JXNET 甲信

JX S9600 系列 三层以太网交换机
配置指南(CLI)
(Rel_01)

北京甲信技术有限公司（以下简称“甲信”）为客户提供全方位的技术支持和服务。直接向甲信购买产品的用户，如果在使用过程中有任何问题，可与甲信各地办事处或用户服务中心联系，也可直接与公司总部联系。

读者如有任何关于甲信产品的问题，或者有意进一步了解公司其他相关产品，可通过下列方式与我们联系：

公司网址：　　www.jiaxinnet.com.cn

技术支持邮箱：　jxhelp@bjjx.cc

技术支持热线：　400-179-1180

公司总部地址：　北京市海淀区丹棱 SOHO 7 层 728 室

邮政编码:　　100080

声　明

前 言

概述

本文档系统介绍了 JX S9600 系列以太网交换机（以下简称 JX S9600 系列）设备支持的特性及其相关配置。主要内容包括基础配置、以太网、环网保护、IP 业务、DHCP、可靠性、安全性、QoS 等基本原理和配置过程，并提供相关的配置案例。在本文档的附录中，提供了该文档所涉及的术语和缩略语。

阅读本文档有助于读者系统掌握设备的原理和各种配置信息，以及如何应用该设备进行组网。

产品版本

与本文档相对应的产品版本如下所示。

产品名称	软件版本	硬件版本
JX S9600 系列 以太网交换机	V7.02	A

约定

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
	以本标志开始的文本表示有潜在危险，如果不能避免，可能导致人员伤害。
	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。
	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。

符号	说明
	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。

通用格式约定

格式	说明
宋体	正文采用宋体表示。
黑体	一级标题、二级标题、三级标题、Block 采用**黑体**表示。
楷体	警告、提示等内容用楷体表示。
“Lucida Console”格式	“Lucida Console”格式表示屏幕输出信息。此外，屏幕输出信息中夹杂的用户从终端输入的信息采用加粗字体表示。

命令行格式约定

格式	说明
粗体	命令行关键字（命令中保持不变、必须照输的部分）采用**粗体**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。

修订记录

修订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 01 (2024-09-30)

第一次正式发布。

目 录

1 基础配置 1
1.1 登录设备 1
1.1.1 简介 1
1.1.2 通过 Console 口登录设备 2
1.1.3 通过 Telnet 登录设备 4
1.1.4 通过 SSH 登录设备 7
1.1.5 通过 Web 登录设备 9
1.1.6 管理登录用户 10
1.1.7 用户密码恢复 12
1.1.8 终端属性设置 12
1.1.9 Bootrom 密码设置 13
1.2 加载与升级 14
1.2.1 简介 14
1.2.2 通过 TFTP 命令行升级方式升级系统软件 14
1.2.3 通过 FTP 命令行升级方式升级系统软件 15
1.2.4 检查配置 15
1.2.5 指定系统启动镜像 15
1.2.6 检查系统版本信息 15
1.3 时间管理 16
1.3.1 简介 16
1.3.2 配置准备 18
1.3.3 配置 NTP 19
1.3.4 配置 SNTP 21
1.3.5 检查配置 22
1.3.6 配置 NTP 功能示例 22
1.4 接口管理 24
1.4.1 简介 24
1.4.2 接口的缺省配置 25
1.4.3 配置接口基本属性 25
1.4.4 配置接口信息统计功能 26
1.4.5 配置接口流控功能 27
1.4.6 配置接口打开或关闭 27

1.4.7 配置接口允许通过的 tagged 报文类型 27
1.4.8 配置管理报文优先级 28
1.4.9 检查配置 28
2 以太网 1
2.1 MAC 地址转发表 1
2.1.1 简介 1
2.1.2 配置准备 3
2.1.3 MAC 地址转发表的缺省配置 4
2.1.4 配置静态 MAC 地址 4
2.1.5 配置黑洞 MAC 地址 4
2.1.6 配置 MAC 地址学习 5
2.1.7 配置 MAC 地址学习数目限制 5
2.1.8 配置 MAC 老化时间 5
2.1.9 配置 MAC 地址漂移检测与保护 5
2.1.10 检查配置 6
2.1.11 维护 7
2.1.12 配置 MAC 地址转发表示例 7
2.2 VLAN 8
2.2.1 简介 8
2.2.2 配置准备 10
2.2.3 VLAN 的缺省配置 11
2.2.4 配置 VLAN 属性 11
2.2.5 配置接口模式 11
2.2.6 配置基于 Access 接口的 VLAN 12
2.2.7 配置基于 Trunk 接口的 VLAN 12
2.2.8 配置基于 Hybrid 接口的 VLAN 13
2.2.9 配置基于 MAC 地址的 VLAN 14
2.2.10 配置基于 IP 子网的 VLAN 14
2.2.11 配置基于协议的 VLAN 15
2.2.12 查询 VLAN 统计信息 15
2.2.13 检查配置 16
2.2.14 配置 VLAN 示例 16
2.3 Voice VLAN 18
2.3.1 简介 18
2.3.2 配置准备 19
2.3.3 Voice VLAN 的缺省配置 19
2.3.4 配置 OUI 地址 20
2.3.5 使能 Voice VLAN 功能 20
2.3.6 检查配置 21

2.3.7 配置端口加入 Voice VLAN 示例 21
2.3.8 配置 LLDP 实现 IP 话机接入 Voice VLAN 示例 22
2.4 QinQ 24
2.4.1 简介 24
2.4.2 配置准备 25
2.4.3 QinQ 的缺省配置 25
2.4.4 配置基本 QinQ 26
2.4.5 配置灵活 QinQ 27
2.4.6 配置网络侧接口为 Trunk 模式 27
2.4.7 配置 TPID 27
2.4.8 检查配置 28
2.4.9 配置基本 QinQ 示例 28
2.4.10 配置灵活 QinQ 示例 29
2.5 VLAN 转换 31
2.5.1 简介 31
2.5.2 配置准备 31
2.5.3 VLAN 转换的缺省配置 32
2.5.4 配置 VLAN 转换 32
2.5.5 检查配置 33
2.5.6 配置 VLAN 转换示例 33
2.6 MRP/MVRP 35
2.6.1 简介 35
2.6.2 配置准备 37
2.6.3 缺省配置 37
2.6.4 配置 MVRP 基本功能 38
2.6.5 检查配置 38
2.6.6 配置 MVRP 应用示例 39
2.7 VTP 41
2.7.1 简介 41
2.7.2 配置准备 43
2.7.3 缺省配置 43
2.7.4 配置 VTP 基本功能 43
2.7.5 检查配置 44
2.7.6 配置 VTP 应用示例 44
3 IP 业务 1
3.1 IP 基础配置 1
3.1.1 简介 1
3.1.2 配置准备 1
3.1.3 VLAN 接口的缺省配置 2

3.1.4 配置 VLAN 接口 IPv4 地址 2
3.1.5 配置 VLAN 接口 IPv6 地址 2
3.1.6 检查配置 3
3.1.7 配置 VLAN 接口 IP 地址实现和主机互通示例 3
3.2 LOOPBACK 接口 5
3.2.1 简介 5
3.2.2 配置准备 5
3.2.3 LOOPBACK 接口的缺省配置 5
3.2.4 配置 LOOPBACK 接口 IP 地址 5
3.2.5 检查配置 6
3.3 ARP 6
3.3.1 简介 6
3.3.2 配置准备 7
3.3.3 ARP 的缺省配置 7
3.3.4 配置静态 ARP 表项 7
3.3.5 配置动态 ARP 表项 7
3.3.6 检查配置 8
3.3.7 维护 8
3.3.8 配置 ARP 示例 8
3.4 NDP 10
3.4.1 简介 10
3.4.2 配置准备 11
3.4.3 NDP 的缺省配置 11
3.4.4 配置静态邻居表项 11
3.4.5 配置动态 NDP 老化时间 12
3.4.6 检查配置 12
3.4.7 维护 12
3.5 静态路由 12
3.5.1 简介 12
3.5.2 配置准备 13
3.5.3 配置静态路由 13
3.5.4 检查配置 13
3.5.5 配置静态路由示例 14
3.6 策略路由 15
3.6.1 简介 15
3.6.2 配置基于 ACL 的策略路由示例 16
3.7 BGP 17
3.7.1 简介 17
3.7.2 配置准备 28
3.7.3 缺省配置 28

3.7.4 配置 BGP 基本功能 29
3.7.5 简化 IBGP 网络连接 30
3.7.6 配置 BGP 路由选路和负载分担 30
3.7.7 控制 BGP 路由的发布与接收 31
3.7.8 控制调整 BGP 网络的收敛速度 31
3.7.9 配置 BGP 可靠性 32
3.7.10 检查配置 32
3.7.11 配置 BGP 的基本功能示例 33
3.7.12 配置 BGP 和 IGP 交互功能示例 36
3.7.13 配置 BGP 路由聚合功能示例 39
3.7.14 配置 BGP 团体功能示例 42
3.7.15 配置 BGP 路由反射器功能示例 45
3.7.16 配置 BGP 联盟功能示例 47
3.7.17 配置 BGP 路径选择示例 50
3.8 OSPFV2 55
3.8.1 简介 55
3.8.2 配置准备 69
3.8.3 缺省配置 69
3.8.4 配置 OSPF 基本功能 69
3.8.5 配置 OSPF STUB 区域 70
3.8.6 配置 OSPF NSSA 区域 71
3.8.7 配置 OSPF 引入外部路由 71
3.8.8 配置 OSPF 路由聚合 72
3.8.9 配置 OSPF 虚连接 72
3.8.10 配置 OSPF 认证功能 73
3.8.11 检查配置 74
3.8.12 配置 OSPF 基本功能示例 74
3.8.13 配置 OSPF STUB 区域示例 77
3.8.14 配置 OSPF NSSA 区域示例 79
3.8.15 配置 OSPF 引入外部路由示例 82
3.8.16 配置 OSPF 路由聚合示例 84
3.8.17 配置 OSPF 虚连接示例 89
3.8.18 配置 OSPF 认证示例 93
3.9 OSPFV3 96
3.9.1 简介 96
3.9.2 配置准备 108
3.9.3 缺省配置 108
3.9.4 配置 OSPFV3 基本功能 109
3.9.5 配置 OSPFV3 STUB 区域 109

3.9.6 配置 OSPFV3 NSSA 区域 110
3.9.7 配置 OSPFV3 引入外部路由 110
3.9.8 配置 OSPFV3 路由聚合 111
3.9.9 配置 OSPFV3 虚连接 111
3.9.10 检查配置 112
3.9.11 配置 OSPFV3 基本功能示例 113
3.9.12 配置 OSPFV3 STUB 区域示例 116
3.9.13 配置 OSPFV3 NSSA 区域示例 118
3.9.14 配置 OSPFV3 引入外部路由示例 122
3.9.15 配置 OSPFV3 路由聚合示例 124
3.9.16 配置 OSPFV3 虚连接示例 129
3.10 IS-IS 134
3.10.3 缺省配置 143
3.10.12 配置 IS-IS 的基本功能示例 147
3.10.13 配置 IS-IS 的 DIS 选择示例 152
3.10.14 配置 IS-IS 引入外部路由 156
3.10.15 配置 IS-IS 验证配置举例 158
3.12.1 简介 171
3.13 RIPng 183
3.13.1 简介 183
3.13.2 配置准备 184
3.13.3 配置 RIPng 基本功能 184
3.13.4 配置 RIPng 的路由特性 184
3.13.5 配置调整和优化 RIPng 网络 186
3.13.6 配置检查和维护 186
3.13.7 配置 RIPng 基本功能实示例 187
3.13.8 配置 RIPng 引入外部路由 191
3.15 ECMP 205
3.15.1 简介 205
3.15.2 配置准备 205
3.15.3 缺省配置 206
3.15.4 配置 ECMP 分流算法 206
3.15.5 配置 ECMP 模版 206
3.15.6 检查配置 207
4 DHCP 1
4.1 DHCP Client 1
4.1.1 简介 1
4.1.2 配置准备 4
4.1.3 配置 DHCP 客户端 5

4.1.4 检查配置 5
4.1.5 配置 DHCP 客户端示例 6
4.2 DHCP Server 7
4.2.1 简介 7
4.2.2 配置准备 10
4.2.3 创建并配置 IPv4 地址池 10
4.2.4 配置接口的 DHCP Server 功能 10
4.2.5 回收 IP 地址或地址池 11
4.2.6 配置 DHCPv4 Server Ping 功能 11
4.2.7 检查配置 11
4.2.8 维护 12
4.2.9 配置 DHCPv4 服务器示例 12
4.3 DHCP Relay 13
4.3.1 简介 13
4.3.2 配置准备 14
4.3.3 DHCP Relay 的缺省配置 14
4.3.4 配置接口的 DHCP Relay 功能 15
4.3.5 配置接口的 DHCPv6 Relay 功能 15
4.3.6 配置 DHCP Relay 支持 Option 82 功能 15
4.3.7 配置 DHCPv6 Relay 支持 Option 18/37 功能 16
4.3.8 检查配置 16
4.3.9 维护 17
4.3.10 配置 DHCPv4 中继示例 17
5 QoS 1
5.1 简介 1
5.1.1 服务模型 1
5.2 优先级映射 2
5.2.1 简介 2
5.2.2 配置准备 3
5.2.3 基本 QoS 的缺省配置 4
5.2.4 配置入接口信任的优先级类型 7
5.2.5 配置 DIFFSERV 模版 7
5.2.6 配置出接口 8021P/DSCP 重标记 8
5.2.7 检查配置 9
5.3 队列调度 9
5.3.1 简介 9
5.3.2 配置准备 11
5.3.3 拥塞管理的缺省配置 11
5.3.4 配置 SP 队列调度 12

5.3.5 配置 WRR 或 SP+WRR 队列调度 12
5.3.6 配置 DRR 或 SP+DRR 队列调度 12
5.3.7 配置队列带宽保证 13
5.3.8 检查配置 13
5.3.9 维护 13
5.3.10 配置队列调度示例 13
5.4 拥塞避免 15
5.4.1 简介 15
5.4.2 配置准备 15
5.4.3 拥塞避免的缺省配置 15
5.4.4 配置 WRED 16
5.4.5 检查配置 16
5.5 流量限速 17
5.5.1 简介 17
5.5.2 配置准备 17
5.5.3 配置流量限速模式 17
5.5.4 配置基于物理接口的流量限速 17
5.5.5 配置基于 VLAN 的流量限速 18
5.5.6 配置基于聚合组的流量限速 18
5.5.7 检查配置 19
5.5.8 配置基于接口的流量限速示例 19
5.6 带宽限速 21
5.6.1 简介 21
5.6.2 配置准备 21
5.6.3 带宽限速功能的缺省配置 22
5.6.4 配置带宽保证功能 22
5.6.5 检查配置 25
6 组播 1
6.1 组播概述 1
6.2 IGMP Snooping 6
6.2.1 简介 6
6.2.2 配置准备 7
6.2.3 IGMP Snooping 的缺省配置 7
6.2.4 配置 IGMP Snooping 基本功能 8
6.2.5 配置 IGMP Snooping 查询器功能 8
6.2.6 配置 IGMP Snooping 报文抑制功能 9
6.2.7 配置 IGMP Snooping 组播复制功能 9
6.2.8 配置 IGMP Snooping 静态组播成员功能 10
6.2.9 配置 IGMP Snooping Proxy 功能 10

6.2.10 配置 IGMP Snooping 接口组播组个数限制 10
6.2.11 配置 IGMP Snooping 组播策略 11
6.2.12 配置 IGMP Snooping SSM Mapping 功能 11
6.2.13 检查配置 12
6.2.14 维护 12
6.2.15 配置 IGMP Snooping 基本功能应用示例 12
6.2.16 配置 IGMP Snooping 静态成员应用示例 14
6.2.17 配置 IGMP Snooping 组播复制应用示例 16
6.2.18 配置 IGMP Snooping Proxy 应用示例 18
6.2.19 配置 IGMP Snooping 策略应用示例 20
6.3 MLD Snooping 22
6.3.1 简介 22
6.3.2 配置准备 22
6.3.3 MLD Snooping 的缺省配置 23
6.3.4 配置 MLD Snooping 基本功能 23
6.3.5 配置 MLD Snooping 查询器功能 24
6.3.6 配置 MLD Snooping 报文抑制功能 25
6.3.7 配置 MLD Snooping 组播复制功能 25
6.3.8 配置 MLD Snooping 静态组播成员功能 25
6.3.9 配置 MLD Snooping Proxy 功能 26
6.3.10 配置 MLD Snooping 接口组播组个数限制 26
6.3.11 配置 MLD Snooping 组播策略 26
6.3.12 配置 MLD Snooping SSM Mapping 功能 27
6.3.13 检查配置 27
6.3.14 维护 28
6.3.15 配置 MLD Snooping 基本功能应用示例 28
6.3.16 配置 MLD Snooping 静态成员应用示例 30
6.3.17 配置 MLD Snooping 组播复制应用示例 32
6.3.18 配置 MLD Snooping Proxy 应用示例 33
6.3.19 配置 MLD Snooping 策略应用示例 35
6.4 IGMP 38
6.4.1 简介 38
6.4.2 配置准备 38
6.4.3 IGMP 的缺省配置 39
6.4.4 配置 IGMP 使能 39
6.4.5 配置 IGMP 查询器参数 40
6.4.6 配置 IGMP 快速离开 40
6.4.7 检查配置 40
6.4.8 维护 40

6.4.9 配置 IGMP 基本功能应用示例 41
6.5 PIM 43
6.5.1 简介 43
6.5.2 配置准备 43
6.5.3 PIM 的缺省配置 44
6.5.4 配置 PIM-DM 功能 45
6.5.5 配置 PIM-SM 功能 45
6.5.6 配置 PIM-SM 得 BSR 管理域功能 45
6.5.7 检查配置 46
6.5.8 维护 46
6.5.9 配置 PIM-DM 功能应用示例 46
6.5.10 配置 PIM-SM 功能应用示例 51
6.6 MLD 56
6.6.1 简介 56
6.6.2 配置准备 56
6.6.3 MLD 的缺省配置 57
6.6.4 配置 MLD 使能 57
6.6.5 配置 MLD 查询器参数 58
6.6.6 配置 MLD 快速离开 58
6.6.7 检查配置 58
6.6.8 维护 58
6.6.9 配置 MLD 基本功能应用示例 59
6.7 PIMv6 59
6.7.1 简介 59
6.7.2 配置准备 59
6.7.3 PIMv6 的缺省配置 59
6.7.4 配置 PIMv6-DM 功能 59
6.7.5 配置 PIM-SM 功能 60
6.7.6 配置 PIM-SM 得 BSR 管理域功能 60
6.7.7 检查配置 61
6.7.8 维护 61
6.7.9 配置 PIMv6-DM 功能应用示例 61
6.7.10 配置 PIM-SM 功能应用示例 61
7 OAM 1
7.1 简介 1
7.2 EFM 2
7.2.1 简介 2
7.2.2 配置准备 3
7.2.3 缺省配置 3

7.2.4 配置 EFM 基本功能 3
7.2.5 配置 EFM 端口环回功能 4
7.2.6 配置 EFM 链路性能监测功能 5
7.2.7 配置 EFM 链路故障检测功能 7
7.2.8 检查配置 7
7.2.9 维护 7
7.3 故障转移 8
7.3.1 简介 8
7.3.2 配置准备 8
7.3.3 故障转移功能的缺省配置 8
7.3.4 配置故障转移 9
7.3.5 检查配置 9
7.3.6 配置故障转移示例 10
7.4 VRRP 11
7.4.1 简介 11
7.4.2 配置准备 14
7.4.3 VRRP 的缺省配置 14
7.4.4 配置 VRRP 备份组 14
7.4.5 配置 VRRP6 备份组 16
7.4.6 配置 VRRP 的 Trap 功能 17
7.4.7 配置 VRRP 监视接口 18
7.4.8 配置 BFD for VRRP 18
7.4.9 检查配置 19
7.5 CFM 19
7.5.1 简介 19
7.5.2 配置准备 22
7.5.3 缺省配置 22
7.5.4 配置 CFM 基本功能 23
7.5.5 配置故障检测功能 24
7.5.6 配置故障确认功能 24
7.5.7 配置故障定位功能 25
7.5.8 配置告警抑制功能 25
7.5.9 配置单向丢包测试功能 26
7.5.10 配置双向时延测试功能 26
7.5.11 检查配置 27
7.5.12 配置 CFM 示例 27
8 安全性 1
8.1 ACL 1
8.1.1 简介 1

