-af-ipv6] peer 10::1 enable
[RouterA-bgp-af-ipv6] network 10:: 64
[RouterA-bgp-af-ipv6] network 8:: 64
```

配置 RouterB。

```
[RouterB] bgp 65009
[RouterB-bgp] peer 10::2 as-number 65008
[RouterB-bgp] ipv6-family unicast
[RouterB-bgp-af-ipv6] peer 10::2 enable
[RouterB-bgp-af-ipv6] network 10:: 64
```

查看 BGP4+对等体的连接状态。

```
[RouterB] display bgp ipv6 peer

 BGP local router ID : 2.2.2.2
 Local AS number : 65009
 Total number of peers : 2                 Peers in established state : 2

  Peer            V          AS  MsgRcvd  MsgSent  OutQ  Up/Down       State PrefRcv

  9:1::2          4       65009       10       14     0 00:07:10 Established       1
  10::2           4       65008        6        6     0 00:02:17 Established       2
```

可以看出，RouterB 到其他路由器的 BGP4+连接均已建立。

显示 RouterA 的路由表。

```
[RouterA] display bgp ipv6 routing-table

 BGP Local router ID is 1.1.1.1
 Status codes: * - valid, > - best, d - damped,
               h - history,  i - internal, s - suppressed, S - Stale
               Origin : i - IGP, e - EGP, ? - incomplete


 Total Number of Routes: 4
 *>  Network  : 8::                                    PrefixLen : 64
     NextHop  : ::                                     LocPrf    :
     MED      : 0                                      PrefVal   : 0
     Label    :
     Path/Ogn : i
 *>  Network  : 9:1::                                  PrefixLen : 64
     NextHop  : 10::1                                  LocPrf    :
     MED      : 0                                      PrefVal   : 0
     Label    :
     Path/Ogn : 65009  i
 *>  Network  : 10::                                   PrefixLen : 64
     NextHop  : ::                                     LocPrf    :
     MED      : 0                                      PrefVal   : 0
     Label    :
     Path/Ogn : i

     NextHop  : 10::1                                  LocPrf    :
     MED      : 0                                      PrefVal   : 0
     Label    :
     Path/Ogn : 65009  i
```

从路由表可以看出，RouterA 学到了 AS65009 中的路由。AS65008 和 AS65009 可以相互交换路由信息。

----结束

配置文件

- RouterA 的配置文件

```
#
 sysname RouterA
#
 ipv6
#
interface Ethernet1/0/0
 ipv6 enable
 ipv6 address 8::1/64
#
interface Ethernet2/0/0
 ipv6 enable
 ipv6 address 10::2/64
#
bgp 65008
 router-id 1.1.1.1
 peer 10::1 as-number 65009
#
 ipv4-family unicast
  undo synchronization
#
 ipv6-family unicast
  undo synchronization
  network 8:: 64
  network 10:: 64
  peer 10::1 enable
#
return
```

- RouterB 的配置文件

```
#
 sysname RouterB
#
 ipv6
#
interface Ethernet1/0/0
 ipv6 enable
 ipv6 address 9:1::1/64
#
interface Ethernet2/0/0
 ipv6 enable
 ipv6 address 10::1/64
#
bgp 65009
 router-id 2.2.2.2
 peer 9:1::2 as-number 65009
 peer 10::2 as-number 65008
#
 ipv4-family unicast
  undo synchronization
#
 ipv6-family unicast
  undo synchronization
  network 9:1:: 64
  network 10:: 64
  peer 9:1::2 enable
  peer 10::2 enable
#
return
```

- RouterC 的配置文件

```
#
 sysname RouterC
#
 ipv6
#
interface Ethernet1/0/0
 ipv6 enable
 ipv6 address 9:1::2/64
#
bgp 65009
 router-id 3.3.3.3
 peer 9:1::1 as-number 65009
#
 ipv4-family unicast
  undo synchronization
#
 ipv6-family unicast
  undo synchronization
  network 9:1:: 64
  peer 9:1::1 enable
#
return
```

10 路由策略配置

关于本章

路由策略是为了改变网络流量所经过的途径而对路由信息采用的方法。

10.1 路由策略概述
使用路由策略可以严格控制网络中路由的发送和接收。

10.2 AR150/200 支持的路由策略特性
配置路由策略时，可选择使用的过滤器：访问控制列表、地址前缀列表、AS 路径过滤器、团体属性过滤器、扩展团体属性过滤器、RD 属性过滤器和 Route-Policy。

10.3 配置地址前缀列表
地址前缀列表在应用时，其匹配对象为路由的目的地址。

10.4 配置 Route-Policy
Route-Policy 的每个节点可以由一组 if-match 子句和 apply 子句组成。

10.5 对接收的路由应用路由过滤器
在路由协议中应用路由策略相关的过滤器，过滤接收的路由。

10.6 对发布的路由应用路由过滤器
在路由协议中应用路由策略相关的过滤器，过滤发布的路由。

10.7 对引入的路由应用路由过滤器
在路由协议中应用路由策略相关的过滤器，过滤引入的路由。

10.8 控制路由策略生效时间
为了保障网络的稳定性，修改路由策略时需要控制路由策略的生效时间。

10.9 路由策略维护
路由策略的维护包括清除地址前缀列表统计数据及路由策略的调试。

10.10 配置举例
路由策略配置举例包括组网需求、组网图、配置注意事项、配置思路和配置步骤。

10.1 路由策略概述

使用路由策略可以严格控制网络中路由的发送和接收。

路由策略

路由策略（Routing Policy）是为了改变网络流量所经过的途径而对路由信息采用的方法。主要通过应用路由属性（包括可达性）来实现。

路由器在发布和接收路由时应用的策略，目的是通过应用路由属性而过滤路由。目前实施路由策略的手段主要是对路由信息进行过滤。例如：

- 只接收或发布一部分满足条件的路由信息。
- 一种路由协议（如 RIP）可能需要引入其它路由协议发现的路由信息，从而丰富自己的路由知识；路由器在引入其它路由协议的路由信息时，可能需要只引入一部分满足条件的路由信息，并对所引入的路由信息的某些属性进行设置，以使其满足本协议的要求。

为实现路由策略：

- 首先定义一组匹配规则和设置规则。满足匹配规则的路由信息将被应用路由策略。
- 然后将它们应用于路由的发布、接收和引入等过程的路由策略中。

路由策略与策略路由的区别

策略路由 PBR（Policy Based Routing）与单纯依照 IP 报文的目的地址查找 FIB 表进行转发不同，是一种依据用户制定的策略而进行路由选择的机制。PBR 支持基于到达报文的源地址和报文长度信息，依据用户制定的策略进行路由选择，可应用于安全、负载分担等目的。

路由策略与策略路由是两种不同的机制，主要区别如表 10-1。

表 10-1 路由策略与策略路由的区别

路由策略	策略路由
基于目的地址按路由表转发	基于策略的转发，失败后再查找路由表转发
基于控制平面，为路由协议和路由表服务	基于转发平面，为转发策略服务
与路由协议结合完成策略	需要手工逐跳配置，以保证报文按策略转发
应用命令 **route-policy**	应用命令 **policy-based-route**

10.2 AR150/200 支持的路由策略特性

配置路由策略时，可选择使用的过滤器：访问控制列表、地址前缀列表、AS 路径过滤器、团体属性过滤器、扩展团体属性过滤器、RD 属性过滤器和 Route-Policy。

过滤器

在 AR150/200 中，提供了访问控制列表、地址前缀列表、AS 路径过滤器、团体属性过滤器、扩展团体属性过滤器、RD 属性过滤器和 Route-Policy 七种过滤器供路由协议引用。下面对各种过滤器逐个进行介绍。

- 访问控制列表 ACL

  访问控制列表包括针对 IPv4 报文的 ACL（Access Control List）。按照 ACL 用途，ACL 可以分为 3 种类型：基于接口的 ACL（Interface-based ACL）、基本 ACL（Basic ACL）和高级 ACL（Advanced ACL）。用户在定义 ACL 时可以指定 IP 地址和子网范围用于匹配路由信息的目的网段地址或下一跳地址。

  ACL 的有关配置请参见《Huawei AR150&200 系列企业路由器 配置指南-IP 业务》中的描述。

- 地址前缀列表（IP-Prefix List）

  地址前缀列表包括 IPv4 地址前缀列表和 IPv6 地址前缀列表，作用比较灵活。

  一个地址前缀列表由前缀列表名标识。每个前缀列表可以包含多个表项，每个表项可以独立指定一个网络前缀形式的匹配范围，并用一个索引号来标识，索引号指明了进行匹配检查的顺序。

  在匹配的过程中，路由器按升序依次检查由 index-number 标识的各个表项。只要有某一表项满足条件，就意味着本次匹配过程结束，而不再进行下一个表项的匹配。具体配置请参见**配置地址前缀列表**。

- AS 路径过滤器（AS-Path-Filter）

  BGP 的路由信息中，包含一个自治系统路径域。AS-Path-Filter 就是针对自治系统路径域指定匹配条件。

  AS 路径过滤器的配置请参见 **BGP 配置**。

- 团体属性过滤器（Community-filter）

  团体属性过滤器仅用于 BGP。BGP 的路由信息中，包含一个团体属性域，用来标识一个团体。团体属性过滤器就是针对团体属性域指定匹配条件。

  团体属性过滤器的配置请参见 **BGP 配置**。

- 扩展团体属性过滤器（Extcommunity-Filter）

  扩展团体属性过滤器仅用于 BGP。BGP 的扩展团体只支持 VPN 的 RT（Route-Target）扩展团体。扩展团体属性过滤器就是针对这种扩展团体属性指定匹配条件。

  扩展团体属性过滤器的配置请参见 **BGP 配置**。

- RD 属性过滤器（Route Distinguisher-Filter）

  VPN 实例通过路由标识符 RD（Route Distinguisher）实现地址空间独立，区分使用相同地址空间的 IPv4 和 IPv6 前缀。RD 属性过滤器针对不同 RD 指定匹配条件。

  RD 属性过滤器的配置请参见《Huawei AR150&200 系列企业路由器 配置指南-VPN》中的描述。

- Route-Policy

  Route-Policy 是一种比较复杂的过滤器，它不仅可以匹配给定路由信息的某些属性，还可以在条件满足时改变路由信息的属性。Route-Policy 可以使用前面几种过滤器定义自己的匹配规则。

  一个 Route-Policy 可以由多个节点（node）构成，不同节点之间是“或”的关系。系统按节点序号依次检查各个节点，如果通过了其中一节点，就意味着通过该策略，不再对其他节点进行匹配。

  每个节点可以由一组 **if-match** 和 **apply** 子句组成。**if-match** 子句定义匹配规则，匹配对象是路由信息的一些属性。同一节点中的不同 **if-match** 子句是“与”的关系，

只有满足节点内所有 **if-match** 子句指定的匹配条件，才能通过该节点的匹配。**apply** 子句指定动作，也就是在通过节点的匹配后，对路由信息的一些属性进行设置。具体配置请参见**配置 Route-Policy**。

路由策略的应用

路由策略主要有两种应用方式：

- 路由协议在引入其它路由协议发现的路由时，通过应用路由过滤器只引入满足条件的路由信息。
- 路由协议在发布或接收本路由协议发现的路由时，通过应用路由过滤器对信息进行过滤，只接收或发布满足给定条件的路由信息。

各协议路由策略应用的具体配置请参考相关的路由协议配置。

说明

路由策略发生变化后，缺省情况下，RM（Routing Management Module）将立即通知协议应用新策略。

10.3 配置地址前缀列表

地址前缀列表在应用时，其匹配对象为路由的目的地址。

10.3.1 建立配置任务

在配置地址前缀列表前了解此特性的应用环境、配置此特性的前置任务和数据准备，有助于快速、准确地完成配置任务。

应用环境

在应用路由策略之前，必须先设置好匹配规则，即过滤器。地址前缀列表的作用类似 ACL，比较灵活。地址前缀列表在应用于路由信息的过滤时，其匹配对象为路由信息的目的地址域。

前置任务

无

数据准备

在配置地址前缀列表之前，需要准备以下数据。

序号	数据
1	地址前缀列表名称
2	匹配的地址范围

10.3.2 配置 IPv4 地址前缀列表

地址前缀列表匹配的对象是 IP 地址前缀，由 IP 地址和掩码长度共同定义。

背景信息

请在需要应用地址前缀列表的路由器上进行如下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **ip ip-prefix** *ip-prefix-name* [**index** *index-number*] { **permit** | **deny** } *ip-address mask-length* [**greater-equal** *greater-equal-value*] [**less-equal** *less-equal-value*]，配置 IPv4 地址前缀列表。

掩码长度范围可以表示为 *mask-length* <= *greater-equal-value* <= *less-equal-value* <= 32。如果只指定了 **greater-equal**，前缀范围为[*greater-equal-value*，32]；如果只指定了 **less-equal**，前缀范围为[*mask-length*，*less-equal-value*]。

IPv4 地址前缀列表由列表名标识，每个前缀列表可以包含多个表项。各表项可以独立指定一个网络前缀形式的匹配范围，并使用索引号标识。比如下面这个名称为 abcd 的 IPv4 地址前缀列表：

```
#
ip ip-prefix abcd index 10 permit 1.0.0.0 8
ip ip-prefix abcd index 20 permit 2.0.0.0 8
```

在匹配过程中，系统按索引号升序依次检查各个表项，只要有一个表项满足条件，就认为通过该过滤列表，不再去匹配其他表项。

AR150/200 默认所有未匹配的路由将被拒绝通过过滤列表。如果所有表项都配置成 **deny** 模式，则任何路由都不能通过该过滤列表。因此，需要在多条 **deny** 模式的表项后定义一条 **permit 0.0.0.0 0 less-equal 32** 表项，允许其它所有 IPv4 路由信息通过。

说明

如果定义了一个以上的前缀列表表项，则至少应该有一个表项的匹配模式为 **permit** 模式。

----结束

10.3.3 配置 IPv6 地址前缀列表

地址前缀列表匹配的对象是 IPv6 地址前缀，由 IPv6 地址和前缀长度共同定义。

背景信息

请在需要应用地址前缀列表的路由器上进行如下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **ip ipv6-prefix** *ipv6-prefix-name* [**index** *index-number*] { **permit** | **deny** } *ipv6-address prefix-length* [**greater-equal** *greater-equal-value*] [**less-equal** *less-equal-value*]，配置 IPv6 地址前缀列表。

前缀长度范围可以表示为 *prefix-length* <= *greater-equal-value* <= *less-equal-value* <= 128。如果只指定了 **greater-equal**，前缀范围为[*greater-equal-value*，128]；如果只指定了 **less-equal**，前缀范围为[*prefix-length*，*less-equal-value*]。

IPv6 地址前缀列表由列表名标识，每个前缀列表可以包含多个表项，每个表项可以独立指定一个网络前缀形式的匹配范围，并用一个索引号标识。比如下面这个名称为 abcd 的 IPv6 地址前缀列表：