8.1.2 配置准备 2
8.1.3 配置 ACL 2
8.1.4 应用 ACL 4
8.1.5 配置统计 4
8.1.6 配置限速 5
8.1.7 配置时间段 5
8.1.8 检查配置 6
8.1.9 维护 6
8.1.10 配置 ACL 示例 6
8.2 AAA 8
8.2.1 简介 8
8.2.2 配置准备 9
8.2.3 缺省配置 9
8.2.4 配置 RADIUS 服务器 10
8.2.5 配置 TACACS+服务器 10
8.2.6 配置 AAA 服务器组 11
8.2.7 配置 AAA 方法 11
8.2.8 检查配置 12
8.2.9 配置 AAA 示例 12
8.2.10 检查结果 14
8.3 802.1x 14
8.3.1 简介 14
8.3.2 配置准备 17
8.3.3 802.1x 功能的缺省配置 17
8.3.4 配置 802.1x 基本功能 17
8.3.5 配置 802.1x 重认证 19
8.3.6 配置 802.1x 定时器 19
8.3.7 检查配置 19
8.3.8 配置 802.1x 示例 20
8.4 PPPoE+ 22
8.4.1 简介 22
8.4.2 配置准备 23
8.4.3 PPPoE+功能的缺省配置 23
8.4.4 配置 PPPoE+基本功能 24
8.4.5 配置 PPPoE+报文信息 24
8.4.6 检查配置 26
8.4.7 维护 26
8.4.8 配置 PPPoE+示例 27
8.5 安全 MAC 28
8.5.1 简介 28

8.5.2 配置准备 30
8.5.3 安全 MAC 功能的缺省配置 30
8.5.4 配置安全 MAC 基本功能 30
8.5.5 配置接口 Sticky 安全 MAC 地址 31
8.5.6 检查配置 32
8.5.7 维护 32
8.5.8 配置安全 MAC 示例 32
8.6 风暴抑制与风暴控制 34
8.6.1 简介 34
8.6.2 配置准备 35
8.6.3 风暴抑制和风暴控制的缺省配置 35
8.6.4 配置风暴抑制功能 36
8.6.5 检查风暴抑制配置 36
8.6.6 配置风暴抑制应用示例 37
8.6.7 配置风暴控制功能 38
8.6.8 检查风暴控制配置 39
8.6.9 配置风暴控制应用示例 39
8.7 ARP 防攻击 40
8.7.1 配置准备 40
8.7.2 ARP 防攻击缺省配置 40
8.7.3 配置 ARP 防攻击功能 40
8.7.4 检查配置 42
8.7.5 配置 ARP 防攻击示例 42
8.8 ND Snooping 44
8.8.1 简介 44
8.8.2 配置准备 44
8.8.3 ND Snooping 的缺省配置 45
8.8.4 配置 ND Snooping 45
8.8.5 检查配置 45
8.8.6 配置 ND Snooping 示例 46
8.9 DHCP Snooping 47
8.9.1 简介 47
8.9.2 配置准备 48
8.9.3 DHCP Snooping 的缺省配置 49
8.9.4 配置 DHCP Snooping 49
8.9.5 配置 DHCP Snooping 支持 Option 82 功能 50
8.9.6 配置 DHCPv6 Snooping 50
8.9.7 检查配置 51
8.9.8 维护 52

8.9.9 配置 DHCP Snooping 示例 52
8.10 IP Source Guard 54
8.10.1 简介 54
8.10.2 配置准备 55
8.10.3 IP Source Guard 功能的缺省配置 55
8.10.4 配置 IP Source Guard 绑定功能 56
8.10.5 配置 IP Source Guard 接口信任状态 56
8.10.6 检查配置 56
8.10.7 配置 IP Source Guard 示例 57
8.11 CPU 防攻击 58
8.11.1 配置准备 58
8.11.2 配置 CPU 防攻击限速功能 58
8.11.3 配置 CPU 防攻击的攻击溯源功能 59
8.11.4 检查配置 60
8.11.5 维护 61
8.11.6 MAC 认证 61
8.11.7 简介 61
8.11.8 缺省配置 61
8.11.9 配置 MAC 认证 62
8.11.10 配置举例 63
8.12 URPF 64
8.12.1 简介 64
8.12.2 配置准备 65
8.12.3 缺省配置 65
8.12.4 配置 URPF 65
8.12.5 检查配置 66
8.13 Timerange 1
8.13.1 简介 1
8.13.2 配置准备 1
8.13.3 缺省配置 1
8.13.4 配置 timerange 1
8.13.5 检查配置 2
8.14 DOS 防攻击 3
8.14.1 简介 3
8.14.2 配置准备 4
8.14.3 DOS 防攻击功能的缺省配置 4
8.14.4 配置畸形报文攻击防范 5
8.14.5 配置分片报文攻击防范 5
8.14.6 配置 TCP SYN 泛洪攻击防范 6
8.14.7 配置 UDP 泛洪攻击防范 6

8.14.8 配置 ICMP 泛洪攻击防范 6
8.14.9 检查配置 6
8.14.10 配置 DOS 防攻击示例 7
9 可靠性 1
9.1 链路聚合 1
9.1.1 简介 1
9.1.2 配置准备 2
9.1.3 缺省配置 2
9.1.4 配置手工链路聚合 3
9.1.5 配置静态 LACP 链路聚合 3
9.1.6 配置主备方式链路聚合 4
9.1.7 配置聚合组负载分担算法 5
9.1.8 检查配置 5
9.1.9 配置静态 LACP 方式的链路聚合示例 6
9.2 G.8031 8
9.2.1 简介 8
9.2.2 配置准备 9
9.2.3 G.8031 的缺省配置 9
9.2.4 创建 G.8031 保护组 10
9.2.5 配置 G.8031 倒换控制 10
9.2.6 检查配置 11
9.2.7 配置 G.8031 保护示例 11
9.3 G.8032 13
9.3.1 简介 13
9.3.2 配置准备 17
9.3.3 G.8032 的缺省配置 17
9.3.4 创建 G.8032 保护环 18
9.3.5 创建 G.8032 保护子环 19
9.3.6 配置 G.8032 倒换控制 21
9.3.7 检查配置 21
9.3.8 维护 21
9.3.9 配置单环 G.8032 保护示例 22
9.3.10 配置相交环 G.8032 保护示例 24
9.4 STP/RSTP 26
9.4.1 简介 26
9.4.2 配置准备 29
9.4.3 STP 的缺省配置 29
9.4.4 使能 STP 功能 30
9.4.5 配置 STP 参数 30

9.4.6 配置 RSTP 边缘接口 31
9.4.7 配置 RSTP 链路类型 31
9.4.8 检查配置 32
9.4.9 配置 STP 示例 32
9.5 MSTP 34
9.5.1 简介 34
9.5.2 配置准备 37
9.5.3 MSTP 的缺省配置 37
9.5.4 使能 MSTP 功能 38
9.5.5 配置 MST 域和 MST 域最大跳数 38
9.5.6 配置根桥/备份根桥 39
9.5.7 配置设备接口和系统的优先级 40
9.5.8 配置接口的路径开销 41
9.5.9 配置接口最大发送速率 41
9.5.10 配置 MSTP 定时器 41
9.5.11 配置边缘接口 42
9.5.12 配置 BPDU 过滤 43
9.5.13 配置 BPDU 保护 43
9.5.14 配置 STP/RSTP/MSTP 模式切换 44
9.5.15 配置链路类型 44
9.5.16 配置根接口保护 45
9.5.17 配置接口环路保护 45
9.5.18 配置端口 TC 报文抑制功能 46
9.5.19 配置 TC 保护功能 46
9.5.20 检查配置 47
9.5.21 维护 47
9.5.22 配置 MSTP 示例 47
9.6 环路检测 50
9.6.1 简介 50
9.6.2 配置准备 52
9.6.3 环路检测的缺省配置 52
9.6.4 配置环路检测功能 53
9.6.5 检查配置 53
9.6.6 配置环路检测内环应用示例 53
9.7 接口备份 55
9.7.1 简介 55
9.7.2 配置准备 57
9.7.3 接口备份的缺省配置 57
9.7.4 配置接口备份基本功能 58

9.7.5 配置接口强制倒换 59
9.7.6 检查配置 59
9.7.7 配置接口备份示例 59
9.8 接口隔离 61
9.8.1 简介 61
9.8.2 配置准备 62
9.8.3 接口隔离的缺省配置 62
9.8.4 配置接口隔离 62
9.8.5 检查配置 62
9.8.6 配置接口保护示例 63
9.9 L2CP 64
9.9.1 简介 64
9.9.2 配置准备 64
9.9.3 缺省配置 65
9.9.4 配置 L2CP 65
9.9.5 检查配置 66
9.9.6 配置 L2CP 示例 66
9.10 BFD 67
9.10.1 简介 67
9.10.2 配置准备 67
9.10.3 BFD 的缺省配置 67
9.10.4 配置 BFD 静态会话检测 67
9.10.5 配置 BFD 单臂回声会话 68
9.10.6 配置 BFD 检测三层 eth-trunk 68
9.10.7 配置 BFD 会话参数 68
9.10.8 检查配置 69
9.10.9 配置 BFD 单跳应用示例 69
9.11 链路震荡保护 70
9.11.1 简介 70
9.11.2 配置准备 70
9.11.3 链路震荡保护的缺省配置 71
9.11.4 配置链路震荡保护 71
9.11.5 检查配置 71
9.11.6 配置链路震荡保护应用示例 72
10 系统管理 1
10.1 SNMP 1
10.1.1 简介 1
10.1.2 配置准备 3
10.1.3 SNMP 的缺省配置 3

10.1.4 配置 SNMP v1/v2c 基本功能 4
10.1.5 配置 SNMP v3 基本功能 5
10.1.6 配置 SNMP 其他信息 6
10.1.7 配置 Trap 7
10.1.8 检查配置 8
10.1.9 配置 SNMP v1/v2c 和 Trap 示例 8
10.1.10 配置 SNMP v3 和 Trap 示例 10
10.2 RMON 12
10.2.1 简介 12
10.2.2 配置准备 13
10.2.3 RMON 的缺省配置 13
10.2.4 配置 RMON 统计功能 13
10.2.5 配置 RMON 历史统计功能 14
10.2.6 配置 RMON 告警组 14
10.2.7 配置 RMON 事件组 15
10.2.8 检查配置 15
10.2.9 维护 15
10.2.10 配置 RMON 告警组应用示例 16
10.3 LLDP 17
10.3.1 简介 17
10.3.2 配置准备 20
10.3.3 LLDP 的缺省配置 20
10.3.4 使能全局 LLDP 功能 20
10.3.5 使能接口 LLDP 功能 21
10.3.6 全局配置 LLDP 基本功能 21
10.3.7 接口配置 LLDP 基本功能 22
10.3.8 配置 LLDP 告警功能 22
10.3.9 配置 TLV 23
10.3.10 检查配置 24
10.3.11 维护 24
10.3.12 配置 LLDP 基本功能示例 25
10.4 端口镜像 27
10.4.1 简介 27
10.4.2 配置准备 28
10.4.3 接口镜像的缺省配置 28
10.4.4 配置接口镜像功能 28
10.4.5 检查配置 29
10.4.6 配置接口镜像应用示例 29
10.4.7 配置远端接口镜像应用示例 30

10.5 电缆诊断 31
10.5.1 简介 31
10.5.2 配置准备 32
10.5.3 配置电缆诊断功能 32
10.5.4 检查配置 32
10.6 UDLD 33
10.6.1 简介 33
10.6.2 配置准备 33
10.6.3 UDLD 功能的缺省配置 33
10.6.4 配置 UDLD 33
10.6.5 检查配置 34
10.7 光模块数字诊断 35
10.7.1 简介 35
10.7.2 配置准备 35
10.7.3 光模块数字诊断的缺省配置 35
10.7.4 配置使能光模块数字诊断 36
10.7.5 配置光模块数字诊断告警发送 Trap 36
10.7.6 检查配置 37
10.8 系统日志 37
10.8.1 简介 37
10.8.2 配置准备 38
10.8.3 系统日志的缺省配置 38
10.8.4 配置系统日志基本信息 39
10.8.5 配置系统日志输出 39
10.8.6 配置系统日志输出 TELNET/SSH 终端 39
10.8.7 检查配置 40
10.8.8 维护 41
10.8.9 配置系统日志输出到当前终端示例 41
10.9 告警管理 42
10.9.1 简介 42
10.9.2 配置准备 42
10.9.3 配置告警基本功能 43
10.9.4 检查配置 43
10.10 CPU 监控 43
10.10.1 简介 43
10.10.2 配置准备 44
10.10.3 CPU 监控的缺省配置 44
10.10.4 配置 CPU 监控告警 44
10.10.5 检查配置 45

10.11 内存监控 45
10.11.1 配置准备 45
10.11.2 配置内存监控 45
10.11.3 检查配置 46
10.12 Ping 46
10.12.1 简介 46
10.12.2 配置 Ping 功能 47
10.13 Trace 48
10.13.1 简介 48
10.13.2 配置 IPv4 Trace 功能 49
10.13.3 配置 IPv6 Trace 功能 49
10.14 硬件监控 49
10.14.1 简介 50
10.14.2 温度监控 50
10.14.3 电源监控 50
10.14.4 检查配置 51
10.15 风扇监控 51
10.15.1 简介 51
10.15.2 配置准备 51
10.15.3 配置风扇监控功能 51
10.15.4 检查配置 52
10.16 设备堆叠 52
10.16.1 简介 52
10.16.2 堆叠拓扑 53
10.16.3 堆叠缺省值 55
10.16.4 配置堆叠 55
10.16.5 配置举例 56
10.17 MAD 58
10.17.1 简介 58
10.17.2 配置准备 59
10.17.3 配置 MAD 60
10.17.4 配置举例 60
10.18 NQA 62
10.18.1 简介 62
10.18.2 配置准备 62
10.18.3 NQA 的缺省配置 62
10.18.4 配置 ICMP-echo 测试 62
10.18.5 配置 UDP-echo 测试 63
10.18.6 配置 TCP 测试 64
10.18.7 配置 DNS 测试 65

10.18.8 配置 HTTP 测试 65
10.18.9 配置 FTP 测试 66
10.18.10 配置 SNMP 测试 67
10.18.11 配置测试历史记录功能 67
10.18.12 配置测试统计功能 68
10.18.13 配置测试告警功能 68
10.18.14 检查配置 68
10.18.15 维护 69
10.18.16 配置 ICMP-echo 测试功能示例 69
10.19 PATCH 补丁功能 71
10.19.1 简介 71
10.19.2 配置准备 71
10.19.3 加载补丁 71
10.19.4 激活补丁 71
10.19.5 去激活补丁 72
10.19.6 删除补丁 72
10.19.7 维护 72
10.19.8 PATCH 示例 72
10.20 定时备份配置功能 73
10.20.1 简介 73
10.20.2 配置准备 73
10.20.3 配置 timerange 73
10.20.4 配置自动上传 74
10.20.5 检查配置 75
10.20.6 定时备份配置示例 75
11 L3VPN 业务 1
11.1 L3VPN 基础配置 1
11.1.1 简介 1
11.1.2 配置准备 3
11.1.3 创建 VPN 实例 3
11.1.4 配置 VLAN 接口绑定 VPN 实例 4
11.1.5 检查配置 4
11.1.6 配置 VLAN 接口绑定 VPN 示例 4
12 附录 1
12.1 术语 1
12.2 缩略语 5

图目录

图 1-1 通过 PC 连接 Console 口登录设备的组网示意图 3
图 1-2 “超级终端”中的通信参数配置示意图 3
图 1-3 交换机作为 Telnet Server 设备的组网示意图 5
图 1-4 交换机设备作为 Telnet Client 设备的组网示意图 5
图 1-5 NTP 基本原理 17
图 1-6 NTP 组网示意图 23
图 2-1 MAC 地址表转发示意图 2
图 2-2 MAC 应用组网示意图 7
图 2-3 VLAN 划分示意图 9
图 2-4 VLAN 和接口保护组网示意图 16
图 2-5 IP 电话单独接入交换机组网示意图 19
图 2-6 基本 QinQ 典型组网示意图 24
图 2-7 基本 QinQ 应用组网示意图 28
图 2-8 灵活 QinQ 应用组网示意图 30
图 2-9 VLAN 转换原理组网示意图 31
图 2-10 VLAN 转换应用组网示意图 34
图 2-11 MVRP 原理示意图 37
图 2-12 MVRP 应用组网示意图 39
VTP 原理示意图 42
图 2-13 VTP 应用组网示意图 44
图 3-1 配置 VLAN 接口组网示意图 3
图 3-2 配置 ARP 组网示意图 9
图 3-3 NDP 地址解析原理示意图 10
图 3-4 配置静态路由组网示意图 14
图 3-5 配置策略路由组网示意图 16

图 3-6 AS_PATH 属性 20
图 3-7 NEXT_HOP 属性 21
图 3-8 MED 属性 22
图 3-9 LOCAL_PREF 属性 23
图 3-10 BGP 路由反射器示意图 25
图 3-11 BGP 路由反射器示意图 26
图 3-12 BGP 联盟示意图 27
图 3-13 BGP 基本配置组网图 33
图 3-14 BGP 与 IGP 交互配置组网图 37
图 3-15 BGP 路由聚合组网图 39
图 3-16 BGP 团体组网图 42
图 3-17 配置 BGP 路由反射器的组网图 45
图 3-18 配置联盟组网图 47
图 3-19 配置 BGP 路径选择的组网图 51
图 3-20 OSPF 区域划分 57
图 3-21 OSPF 路由器类型示例 58
图 3-22 OSPF 邻居状态机 60
图 3-23 OSPF 接口状态机 61
图 3-24 OSPF 广播网络建邻示意图 63
图 3-25 OSPFNSSA 区域示意图 66
图 3-26 OSPF 基本配置组网图 75
图 3-27 OSPF STUB 区域示例组网图 78
图 3-28 OSPF NSSA 区域示例组网图 80
图 3-29 OSPF 引入外部路由组网图 83
图 3-30 OSPF 路由聚合组网图 85
图 3-31 配置 OSPF 虚连接的组网图 89
图 3-32 配置 OSPF 虚连接的组网图 93
图 3-33 OSPFV3 区域划分 97
图 3-34 OSPFV3 路由器类型示例 98
图 3-35 OSPFV3 邻居状态机 99
图 3-36 OSPFV3 接口状态机 101
图 3-37 OSPFV3 广播网络建邻示意图 103

图 3-38 OSPFV3 NSSA 区域示意图 106
图 3-39 OSPFV3 基本配置组网图 113
图 3-40 OSPFV3 STUB 区域示例组网图 117
图 3-41 OSPFV3 NSSA 区域示例组网图 119
图 3-42 OSPV3F 引入外部路由组网图 123
图 3-43 OSPFV3 路由聚合组网图 125
图 3-44 配置 OSPFV3 虚连接的组网图 130
图 3-45 P2P 链路上 LSDB 数据库的同步过程 141
图 3-46 P2P 链路上 LSDB 数据库的同步过程 142
图 3-47 RIP 基本功能组网图 177
图 3-48 RIP 引入外部路由配置组网图 181
图 3-49 RIPng 基本功能组网图 187
图 3-50 RIPng 引入外部路由组网图 192
图 3-51 ipv4 路由引入中应用路由策略组网图 200
图 3-52 ipv6 路由引入中应用路由策略组网图 204
图 4-1 DHCP 典型应用组网示意图 2
图 4-2 DHCP 报文结构示意图 2
图 4-3 DHCP 客户端组网示意图 4
图 4-4 配置 DHCP 客户端组网示意图 6
图 4-5 DHCP Server 和 DHCP Client 应用组网示意图 8
图 4-6 DHCP 报文结构示意图 8
图 4-7 配置 DHCP 服务器组网示意图 12
图 4-8 DHCP Relay 工作原理示意图 14
图 4-9 配置 DHCP 中继组网示意图 17
图 5-1 SP 调度示意图 10
图 5-2 WRR 调度示意图 10
图 5-3 WFQ 调度示意图 11
图 5-4 配置队列调度组网示意图 14
图 5-5 配置基于接口的流量限速组网示意图 20
图 6-1 组播方式传输信息示意图 2
图 6-2 组播基本概念在网络中相应位置标示示意图 3
图 6-3 IPv4 组播地址和组播 MAC 地址的映射关系 4