```
#
ip ipv6-prefix abcd index 10 permit 1:: 64
ip ipv6-prefix abcd index 20 permit 2:: 64
```

在匹配的过程中，系统按索引号升序依次检查各个表项，只要有某一表项满足条件，就认为通过该过滤列表，不再去匹配其他表项。

AR150/200 默认所有未匹配的路由将被拒绝通过过滤列表。如果所有表项都配置成 **deny** 模式，则任何路由都不能通过该过滤列表。因此，需要在多条 **deny** 模式的表项后定义一条 **permit :: 0 less-equal 128** 的表项，以允许其它所有 IPv6 路由信息通过。

说明

如果定义了一个以上的前缀列表表项，则至少应该有一个表项的匹配模式是 **permit** 模式。

----结束

10.3.4 检查配置结果

地址前缀列表配置成功后，可以查看地址前缀列表的相关信息。

前提条件

已经完成地址前缀列表的所有配置。

操作步骤

- 使用 **display ip ip-prefix** [*ip-prefix-name*]命令查看 IPv4 地址前缀列表信息。
- 使用 **display ip ipv6-prefix** [*ipv6-prefix-name*]命令查看 IPv6 地址前缀列表信息。

----结束

10.4 配置 Route-Policy

Route-Policy 的每个节点可以由一组 if-match 子句和 apply 子句组成。

10.4.1 建立配置任务

在配置 Route-Policy 前了解此特性的应用环境、配置此特性的前置任务和数据准备，有助于快速、准确地完成配置任务。

应用环境

Route-Policy 用来匹配给定的路由信息或者路由信息的某些属性，并在条件满足时改变这些路由信息的属性。

一个 Route-Policy 可由多个节点构成，每个节点又分为：

- **If-match** 子句：定义匹配规则，即路由信息通过当前 Route-Policy 所需满足的条件，匹配对象是路由信息的某些属性。
- **Apply** 子句：指定动作，即执行的一些配置命令，对路由的一些属性进行修改。

Route-Policy 的特性描述请参见《Huawei AR150&200 系列 企业路由器 特性描述-IP 路由》中的描述。

前置任务

在配置 Route-Policy 之前，需完成以下任务：

- 配置地址前缀列表
- 配置路由协议

数据准备

在配置 Route-Policy 之前，需要准备以下数据。

序号	数据
1	Route-Policy 的名称、节点序号
2	匹配条件
3	要修改的路由属性值

10.4.2 创建 Route-Policy

通过应用 Route-Policy，可根据组网需求来设置引入路由的相关属性。

背景信息

请在需要应用 Route-Policy 的路由器上进行如下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **route-policy** *route-policy-name* { **permit** | **deny** } **node** *node*，创建 Route-Policy 的节点，并进入 Route-Policy 视图。

- **permit** 指定节点的匹配模式为允许。当路由项通过该节点的过滤后，将执行该节点的 **apply** 子句，不进入下一个节点的过滤；如果路由项没有通过该节点过滤，将进入下一个节点继续过滤。
- **deny** 指定节点的匹配模式为拒绝，这时 **apply** 子句不会被执行。当路由项满足该节点的所有 **if-match** 子句时，将被拒绝通过该节点，不进入下一个节点；如果路由项不满足该节点的任何 **if-match** 子句，将进入下一个节点继续过滤。

说明

AR150/200 默认所有未匹配的路由将被拒绝通过 Route-Policy。如果 Route-Policy 中定义了一个以上的节点，则各节点中至少应该有一个节点的匹配模式是 **permit**。

当 Route-Policy 用于路由信息过滤时，如果某路由信息没有通过任一节点，则认为该路由信息没有通过该 Route-Policy。如果 Route-Policy 的所有节点都是 **deny** 模式，则没有路由信息能通过该 Route-Policy。

当引用 Route-Policy 进行路由信息过滤时，*node* 的值小的节点先进行过滤。

步骤 3 （可选）执行命令 **description** *text*，配置路由策略的描述信息。

----结束

10.4.3 （可选）配置 If-match 子句

If-match 子句用来定义 Route-Policy 匹配条件，匹配对象是路由信息的一些属性。

背景信息

请在需要应用 Route-Policy 的路由器上进行如下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **route-policy** *route-policy-name* { **permit** | **deny** } **node** *node*，进入 Route-Policy 视图。

步骤 3 选择执行下列命令，配置 Route-Policy 中的 If-match 子句。

- 匹配 ACL：**if-match acl** { *acl-number* | *acl-name* }
- 匹配路由信息的路由权值：**if-match cost** *cost*
- 匹配路由信息的出接口：**if-match interface** *interface-type interface-number*
- 匹配 IPv4 的路由信息（下一跳、源地址或组播组地址）：**if-match ip** { **next-hop** | **route-source** | **group-address** } { **acl** { *acl-number* | *acl-name* } | **ip-prefix** *ip-prefix-name* }
- 匹配地址前缀列表：**if-match ip-prefix** *ip-prefix-name*

  说明

  对于同一个 Route-Policy 节点，命令 **if-match acl** 和命令 **if-match ip-prefix** 不能同时配置，后配置的命令会覆盖先配置的命令。
- 匹配 IPv6 的路由信息：**if-match ipv6** { **address** | **next-hop** | **route-source** } **prefix-list** *ipv6-prefix-name*
- 匹配路由信息的类型
  - 匹配 OSPF 各类型路由信息：**if-match route-type** { **external-type1** | **external-type1or2** | **external-type2** | **internal** | **nssa-external-type1** | **nssa-external-type1or2** | **nssa-external-type2** }
  - 匹配 IS-IS 各 level 路由信息：**if-match route-type** { **is-is-level-1** | **is-is-level-2** }
- 匹配路由信息的标记域：**if-match tag** *tag*

步骤 3 中各命令之间无顺序关系。在一个节点中，可以没有 **if-match** 子句，也可以有多个 **if-match** 子句。

说明

- 对于同一个 Route-Policy 节点，在匹配的过程中，各个 **if-match** 子句间是“与”的关系，即路由信息必须同时满足所有匹配条件，才可以执行 **apply** 子句的动作。但命令 **if-match route-type** 和 **if-match interface** 除外，这两个命令的各自 **if-match** 子句间是“或”的关系，与其它命令的 **if-match** 子句间仍是“与”的关系。
- 如不指定 **if-match** 子句，则所有路由信息都会通过该节点的过滤。

----结束

10.4.4 （可选）配置 Apply 子句

Apply 子句用来为 Route-Policy 指定动作，来对路由信息的属性进行设置。

背景信息

请在需要应用 Route-Policy 的路由器上进行如下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **route-policy** *route-policy-name* { **permit** | **deny** } **node** *node*，进入 Route-Policy 视图。

步骤 3 选择执行下列命令，配置 Route-Policy 中的 apply 子句。

- 设置路由信息的开销值：**apply cost** [+ | -] *cost*
- 设置路由信息的开销类型
  - 设置 IS-IS 的开销类型：**apply cost-type** { **external** | **internal** }
  - 设置 OSPF 的开销类型：**apply cost-type** { **type-1** | **type-2** }

  apply cost-type { **external** | **internal** }命令和 **apply cost-type** { **type-1** | **type-2** }命令互斥，不能同时配置。
- 设置 IPv4 路由信息的下一跳地址：**apply ip-address next-hop** { **peer-address** | *ipv4-address* }
- 设置 IPv6 路由信息的下一跳地址：**apply ipv6 next-hop** { **peer-address** | *ipv6-address* }
- 设置 IS-IS 的路由级别：**apply isis** { **level-1** | **level-1-2** | **level-2** }
- 设置引入路由到 OSPF 的特定区域：**apply ospf** { **backbone** | **stub-area** }
- 设置路由协议的优先级：**apply preference** *preference*

  设置的优先级的值越小，优先级越高。
- 设置路由信息的标记域：**apply tag** *tag*

步骤 3 各命令之间没有顺序关系。

----结束

10.4.5 检查配置结果

Route-Policy 配置成功后，可以查看 Route-Policy 的相关信息。

前提条件

已经完成 Route-Policy 的所有配置。

操作步骤

- 使用 **display route-policy** [*route-policy-name*]命令查看 Route-Policy。

----结束

10.5 对接收的路由应用路由过滤器

在路由协议中应用路由策略相关的过滤器，过滤接收的路由。

10.5.1 建立配置任务

在对接收的路由应用路由过滤器前了解此特性的应用环境、配置此特性的前置任务和数据准备，有助于快速、准确地完成配置任务。

应用环境

在定义了路由策略相关的过滤器（地址前缀列表、ACL、Route-Policy 等）后，需要在协议中引入这些过滤器。

- 对接收的路由进行过滤

  应用各协议中的 **filter-policy** 命令引用 ACL 或地址前缀列表，对接收的路由进行过滤，仅接收满足条件的部分路由。

  过滤接收路由的命令是 **filter-policy import**。

  对于距离矢量协议和链路状态协议，**filter-policy** 命令的操作过程是不同的：

  - 距离矢量协议

    距离矢量协议是基于路由表生成路由的。因此过滤器会影响从邻居接收的路由和向邻居发布的路由。

  - 链路状态协议

    链路状态路由协议是基于链路状态数据库来生成路由的，**filter-policy** 不影响链路状态通告或链路状态数据库的完整性，因此在接收和发布时的影响是不同的。