图 6-4 IGMP 和二层组播特性运行位置示意图 5
图 6-5 IGMP Snooping 应用场景 7
图 6-6 IGMP Snooping 基本功能组网应用示意图 13
图 6-7 IGMP Snooping 静态成员功能组网应用示意图 14
图 6-8 IGMP Snooping 组播复制功能组网应用示意图 16
图 6-9 IGMP Snooping Proxy 功能组网应用示意图 18
图 6-10 IGMP Snooping 组播策略功能组网应用示意图 20
图 6-11 MLD Snooping 应用场景 23
图 6-12 MLD Snooping 基本功能组网应用示意图 28
图 6-13 MLD Snooping 静态成员功能组网应用示意图 30
图 6-14 MLD Snooping 组播复制功能组网应用示意图 32
图 6-15 MLD Snooping Proxy 功能组网应用示意图 34
图 6-16 MLD Snooping 组播策略功能组网应用示意图 36
图 6-17 IGMP 在组播网络中的部署位置 38
图 6-18 IGMP 基本功能功能组网应用示意图 41
图 6-19 PIM 在组播网络中的部署位置 44
图 6-20 PIM-DM 功能组网应用示意图 47
图 6-21 PIM-SM 功能功能组网应用示意图 52
图 7-1 OAM 环回示意图 2
图 7-2 故障转移应用组网示意图 10
图 7-3 VRRP 原理示意图 12
图 7-4 VRRP 负载分担原理示意图 13
图 7-5 不同级别 MD 网络示意图 20
图 7-6 不同级别 MD 网络示意图 21
图 7-7 CFM 典型配置组网图 28
图 8-1 ACL 示例 7
图 8-2 分域认证应用组网示意图 13
图 8-3 802.1x 认证体系结构 15
图 8-4 802.1x 应用组网示意图 20
图 8-5 用户通过 PPPoE 认证连接网络示意图 22
图 8-6 PPPoE+应用组网示意图 27
图 8-7 安全 MAC 应用组网示意图 33

图 8-8 风暴抑制应用组网示意图 37
图 8-9 动态 ARP 检测应用组网示意图 43
图 8-10 配置 ND Snooping 组网示意图 46
图 8-11 DHCP Snooping 组网示意图 48
图 8-12 配置 DHCP Snooping 组网示意图 53
图 8-13 IP Source Guard 功能示意图 55
图 8-14 IP Source Guard 应用组网示意图 57
图 8-15 MAC 认证组网示意图 63
图 8-16 DOS 防攻击应用组网示意图 7
图 9-1 静态 LACP 方式链路聚合应用组网示意图 6
图 9-2 G.8031 组网示意图 11
图 9-3 G.8032 环网示意图 13
图 9-4 环网空闲状态 15
图 9-5 环网保护状态 15
图 9-6 子环模型 16
图 9-7 单环 G.8032 应用组网示意图 22
图 9-8 相交环 G.8032 应用组网示意图 24
图 9-9 网络环路造成网络风暴示意图 27
图 9-10 运行 STP 协议的环网示意图 28
图 9-11 RSTP 协议造成 VLAN 报文无法转发示意图 29
图 9-12 STP 应用组网示意图 32
图 9-13 MSTP 网络基本概念示意图 35
图 9-14 MSTI 概念示意图 36
图 9-15 MST 域内多生成树实例组网示意图 37
图 9-16 MSTP 应用组网示意图 48
图 9-17 环路类型示意图 51
图 9-18 环路检测内环应用组网示意图 54
图 9-19 接口备份原理示意图 56
图 9-20 接口备份在不同 VLAN 上的应用原理示意图 57
图 9-21 接口备份应用组网示意图 60
图 9-22 接口保护组网示意图 63
图 9-23 L2CP 拓扑图 64

图 9-24 BFD 单跳应用 69
图 10-1 SNMP 工作机制示意图 2
图 10-2 SNMP v3 认证机制示意图 5
图 10-3 SNMP v1/v2c 组网示意图 8
图 10-4 SNMP v3 和 Trap 组网示意图 10
图 10-5 RMON 应用示意图 12
图 10-6 RMON 典型应用组网示意图 16
图 10-7 LLDPDU 结构图 18
图 10-8 基本 TLV 结构图 18
图 10-9 配置 LLDP 基本功能组网示意图 25
图 10-10 接口镜像功能原理示意图 27
图 10-11 接口镜像应用组网示意图 30
图 10-12 接口远端镜像应用组网示意图 30
图 10-13 系统日志输出到日志主机组网示意图 41
图 10-14 Ping 功能实现原理组网示意图 47
图 10-15 Trace 功能实现原理组网示意图 48
图 10-16 3 台成员设备组成链型拓扑 54
图 10-17 3 台成员设备组成环形型拓扑 54
图 10-18 3 台堆叠拓扑图 56
图 10-19 MAD 直连模式 59
图 10-20 MAD 代理模式 59
图 10-21 配置 ICMP-echo 测试功能组网示意图 69
图 11-1 部署 MCE 前的组网 2
图 11-2 部署 MCE 后的组网 3

表格目录

表 2-1 接口模式与报文转发 9
表 4-1 DHCP 报文字段含义列表 2
表 4-2 DHCP 报文字段含义列表 8
表 5-1 缺省情况下 8021p 入方向和本地优先级及颜色的映射关系 4
表 5-2 缺省情况下 8021p 出方向和本地优先级及颜色的映射关系 4
表 5-3 缺省情况下 DSCP 入方向和本地优先级及颜色的映射关系 5
表 5-4 缺省情况下 DSCP 出方向和本地优先级及颜色的映射关系 6
表 10-1 TLV 类型 18
表 10-2 IEEE 802.1 组织定义的 TLV 19
表 10-3 IEEE 802.3 组织定义的 TLV 19
表 10-4 信息级别 37

1 基础配置

本章介绍交换机设备的基础配置信息及配置过程，并提供相关的配置案例。

- 登录设备
- 加载与升级
- 时间管理
- 接口管理

1.1 登录设备

1.1.1 简介

登录交换机设备进行配置和管理，可以采用 CLI（Command-Line Interface，命令行界面）方式、Web 方式。

交换机命令行方式下有多种配置方式：

- Console 方式：第一次配置时必须采用 Console 方式，甲信设备支持 RJ45、M12、Micro USB 和 Mini-USB 类型的 Console 口。
- Telnet 方式：设备默认 IP 地址为 192.168.0.1。如需修改 IP 地址，需要先通过 Console 方式登录，在设备上配置 IP 地址，以及设置用户名和密码，再使用新 IP 地址进行远程 Telnet 配置。
- SSH 方式：在通过 SSH 登录设备之前，需要先通过 Console 接口登录设备并启动 SSH 服务。

当需要在 Web 方式下配置时，也必须先通过命令行方式，配置 VLAN 接口 IP 地址，然后才可以通过 NView NNM 网管平台对设备进行配置。

1.1.2 通过 Console 口登录设备

说明

- 设备均支持 RJ45 类型的 Console 口。
- 设备使用黑色线序 Console 线缆，如不确定，请查看该设备系列对应的《用户手册》或《产品描述》手册，或咨询我司技术人员。
- 以下均以 RJ45 类型 Console 口为例进行说明。

简介

Console 口是网络设备用来与运行终端仿真程序的 PC 进行连接的常用接口，用户可以借助此接口对本地设备进行配置和管理。这种管理方式不需借助网络进行通信，所以被称为带外（out-of-band）管理方式，在网络运行异常的情况下，用户也可以通过 Console 口对设备进行配置和管理。

在以下两种情况中，只能通过 Console 口登录设备进行配置：

- 设备第一次加电启动
- 无法通过 Telnet 方式登录设备

缺省配置

设备上 Console 口的缺省配置如下。

功能	缺省值
传输速率	115200
流控方式	无流控
验证方式	不验证
停止位	1
数据位	8

通过 Console 口登录

当用户希望通过 PC 连接 Console 口登录设备时，首先需要通过配置线缆将设备的 Console 口和 PC 的 RS-232 串口相连，如图 1-1 所示，然后在 PC 上运行终端仿真程序，如微软公司的 Windows XP 操作系统自带的“超级终端”程序，将通信参数如图 1-2 配置，完成后即可登录设备。

图 1-1 通过 PC 连接 Console 口登录设备的组网示意图

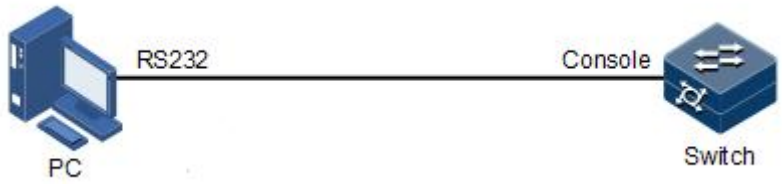


图 1-2 “超级终端”中的通信参数配置示意图

说明

初始情况下串口波特率为 115200。

配置 Console 口波特率

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**configure** JX(config)#**line console** JX(config-line-console)#**baudrate { 115200 | 9600 }**	修改串口登录波特率。

配置 Console 密码和认证方式

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**configure** JX(config)#**line console password**	设置串口密码。
2	JX(config)#**line console** JX(config-line-console)#**login authentication { local | password | none}**	设置串口认证方式 local：用户名密码 password：串口密码 none：不认证

检查配置

配置完成后，请在设备上执行以下命令检查配置结果。

序号	检查项	说明
1	JX#**show line console information**	查看串口波特率配置信息。

1.1.3 通过 Telnet 登录设备

简介

说明

初始情况下，设备缺省管理 IP 地址为 192.168.0.1，子网掩码为 255.255.255.0。如需修改设备 IP 地址，用户可以通过 Console 口登录设备，并对设备进行配置。设备的缺省用户名和密码均为 JX。Telnet 连接状态下输错 3 次密码自动断开连接。

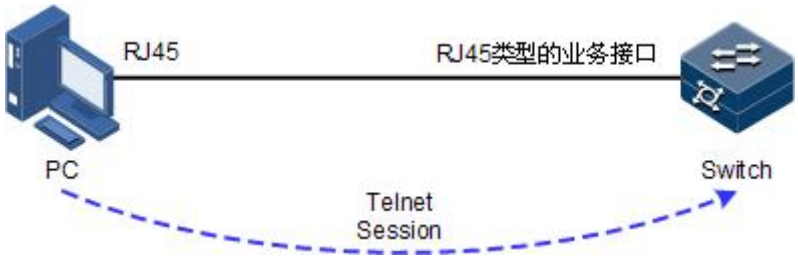
Telnet 提供了一种通过 PC 远程登录设备的方式。用户可以先通过 PC 登录到一台网络设备，然后再通过 Telnet 方式远程登录到联网的其他网络设备，而不需要为每一台网络设备都连接一台 PC。

有 SNMP 接口的设备，需使用 SNMP 接口进行 telnet 登录。无 SNMP 接口的设备，可以使用任意接口进入管理 VLAN 进行 telnet 登录。

交换机设备提供的 Telnet 服务包括：

- Telnet Server：用户在 PC 上运行 Telnet 客户端程序登录到设备，对设备进行配置管理。如下图所示，交换机此时提供的是 Telnet Server 服务。

图 1-3 交换机作为 Telnet Server 设备的组网示意图

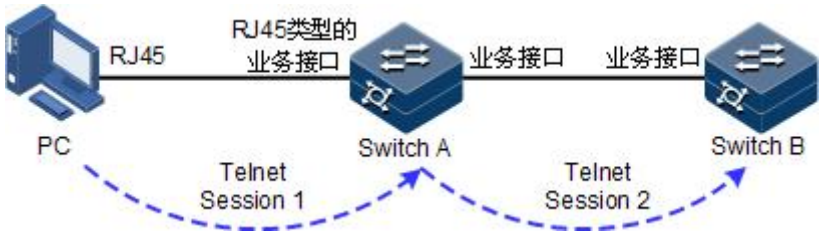


说明

设备支持 Telnet 最大用户数为 10 个。

- Telnet Client：用户在 PC 上通过终端仿真程序或 Telnet 客户端程序建立与设备的连接后，再通过 telnet 命令登录到其它设备，对其进行配置管理。如下图所示，Switch A 此时既作为 Telnet Server，也同时提供 Telnet Client 服务。

图 1-4 交换机设备作为 Telnet Client 设备的组网示意图



缺省配置

设备上 Telnet 服务器功能的缺省配置如下。

功能	缺省值
Telnet 服务器功能状态	使能
Telnet 服务器监听端口号	23
使能 Telnet 服务器功能的接口	所有接口
最大 Telnet 连接数	10

说明

通过 Telnet 配置设备时，建议不要频繁修改设备的 IP 地址。修改 IP 地址可能导致当前 Telnet 连接断开，需根据新的 IP 地址重新建立 Telnet 连接。

配置 Telnet 服务器

在通过 Telnet 登录设备之前，用户需要通过 Console 接口登录设备并启动 Telnet 服务，请在设备上进行以下配置。

步骤	配置	说明
1	JX#**configure**	进入全局配置模式。
2	JX(config)#**interface vlan** *vlan-id* JX(config)#**interface ge 1/0/1**	进入 VLAN 接口或带外接口配置模式，以下使用进入 VLAN 接口为例。
3	JX(config-vlan*)#**ip address** *ip-address* [*ip-mask*] [**sub**] JX(config-vlan*)#**ipv6 address** *ipv6-address/prefix-length* [**eui-64**] JX(config-vlan*)#**exit**	配置设备 IP 地址。
4	JX(config)#**telnet server start**	开启设备 Telnet Server 功能。
5	JX(config)#**telnet server stop**	断开指定的 Telnet 连接。
6	JX(config)#**telnet-ipv6 server start**	开启设备 IPv6 Telnet Server 功能。
7	JX(config)#**telnet-ipv6 server stop**	断开指定的 IPv6 Telnet 连接。

配置 Telnet 服务器端口号

Telnet 缺省端口号为 23，可进行以下配置修改端口号。

步骤	配置	说明
1	JX#**configure**	进入全局配置模式。
2	JX(config)#**telnet server port** {*port-number* | **default**}	修改 Telnet IPv4 服务端口号。
3	JX(config)#**telnet-ipv6 server start**	开启设备 IPv6 Telnet Server 功能。
4	JX(config)#**telnet-ipv6 server port** {*port-number* | **default**}	修改 Telnet IPv6 服务端口号。

配置 Telnet 客户端

请在作为 Telnet Client 的设备上进行以下配置。

步骤	配置	说明
1	JX#**telnet** *ipv4-address* [**-p** *port-id* | **-s** *source-ipv4-address*] JX#**telnet-ipv6** *ipv6-address* [**-p** *port-id* | **-s** *source-ipv6-address*]*	以 Telnet 方式登录其他设备。

检查配置

配置完成后，请在设备上执行以下命令检查配置结果。

序号	检查项	说明
1	JX#**show running-config**	查看 Telnet Server 的配置情况。

1.1.4 通过 SSH 登录设备

简介

Telnet 缺少安全的认证方式，而且传输过程采用 TCP（Transmission Control Protocol，传输控制协议）进行明文传输，存在很大的安全隐患。单纯提供 Telnet 服务容易招致 DoS（Deny of Service，拒绝服务）、主机 IP 地址欺骗、路由欺骗等恶意攻击。

传统的 Telnet 和 FTP（File Transfer Protocol，文件传输协议）通过明文传送密码和数据的方式，已经慢慢不被用户所接受。SSH 是一个网络安全协议，通过对网络数据的加密，可以有效防止远程管理过程中的信息泄露问题，在网络环境中为远程登录和其他网络服务提供了更高的安全性。

SSH 通过 TCP 进行数据交互，它在 TCP 之上构建了一个安全的通道。另外，SSH 服务除了支持标准端口 22 以外，还支持其他服务端口，以防止设备受到来自网络的非法攻击。

在通过 SSH 登录设备之前，用户需要通过 Console 接口登录设备并启动 SSH 服务。

设备支持基于密码和公钥两种认证方式。

- 基于密码认证方式：与登录设备认证使用同一数据库。SSH 客户端只需输入用户名和密码，就可以登录到远程 SSHv2 服务器。所有传输的数据都会被加密，但是可能会有其他服务器伪冒真正的服务器，无法避免伪冒服务器攻击。
- 基于公钥认证方式：SSHv2 客户端除需要输入用户名和密码外，还需要依靠密钥进行认证。登录前在 SSHv2 客户端创建一对密钥，包括主机公钥和主机私钥，并将公钥存入 SSHv2 服务器。登录认证过程和传输的数据都会被加密，避免了伪冒服务器攻击。

缺省配置

设备上 SSH 登录设备的缺省配置如下。

功能	缺省值
SSH 服务器功能状态	禁止
本地 SSH 密钥对长度	512bit
密钥重协商周期	0h
SSH 采用的认证方式	password
SSH 认证超时时间	600s
SSH 侦听端口号	22
SSH 会话功能状态	禁用
SSH 协议版本	v2
SSH 安全算法模式	禁止

配置 SSH 服务器

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**ssh server start**	开启设备 IPv4 SSH Server 功能
3	JX(config)#**ssh server stop**	关闭设备 IPv4 SSH Server 功能
2	JX(config)#**ssh-ipv6 server start**	开启设备 IPv6 SSH Server 功能
3	JX(config)#**ssh-ipv6 server stop**	关闭设备 IPv6 SSH Server 功能

检查配置

配置完成后，请在设备上执行以下命令检查配置结果。

序号	检查项	说明
1	JX#**show ssh config**	查看 SSH 配置信息。

1.1.5 通过 Web 登录设备

简介

为了方便用户对设备进行配置和维护，设备支持 Web 网管功能。用户可以利用 Web 网管在图形界面下直观的管理和配置设备。

Web 网管支持以下两种文本传输协议：

- HTTP（Hypertext Transfer Protocol，超文本传输协议）：用来在网络上传递 Web 页面信息。在设备上使能 HTTP 服务功能后，用户就可以通过 HTTP 协议登录设备，并利用 Web 界面访问、控制设备。
- HTTPS（Secure Hypertext Transfer Protocol，安全的超文本传输协议）：利用 SSL（Secure Sockets Layer，安全套接层）协议保证合法客户端可以安全访问设备。客户端与设备之间交互的数据需要经过加密，保证了数据传输的安全性和完整性，从而实现对设备的安全管理。

配置 Web 网管功能后，远程用户可以通过 Web 浏览器登录设备并对其进行管理。如果禁止 Web 网管功能，则断开所有已经建立的 HTTP 连接。

缺省配置

设备上 Web 网管的缺省配置如下。

功能	缺省值
HTTP 功能状态	使能

配置 Web 网管功能

请在设备上进行以下配置。

步骤	配置	说明
1	`JX#configure`	进入全局配置模式。
2	`JX(config)#http server { start | stop }`	使能 HTTP 功能，使用 **stop** 格式禁止该功能。
3	`JX(config)#https server { start | stop }`	使能 HTTPS 功能，使用 **stop** 格式禁止该功能。

检查配置

配置完成后，请在设备上执行以下命令检查配置结果。

序号	检查项	说明
1	JX#show running-config	查看设备配置信息。

1.1.6 管理登录用户

简介

第一次启动交换机设备时，用户只要将 PC 通过 Console 接口与设备连接，在超级终端中输入初始的用户名和密码，即可以登录设备并对其进行配置。

说明

初始情况下，设备的用户名为 JX，密码为 JX。

如果为设备的业务接口配置了 IP 地址，在没有任何权限控制的情况下，任意远端用户都可以通过 Telnet 方式登录设备，或者通过与设备建立 PPP（Point to Point Protocol，点对点协议）连接来访问网络，这显然对设备和网络都是不安全的。为此需要为设备创建用户并设置密码和权限，对登录用户进行管理。

缺省配置

设备上用户管理的缺省配置如下。

功能	缺省值
本地用户信息	• 用户名：JX • 密码：JX • 用户权限：15
新建用户权限	15
新建用户激活状态	**active**
新建用户服务类型	console、telnet、ssh、ftp、http
密码复杂度	**3**
密码复杂度	**3**
用户名复杂度	**1**
用户名最大长度	**64**
密码最大长度	**64**
缺省支持特殊字符	**`~!@#$%^&*()_-+={}[]\:;'<>",./**

配置本地用户管理

请在设备上进行以下配置。

步骤	配置	说明
1	JX(config)#**username** *user-name* **password** { **cipher** | **reversible-cipher** | **plain** } *password* **group** { **administrators** | **operators** | **users** | **guests** } [**domain** [**telnet** | **ssh** | **http** | **ftp** | **console** | **default** | **all**]]	创建或修改登录用户的用户名和密文形式密码。
2	JX(config)#**username** *user-name* **domain** [**telnet** | **ssh** | **http** | **ftp** | **console** | **default** | **all**]*	修改用户组域。
3	JX(config)#**username** *user-name* **group** { **administrators** | **operators** | **users** | **guests** }	修改用户组。
4	JX(config)#**user password-complex** { *complex* | **default** }	配置用户密码复杂度。
4	JX(config)#**user password-length** { *length* | **default** }	配置用户密码最小长度。
5	JX(config)#**user name-complex** { *complex* | **default** }	配置用户名复杂度，默认值为 1。
6	JX(config)#**user name-length** { *length* | **default** }	配置用户名长度，默认值为 1。
7	JX(config)#**username** *user-name* { **login-lock** | **manual-lock** | **unlock** } [**reauth-interval** { *interval* | **default** } | **fail-count** { *count* | **default**]*	配置用户锁定类型为 登录锁定（login-lock）时可以配置登录失败次数和重认证间隔。
8	JX#**show user name** *user-name*	查看配置用户的信息
9	JX#**user special-characters** *CHARLIST*	配置用户名称和用户密码可包含的特殊字符

- 除缺省用户外，设备最多可再创建 30 个本地用户。
- 用户启用登录锁定（login-lock）时可配置最大登录失败次数和重认证间隔，默认失败次数上限是 3 次，重认证时间间隔为 10s。当登录失败达上限并且在静默时间内设备处于登录锁定，此时无法登录。超过静默时间，解除锁定，也可通过 unlock 手动解除锁定。
- 用户启动手工锁定（manual-lock）将会一直锁定，不受失败次数和重认证间隔限制。可通过 unlock 手动解除锁定

1.1.7 用户密码恢复

简介

当用户忘记设备登录用户密码时，用户只要将 PC 通过 Console 接口与设备连接，在超级终端中进入临时密码视图，获取临时密码序列号，可通过临时密码序列号生成临时密码。临时密码校验成功登录设备后，需要用户再次修改本地用户的密码。

密码恢复操作只能在 Console 终端进行。在登录界面输入快捷键“CTRL+P”进入临时密码视图。

临时密码恢复功能缺省开启状态。

配置临时密码恢复

请在设备上进行以下配置。

步骤	配置	说明
1	快捷键 CTRL+P	在 Console 登录界面键入快捷键
2	JX(key)#**get temporary-password-serial**	获取临时密码序列号
3	JX(key)#**check temporary-password** *password*	校验临时密码
4	JX(config)#**temporary-password** { enable | disable }	开启关闭临时密码恢复功能。

1.1.8 终端属性设置

简介

用户可以设置 Console，Telnet，SSH 终端的超时时间，屏幕逐页滚动控制，终端颜色，大小写敏感，交互模式等终端属性。

配置终端属性

请在设备上进行以下配置。

步骤	配置	说明
1	JX(config)#**terminal length** { 0 | *length* | default }	设置屏幕逐页滚动的条数。 0：不逐页滚动，直接全部输出 缺省逐页滚动为 24 行
2	JX(config)#**terminal timeout** { 0 | *timeout* }	设置终端超时时间。 0：终端不超时 缺省超时时间 10 分钟
3	JX(config)#**terminal monitor** JX(config)#**no terminal monitor**	开启关闭终端信息监控
4	JX(config)#**terminal mmi-mode** { enable | disable }	开启关闭设备人机交互模式。
5	JX(config)#**case-sensitive** { enable | disable }	开启关闭输入大小写字母敏感功能

1.1.9 Bootrom 密码设置

简介

用户可以设置 bootrom 密码。

配置终端属性

请在设备上进行以下配置。

步骤	配置	说明
1	JX(config)#**bootrom password** *PASSWORD*	设置 bootrom 密码，密码字符串
2	JX(config)#**no bootrom password**	清除 bootrom 密码
3	JX(config)#**show bootrom password**	查看 bootrom 密码

1.2 加载与升级

1.2.1 简介

加载

传统的配置文件加载方式为串口加载，该方式加载速度慢、耗时长、不具备远程加载功能，导致操作很不方便。为了解决这些问题，引入了 FTP 加载方式、TFTP 加载方式等。

设备提供多种方法用于确定设备在 TFTP/FTP 服务器上的配置文件名称，比如手动输入、使用 DHCP 客户端获取、使用默认的配置文件名。除此之外，用户还可以指定某种配置文件命名规则，根据规则使用设备自身的属性（例如设备型号、MAC 地址、软件版本号）确定与指定设备对应的配置文件名称。

升级

当需要为设备增加新特性、优化原有功能或解决当前软件版本的 BUG 时，可以对设备进行升级。

设备支持以下升级方式：

- 命令行升级方式

1.2.2 通过 TFTP 命令行升级方式升级系统软件

在通过命令行升级方式升级系统软件前，需要首先搭建 TFTP 环境，PC 作为 TFTP 服务器，交换机设备作为客户端，基本要求如下：

- 将 TFTP 服务器网口和交换机接口用网线连接，交换机设备默认 IP 地址为 192.168.0.1。
- 配置 TFTP 服务器端，确保服务器处于可用状态。
- 配置 TFTP 服务器的 IP 地址，使之与设备的 IP 地址处于同一网段，使设备可以访问服务器。

通过命令行升级方式升级系统软件的步骤如下：

步骤	配置	说明
1	JX#{**tftp | tftp-ipv6 } get** {*ipv4-address | ipv6-address* } *remote-file-name* [**localfile** *local-file-name*]	通过 TFTP 协议下载系统启动软件。支持 IPv6 地址。
2	JX#**upgrade os [localfile** *local-file-name*]	升级系统软件。
3	JX#**reboot**	重新启动设备。

1.2.3 通过 FTP 命令行升级方式升级系统软件

在通过命令行升级方式升级系统软件前，需要首先搭建 FTP 环境，PC 作为 FTP 服务器，交换机设备作为客户端，基本要求如下：

- 将 FTP 服务器网口和交换机接口用网线连接，交换机设备默认 IP 地址为 192.168.0.1。
- 配置 FTP 服务器端，确保服务器处于可用状态。
- 配置 FTP 服务器的 IP 地址，使之与设备的 IP 地址处于同一网段，使设备可以访问服务器。

通过命令行升级方式升级系统软件的步骤如下：

步骤	配置	说明
1	JX#{**ftp | ftp-ipv6 } get** {*ipv4-address | ipv6-address* } *user-name user-password remote-file-name* [**localfile** *local-file-name*]	通过 FTP 协议下载系统启动软件。支持 IPv6 地址。
2	JX#**upgrade os [localfile** *local-file-name*]	升级系统软件。
3	JX#**reboot**	重新启动设备。

1.2.4 检查配置

配置完成后，请在设备上执行以下命令检查配置结果。

序号	检查项	说明
1	JX#**show startup-config**	查看设备启动时加载的配置信息。
2	JX#**show running-config**	查看设备的当前配置信息。
3	JX#**show version**	查看系统的版本信息。

1.2.5 指定系统启动镜像

指定启动镜像为主或备镜像，请在设备上执行以下命令。

序号	检查项	说明
1	JX#**boot os {main | backup}**	指定系统启动镜像为主或备镜像，设置完成后下一次启动生效。仅设备支持双镜像时可选择，缺省系统从主镜像启动。

1.2.6 检查系统版本信息

查看系统版本信息，请在设备上执行以下命令查看结果。

序号	检查项	说明
1	JX#show os-package	查看系统版本信息，包括主备镜像版本信息、版本大小、编译时间等。

1.3 时间管理

1.3.1 简介

随着互联网在社会各个方面的发展和延伸，网络上多种涉及时间的应用，如网上实时交易、分布性的网络计算和处理、交通航班航路管理、数据库管理等，都需要精确、可靠的时间。

为了保证设备系统时间的精准，设备提供了完善的时间管理功能,，包括手动配置系统时间和时区、手动配置夏令时、NTP 功能以及 SNTP 功能。

时间和时区

通常情况下设备时间配置为设备所在地的实时时间，将时区配置为以格林尼治标准时间为基础的所在地时区（如中国北京在格林尼治为基准的东八区，即配置为+08:00）。

设备支持“年月日时分秒”的时间显示和时区偏移显示，可手动配置设备的时间和时区。

夏令时

夏令时（DST，Daylight Saving Time）是一种为节约资源而人为规定地方时间的制度。一般在夏季人为将时间调整提前一小时，可以使人早起早睡以减少照明量。但各个国家对夏令时的具体规定不同，所以在配置夏令时前需要考虑当地的具体情况。

设备支持配置夏令时开始的时间和结束的时间，以及调整时间的偏移量。

NTP

NTP（Network Time Protocol，网络时间协议）是用于互联网中时间同步的标准互联网协议，用于快速使网络内所有具有时钟的设备进行时钟同步。NTP 基于 UDP 进行传输，使用的端口号是 123，保证较高的精度（误差在 10ms 左右）。

NTP 基本原理如下图所示，时钟同步的工作过程如下：

步骤 1 Switch A 发送一个 NTP 消息包给 Switch B，该消息包带有离开 Switch A 时的时间戳，该时间戳为 10:00:00am，记为 t1。

步骤 2 当此 NTP 消息包到达 Switch B 时，Switch B 加上自己的时间戳，该时间戳为 11:00:01am，记为 t2。

步骤 3 当此 NTP 消息包离开 Switch B 时，Switch B 再加上自己的时间戳，该时间戳为 11:00:02am，记为 t3。

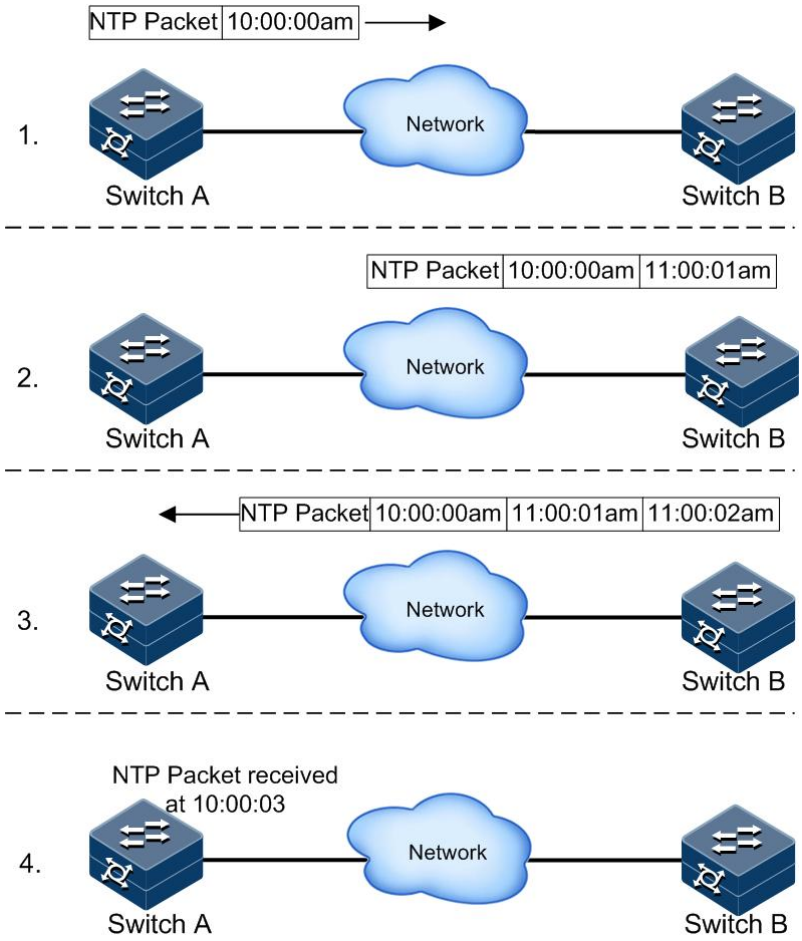
步骤 4 当 Switch A 接收到该响应消息包时，加上一个新的时间戳，该时间戳为 10:00:03am，记为 t4。

至此，Switch A 已经拥有足够的信息来计算两个重要的参数：

- NTP 消息来回一个周期的时延：Delay=(t4-t1)-(t3-t2)。
- Switch A 和 Switch B 之间的时间差：Offset=((t2-t1)+(t3-t4))/2。

Switch A 根据这些信息来设定自己的时钟，实现与 Switch B 的时钟同步。

图 1-5 NTP 基本原理



NTP 支持多种工作模式进行时间同步：

- 服务器/客户端模式

在服务器/客户端模式中，客户端向不同的服务器发送时钟同步报文。服务器收到报文后会发送应答报文。客户端收到应答报文后，进行时钟过滤和选择，并同步到优选的服务器。

在该模式下，客户端仅能同步服务器的时间，而服务器无法同步客户端的时间。设备不能既作为客户端又作为服务器。

- 对等体模式

在对等体模式中，配置对等体的设备可以向层级高的设备或者服务器进行对时。

已配置 NTP 服务器的设备不能再配置对等体。

SNTP

SNTP（Simple Network Time Protocol，简单网络时间协议）将设备系统时间同步为格林尼治时间，然后根据系统时区的设置来转化成本地时间。当 SNTP 客户端与服务器不在同一地区时，SNTP 客户端同步的时间为格林威治时间，然后根据系统时区的设置来转化成本地时间。

SNTP 客户端获取时间有两种方式，即主动发送请求报文和被动监听报文，通过不同模式实现：

- 单播模式：SNTP 客户端主动发送请求报文。指定设备的 SNTP 单播服务器地址后，设备将每隔 10s 尝试一次从 SNTP 服务器获取时钟信息，并且每次从 SNTP 服务器获取时钟信息的最大超时时间为 3 秒。
- 组播或广播模式：SNTP 客户端被动监听报文。
  - 配置 SNTP 客户端为组播模式后，设备将时刻监听组播地址 224.0.1.1，从 SNTP 组播服务器获取时钟信息，并且每次从 SNTP 获取时钟信息的最大超时时间为服务器发送周期的 1.5 倍。
  - 配置 SNTP 客户端为广播模式后，设备将时刻监听广播地址 255.255.255.255，从 SNTP 广播服务器获取时钟信息，并且每次从 SNTP 获取时钟信息的最大超时时间为服务器发送周期的 1.5 倍。

1.3.2 配置准备

场景

配置设备的系统时间，保证系统时间的精准。

- 无论何时，手动配置时间和时区立即生效。
- 开启 NTP 或 SNTP 功能，经过同步周期后，设备同步到的时间会实时刷新，覆盖当前系统时间。
- NTP 功能与 SNTP 功能互斥，不能同时配置。

前提

无

NTP

设备上 NTP 的缺省配置如下。

功能	缺省值
设备是否作为 NTP 主时钟	否

功能	缺省值
全局 NTP 服务器	无
全局 NTP 对等体	无
参考时钟源	0.0.0.0
身份验证功能	关闭
身份验证密钥 ID	无
可信密钥	无

SNTP

设备上 SNTP 的缺省配置如下。

功能	缺省值
SNTP 服务器地址	无

1.3.3 配置 NTP

配置 NTP 基本功能

请在设备上进行以下配置。

注意

NTP 功能和 SNTP 功能互斥，二者不能同时使用。

步骤	配置	说明
1	JX#**configure**	进入全局配置模式。
2	JX(config)#**ntp client update-interval** { *interval* | **default** }	配置 ntp client 通告时间 interval：通告间隔时间 范围 4-17 秒 缺省为 6 秒
3	JX(config)#**ntp server broadcast-interval** { *interval* | **default** }	配置 ntp server 通告时间 interval：通告间隔时间 范围 4-17 秒 缺省为 6 秒
4	JX(config)#**ntp master**	指定为主时钟

步骤	配置	说明
5	JX(config)#**ntp stratum** *value*	配置系统时钟的层数 value：层数 范围 1-15 缺省为 16
6	JX(config)#**ntp unicast-peer** *peer-ipv4* { **version** *version-id* | **authentication-keyid** *key-value* | **vpn** *vpn-name* | **port** *port-id* }	配置指定 ipv4 对等体 peer-ipv4：对等体 ipv4 地址 version-id：ntp 版本号 key-value：使用的认证索引 vpn-name：vpn 名称 port-id：使用的 udp 端口号
7	JX(config)#**ntp unicast-server** sserver-*ipv4* { **version** *version-id* | **authentication-keyid** *key-value* | **vpn** *vpn-name* | **port** *port-id* }	配置指定 ipv4 服务器 server-ipv4：服务器 ipv4 地址 version-id：ntp 版本号 key-value：使用的认证索引 vpn-name：vpn 名称 port-id：使用的 udp 端口号

说明

如果设备被配置为 NTP 参考时钟源，则无法配置 NTP 服务器或 NTP 对等体；反之亦然，如果配置了 NTP 服务器或对等体，则无法将设备配置为 NTP 参考时钟源。

配置 NTP 身份验证功能

在对安全性要求较高的网络中，使用 NTP 协议时需要进行身份验证。NTP 客户端使能身份验证功能后只与通过验证的服务器进行同步，保证了网络的安全性。NTP 客户端只有使能了身份验证功能才会对服务器进行验证，若未使能身份验证功能，即使服务器携带密钥信息，客户端也不会进行验证，直接与服务器进行时间同步。

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**configure**	进入全局配置模式。
2	JX(config)#**ntp authentication { enable | disable }**	（可选）ntp 认证全局使能 enable：使能 diabale：去使能。

步骤	配置	说明
3	JX(config)#**ntp authentication-keyid** *key-value* **md5 key** { **cipher** | **plain** } *string*	（可选）添加 ntp 验证密钥 key-value：密钥索引值 cipher：密文 plain：明文 STRING：密钥字符串
4	JX(config)#**ntp trusted-keyid** *key-value* **enable**	（可选）指定已创建的密钥是可信的 key-value：密钥索引值

1.3.4 配置 SNTP

配置 SNTP 客户端功能

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**configure**	进入全局配置模式。
2	JX(config-sntp)#**sntp client update-interval** <4-17>	配置 sntp client 通告时间。
3	JX(config-sntp)#**sntp server broadcast-interval** <4-17>	配置 sntp server 通告时间
4	JX(config-sntp)#**sntp authentication** { **enable** | **disable** }	SNTP 认证全局使能
5	JX(config-sntp)#**sntp authentication-keyid** <1-4294967295> **md5 key string**	添加 SNTP 验证密钥
6	JX(config-sntp)#**sntp master**	指定为主时钟
7	JX(config-sntp } **sntp stratum <1-15>**	配置系统时钟的层数
8	JX(config-sntp)#**sntp trusted-keyid** <1-4294967295>	指定已创建的密钥是可信的
9	JX(config-sntp)#**sntp unicast-peer** { **A.B.C.D** } JX(config-sntp)#**sntp unicast-peer** { **A.B.C.D** } **version** { **3** | **4** } JX(config-sntp)#**sntp unicast-peer** { **A.B.C.D** } **version** { **3** | **4** } **authentication-keyid** <1-4294967295> JX(config-sntp)#**sntp unicast-peer** { **A.B.C.D** } **authentication-keyid** <1-4294967295>	为设备指定 IPv4 对等体