    在接收路由时，**filter-policy** 只能决定哪些路由从协议路由表安装到本地核心路由表，即只影响本地核心路由表，不影响协议路由表；

说明

- BGP 具有强大的过滤功能，BGP 相关的策略配置请参见 **BGP 配置**。
- 在路由协议 RIP、OSPF、IS-IS、BGP 中，都有相应的 **filter-policy** 和 **import-route** 命令及其应用，请参见各章节中相关配置。

前置任务

在应用路由策略相关过滤器之前，需完成以下任务：

- **配置地址前缀列表**
- 配置 ACL 列表
- **配置 Route-Policy**

数据准备

在应用路由策略相关过滤器之前，需要准备以下数据。

序号	数据
1	地址前缀列表名称

序号	数据
2	ACL 列表名称
3	Route-policy 的名称和 node 号

10.5.2 对 RIP 接收的路由进行过滤

应用过滤器可以控制 RIP 路由的接收。

背景信息

请在配置 RIP 的路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **rip** [*process-id*] [**vpn-instance** *vpn-instance-name*]，启动 RIP 进程，进入 RIP 视图。

步骤 3 执行命令（根据实际情况选择其中之一）：

- **filter-policy** { *acl-number* | **acl-name** *acl-name* } **import** [*interface-type interface-number*]
- **filter-policy gateway** *ip-prefix-name* **import**
- **filter-policy ip-prefix** *ip-prefix-name* [**gateway** *ip-prefix-name*] **import** [*interface-type interface-number*]

配置对接收的路由进行过滤。

该命令在 RIP 进程下配置，如果基于接口对路由进行过滤，则一个接口只能配置一个路由策略；如果不指定接口，就认为是配置全局过滤策略，同样每次只能配置一个策略，如果重复配置，新的策略将覆盖之前的策略。

----结束

10.5.3 对 OSPF 接收的路由进行过滤

应用过滤器可以控制 OSPF 路由的接收。

背景信息

请在配置 OSPF 的路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **ospf** [*process-id*]，启动 OSPF 进程，进入 OSPF 视图。

步骤 3 执行命令 **filter-policy** { *acl-number* | **acl-name** *acl-name* | **ip-prefix** *ip-prefix-name* } **import**，配置对接收的路由进行过滤。

----结束

10.5.4 对 IS-IS 接收的路由进行过滤

应用过滤器可以控制 IS-IS 路由的接收。

背景信息

请在配置 IS-IS 的路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **isis** [*process-id*] [**vpn-instance** *vpn-instance-name*]，启动 IS-IS 进程，进入 IS-S 视图。

步骤 3 执行命令 **filter-policy** { *acl-number* | **acl-name** *acl-name* | **ip-prefix** *ip-prefix-name* | **route-policy** *route-policy-name* } **import**，配置对接收的路由进行过滤。

----结束

10.5.5 对 BGP 接收的路由进行过滤

应用过滤器可以控制 BGP 路由的接收。

操作步骤

- 过滤从全局接收的路由

  请在配置 BGP 的路由器上进行以下配置。

  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **bgp** *as-number*，进入 BGP 视图。
  3. 执行命令 **filter-policy** { *acl-number* | **acl-name** *acl-name* | **ip-prefix** *ip-prefix-name* } **import**，配置对接收的路由进行过滤。

- 过滤从对等体（组）接收的路由

  请在配置 BGP 的路由器上进行以下配置。

  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **bgp** *as-number*，进入 BGP 视图。
  3. 执行命令 **ipv4-family unicast**，进入 IPv4 单播地址族视图。
  4. 执行命令 **peer** { *group-name* | *ipv4-address* } **filter-policy** { *acl-number* | **acl-name** *acl-name* } **import**，配置从对等体（组）接收的路由进行过滤。

----结束

10.5.6 检查配置结果

对接收的路由应用路由过滤器配置成功后，可以查看各协议路由表的相关信息。

前提条件

已经完成对接收的路由应用路由过滤器的所有配置。

操作步骤

- 使用 **display rip** *process-id* **route** 命令查看 RIP 协议路由表信息。
- 使用 **display ospf** [*process-id*] **routing** 命令查看 OSPF 协议路由表信息。
- 使用 **display isis** [*process-id*] **route** 命令查看 ISIS 协议路由表信息。
- 使用 **display bgp routing-table** 命令查看 BGP 协议路由表信息。
- 使用 **display ip routing-table** 命令查看公网 IPv4 路由表信息。

  在邻居路由器上执行命令 **display ip routing-table**，可以看到匹配邻居过滤条件的路由已经被过滤掉或已经执行了 **apply** 动作。

----结束

10.6 对发布的路由应用路由过滤器

在路由协议中应用路由策略相关的过滤器，过滤发布的路由。

10.6.1 建立配置任务

在对发布的路由应用路由过滤器前了解此特性的应用环境、配置此特性的前置任务和数据准备，有助于快速、准确地完成配置任务。

应用环境

在定义了路由策略相关的过滤器（地址前缀列表、ACL、Route-Policy 等）后，需要在协议中引入这些过滤器。

- 对发布的路由进行过滤

  应用各协议中的 **filter-policy** 命令引用 ACL 和地址前缀列表，对发布的路由进行过滤，仅发布满足条件的部分路由。

  过滤发布路由的命令是 **filter-policy export**。

  对于距离矢量协议和链路状态协议，**filter-policy** 命令的操作过程是不同的：

  - 距离矢量协议

    距离矢量协议是基于路由表生成路由的。因此过滤器会影响从邻居接收的路由和向邻居发布的路由。

  - 链路状态协议

    链路状态路由协议是基于链路状态数据库来生成路由的，**filter-policy** 不影响链路状态通告或链路状态数据库的完整性，因此在接收与发布时的影响是不同的。

    在发布路由时，**filter-policy export** 命令可以用来控制是否发布本协议从其他路由协议引入的路由（如引入的 RIP 路由）。如果没有通过 **Import** 方式引入路由，则不会将引入路由的 LSA/LSP 加入到 LSDB 中，而且不影响向其他路由器发布的链路状态通告。

说明

- BGP 具有强大的过滤功能，BGP 相关的策略配置请参见 **BGP 配置**。
- 在路由协议 RIP、OSPF、IS-IS、BGP 中，都有相应的 **filter-policy** 和 **import-route** 命令及其应用，请参见各章节中相关配置。

前置任务

在应用路由策略相关过滤器之前，需完成以下任务：

- **配置地址前缀列表**
- 配置 ACL 列表
- **配置 Route-Policy**

数据准备

在应用路由策略相关过滤器之前，需要准备以下数据。

序号	数据
1	地址前缀列表名称
2	ACL 列表名称
3	Route-policy 的名称和 node 号

10.6.2 对 RIP 发布的路由进行过滤

应用过滤器可以控制 RIP 路由的发布。

背景信息

请在配置 RIP 的路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **rip** [*process-id*]，启动 RIP 进程，进入 RIP 视图。

步骤 3 执行命令 **filter-policy** { *acl-number* | **acl-name** *acl-name* | **ip-prefix** *ip-prefix-name* } **export** [*protocol* [*process-id*] | *interface-type interface-number*]，配置对发布的路由进行过滤。

该命令在 RIP 进程下配置，如果基于接口对路由进行过滤，则一个接口只能配置一个路由策略；如果不指定接口，就认为是配置全局过滤策略，同样每次只能配置一个策略，如果重复配置，新的策略将覆盖之前的策略。

----结束

10.6.3 对 OSPF 发布的路由进行过滤

应用过滤器可以控制 OSPF 路由的发布。

背景信息

请在配置 OSPF 的路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **ospf** [*process-id*]，启动 OSPF 进程，进入 OSPF 视图。

步骤 3 执行命令 **filter-policy** { *acl-number* | **acl-name** *acl-name* | **ip-prefix** *ip-prefix-name* } **export** [*protocol* [*process-id*]]，配置 OSPF 按照过滤策略，对引入的路由在向外发布时进行过滤。

----结束

10.6.4 对 IS-IS 发布的路由进行过滤

应用过滤器可以控制 IS-IS 路由的发布。

背景信息

请在配置 IS-IS 的路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **isis** [*process-id*]，启动 IS-IS 进程，进入 IS-S 视图。

步骤 3 执行命令 **filter-policy** { *acl-number* | **acl-name** *acl-name* | **ip-prefix** *ip-prefix-name* | **route-policy** *route-policy-name* } **export** [*protocol* [*process-id*]]，配置对发布的路由进行过滤。

----结束

10.6.5 对 BGP 发布的路由进行过滤

应用过滤器可以控制 BGP 路由的发布。

操作步骤

- 过滤全局的路由

  请在配置 BGP 的路由器上进行以下配置。

  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **bgp** *as-number*，进入 BGP 视图。
  3. 执行命令 **ipv4-family unicast**，进入 IPv4 单播地址族视图。
  4. 执行命令 **filter-policy** { *acl-number* | **acl-name** *acl-name* | **ip-prefix** *ip-prefix-name* } **export** [*protocol* [*process-id*]]，对发布的路由信息进行过滤。

     对于 BGP 引入的路由，只有符合条件的路由信息才能进入 BGP 本地路由表，发布给 BGP 对等体。