步骤	配置	说明
10	JX(config-sntp)# **sntp unicast-server { A.B.C.D }** JX(config-sntp)#**sntp unicast-server { A.B.C.D } version { 3 | 4 }** JX(config-sntp)#**sntp unicast-server { A.B.C.D } version { 3 | 4 } authentication-keyid** <1-4294967295> JX(config-sntp)#**sntp unicast-server { A.B.C.D } authentication-keyid** <1-4294967295>	为设备指定单播 IPv4 SNTP 服务器

1.3.5 检查配置

配置完成后，请在设备上执行以下命令检查配置结果。

序号	检查项	说明
1	JX#**show ntp information**	显示 ntp 本地信息
2	JX#**show ntp peer**	显示 ntp 对端信息
3	JX#**show ntp key**	显示 ntp 密钥信息
4	JX#**show ntp session**	显示 ntp 会话信息
5	JX#**show sntp**	显示 sntp 全局配置
6	JX#**show sntp service**	显示 sntp 业务配置
7	JX#**show sntp service verbose**	显示 sntp 业务详细配置

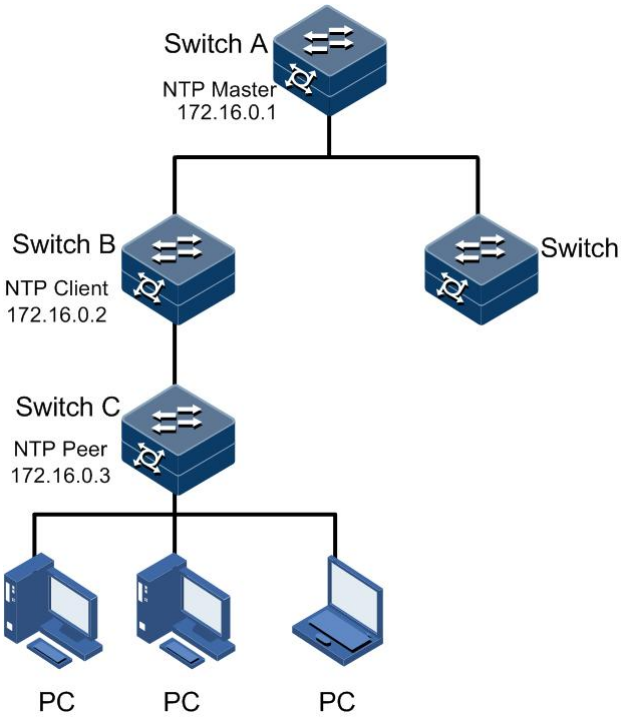
1.3.6 配置 NTP 功能示例

组网需求

某公司搭建稳定的时钟同步系统，使公司内部设备保持系统时间的一致和精准。基本规划为：

- Switch A 作为时钟同步系统的主时钟。
- Switch B 作为时钟同步系统的客户端，需设置上层的 Switch A 为 NTP 服务器。
- 设置 Switch C 为 Switch B 的 NTP 对等体，接收 Switch B 发出的下行同步数据流。

图 1-6 NTP 组网示意图



配置步骤

步骤 1 配置 Switch A。

```
JX#hostname SwitchA
SwitchA#config
SwitchA(config)#ntp
SwitchA(config-ntp)#master
SwitchA(config-ntp)#ntp stratum 2
```

步骤 2 配置 Switch B。

```
JX#hostname SwitchB
SwitchB#config
SwitchB(config)#ntp
SwitchB(config-ntp)#ntp unicast-server 172.16.0.1
SwitchB(config-ntp)#ntp unicast-peer 172.16.0.3
SwitchB(config-ntp)#ntp stratum 3
```

步骤 3 配置 Switch C

```
JX#hostname SwitchC
SwitchC#config
SwitchC(config)#ntp
SwitchC(config-ntp)#ntp unicast-peer 172.16.0.2
SwitchC(config-ntp)#ntp stratum 4
```

检查结果

- 查看 Switch A。

通过 **show ntp** 查看 Switch A 配置是否正确。

```
SwitchA#show ntp
```

- 查看 Switch B。

通过 **show ntp** 查看 Switch B 配置是否正确。

```
SwitchB#show ntp
```

通过 **show ntp service** 查看 Switch B 的 NTP 会话信息。

```
SwitchB#show ntp service
```

- 查看 Switch C。

通过 **show ntp** 查看 Switch C 配置是否正确。

```
JX#show ntp
```

通过 **show ntp service** 查看 Switch C 的 NTP 会话信息。

```
JX#show ntp service
```

1.4 接口管理

1.4.1 简介

以太网接口

以太网以其高度灵活、相对简单、易于实现的特点，成为重要的局域网组网技术。以太网接口分为：以太网电接口和以太网光接口。

交换机设备支持以太网电接口和以太网光接口。

- 自协商功能

自动协商的主要功能就是使物理链路两端的设备通过交互信息自动选择同样的工作参数。自动协商的内容主要包括双工模式、运行速率等参数。一旦协商通过，链路两端的设备就锁定在同样的双工模式和运行速率。

- 连接线缆

一般以太网标准网线分为直通线 MDI（Medium Dependent Interface）和交叉线 MDI-X（Medium Dependent Interface cross-over）两种。MDI 提供终端到网络中继设备的物理和电路连接。MDI-X 提供同种设备（终端到终端）的连接。主机和路由器的接口类型为 MDI，集线器和交换机的端口类型为 MDI-X。一般情况下，异类设备互连用直通线，同类设备互连用交叉线。自适应连接则无需考虑直通线或交叉线。

设备以太网线连接支持自适应 MDI/MDI-X。

VLAN 接口

VLAN 接口是一种逻辑接口，用于实现 VLAN 间的三层互通。每个 VLAN 对应一个 VLAN 接口，在为 VLAN 接口配置了 IP 地址后，该接口即可作为本 VLAN 内网络设备的网关，实现跨网段的报文进行基于 IP 地址的三层转发。

LoopBack 接口

LoopBack 接口是一种逻辑虚拟接口，由于该接口物理层状态和链路层协议始终处于 Up 状态，稳定性强，因此可以配置 IP 地址，并常用于动态路由协议中，作为设备的 Router ID。

NULL 接口

NULL 接口是一种逻辑虚拟接口，永远处于 Up 状态，但不能转发报文，也不能配置 IP 地址和链路层协议。NULL 接口可以对报文进行过滤，将不需要的网络流量发送到 NULL 接口，免去配置 ACL 的复杂操作。例如，在路由协议中指定到达某一网段的下一跳为 NULL 接口，则任何发送到该网段的网络数据报文都会被丢弃。

1.4.2 接口的缺省配置

设备上物理层接口的缺省配置如下。

功能	缺省值
接口的双工模式	自协商
接口的速率	自协商
接口速率统计功能状态	打开
接口的流控功能状态	禁止
接口状态	打开

1.4.3 配置接口基本属性

当互连的两个设备的接口属性，如 MTU（Maximum Transfered Unit，最大传输单元）、双工模式、速率等参数不一致时，会造成设备间无法正常通信，此时需要调整接口的属性使两端设备互相匹配。

以太网物理层分为半双工、全双工和自协商三种工作模式。

- 半双工在任意时刻只能接收或发送报文。
- 全双工在任意时刻可以同时接收和发送报文。
- 自协商是指链路两端的设备通过交互信息自动选择双工模式，一旦协商通过，两端的设备就使用同样的双工模式进行报文传输。

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**interface** *interface-type interface-number*	进入物理接口配置模式。
3	JX(config-ge-1/0/1)#**description** *string*	配置接口的描述信息
4	JX(config-ge-1/0/1)#**mtu** *max-frame-length*	配置接口的最大传输单元
5	JX(config-ge-1/0/1)#**negotiation auto** { **enable** | **disable** }	配置接口自协商功能使能或关闭
6	JX(config-ge-1/0/1)#**duplex** { **full** | **half** | **default**}	配置接口的双工模式。 需要先去使能接口自协商
7	JX(config-ge-1/0/1)#**speed** { **10** | **100** | **1000** | **default**}	配置接口的速率。 对于光接口而言，接口的速率还取决于光模块的规格。
8	JX(config-ge-1/0/1)#**transceiver type** { **1000BASE-T** | **1000BASE-X** | **100GBASE-COPPER** | **100GBASE-FIBER** | **10GBASE-COPPER** | **10GBASE-FIBER** }	配置 SFP 接口连接模式。
9	JX(config-ge-1/0/1)#**port up-hold-time** *delay-time-value*	配置端口状态变化为 UP 时延迟处理时间
10	JX(config-ge-1/0/1)#**port down-hold-time** *delay-time-value*	配置端口状态变化为 DOWN 时延迟处理时间。

1.4.4 配置接口信息统计功能

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**flow-statistic interval** *time-value*	配置全局接口信息统计周期。 缺省对所有接口生效，具体接口可以配置相应命令覆盖全局
3	JX(config)#**interface** *interface-type interface-number*	进入物理接口配置模式。
4	JX(config-ge-1/0/1)#**port flow-statistic interval** *time-value*	配置接口信息统计周期。 该配置覆盖全局的接口信息统计周期配置
5	JX(config-ge-1/0/1)#**reset statistics**	清除接口的统计信息。

1.4.5 配置接口流控功能

IEEE802.3x 是全双工以太网数据链路层的流量控制方法。当客户端向服务器发出请求后，自身系统或网络产生拥塞时，客户端会向服务器发出 PAUSE 帧，以延缓服务器向客户端的数据传输。

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**interface** *interface-type interface-number*	进入物理接口配置模式。
3	JX(config-ge-1/0/1)#**flow-control** { **enable**| **disable** }	配置接口流控功能使能或关闭
4	JX#**show interface** *interface-type interface-number*	查询接口详细信息，查看流控功能使能状态。

1.4.6 配置接口打开或关闭

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**interface** *interface-type interface-number*	进入物理接口配置模式/VLAN 接口配置模式/链路聚合接口配置模式。
3	JX(config-ge-1/0/1)#**shutdown**	关闭当前接口。 可以使用 **no shutdown** 命令再次打开接口。

1.4.7 配置接口允许通过的 tagged 报文类型

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**interface** *interface-type interface-number*	进入物理接口配置模式/链路聚合接口配置模式。

步骤	配置	说明
3	JX(config-ge-1/0/1)#**accept-frame-type { all | only-tagged }**	all：配置接口 tagged 和 untagged 报文通过。 only-tagged:配置接口只允许 tagged 报文通过。 缺省是 all。

1.4.8 配置管理报文优先级

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**interface vlan** *vlan-id*	进入 VLAN 接口配置模式
3	JX(config-vlanif-*)#**packet-priority 8021p** *value*	配置报文优先级。
4	JX(config-vlanif-*)#**no packet-priority 8021p**	恢复报文缺省优先级

1.4.9 检查配置

配置完成后，请在设备上执行以下命令检查配置结果。

序号	检查项	说明
1	JX#**show interface**	查看接口列表概要信息。
2	JX#**show interface verbose**	查看接口列表详细信息。
3	JX#**show interface** *interface-type interface-number*	查看具体接口的详细信息
4	JX#**show interface** [*interface-type interface-number*] **config**	查看接口配置信息。

2 以太网

本章介绍以太网特性的原理和配置过程，并提供相关的配置案例。

- MAC 地址转发表
- VLAN
- Voice VLAN
- QinQ
- VLAN 转换
- MRP/MVRP
- VTP

2.1 MAC 地址转发表

2.1.1 简介

以太网设备转发以太网报文是通过 MAC 地址转发规则实现的快速转发，每台设备都有一个 MAC 地址与每个接口的转发对应表，这就是 MAC 地址转发表。所有入接口报文都会依据 MAC 地址转发表进行转发，是以太网设备实现二层报文快速转发的基础。MAC 地址转发表存储在设备的缓存中，缓存容量决定设备能存储多少 MAC 地址。

MAC 地址转发表的表项中包含如下信息：

- 目的 MAC 地址
- 目的 MAC 地址所对应的接口号
- 接口所属的 VLAN ID
- 标志位 Flag

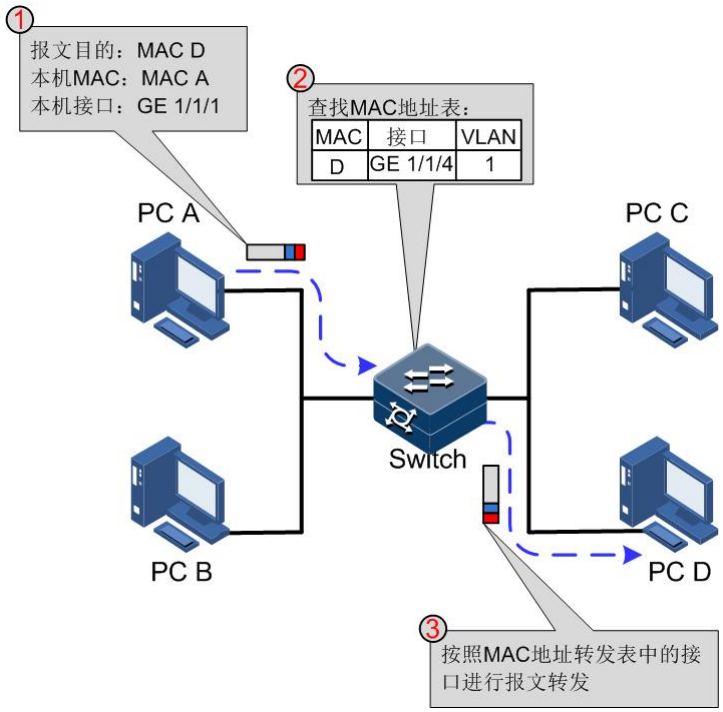
设备可基于设备、接口和 VLAN 来查看 MAC 地址表信息。

MAC 地址转发方式

以太网设备在转发报文时，根据 MAC 地址表项信息，会采取以下转发方式：

- 单播方式：当 MAC 地址转发表中包含与报文目的 MAC 地址对应的表项时，设备直接将报文从该表项中的转发接口发送；如不包含对应表项，则以广播方式向除接收接口外的所有接口转发。如下图所示。

图 2-1 MAC 地址表转发示意图



- 组播方式：当设备收到目的地址为组播 MAC 地址的报文时，组播以广播形式发送，如果开启了组播功能并设置了未知组播过滤，则发送到指定 Report 接口。如无指定 Report 接口，则报文丢弃不转发。
- 广播方式：当设备收到目的地址为全 F 的报文，或 MAC 地址转发表中没有包含对应报文目的 MAC 地址的表项时，设备将采取广播方式将报文向除接收接口外的所有接口转发。

MAC 地址表项的分类

MAC 地址转发表分为静态地址表和动态地址表两项。

- 静态 MAC 地址表项：也称为“永久地址”，由用户手工添加和删除，不会随着时间老化。对于一个设备变动较小的网络，手工添加静态地址表项可以减少网络中的广播流量，能够提高接口的安全性，且系统复位后，表项不丢失。
- 动态 MAC 地址表项：交换机可以通过 MAC 地址学习机制学习动态 MAC 地址表项，该表项会按照用户配置的老化时间而老化掉。系统复位后，表项会清空。

MAC 地址老化时间

以太网交换机的 MAC 地址转发表是有容量限制的。为了最大限度利用地址转发表资源，以太网交换机利用老化机制更新 MAC 地址转发表，即：系统在动态创建某条表

项的同时，开启老化定时器，如果在老化时间内没有再次收到来自该表项中的 MAC 地址的报文，交换机就会把该 MAC 地址表项删除。

设备支持 MAC 地址自动老化，老化时间设置范围为 0 或 10s～1 000 000s，其中 0 表示永不老化。

说明

MAC 地址的老化机制只对动态 MAC 地址表项生效。

MAC 地址转发策略

MAC 地址转发表有 2 种转发策略：

- 当报文进入设备接口时，会在 MAC 地址表查找报文中的目的 MAC 地址所关联的接口，如果在 MAC 地址表中有目的 MAC，并且目的 MAC 对应的接口与报文进入设备的接口不同，则从目的 MAC 对应的接口转发报文，并会将报文中的源 MAC 地址记录下来，与入报文的接口号、VLAN ID 相关联记录到 MAC 地址表中。当其它接口有去往该 MAC 地址时，可以通过 MAC 对应表将报文直接转发到对应的接口。
- 如果在 MAC 地址表中没有这个报文的目的 MAC，就会向除源接口外所有相同广播域的接口转发数据包，并记录该数据包中的源 MAC 地址到设备 MAC 地址表中。

MAC 地址学习数目限制

MAC 地址学习数目限制功能主要是为了限制 MAC 地址条目数，避免了因 MAC 地址表过于庞大，可能延长查找转发表项的时间，导致以太网交换机的转发性能下降的情况，是一种管理 MAC 地址表的有效方法。

MAC 地址学习数目限制主要用于限制 MAC 地址转发表的大小，提高交换机芯片的转发速度。

2.1.2 配置准备

场景

在以下情况，需要配置静态 MAC 地址转发表：

- 在公司内固定服务器、以及特定人员（经理、财务等）位置相对固定且较重要的主机上配置固定的静态 MAC 地址。以保证所有去往该 MAC 地址的数据流优先从静态 MAC 地址对应的接口进行转发。
- 对于固定静态 MAC 地址的接口，可以设置禁止 MAC 地址学习，防止其他主机通过该接口访问局域网数据。

同时，为避免 MAC 地址转发表中保存过多的 MAC 地址表项，而耗尽 MAC 地址表资源，需要配置 MAC 地址转发表的老化时间，以实现动态 MAC 地址的老化功能。

前提

无

2.1.3 MAC 地址转发表的缺省配置

设备上 MAC 地址转发表的缺省配置如下。

功能	缺省值
MAC 地址学习功能状态	使能
MAC 地址老化时间	300s
MAC 地址学习数目限制	无限制
MAC 地址漂移检测	使能
MAC 地址漂移保护	关闭

2.1.4 配置静态 MAC 地址

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**mac-address static vlan** *vlan-id* **mac** *mac-address interface-type interface-number*	配置静态单播 MAC 地址

说明

设备的 MAC 地址、组播地址、FFFF.FFFF.FFFF 及 0000.0000.0000 不能被配置为静态单播 MAC 地址。

2.1.5 配置黑洞 MAC 地址

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**mac-address blackhole vlan** *vlan-id* **mac** *mac-address*	配置黑洞 MAC 地址。

2.1.6 配置 MAC 地址学习

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**interface** *interface-type interface-number*	进入物理接口配置模式。
3	JX(config-ge-1/0/1)#**mac-learning** { **disable** | **enable** }	使能或禁止 MAC 地址学习功能

2.1.7 配置 MAC 地址学习数目限制

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**interface** *interface-type interface-number*	进入物理接口配置模式。
3	JX(config-ge-1/0/1)#**mac-limit** *limit-value*	配置接口下 MAC 地址学习数目限制阈值。

2.1.8 配置 MAC 老化时间

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**mac-address aging-time** { **0** | **default** | *period* }	配置 MAC 地址老化时间。

2.1.9 配置 MAC 地址漂移检测与保护

请在需要的设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**mac-address flapping detection** { **enable** | **disable** }	使能或关闭 MAC 地址漂移检测功能

步骤	配置	说明
3	JX(config)#**mac-address flapping detection exclude-vlan** *vlan-list*	配置不检测 MAC 地址漂移的 VLAN
4	JX(config)#**mac-address flapping detection vlan** *vlan-list* **security-level** { **high** | **middle** | **low** }	配置检测 MAC 地址漂移的 VLAN 并配置触发 MAC 地址漂移保护的安全等级 安全等级越高，触发漂移保护需要的 MAC 地址漂移次数越少
5	JX(config)#**mac-address flapping error-down recovery-interval** *period*	配置 MAC 地址漂移触发接口 error-down 的自动恢复的时间
6	JX(config)#**mac-address flapping quit-vlan recover-time** *period*	配置 MAC 地址漂移触发接口 quit-vlan 的自动恢复的时间
7	JX(config)#**interface** *interface-type interface-number*	进入物理接口配置模式。
8	JX(config-ge-1/0/1)#**mac-address flapping action** { **error-down** | **quit-vlan** }	配置接口下 MAC 地址漂移保护动作
9	JX(config-ge-1/0/1)#**mac-address flapping action priority** *priority*	配置接口下 MAC 地址漂移保护动作的端口优先级 优先级数值越高，优先触发保护动作