     - 指定 *protocol* 参数可以只对特定路由协议的信息进行过滤；

- 如果没有指定此参数，则对所有要发布的 BGP 路由信息进行过滤，包括引入的路由和使用 **network** 命令发布的本地路由。

  说明

  不同协议的 **filter-policy export** 命令对待发布路由的影响范围不同：

  - 对于链路状态协议，只对引入的路由信息进行过滤。
  - 对于距离矢量协议，会对引入的路由信息、本协议发现的路由信息进行过滤。

- 过滤向对等体发布的路由

  请在配置 BGP 的路由器上进行以下配置。

  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **bgp** *as-number*，进入 BGP 视图。
  3. 执行命令 **ipv4-family unicast**，进入 IPv4 单播地址族视图。
  4. 执行命令 **peer** { *group-name* | *ipv4-address* } **filter-policy** { *acl-number* | **acl-name** *acl-name* } **export**，配置向对等体（组）发布的路由进行过滤。

----结束

10.6.6 检查配置结果

对发布的路由应用路由过滤器配置成功后，可以查看各协议路由表的相关信息。

前提条件

已经完成对发布的路由应用路由过滤器的所有配置。

操作步骤

- 使用 **display rip** *process-id* **route** 命令查看 RIP 协议路由表信息。
- 使用 **display ospf** [*process-id*] **routing** 命令查看 OSPF 协议路由表信息。
- 使用 **display isis** [*process-id*] **route** 命令查看 ISIS 协议路由表信息。
- 使用 **display bgp routing-table** 命令查看 BGP 协议路由表信息。
- 使用 **display ip routing-table** 命令查看公网 IPv4 路由表信息。

  在邻居路由器上执行命令 **display ip routing-table**，可以看到匹配邻居过滤条件的路由已经被过滤掉或已经执行了 **apply** 动作。

----结束

10.7 对引入的路由应用路由过滤器

在路由协议中应用路由策略相关的过滤器，过滤引入的路由。

10.7.1 建立配置任务

在对引入的路由应用路由过滤器前了解此特性的应用环境、配置此特性的前置任务和数据准备，有助于快速、准确地完成配置任务。

应用环境

在定义了路由策略相关的过滤器（地址前缀列表、ACL、Route-Policy 等）后，需要在协议中引入这些过滤器。

- 引入外部路由时应用策略
  - 应用各协议中的 **import-route** 命令，在各个协议中引入所需的外部路由，同时可以使用 Route-Policy 过滤器对引入的路由进行严格控制。
  - 通常在引入外部路由后，可以应用 **filter-policy export** 命令对引入的路由信息在向外发布时，进行过滤。

说明

- BGP 具有强大的过滤功能，BGP 相关的策略配置请参见 **BGP 配置**。
- 在路由协议 RIP、OSPF、IS-IS、BGP 中，都有相应的 **filter-policy** 和 **import-route** 命令及其应用，请参见各章节中相关配置。

前置任务

在应用路由策略相关过滤器之前，需完成以下任务：

- **配置地址前缀列表**
- 配置 ACL 列表
- **配置 Route-Policy**

数据准备

在应用路由策略相关过滤器之前，需要准备以下数据。

序号	数据
1	地址前缀列表名称
2	ACL 列表名称
3	Route-policy 的名称和 node 号

10.7.2 对 RIP 引入外部路由时应用策略

通过应用过滤器，RIP 可以选择引入外部路由。

背景信息

请在配置 RIP 的路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **rip** [*process-id*]，使能 RIP 路由进程，进入 RIP 视图。

步骤 3 执行命令 **import-route bgp** [**cost** { *cost* | **transparent** } | **route-policy** *route-policy-name*] * 或 **import-route** { { **static** | **direct** | **unr** } | { { **rip** | **ospf** | **isis** } [*process-id*] } } [**cost** *cost* | **route-policy** *route-policy-name*] *，引入外部路由信息。

----结束

10.7.3 对 OSPF 引入外部路由时应用策略

通过应用过滤器，OSPF 可以选择引入外部路由。

背景信息

请在配置 OSPF 的路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **ospf** [*process-id*]，使能 OSPF 路由进程，进入 OSPF 视图。

步骤 3 执行命令 **import-route** { **limit** *limit-number* | { **bgp** [**permit-ibgp**] | **direct** | **unr** | **rip** [*process-id-rip*] | **static** | **isis** [*process-id-isis*] | **ospf** [*process-id-ospf*] } [**cost** *cost* | **type** *type* | **tag** *tag* | **route-policy** *route-policy-name*] * }，引入外部路由信息。

----结束

10.7.4 对 IS-IS 引入外部路由时应用策略

通过应用过滤器，IS-IS 可以选择引入外部路由。

背景信息

请在配置 IS-IS 的路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **isis** [*process-id*]，使能 IS-IS 路由进程，进入 IS-IS 视图。

步骤 3 配置 IS-IS 引入外部路由。

- 当需要对引入路由的开销进行设置时，执行命令 **import-route** *protocol* [*process-id*] [**cost-type** { **external** | **internal** } | **cost** *cost* | **tag** *tag* | **route-policy** *route-policy-name* | [**level-1** | **level-2** | **level-1-2**]] * 引入外部路由。
- 当需要保留引入路由的原有开销时，执行命令 **import-route** { { **rip** | **isis** | **ospf** } [*process-id*] | **bgp** } **inherit-cost** [**tag** *tag* | **route-policy** *route-policy-name* | [**level-1** | **level-2** | **level-1-2**]] * 引入外部路由。

----结束

10.7.5 对 BGP 引入外部路由时应用策略

通过应用过滤器，BGP 可以选择引入外部路由。

背景信息

请在配置 BGP 的路由器上进行以下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **bgp** *as-number*，进入 BGP 视图。

步骤 3 执行命令 **ipv4-family unicast**，进入 IPv4 单播地址族视图。

步骤 4 执行命令 **import-route** *protocol* [*process-id*] [**med** *med* | **route-policy** *route-policy-name*] *，引入外部路由信息。

----结束

10.7.6 检查配置结果

对引入的路由应用路由过滤器配置成功后，可以查看各协议路由表的相关信息。

前提条件

已经完成对引入的路由应用路由过滤器的所有配置。

操作步骤

- 使用 **display rip** *process-id* **route** 命令查看 RIP 协议路由表信息。
- 使用 **display ospf** [*process-id*] **routing** 命令查看 OSPF 协议路由表信息。
- 使用 **display isis** [*process-id*] **route** 命令查看 ISIS 协议路由表信息。
- 使用 **display bgp routing-table** 命令查看 BGP 协议路由表信息。
- 使用 **display ip routing-table** 命令查看公网 IPv4 路由表信息。

  在邻居路由器上执行命令 **display ip routing-table**，可以看到匹配邻居过滤条件的路由已经被过滤掉或已经执行了 **apply** 动作。

----结束

10.8 控制路由策略生效时间

为了保障网络的稳定性，修改路由策略时需要控制路由策略的生效时间。

10.8.1 建立配置任务

在配置控制路由策略生效时间前了解此特性的应用环境、配置此特性的前置任务和数据准备，有助于快速、准确地完成配置任务。

应用环境

在实际应用中，当多条相互配合的路由策略的配置发生变化时，如果每完成一条配置，RM（Routing Management Module）就立即通知各协议重新应用策略。不完整的策略则会造成路由振荡和中间结果处理时间的浪费，造成网络的不稳定。

AR150/200 系统对路由策略的变化处理规则如下：

- 缺省情况下，路由策略变化后，RM 将立即通知协议应用新策略。
- 如果配置了路由策略生效时间，当路由策略的相关命令配置变化后，RM 并不立即通知协议进行处理，而是等待所配置的时长，然后再通知各协议应用变化后的策略。
- 如果在等待时间内路由策略的配置又发生了改变，RM 将重置定时器，重新开始计时。

可以根据实际情况，应用命令选择延迟等待时间的长短。

前置任务

无

数据准备

在配置路由策略生效时间之前，需要准备以下数据。

序号	数据
1	路由策略应用的延迟时间

10.8.2 配置路由策略应用延迟时间

修改多条相互配合的路由策略时，需要控制路由策略的生效时间。

背景信息

请在需要改变路由策略应用延迟时间的路由器上进行如下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **route-policy-change notify-delay** *delay-time*，设置路由策略应用延迟时间。

延迟时间的取值范围是 1 秒～ 180 秒。

缺省情况下，路由策略变化后，RM 将立即通知协议应用新策略。

步骤 3 执行命令 **quit**，退回用户视图。

步骤 4 （可选）执行命令 **refresh bgp all** { **export** | **import** }，配置 BGP 协议立即应用新策略。

如果配置策略命令后，需要立即看到策略过滤的效果。可以通过执行这个命令，配置 BGP 协议立即应用新策略。

受该定时器影响的相关的策略有访问控制列表、地址前缀列表、AS 路径过滤器、团体属性过滤器、扩展团体属性过滤器、RD 属性过滤器和 Route-Policy。

----结束

10.8.3 检查配置结果

路由策略生效时间配置成功后，可以查看配置的路由策略生效时间。

前提条件

已经完成控制路由策略生效时间的所有配置。

操作步骤

- 使用 **display current-configuration | include notify-delay** 命令查看路由策略应用延迟时间。

----结束

任务示例

执行 **display current-configuration** 命令，可以看到目前配置的路由策略延迟时间，例如：