2.1.10 检查配置

配置完成后，请在设备上执行以下命令检查配置结果。

序号	检查项	说明
1	JX#**show mac-address**	查看 MAC 地址表项信息。
2	JX#**show mac-address vlan** *vlan-id*	查看指定 VLAN 的 MAC 地址表项信息。
3	JX#**show mac-address interface** *interface-type interface-number*	查看指定接口的 MAC 地址表项信息。
3	JX#**show mac-address** { **static** | **dynamic** | **blackhole** } [**vlan** *vlan-id*]	查看静态/动态/黑洞 MAC 地址表项信息。
4	JX#**show mac-address aging-time**	查看 MAC 地址老化时间。
5	JX#**show mac-address flapping record**	查看 MAC 地址漂移检测记录。
6	JX#**show mac-limit** [**interface** *interface-type interface-number*]	查看 MAC 地址学习数目限制的配置信息。

2.1.11 维护

用户可以通过以下命令维护 MAC 地址转发表特性。

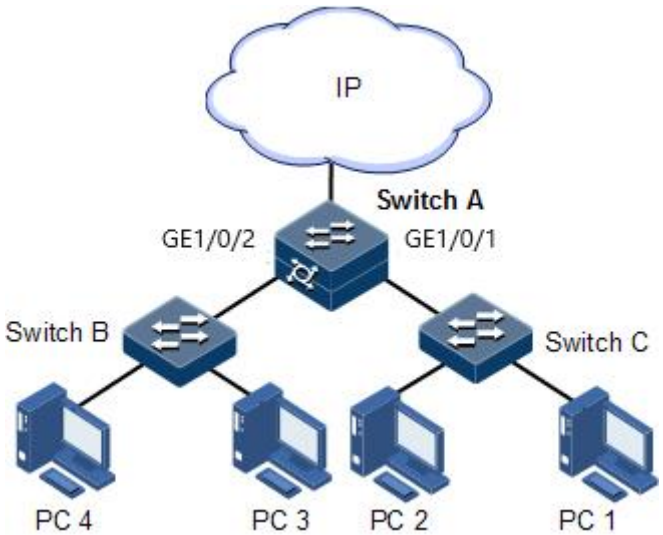
命令	描述
JX(config)#**no mac-address** [**vlan** *vlan-id*] [**mac** *mac-address*]	清除指定参数的 MAC 地址。
JX(config)#**no mac-address** *interface-type interface-number*	清除指定接口的 MAC 地址。
JX(config)#**no mac-address** { **static** | **dynamic** | **blackhole** } [**vlan** *vlan-id*] [**mac** *mac-address*]	清除指定参数的静态/动态/黑洞 MAC 地址
JX(config)#**no mac-address** { **static** | **dynamic** } [*interface-type interface-number*]	清除指定接口的静态/动态 MAC 地址
JX(config)#**reset mac-address flapping record**	清除 MAC 地址漂移记录

2.1.12 配置 MAC 地址转发表示例

组网需求

如下图所示，在 Switch A 上进行操作，在 GE 1/0/2 配置一条静态单播 MAC 地址 0001.0203.0405，所属 VLAN 为 VLAN 10；配置 MAC 地址老化时间为 500s。

图 2-2 MAC 应用组网示意图



配置步骤

步骤 1 创建 VLAN 10 并激活，将 GE 1/0/2 加入 VLAN 10。

```
JX#config
JX(config)#vlan 10
JX(config-vlan-10)#quit
JX(config)#interface ge 1/0/2
JX(config-ge-1/0/2)#port link-type access
JX(config-ge-1/0/2)#port default vlan 10
JX(config-ge-1/0/2)#quit
```

步骤 2 在 GE 1/0/2 配置一条静态单播 MAC 地址 0001.0203.0405，所属 VLAN 10。

```
JX(config)#mac-address static vlan 10 mac 00:01:02:03:04:05 ge 1/0/2
```

步骤 3 配置 MAC 地址老化时间为 500s。

```
JX(config)#mac-address aging-time 500
```

检查结果

通过 **show mac-address** 命令查看 MAC 地址配置。

```
JX#show mac-address
  MacAddress        VLAN/VSI/BD Learned-From    Type       Valid
------------------------------------------------------------------------------
  0001:0203:0405     10/--/--    ge-1/0/2       static     yes
------------------------------------------------------------------------------
  Total:1      Static:1       Dynamic:0      Blackhole:0
  Sticky:0     Security:0     Snooping:0
```

2.2 VLAN

2.2.1 简介

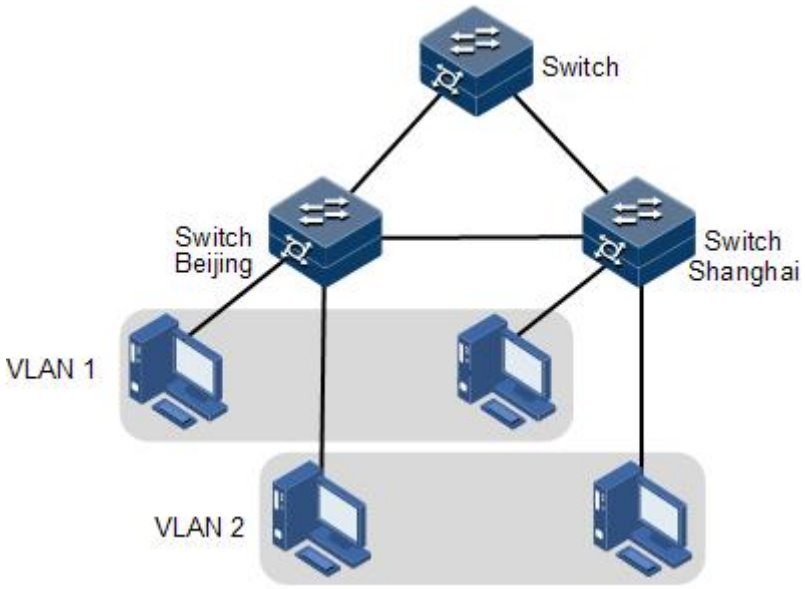
VLAN 概述

VLAN 是为解决以太网的广播问题和安全性而提出的一种协议。它是一种通过将局域网内的设备逻辑地而不是物理地划分成不同的广播域，从而实现多个互不影响的虚拟工作组的二层隔离技术。从功能上看，VLAN 和 LAN 有着相同的特性，但两者的主要区别在于同一 VLAN 内的成员可以不受物理位置的限制进行相互访问。

VLAN 划分

VLAN 划分有多种方式，例如基于接口、基于 MAC 地址、基于 IP 子网、基于协议划分等，如下图所示。

图 2-3 VLAN 划分示意图



VLAN 技术允许将一个物理的 LAN 逻辑地划分成不同的广播域。通过划分 VLAN，将没有互通需求的主机进行隔离，在增强网络安全性、减少广播流量的同时也减少了广播风暴的发生。

设备符合 IEEE 802.1Q 标准的 VLAN，支持 4094 个并发 VLAN。

- 基于接口划分 VLAN

设备支持基于接口划分 VLAN。交换机设备的接口模式分为 Access 和 Trunk 两种，接口模式与报文转发处理方式比较请参考下表。

表 2-1 接口模式与报文转发

接口类型	入报文处理		出报文处理
	Untag 报文	带 Tag 报文	
Access	为报文打上 Access VLAN 的 Tag	• 报文 VLAN ID＝Access VLAN ID，接收该报文 • 报文 VLAN ID≠Access VLAN ID，丢弃该报文	• 报文 VLAN ID＝Access VLAN ID，去掉 Tag 发送该报文 • 接口允许通过的 VLAN ID 列表中不包含报文 VLAN ID，丢弃该报文
Trunk	为报文打上 Native VLAN 的 Tag	• 接口允许通过的 VLAN ID 列表中包含报文 VLAN ID，接收该报文 • 接口允许通过的 VLAN ID 列表中不包含报文 VLAN ID，丢弃该报文	• 报文 VLAN ID＝Native VLAN ID，去掉 Tag 发送该报文 • 报文 VLAN ID≠Native VLAN ID，且接口允许通过时，保持原有 Tag 发送该报文

- 基于 MAC 地址划分 VLAN

基于 MAC 地址划分 VLAN 是指根据报文源 MAC 地址对 VLAN 进行划分。

  - 当接口收到的报文为 Untag 报文时，根据报文的源 MAC 地址匹配 MAC VLAN 表项。如果报文中的源 MAC 地址与 MAC VLAN 表项中的 MAC 地址完全相同，则匹配成功，给报文添加表项中指定的 VLAN ID 并转发该报文。如果没有找到匹配 MAC VLAN 表项，则继续匹配基于 IP 子网 VLAN、基于接口 VLAN。
  - 当接口收到的报文为 Tag 报文时，如果 VLAN ID 在接口允许通过的 VLAN ID 列表里时，则接收该报文；如 VLAN ID 不在接口允许通过的 VLAN ID 列表里时，则丢弃该报文。

- 基于 IP 子网划分 VLAN

基于 IP 子网划分 VLAN 是指根据报文源 IP 地址及子网掩码对 VLAN 进行划分。

  - 设备从接口接收到 Untag 报文后，会根据报文的源 IP 地址及子网掩码确定报文所属的 VLAN，然后将报文自动划分到指定 VLAN 中传输。
  - 当接口收到的报文为 Tag 报文时，如果 VLAN ID 在接口允许通过的 VLAN ID 列表里时，则接收该报文；如 VLAN ID 不在接口允许通过的 VLAN ID 列表里时，则丢弃该报文。

- 基于协议划分 VLAN

基于协议划分 VLAN 是指根据数据帧所属的协议类型及封装格式对 VLAN 进行划分。

  - 设备从接口接收到 Untag 报文后，会根据报文的协议域确定报文所属的 VLAN，然后将报文自动划分到指定 VLAN 中传输。
  - 当接口收到的报文为 Tag 报文时，如果 VLAN ID 在接口允许通过的 VLAN ID 列表里时，则接收该报文；如 VLAN ID 不在接口允许通过的 VLAN ID 列表里时，则丢弃该报文。

2.2.2 配置准备

场景

VLAN 最主要功能是划分逻辑网段，通常有 2 种典型应用模式。

- 一种是在小型局域网中，一台设备下划分多个 VLAN，用 VLAN 将所有连接到设备的主机逻辑的划分开，同一个 VLAN 内的主机之间可以相互通信，不同 VLAN 间的主机之间无法通信。例如财务部门与其他部门需要划分开，互相不能访问，通常连接主机的接口设置为 Access 模式。
- 另一种是在稍大型局域网或企业网中，有多台设备连接较多的主机，并且设备之间级联，转发数据报文时都携带 VLAN Tag，多台设备的相同 VLAN 的接口可以相互通信，不同 VLAN 的主机之间无法通信。主要用在公司人员及主机数量较多，并且同一个部门所在位置不同，但是要求部门内的主机可以互相访问，需要在多台设备上划分 VLAN。如果不同 VLAN 之间需要通信，则要通过路由器等 3 层设备。设备之间级联的接口设置为 Trunk 模式。

当需要为 VLAN 配置 IP 地址时，可以为其关联一个三层接口，每一个三层接口将对应一个 IP 地址并关联一个 VLAN。

前提

无

2.2.3 VLAN 的缺省配置

设备上 VLAN 的缺省配置如下。

功能	缺省值
创建 VLAN	存在 VLAN 1
接口模式	Hybrid
PVID	1
Trunk 接口的 VLAN	VLAN 1
Access 接口的 VLAN	VLAN 1

2.2.4 配置 VLAN 属性

请在需要配置 VLAN 属性信息的设备上进行以下配置。

步骤	配置	说明
1	JX#**configure**	进入全局配置模式。
2	JX(config)#**vlan** *vlan-list*	创建 VLAN。 该命令也可用于批量创建 VLAN。
3	JX(config)#**vlan** v*lan-id*	进入 VLAN 配置模式。

说明

- 用 **vlan** *vlan-id* 命令新创建的 VLAN 为活动状态。
- VLAN 的所有配置仅在该 VLAN 被激活后才会在系统中生效。

2.2.5 配置接口模式

请在需要配置接口模式的设备上进行以下配置。

步骤	配置	说明
1	JX#**configure**	进入全局配置模式。

步骤	配置	说明
2	JX(config)#**interface** *interface-type interface-number*	进入二层物理接口配置模式或聚合组配置模式。以下步骤以物理接口配置模式为例。
3	JX(config-ge-1/0/*)#**port link-type** { **access** | **default** | **dot1q-tunnel** | **hybrid** | **trunk** }	配置接口模式为 Access 或 Trunk 或 hybrid 或 dotq-tunnel。

2.2.6 配置基于 Access 接口的 VLAN

请在需要配置基于 Access 接口 VLAN 的设备上进行以下配置。

步骤	配置	说明
1	JX#**configure**	进入全局配置模式。
2	JX(config)#**interface** *interface-type interface-number*	进入二层物理接口配置模式或聚合组配置模式。以下步骤以二层物理接口配置模式为例。
3	JX(config-ge-1/0/*)#**port link-type accsess** JX(config-ge-1/0/*)#**port default vlan** *vlan-id*	配置接口模式为 Access，并将 Access 接口加入 VLAN。

说明

- 无论 Access 接口允许通过的 VLAN 列表如何配置，该接口都允许 Access VLAN 的数据包通过，且转发出去的数据包不携带相应 VLAN TAG 标记。
- 设置 Access VLAN 时，如果该 VLAN 没有创建并激活，将无法配置成功。
- 如果 Access VLAN 被用户手动删除或挂起，系统不会自动设置该接口 Access VLAN 为缺省 VLAN。
- 当配置接口 Access VLAN 为非缺省的 Access VLAN 时，缺省 Access VLAN 1 为 Access 出接口允许通过的 VLAN，通过配置删除 Access 出接口允许通过的 VLAN，可以将缺省 Access VLAN 1 从 Access 出接口允许通过 VLAN 列表中删除。
- 如果配置 Access VLAN 不是缺省 VLAN，且 Access 接口允许通过的 VLAN 列表中没有缺省 VLAN，该接口将不允许缺省 VLAN 的数据包通过。

2.2.7 配置基于 Trunk 接口的 VLAN

请在需要配置基于 Trunk 接口 VLAN 的设备上进行以下配置。

步骤	配置	说明
1	JX#**configure**	进入全局配置模式。
2	JX(config)#**interface** *interface-type interface-number*	进入二层物理接口配置模式或聚合组配置模式。以下步骤以物理接口配置模式为例。
3	JX(config-ge-1/0/*)#**port link-type trunk**	配置接口模式为 Trunk。
4	JX(config-ge-1/0/*)#**port trunk pvid** *vlan-id*	配置接口 PVID。
5	JX(config-ge-1/0/*)#**port trunk allowed-pass vlan** { **all** | *vlan-list* }	配置 Trunk 接口允许通过的 VLAN。

说明

- 无论 Trunk 接口允许通过的 VLAN 列表列表如何配置，该接口都允许 VLAN 的数据包通过，且转发出去的数据包不携带相应 PVID 标记。
- 设置 Native VLAN 时，如果该 VLAN 没有创建并激活，系统不会自动创建并激活该 VLAN。
- 如果 Native VLAN 被用户手动删除或阻塞，系统不会自动设置该接口 Trunk Native VLAN 为缺省 VLAN。
- 接口允许 Trunk Allowed VLAN 报文出入。
- 如果配置 Native VLAN 不是缺省 VLAN，且 Trunk 接口允许通过的 VLAN 列表中没有缺省 VLAN，该接口将不允许缺省 VLAN 的数据包通过。
- Trunk 允许 VLAN 列表都只对静态 VLAN 生效，对集群 VLAN、GVRP 动态 VLAN 等不生效。

2.2.8 配置基于 Hybrid 接口的 VLAN

请在需要配置基于 Hybrid 接口 VLAN 的设备上进行以下配置。

步骤	配置	说明
1	JX#**configure**	进入全局配置模式。
2	JX(config)#**interface** *interface-type interface-number*	进入二层物理接口配置模式或聚合组配置模式。以下步骤以物理接口配置模式为例。
3	JX(config-ge-1/0/*)#**port link-type hybrid**	配置接口模式为 hybrid。
4	JX(config-ge-1/0/*)#**port hybrid pvid** *vlan-id*	配置接口 PVID。

步骤	配置	说明
5	JX(config-ge-1/0/*)#**port hybrid vlan** { **all** | *vlan-list* } { **tagged** | **untagged** }	配置 hybrid 接口以{tagged|untagged}方式允许通过的 VLAN。

- Hybrid 端口根据配置的不同将有不同的工作方式，untagged 方式接口工作模式与 Access 模式端口相同，但可配置多个可通过 VLAN，tagged 方式接口工作模式与 trunk 模式端口相同。

2.2.9 配置基于 MAC 地址的 VLAN

请在需要配置基于 MAC 地址的 VLAN 的设备上进行以下配置。

步骤	配置	说明
1	JX#**configure**	进入全局配置模式。
2	JX(config)#**mac-vlan** *mac-address vlan-id*	配置 MAC 地址与 VLAN 的关联
3	JX(config)#**interface** *interface-type interface-number*	进入二层物理接口配置模式。
4	JX(config-ge-1/0/*)#**mac-vlan enable**	使能 MAC-VLAN 功能。

- MAC 地址为组播 MAC 地址、全 0 或全 F 时，配置失败。
- 创建的 MAC 地址与 VLAN 关联与已经存在的关联冲突（例如同一 MAC 地址关联到不同的 VLAN），则配置失败。

2.2.10 配置基于 IP 子网的 VLAN

请在需要配置基于 IP 子网的 VLAN 的设备上进行以下配置。

步骤	配置	说明
1	JX#**configure**	进入全局配置模式。
2	JX(config)#**ip-subnet-vlan** *ip-address* [*ip-mask*] **vlan** *vlan-id*	配置 VLAN 与 IP 子网地址的关联。

步骤	配置	说明
3	JX(config)#**interface** *interface-type interface-number*	进入二层物理接口配置模式。
4	JX(config-ge-1/0/*)#**ip-subnet-vlan enable**	使能基于 IP 子网划分 VLAN 功能。

- IP 地址或掩码无效时，配置失败。
- 创建的 IP 子网与 VLAN 关联与已经存在的关联冲突（例如同一子网关联到不同的 VLAN 时），配置失败。

2.2.11 配置基于协议的 VLAN

请在需要配置基于协议划分 VLAN 的设备上进行以下配置。

步骤	配置	说明
1	JX#**configure**	进入全局配置模式。
2	JX(config)#**protocol-vlan** *protocol-index* { **ethernet2** | **llc** | **snap** *protocol-id* }	配置协议 VLAN 与以太报文关联规则
3	JX(config)#**interface** *interface-type interface-number*	进入二层物理接口配置模式。
4	JX(config-ge-1/0/*)#**protocol-vlan** *protocol-index* **vid** *vlan-id*	配置接口和协议 VLAN 的关联规则

2.2.12 查询 VLAN 统计信息

查询 VLAN 统计，请在设备上执行以下命令获取统计结果。

序号	检查项	说明
1	JX(config-vlan-*)#**statistics { enable | disable }**	使能 VLAN 统计功能。
2	JX#**show vlan** [*vlan-id*] **statistics**	查看 VLAN 统计信息。
3	JX(config)#**reset vlan** [*vlan-id*] **statistics**	清除 VLAN 统计信息。

2.2.13 检查配置

配置完成后，请在设备上执行以下命令检查配置结果。

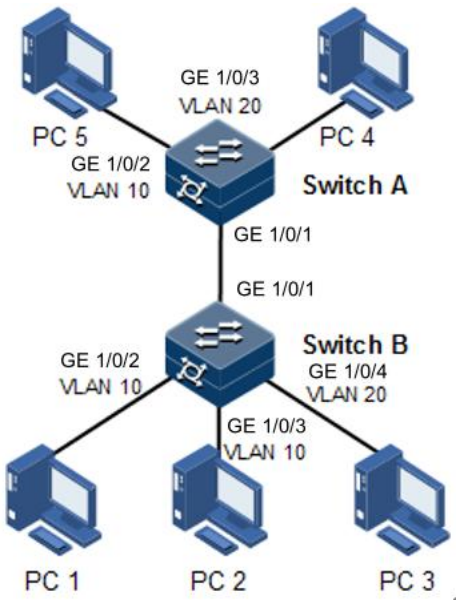
序号	检查项	说明
1	JX#**show vlan** [*vlan-list* | *vlan-id*]	查看 VLAN 配置。
2	JX#**show port vlan**	查看接口 VLAN 配置信息。
3	JX#**show mac-vlan** [**id** vlan *vlan-id*]	查看 MAC VLAN 配置信息
4	JX#**show ip-subnet-vlan** [**vlan** *vlan-id*]	查看 IP 子网 VLAN 的配置信息。
5	JX#**show protocol-vlan all**	查看全部协议 VLAN 配置信息。
6	JX#**show protocol-vlan interface**	查看接口的协议 VLAN 配置信息。

2.2.14 配置 VLAN 示例

组网需求

如图 2-4 所示，PC 1、PC 2、PC 5 属于 VLAN 10，PC3、PC4 属于 VLAN 20；两台设备相连的接口为 Trunk 模式，但不允许 VLAN 20 的报文通过，使 PC 3 和 PC 4 无法通信；在同一 Switch B 设备下的 PC 1 和 PC 2 的接口开启接口保护功能，使 PC 1 和 PC 2 无法通信，但 PC 1 和 PC 2 分别可以和 PC 5 通信。

图 2-4 VLAN 和接口保护组网示意图



配置步骤

步骤 1 在两台设备上分别创建 VLAN 10 和 VLAN 20 并激活。

配置 Switch A。

```
JX#hostname SwitchA
SwitchA#configure
SwitchA(config)#vlan 10,20
```

配置 Switch B。

```
JX#hostname SwitchB
SwitchB#configure
SwitchB(config)#vlan 10,20
```

步骤 2 将 Switch B 的接口 GE 1/0/2 和 GE 1/0/3 以 Access 模式加入 VLAN 10，接口 GE 1/0/4 以 Access 模式加入 VLAN 20，接口 GE 1/0/1 为 Trunk 模式允许 VLAN 10 通过。

```
SwitchB(config)#interface ge 1/0/2
SwitchB(config-ge-1/0/2)#port link-type access
SwitchB(config-ge-1/0/2)#port default vlan 10
SwitchB(config-ge-1/0/2)#exit
SwitchB(config)#interface ge 1/0/3
SwitchB(config-ge-1/0/3)#port link-type access
SwitchB(config-ge-1/0/3)#port default vlan 10
SwitchB(config-ge-1/0/3)#exit
SwitchB(config)#interface ge 1/0/4
SwitchB(config-ge-1/0/4)#port link-type access
SwitchB(config-ge-1/0/4)#port default vlan 20
SwitchB(config-ge-1/0/4)#exit
SwitchB(config)#interface ge 1/0/1
SwitchB(config-ge-1/0/1)#port link-type trunk
SwitchB(config-ge-1/0/1)#port trunk allow-pass vlan 10
SwitchB(config-ge-1/0/1)#exit
```

步骤 3 将 Switch A 的接口 GE 1/0/2 以 Access 模式加入 VLAN 10，接口 GE 1/0/3 以 Trunk 模式加入 VLAN 20，接口 GE 1/0/1 为 Trunk 模式允许 VLAN 10 通过。

```
SwitchA(config)#interface ge 1/0/2
SwitchA(config-ge-1/0/2)#port link-type access
SwitchA(config-ge-1/0/2)#port access vlan 10
SwitchA(config-ge-1/0/2)#exit
SwitchA(config)#interface ge 1/0/3
SwitchA(config-ge-1/0/3)#port mode trunk
SwitchA(config-ge-1/0/3)#port trunk pvid 20
SwitchA(config-ge-1/0/3)#port trunk allow-pass 20
SwitchA(config-ge-1/0/3)#exit
SwitchA(config)#interface ge 1/0/1
SwitchA(config-ge-1/0/1)#port link-type trunk
SwitchA(config-ge-1/0/1)#port trunk allowed-pass vlan 10
```

检查结果

通过 **show vlan** 命令查看 VLAN 的配置信息是否正确。

以 Switch B 为例。

```
SwitchB#show vlan
    S: supervlan       P: pvlan        N: normal


 Vlan  Type       Ports('M': member , '-': not member)
--------------------------------------------------------------------------------
 1     N/static   untagged: ge-1/0/1<->ge-1/0/16      M---MMMM MMMMMMMM
                         ge-1/0/17<->ge-1/0/24        MMMMMMMM
                         10ge-1/0/25<->10ge-1/0/30    MMMMMM
 10    N/static   tagged:   ge-1/0/1<->ge-1/0/16      M------- --------
                  untagged: ge-1/0/1<->ge-1/0/16      -MM----- --------
 20    N/static   untagged: ge-1/0/1<->ge-1/0/16      ---M---- --------
--------------------------------------------------------------------------------
  Total: 3    Static: 3    Dynamic: 0
```

通过 **show interface** *interface-type interface-number* **config** 查看接口 VLAN 配置是否正确。

以 Switch B 为例。

```
SwitchB#show interface ge 1/0/2 config
!
interface ge 1/0/2
 port link-type access
 port default vlan 10
```

通过 PC 1 ping PC 5、PC 2 ping PC 5、PC 3 ping PC 4 是否能够 ping 通，查看 Trunk 接口允许通过 VLAN 是否正确。

- PC 1 ping PC 5，可以 ping 通，VLAN 10 通信正常
- PC 2 ping PC 5，可以 ping 通，VLAN 10 通信正常
- PC 3 ping PC 4，不能 ping 通，VLAN 20 无法通信

2.3 Voice VLAN

2.3.1 简介

随着语音技术的日益发展，语音设备应用越来越广泛，尤其在宽带小区，网络中经常同时存在语音、数据和业务数据两种流量。语音数据在传输时需要具有比业务数据更高的优先级，以减少传输过程中可能产生的时延和丢包现象。

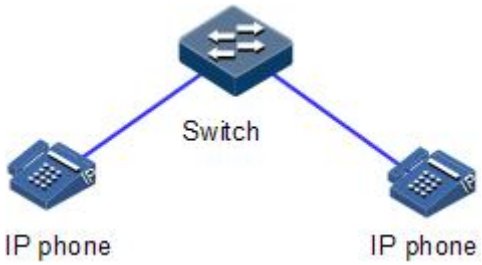
Voice VLAN 指为用户的语音数据流而专门划分的 VLAN。通过划分 Voice VLAN 并将连接语音设备的端口加入 Voice VLAN，可以为语音数据配置 QoS（Quality of Service，服务质量），提高语音流量的传输优先级，保证通话质量。

相对于使用其他语音流的方法，Voice VLAN 对语音流的管理具有以下一些优势：

- 配置简单：用户只需要在全局和端口下进行简单的配置，开启 Voice VLAN 功能，即可对语音数据进行分类处理。
- 便于维护：用户可以在全局配置对语音数据的匹配规则（Voice VLAN OUI 地址）进行修改，在新增 IP 语音设备的情况下，各端口能够迅速根据更新的匹配规则识别语音流。
- 实现灵活： Voice VLAN 功能在全局提供了安全/普通两种模式，端口上又可以分为自动/手动模式，实现更为灵活，用户可以根据自己需要进行组合，最大限度满足用户的需求。

适用于 IP 电话单独接入交换机（端口仅传输语音报文）的组网方式（如图 2 所示），这种静态加入的方式可以使该端口专用于传输语音数据，最大限度避免业务数据对语音数据传输的影响。

图 2-5 IP 电话单独接入交换机组网示意图



2.3.2 配置准备

场景

语音流量可通过专属 VLAN（Voice VLAN）传输，在一段时间内，如果语音设备发生故障或语音设备退出网络，连接语音设备的端口会自动从 Voice VLAN 中退出。

前提

已经创建 VLAN，并且正确设备 VLAN 属性。

2.3.3 Voice VLAN 的缺省配置

设备上 Voice VLAN 的 OUI（Organizationally Unique Identifier，全球统一标识符地址）缺省配置如下。

OUI-Address	Mask address	Description
0001.E300.0000	FFFF.FF00.0000	Siemens-phone
0003.6B00.0000	FFFF.FF00.0000	Cisco-phone
0004.0D00.0000	FFFF.FF00.0000	Avaya-phone
00D0.1E00.0000	FFFF.FF00.0000	Pingtel-phone

OUI-Address	Mask address	Description
0060.B900.0000	FFFF.FF00.0000	Philips/NEC-phone
00E0.7500.0000	FFFF.FF00.0000	Verilink-phone
00E0.BB00.0000	FFFF.FF00.0000	NBX-phone

设备上 Voice VLAN 的其他缺省配置如下。

功能	缺省值
Voice VLAN 功能	禁止
Voice Vlan 工作模式为安全模式	禁止
Voice Vlan 工作模式为普通模式	使能
端口加入 Voice Vlan 的工作模式为自动模式	禁止
端口加入 Voice Vlan 的工作模式为手工模式	使能
Voice Vlan 报文的 Cos 和 DSCP 值	Voice VLAN 报文 COS 为 6，DSCP 为 46
Voice Vlan QoS 优先级信任	无
Voice Vlan 老化时间	5min

2.3.4 配置 OUI 地址

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**voice-vlan oui** *mac-address* [*description word*]	配置 Voice VLAN 的 OUI。

2.3.5 使能 Voice VLAN 功能

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。

步骤	配置	说明
2	JX(config)#**interface** *interface-type interface-number*	进入物理层接口配置模式。
3	JX(config-ge-1/0/*)#**voice-vlan** *vlan-id* **enable**	配置 Voice VLAN 功能使能。
4	JX(config-ge-1/0/*)#**voice-vlan mode** { **auto** | **manual** }	配置端口加入 Voice VLAN 的工作模式。
5	JX(config)#**voice-vlan aging-time** *time*	配置端口在自动模式下离开 Voice VLAN 的老化时间。
6	JX(config)#**interface** *interface-type interface-number* JX(config-ge-1/0/*)#**voice-vlan security enable**	配置 Voice Vlan 工作模式为安全模式。

2.3.6 检查配置

配置完成后，请在设备上执行以下命令检查配置结果。

序号	检查项	说明
1	JX#**show voice-vlan oui**	查看当前设备上的 OUI 地址、OUI 地址掩码和描述信息。
2	JX#**show voice-vlan interface**	查看当前设备上 Voice VLAN 的接口状态。

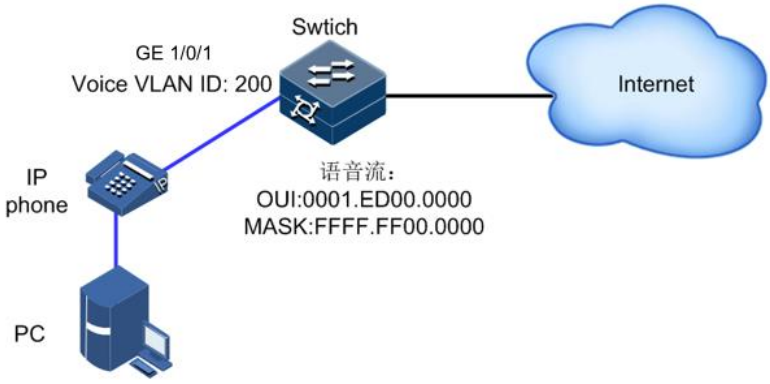
2.3.7 配置端口加入 Voice VLAN 示例

配置端口加入 Voice VLAN，工作于手动模式

组网需求

Switch 的 GE 1/0/1 接口接入 IP 电话和 PC，要求 GE 1/0/1 端口同时转发并且隔离语音流和数据流。

可将端口配置为 Trunk 口，Native Vlan 转发数据流，Voice Vlan 转发语音流。PC 发出的是 Untagged 的报文，因此会在端口的 Native VLAN 内传输，将 Native Vlan 设置为 100，用来传输 PC 发出的数据流；IP 电话发出的也是 Untagged 报文，源 mac 配置为 Voice Vlan 的 OUI 地址，则报文通过 Voice Vlan 端口时会添加 Voice Vlan Tag，Voice Vlan 设置为 200，用来传输 IP 电话语音流。



配置步骤

步骤 1 在交换机上设置 IP Phone 的 MAC 地址（支持掩码）为 Voice Vlan OUI 地址，地址为 0001.ED00.0000，掩码是 FFFF.FF00.0000。设备默认支持的 OUI 参见 VoiceVLAN 缺省配置章节。

```
JX(config)#voice-vlan oui 00:01:ED:00:00:00/24
```

步骤 2 创建 VLAN100，VLAN200 并激活，设置 VLAN 200 为 Voice VLAN。

```
JX(config)#vlan 100,200
JX(config)#interface ge 1/0/1
JX(config-ge-1/0/1)#port link-type trunk
JX(config-ge-1/0/1)#port trunk allow-pass vlan 100,200
JX(config-ge-1/0/1)#voice-vlan 200 enable
```

检查配置

使用命令 **show voice-vlan interface** 查看设备当前的 Voice VLAN 接口状态

```
JX(config)#show voice-vlan interface
  Support max interface      :64
  Current enable interface   :1
  Interface        VID    Mode    Security  RemainTime(s)
------------------------------------------------------------
  ge-1/0/1         200      manual   disable    N/A
------------------------------------------------------------
```

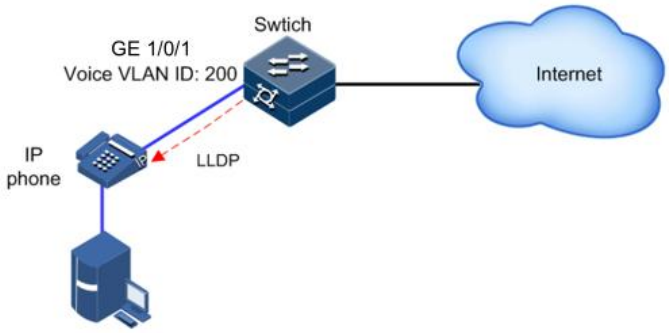
2.3.8 配置 LLDP 实现 IP 话机接入 Voice VLAN 示例

组网需求

当语音设备支持 LLDP 协议，如图所示，支持通过 LLDP 协议获取语音 VLAN。因此可通过在交换机上配置 LLDP 和 Voice VLAN 功能，实现 IP 话机接入。在交换机上通过配置 LLDP 协议向语音设备发布端口的 Voice VLAN；为保证通话质量，配置 Voice VLAN 功能提升语音报文的优先级。

Switch 的 GE 1/0/1 接口接入 IP 电话和 PC，要求 GE 1/0/1 端口同时转发并且隔离语音流和数据流。可将端口可配置为 Trunk 口，Native Vlan 转发数据流，Voice Vlan 转发语音流。PC 发出的是 Untagged 的报文，因此会在端口的 Native VLAN 内传输，将 Native Vlan 设置为 100，用来传输 PC 发出的数据流；Voice Vlan 设置为 200，用来传

输 IP 电话语音流，IP 电话通过 LLDP 协议获取到 Voice VLAN，发出的是带 Voice VLAN Tag 的报文。



配置步骤

步骤 1 在交换机上设置 IP Phone 的 MAC 地址（支持掩码）为 Voice Vlan OUI 地址，地址为 0001.ED00.0000，掩码是 FFFF.FF00.0000。设备默认支持的 OUI 参见 VoiceVLAN 缺省配置章节。

```
JX(config)#voice-vlan oui 0001.ED00.0000/24
```

步骤 2 创建 VLAN100，VLAN200 并激活，设置 VLAN 200 为 Voice VLAN。

```
JX(config)#vlan 100,200
JX(config)#interface ge 1/0/1
JX(config-ge-1/0/1)#port link-type trunk
JX(config-ge-1/0/1)#port trunk allow-pass vlan 100
JX(config-ge-1/0/1)#voice-vlan 200 enable
JX(config-ge-1/0/1)#exit
```

步骤 3 全局和接口使能 LLDP 功能，向 IP 话机发布端口的 Voice VLAN。

```
JX(config)#lldp start
JX(config)#interface ge 1/0/1
JX(config-ge-1/0/1)#lldp admin-status rx-tx
```

检查配置

使用命令 **show voice-vlan interface** 查看设备的 Voice VLAN 接口状态

```
JX(config)#show voice-vlan interface
  Support max interface      :64
  Current enable interface   :1
  Interface        VID    Mode    Security  RemainTime(s)
-------------------------------------------------------------
  ge-1/0/1         200      auto   enable    N/A
-------------------------------------------------------------
```

2.4 QinQ

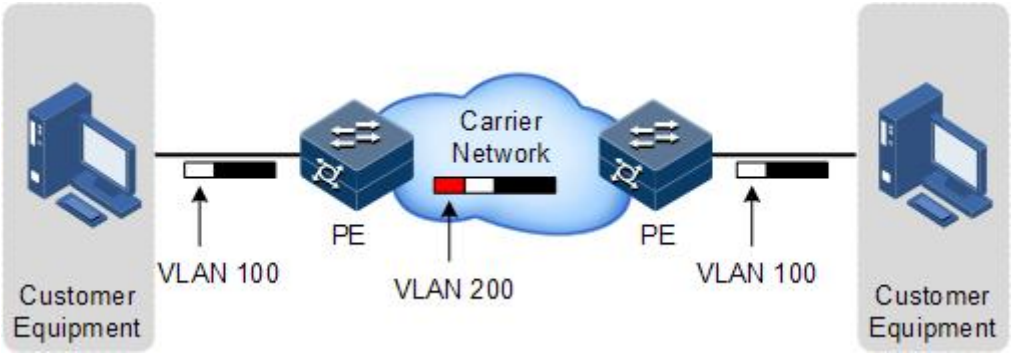
2.4.1 简介

QinQ（也称 Stacked VLAN 或 Double VLAN）技术是对 802.1Q 的扩展，是 IEEE 在 802.1ad 标准中定义的。

基本 QinQ

基本 QinQ 是一种简单的二层 VPN 隧道技术，它通过在运营商接入端为用户的私网报文封装外层 VLAN Tag，使报文携带两层 VLAN Tag 穿越运营商的骨干网络（公网）。在公网中，报文只根据外层 VLAN Tag（即公网 VLAN Tag）进行传输，用户的私网 VLAN Tag 则当作报文中的数据部分来进行传输。

图 2-6 基本 QinQ 典型组网示意图



基本 QinQ 典型组网应用如上图所示，设备作为运营商 PE（Provider Edge）。

报文从用户设备中传送到达 PE 设备，此时报文携带标签的 VLAN ID 为 100。经过 PE 设备用户侧接口时打上外层标签 VLAN 200，经由 PE 设备的网络侧接口进入运营商网络。

带外层标签 VLAN 200 的报文经过运营商传送到达另一端 PE 设备，另一端的 PE 把外层签 VLAN 200 剥去，然后发送到用户设备。报文此时又恢复到只携带一层标签 VLAN 100。

通过该技术可以缓解日益紧缺的公网 VLAN ID 资源，使得用户可以规划自己的私网 VLAN ID，不会导致和公网 VLAN ID 冲突。

灵活 QinQ

灵活 QinQ 是基本 QinQ 的一种增强应用，可以根据一些特性对用户数据进行流分类，然后对不同的类别封装不同的外层 VLAN 标签，它基于接口与 VLAN 结合的方式来实现。除了能实现所有基本 QinQ 的功能外，灵活 QinQ 还能够对同一个接口收到的报文根据不同的 VLAN Tag 做不同的动作，为报文的不同内层 VLAN ID 添加不同的外层 VLAN ID。通过配置内外层 Tag 映射规则，还可以为具有不同内层 Tag 的报文按映射规则封装不同的外层 Tag。

灵活 QinQ 功能可以使运营商的网络构架更为灵活，在连接接入层设备的接口上可以根据 VLAN Tag 对不同的终端用户进行分类，为各类用户封装不同的外层 Tag，并可以在公网中按外层 Tag 配置 QoS 策略，灵活配置数据的传输优先级，使各类用户获得相应的服务。