```
<Huawei> display current-configuration | include notify-delay
route-policy-change notify-delay 10
```

10.9 路由策略维护

路由策略的维护包括清除地址前缀列表统计数据及路由策略的调试。

背景信息

注意

清除地址前缀列表的统计数据后，以前的数据信息将无法恢复，务必仔细确认。

缺省情况下，不清除地址前缀列表的统计数据。

操作步骤

- 在确认需要清除 IPv4 地址前缀列表统计数据，请在用户视图下执行 **reset ip ip-prefix** [*ip-prefix-name*]命令。
- 在确认需要清除 IPv6 地址前缀列表统计数据，请在用户视图下执行 **reset ip ipv6-prefix** [*ipv6-prefix-name*]命令。

----结束

10.10 配置举例

路由策略配置举例包括组网需求、组网图、配置注意事项、配置思路和配置步骤。

10.10.1 对接收和发布的路由进行过滤示例

网络中可根据通信需求，对接收和发布的路由应用过滤器。

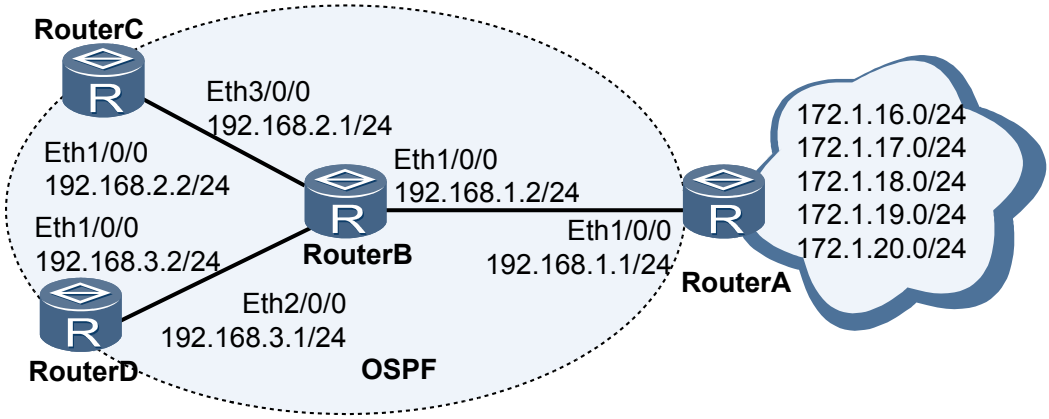
组网需求

如图 10-1，运行 OSPF 协议的网络中，RouterA 从 Internet 网络接收路由，并为 RouterB 提供了部分 Internet 路由。要求 RouterA 仅提供 172.1.17.0/24、172.1.18.0/24、172.1.19.0/24 给 RouterB，RouterC 仅接收路由 172.1.18.0/24，RouterD 接收 RouterB 提供的全部路由。

说明

AR150/200 仅可作为 RouterC 或 RouterD。

图 10-1 配置对接收和发布的路由过滤组网图



配置思路

采用如下的思路配置对路由进行过滤：

1. 在 RouterA、RouterB、RouterC 和 RouterD 上配置 OSPF 基本功能。
2. 在 RouterA 上配置静态路由，并将这些路由引入 OSPF 路由。
3. 在 RouterA 上配置路由发布策略，在 RouterB 上查看过滤结果。
4. 在 RouterC 上配置路由接收策略，在 RouterC 上查看过滤结果。

数据准备

为完成此配置例，需准备如下数据：

- RouterA 引入的 5 条静态路由。
- RouterA、RouterB、RouterC 和 RouterD 位于 OSPF 骨干区域（Area0）。
- 地址前缀列表名称，待过滤路由。

操作步骤

步骤 1 配置各接口的 IP 地址（略）

步骤 2 配置 OSPF 基本功能

RouterA 的配置

```
[RouterA] ospf
[RouterA-ospf-1] area 0
[RouterA-ospf-1-area-0.0.0.0] network 192.168.1.0 0.0.0.255
[RouterA-ospf-1-area-0.0.0.0] quit
[RouterA-ospf-1] quit
```

RouterB 的配置

```
[RouterB] ospf
[RouterB-ospf-1] area 0
[RouterB-ospf-1-area-0.0.0.0] network 192.168.1.0 0.0.0.255
[RouterB-ospf-1-area-0.0.0.0] network 192.168.2.0 0.0.0.255
[RouterB-ospf-1-area-0.0.0.0] network 192.168.3.0 0.0.0.255
[RouterB-ospf-1-area-0.0.0.0] quit
```

RouterC 的配置

```
[RouterC] ospf
[RouterC-ospf-1] area 0
[RouterC-ospf-1-area-0.0.0.0] network 192.168.2.0 0.0.0.255
[RouterC-ospf-1-area-0.0.0.0] quit
[RouterC-ospf-1] quit
```

RouterD 的配置

```
[RouterD] ospf
[RouterD-ospf-1] area 0
[RouterD-ospf-1-area-0.0.0.0] network 192.168.3.0 0.0.0.255
[RouterD-ospf-1-area-0.0.0.0] quit
```

步骤 3 在 RouterA 上配置 5 条静态路由，并在将这些静态路由引入到 OSPF 协议中

```
[RouterA] ip route-static 172.1.16.0 24 NULL 0
[RouterA] ip route-static 172.1.17.0 24 NULL 0
[RouterA] ip route-static 172.1.18.0 24 NULL 0
[RouterA] ip route-static 172.1.19.0 24 NULL 0
[RouterA] ip route-static 172.1.20.0 24 NULL 0
[RouterA] ospf
[RouterA-ospf-1] import-route static
[RouterA-ospf-1] quit
```

在 RouterB 上查看 IP 路由表，可以看到 OSPF 引入的 5 条静态路由。

```
[RouterB] display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 16        Routes : 16

Destination/Mask    Proto   Pre  Cost      Flags NextHop         Interface

        127.0.0.0/8   Direct 0    0           D   127.0.0.1       InLoopBack0
        127.0.0.1/32  Direct 0    0           D   127.0.0.1       InLoopBack0
      172.1.16.0/24  O_ASE  150  1           D   192.168.1.1     Ethernet1/0/0
      172.1.17.0/24  O_ASE  150  1           D   192.168.1.1     Ethernet1/0/0
      172.1.18.0/24  O_ASE  150  1           D   192.168.1.1     Ethernet1/0/0
      172.1.19.0/24  O_ASE  150  1           D   192.168.1.1     Ethernet1/0/0
      172.1.20.0/24  O_ASE  150  1           D   192.168.1.1     Ethernet1/0/0
      192.168.1.0/24  Direct 0    0           D   192.168.1.2     Ethernet1/0/0
      192.168.1.1/32  Direct 0    0           D   192.168.1.1     Ethernet1/0/0
      192.168.1.2/32  Direct 0    0           D   127.0.0.1       InLoopBack0
      192.168.2.0/24  Direct 0    0           D   192.168.2.1     Ethernet3/0/0
      192.168.2.1/32  Direct 0    0           D   127.0.0.1       InLoopBack0
      192.168.2.2/32  Direct 0    0           D   192.168.2.2     Ethernet3/0/0
      192.168.3.0/24  Direct 0    0           D   192.168.3.1     Ethernet2/0/0
      192.168.3.1/32  Direct 0    0           D   127.0.0.1       InLoopBack0
      192.168.3.2/32  Direct 0    0           D   192.168.3.2     Ethernet2/0/0
```

步骤 4 配置路由发布策略

在 RouterA 上配置地址前缀列表 a2b。

```
[RouterA] ip ip-prefix a2b index 10 permit 172.1.17.0 24
[RouterA] ip ip-prefix a2b index 20 permit 172.1.18.0 24
[RouterA] ip ip-prefix a2b index 30 permit 172.1.19.0 24
```

在 RouterA 上配置发布策略，引用地址前缀列表 a2b 进行过滤。

```
[RouterA] ospf
[RouterA-ospf-1] filter-policy ip-prefix a2b export static
```

在 RouterB 上查看 IP 路由表，可以看到 RouterB 仅接收到列表 a2b 中定义的 3 条路由。

```
[RouterB] display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 14        Routes : 14

Destination/Mask    Proto   Pre  Cost      Flags NextHop         Interface

        127.0.0.0/8   Direct 0    0           D   127.0.0.1       InLoopBack0
        127.0.0.1/32  Direct 0    0           D   127.0.0.1       InLoopBack0
      172.1.17.0/24  O_ASE  150  1           D   192.168.1.1     Ethernet1/0/0
      172.1.18.0/24  O_ASE  150  1           D   192.168.1.1     Ethernet1/0/0
      172.1.19.0/24  O_ASE  150  1           D   192.168.1.1     Ethernet1/0/0
      192.168.1.0/24  Direct 0    0           D   192.168.1.2     Ethernet1/0/0
      192.168.1.1/32  Direct 0    0           D   192.168.1.1     Ethernet1/0/0
      192.168.1.2/32  Direct 0    0           D   127.0.0.1       InLoopBack0
      192.168.2.0/24  Direct 0    0           D   192.168.2.1     Ethernet3/0/0
      192.168.2.1/32  Direct 0    0           D   127.0.0.1       InLoopBack0
      192.168.2.2/32  Direct 0    0           D   192.168.2.2     Ethernet3/0/0
      192.168.3.0/24  Direct 0    0           D   192.168.3.1     Ethernet2/0/0
      192.168.3.1/32  Direct 0    0           D   127.0.0.1       InLoopBack0
      192.168.3.2/32  Direct 0    0           D   192.168.3.2     Ethernet2/0/0
```

步骤 5 配置路由接收策略

在 RouterC 上配置地址前缀列表 in。

```
[RouterC] ip ip-prefix in index 10 permit 172.1.18.0 24
```

在 RouterC 上配置接收策略，引用地址前缀列表 in 进行过滤。

```
[RouterC] ospf
[RouterC-ospf-1] filter-policy ip-prefix in import
```

查看 RouterC 的 IP 路由表，可以看到 RouterC 的本地核心路由表中，仅接收了列表 in 定义的 1 条路由。

```
[RouterC] display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 6        Routes : 6

Destination/Mask    Proto   Pre  Cost      Flags NextHop         Interface

        127.0.0.0/8   Direct  0    0           D   127.0.0.1       InLoopBack0
        127.0.0.1/32  Direct  0    0           D   127.0.0.1       InLoopBack0
      172.1.18.0/24  O_ASE  150  1           D   192.168.2.1     Ethernet1/0/0
      192.168.2.0/24  Direct  0    0           D   192.168.2.2     Ethernet1/0/0
      192.168.2.1/32  Direct  0    0           D   192.168.2.1     Ethernet1/0/0
      192.168.2.2/32  Direct  0    0           D   127.0.0.1       InLoopBack0
```

查看 RouterD 的 IP 路由表，可以看到 RouterD 的本地核心路由表中，接收了 RouterB 发送的所有路由。

```
[RouterD] display ip routing-table
```

```
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 10        Routes : 10

Destination/Mask    Proto   Pre  Cost      Flags NextHop         Interface

      127.0.0.0/8   Direct  0    0           D   127.0.0.1       InLoopBack0
      127.0.0.1/32  Direct  0    0           D   127.0.0.1       InLoopBack0
     172.1.17.0/24  O_ASE   150  1           D   192.168.3.1     Ethernet1/0/0
     172.1.18.0/24  O_ASE   150  1           D   192.168.3.1     Ethernet1/0/0
     172.1.19.0/24  O_ASE   150  1           D   192.168.3.1     Ethernet1/0/0
    192.168.1.0/24  OSPF    10   1           D   192.168.3.1     Ethernet1/0/0
    192.168.2.0/24  OSPF    10   1           D   192.168.3.1     Ethernet1/0/0
    192.168.3.0/24  Direct  0    0           D   192.168.3.2     Ethernet1/0/0
    192.168.3.1/32  Direct  0    0           D   192.168.3.1     Ethernet1/0/0
    192.168.3.2/32  Direct  0    0           D   127.0.0.1       Ethernet1/0/0
```

查看 RouterC 的 OSPF 路由表，可以看到 OSPF 路由表中接收到 3 条列表 a2b 中定义的路由。因为在链路状态协议中，**filter-policy import** 命令用于过滤从协议路由表加入本地核心路由表的路由。

```
[RouterC] display ospf routing

          OSPF Process 1 with Router ID 192.168.2.2
                  Routing Tables

 Routing for Network
 Destination        Cost  Type       NextHop         AdvRouter       Area
 192.168.2.0/24     1     Stub       192.168.2.2     192.168.2.2     0.0.0.0
 192.168.1.0/24     2     Stub       192.168.2.1     192.168.2.1     0.0.0.0
 192.168.3.0/24     2     Stub       192.168.2.1     192.168.2.1     0.0.0.0

 Routing for ASEs
 Destination        Cost      Type       Tag         NextHop         AdvRouter
 172.1.17.0/24      1         Type2      1           192.168.2.1     192.168.1.1
 172.1.18.0/24      1         Type2      1           192.168.2.1     192.168.1.1
 172.1.19.0/24      1         Type2      1           192.168.2.1     192.168.1.1

 Total Nets: 6
 Intra Area: 3  Inter Area: 0  ASE: 3  NSSA: 0
```

----结束

配置文件

- RouterA 的配置文件

```
#
 sysname RouterA
#
interface Ethernet1/0/0
 ip address 192.168.1.1 255.255.255.0
#
ospf 1
 filter-policy ip-prefix a2b export static
 import-route static
 area 0.0.0.0
  network 192.168.1.0 0.0.0.255
#
 ip ip-prefix a2b index 10 permit 172.1.17.0 24
 ip ip-prefix a2b index 20 permit 172.1.18.0 24
 ip ip-prefix a2b index 30 permit 172.1.19.0 24
#
 ip route-static 172.1.16.0 255.255.255.0 NULL0
 ip route-static 172.1.17.0 255.255.255.0 NULL0
 ip route-static 172.1.18.0 255.255.255.0 NULL0
 ip route-static 172.1.19.0 255.255.255.0 NULL0
 ip route-static 172.1.20.0 255.255.255.0 NULL0
```

```
#
return
```

- RouterB 的配置文件

```
#
 sysname RouterB
#
interface Ethernet1/0/0
 ip address 192.168.1.2 255.255.255.0
#
interface Ethernet2/0/0
 ip address 192.168.3.1 255.255.255.0
#
interface Ethernet3/0/0
 ip address 192.168.2.1 255.255.255.0
#
ospf 1
 area 0.0.0.0
  network 192.168.1.0 0.0.0.255
  network 192.168.2.0 0.0.0.255
  network 192.168.3.0 0.0.0.255
#
return
```

- RouterC 的配置文件

```
#
 sysname RouterC
#
interface Ethernet1/0/0
 ip address 192.168.2.2 255.255.255.0
#
ospf 1
 filter-policy ip-prefix in import
 area 0.0.0.0
  network 192.168.2.0 0.0.0.255
#
 ip ip-prefix in index 10 permit 172.1.18.0 24
#
return
```

- RouterD 的配置文件

```
#
 sysname RouterD
#
interface Ethernet1/0/0
 ip address 192.168.3.2 255.255.255.0
#
ospf 1
 area 0.0.0.0
  network 192.168.3.0 0.0.0.255
#
return
```