2.4.2 配置准备

场景

对设备进行基本 QinQ 配置还是灵活 QinQ 配置，需要根据不同的业务需求进行选择。

- 基本 QinQ

通过基本 QinQ 技术应用，用户可以添加外层 VLAN Tag 自由规划自己的私网 VLAN ID，使得运营商网络两端的用户设备之间的数据可以通过运营商网络进行透明输，而不会导致和服务提供商网络中 VLAN ID 冲突。

- 灵活 QinQ

与基本 QinQ 不同，外层 VLAN Tag 是可以根据业务不同进行选择的。用户网络中有多种业务，并设定不同的私网 VLAN ID，针对语音、视频、或数据业务等在运营商网络中通过加不同的外层 VLAN Tag 区分出来，针对不同业务转发的同时实现不同的分流，实现内外层的 VLAN 映射。

前提

在配置 QinQ 之前，需完成以下任务：

- 连接接口并配置接口的物理参数，使接口的物理层状态为 Up。
- 创建 VLAN。

说明

- 基本 QinQ 和 1:1VLAN 转换命令行可同时配置，VLAN 转换在基本 QinQ 使能前后功能正常。
- 灵活 QinQ 和 1:1 VLAN 转换命令行可同时配置：同时配置时，灵活 QinQ 和 VLAN 转换功能正常；开启/关闭基本 QinQ，灵活 QinQ 和 VLAN 转换功能正常；取消其中一个配置命令，不影响其他配置正常使用。
- 基本 QinQ、灵活 QinQ 和 2:2VLAN 转换互斥：同时配置灵活 QinQ 和 1:1VLAN 转换时，匹配 VLAN 不能相同，转换后的 VLAN 不能相同。

2.4.3 QinQ 的缺省配置

设备上 QinQ 的缺省配置如下。

功能	缺省值
外层 VLAN Tag 的 TPID 值	0x8100
基本 QinQ 功能状态	禁用

功能	缺省值
灵活 QinQ 功能状态	禁用

2.4.4 配置基本 QinQ

请在需要的设备的用户侧接口上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**interface** *interface-type interface-number*	进入二层物理接口配置模式或聚合组配置模式。以下步骤以物理接口配置模式为例。
3	JX(config-ge-1/0/1)#**port link-type dot1q-tunnel**	使能接口基本 QinQ 功能。
4	JX(config-ge-1/0/1)#**port default vlan** *vlan-id*	配置使能接口的基本 QinQ 功能，添加双层 TAG，指定 CVLAN，SVLAN 使用 PVID。
5	JX(config-ge-1/0/1)#**tpid** *tpid*	配置接口的 VLAN Tag 的 TPID，用于识别用户内层 VLAN 或者 QinQ 使能时识别用户外层 VLAN， tpid：TPID 值，十六进制数，整数形式，取值范围是 0x8100、0x88a8 或 0x9100
6	JX(config-ge-1/0/1)#**inner tpid** *tpid*	配置接口的内层 VLAN Tag 的 TPID，用于识别用户内层 VLAN 或者 QinQ 使能时识别用户外层 VLAN， tpid：TPID 值，十六进制数，整数形式，取值范围是 0x8100、0x88a8 或 0x9100

说明

- 当接口开启基本 QINQ 时，所有报文均当做 Untagged 报文处理，若同时配置 Untagged 报文丢弃，那么 Tagged 的报文也会同时丢弃。

2.4.5 配置灵活 QinQ

请在需要的设备用户侧接口上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**interface** *interface-type interface-number*	进入二层物理接口配置模式或聚合组配置模式。以下步骤以物理接口配置模式为例。
3	JX(config-ge-1/0/*)#**vlan-stacking vlan** *vlan-id* **stack-vlan** *vlan-id* [**remark-8021p** *cos-id*]	配置灵活 qinq 功能。

2.4.6 配置网络侧接口为 Trunk 模式

请在需要配置基本 QinQ 或灵活 QinQ 的设备网络侧接口上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**interface** *interface-type interface-number*	进入二层物理接口配置模式或聚合组配置模式。以下步骤以物理接口配置模式为例。
3	JX(config-ge-1/0/*)#**switchport mode trunk**	配置接口 Trunk 模式，可允许双 Tag 报文通过。

2.4.7 配置 TPID

请在需要设备的网络侧接口进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**interface** *interface-type interface-number*	进入二层物理接口配置模式或聚合组配置模式。以下步骤以物理接口配置模式为例。
3	JX(config-ge-1/0/1)#**tpid** *tpid*	配置接口的外层 VLAN Tag 的 TPID。

2.4.8 检查配置

配置完成后，请在设备上执行以下命令检查配置结果。

序号	检查项	说明
1	JX#show vlan-stacking config	查看接口基本 QinQ 配置信息。
2	JX#show vlan-stacking interface	查看接口的 QinQ 配置信息。
3	JX#show vlan-stacking resource	查看 QinQ 的资源信息

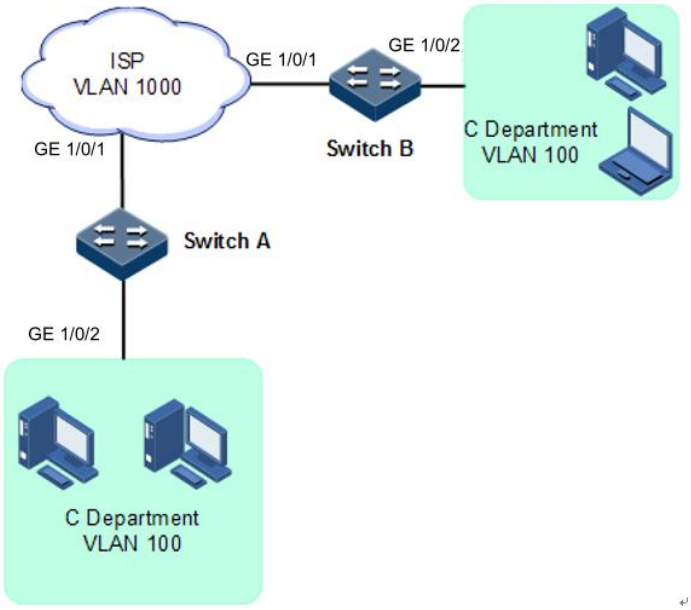
2.4.9 配置基本 QinQ 示例

组网需求

如下图所示，Switch A 和 Switch B 分别连接了 C 部门的 2 个不同地点的网络。C 部门为 VLAN100，要通过运营商 VLAN1000 网络进行通信。运营商的 TPID 为 9100。

通过在 Switch A 和 Switch B 配置基本 QinQ 功能来实现部门内部通过运营商网络的正常通信。

图 2-7 基本 QinQ 应用组网示意图



配置步骤

配置 Switch A 和 Switch B。

Switch A 和 Switch B 配置步骤完全相同，仅以配置 Switch A 为例。

步骤 1 创建 VLAN 100 和 VLAN 1000 并激活。

```
JX#config
JX(config)#vlan 100,1000
JX(config)#interface ge 1/0/1
JX(config-ge-1/0/1)#port link-type trunk
JX(config-ge-1/0/1)#port trunk allow-pass vlan 1000
JX(config-ge-1/0/1)#tpid 0x9100
JX(config-ge-1/0/1)#exit
```

步骤 2 使能接口的基本 QinQ 功能。

```
JX(config)#interface ge 1/0/2
JX(config-ge-1/0/2)#port link-type dot1q-tunnel
JX(config-ge-1/0/2)#port default vlan 1000
JX(config-ge-1/0/2)#exit
```

检查结果

通过 **show dot1q-tunnel** 命令查看 QinQ 配置信息。

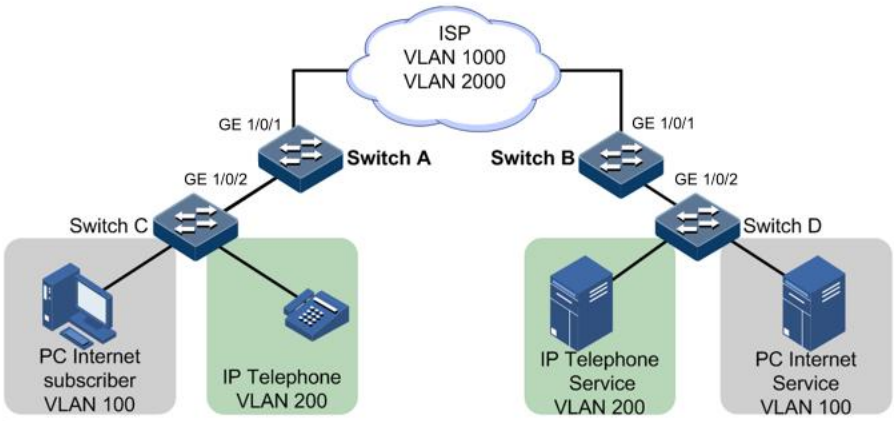
```
JX#show interface ge 1/0/2 config
!
interface ge 1/0/2
 port link-type dot1q-tunnel
 port default vlan 1000
```

2.4.10 配置灵活 QinQ 示例

组网需求

如下图所示，运营商网络区分普通 PC 上网业务和 IP 电话业务，故将 PC 上网业务分配 VLAN 1000，IP 电话业务分配的 VLAN 是 2000。在 Switch A 和 Switch B 上设置灵活 QinQ，将分配给 PC 上网业务的 VLAN 100 添加外层 Tag VLAN 1000，将分配给 IP 电话业务的 VLAN 200 添加外层 Tag VLAN 2000，使用户和服务器间通过运营商的网络能够正常通信。运营商的 TPID 为 9100。

图 2-8 灵活 QinQ 应用组网示意图



配置步骤

配置 Switch A 和 Switch B。

Switch A 和 Switch B 配置步骤完全相同，仅以配置 Switch A 为例。

步骤 1 创建 VLAN 100，200，1000，2000 并激活，TPID 为 9100。

```
JX#hostname SwitchA
SwitchA#config
SwitchA(config)#vlan 100,200,1000,2000
SwitchA(config)#interface ge 1/0/1
SwitchA(config-ge-1/0/1)#port link-type trunk
SwitchA(config-ge-1/0/1)#port trunk allow-pass vlan 1000,2000
SwitchA(config-ge-1/0/1)#tpid 0x9100
SwitchA(config-ge-1/0/1)#exit
```

步骤 2 使能接口 GE 1/0/2 的灵活 QinQ 功能。

```
SwitchA(config)#interface ge 1/0/2
SwitchA(config-ge-1/0/2)#port link-type trunk
SwitchA(config-ge-1/0/2)#port trunk allow-pass vlan 100,200,1000,2000
SwitchA(config-ge-1/0/2)#vlan-stacking enable
SwitchA(config-ge-1/0/2)#vlan-stacking vlan 100 stack-vlan 1000
SwitchA(config-ge-1/0/2)#vlan-stacking vlan 200 stack-vlan 2000
SwitchA(config-ge-1/0/2)#exit
```

检查结果

通过 **show vlan-stacking interface** 命令查看灵活 QinQ 配置信息。

以 Switch A 为例。

```
SwitchA#show vlan-stacking interface
 Interface         MatchVlan(outer/8021p)
StackVlan(inner/outer/8021p)
```

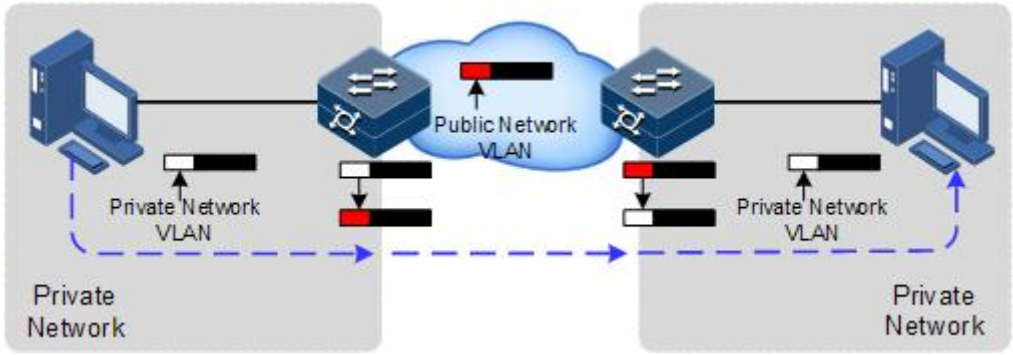
```
--------------------------------------------------------------------------
-------
 ge-1/0/2          100/-                    -/1000/-
 ge-1/0/2          200/-                    -/2000/-
```

2.5 VLAN 转换

2.5.1 简介

VLAN 转换主要功能是将以太网业务报文中的私网 VLAN Tag 替换为运营商的 VLAN Tag，使其按照运营商的 VLAN 转发规则进行传输。在报文从运营商网络转发到对端用户私网时，再按照同样的规则将 VLAN Tag 恢复为原有的用户私网 VLAN Tag，使报文正确到达目的地。VLAN 转换原理如下图所示。

图 2-9 VLAN 转换原理组网示意图



交换机收到带有用户私网报文的 VLAN Tag 后，根据配置的 VLAN 转换规则对用户私网报文的 VLAN Tag 进行匹配，如果匹配成功，则按照 VLAN 转换规则将私网 VLAN Tag 进行替换。设备支持 1:1 的 VLAN 转换，即将来自某一特定 VLAN 的报文所携带的 VLAN Tag 替换为新的 VLAN Tag。

与 QinQ 功能不同的是，VLAN 转换功能不需要对报文进行多层 VLAN Tag 的封装，只需要更改 VLAN Tag 标记即可，使其按照运营商的 VLAN 转发规则进行传输。

2.5.2 配置准备

场景

与 QinQ 区别的是，VLAN 转换只是变更 VLAN 标签，不额外进行多层 VLAN Tag 的封装，只需要更改 VLAN Tag 标记即可，使其按照运营商的 VLAN 转发规则进行传输，不会增加原报文的帧长度。VLAN 转换可以用在如下场景：

- 用户业务转换为一个运营商的 VLAN ID。
- 多种用户业务转换为一个运营商的 VLAN ID。

前提

在配置 VLAN 转换之前，需完成以下任务：

- 连接接口并配置接口的物理参数，使接口的物理层状态为 Up。
- 创建 VLAN。

2.5.3 VLAN 转换的缺省配置

设备上 VLAN 转换的缺省配置如下。

功能	缺省值
VLAN 转换功能状态	禁止

2.5.4 配置 VLAN 转换

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**interface** *interface-type interface-number*	进入二层物理接口配置模式。
3	JX(config-ge-1/0/*)#**vlan-mapping enable**	使能 VLAN 转换功能
4	JX(config-ge-1/0/*)#**vlan-mapping vlan** *vlan-id* **map-vlan** *vlan-id* **[remark-8021p** *cos-id***]**	配置单层 1:1 VLAN 转换规则。
5	JX(config-ge-1/0/*)#**vlan-mapping vlan** *vlan-id* **8021p** *cos-id* **map-vlan** *vlan-id* [**remark-8021p** *cos-id*]	配置单层 1:1 VLAN 匹配 cos 转换规则。
6	JX(config-ge-1/0/*)#**vlan-mapping vlan** *vlan-id* **inner-vlan** *vlan-id* **map-vlan** *vlan-id* **map-inner-vlan** *vlan-id* [**remark-8021p** *cos-id*]	配置内外层 2:2 VLAN 转换规则。
7	JX(config-ge-1/0/*)#**vlan-mapping vlan** *vlan-id* **inner-vlan** *vlan-id* **map-vlan** *vlan-id* [**remark-8021p** *cos-id*]	配置内外层 2:1 VLAN 转换规则。
8	JX(config-ge-1/0/*)#**vlan-mapping 8021p** *802.1p priority* **map-vlan** *vlan-id* [**remark-8021p** *cos-id*]	配置匹配 cos 转换规则。

说明

- 基本 QinQ 和 1:1 VLAN 转换命令行可同时配置，VLAN 转换在基本 QinQ 使能前后功能正常。
- 灵活 QinQ 和 1:1 VLAN 转换命令行可同时配置：同时配置时，灵活 QinQ 和 VLAN 转换功能正常；开启/关闭基本 QinQ，灵活 QinQ 和 VLAN 转换功能正常；取消其中一个配置命令，不影响其他配置正常使用。
- 基本 QinQ、灵活 QinQ 和 2:2 VLAN 转换互斥：同时配置灵活 QinQ 和 1:1 VLAN 转换时，匹配 VLAN 不能相同，转换后的 VLAN 不能相同。

2.5.5 检查配置

配置完成后，请在设备上执行以下命令检查配置结果。

序号	检查项	说明
1	JX#show vlan-mapping interface	查看接口的 VLAN 转换规则配置信息。

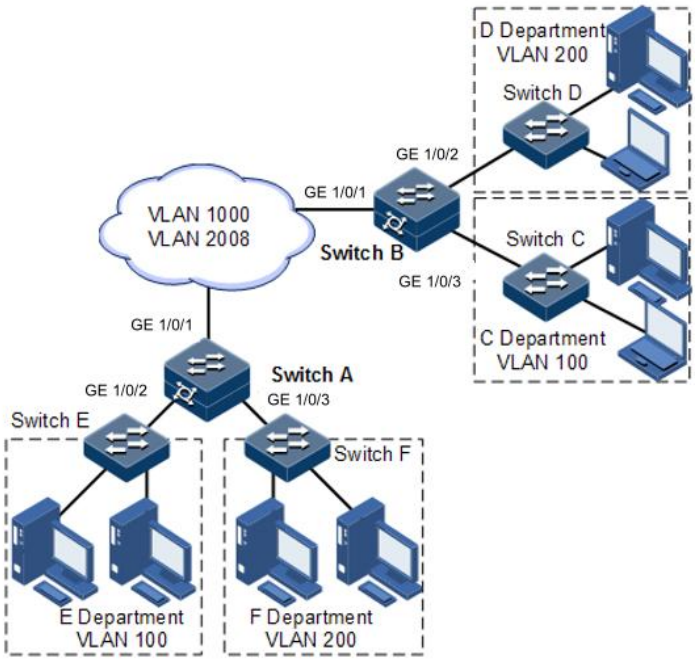
2.5.6 配置 VLAN 转换示例

组网需求

如下图所示，Switch A 的 GE1/0/2 和 GE1/0/3 分别连接使用 VLAN 100 的 E 部门和使用 VLAN 200 的 F 部门，Switch B 的 GE 1/0/2 和 GE 1/0/3 分别连接使用 VLAN 100 的 C 部门和使用 VLAN 200 的 D 部门。在运营商网络中为 E 部门和 C 部门分配 VLAN 1000 来传输，为 F 部门和 D 部门分配使用 VLAN 2008 来传输。

通过在 Switch A 和 Switch B 配置 1:1 的 VLAN 转换功能来实现 PC 用户和终端用户与其服务器之间的正常通信。

图 2-10 VLAN 转换应用组网示意图



配置步骤

Switch A 和 Switch B 配置相同，在这里只描述 Switch A 的配置信息。

步骤 1 创建 VLAN 并激活。

```
JX#hostname SwitchA
SwitchA#config
SwitchA(config)#vlan 100,200,1000,2008
```

步骤 2 配置接口 GE 1/0/1 为 Trunk 模式，允许 VLAN 1000 和 VLAN 2008 通过。

```
SwitchA(config)#interface ge 1/0/1
SwitchA(config-ge-1/0/1)#port link-type trunk
SwitchA(config-ge-1/0/1)#port trunk allow-pass vlan 1000,2008
SwitchA(config-ge-1/0/1)#exit
```

步骤 3 配置接口 GE 1/0/2 为 Trunk 模式，允许 VLAN 100 通过，配置 VLAN 转换规则。

```
SwitchA(config)#interface ge 1/0/2
SwitchA(config-ge-1/0/2)#port link-type trunk
SwitchA(config-ge-1/0/2)#port trunk allow-pass vlan 100
SwitchA(config-ge-1/0/2)#vlan-mapping vlan 100 map-vlan 1000
SwitchA(config-ge-1/0/2)#exit
```

步骤 4 配置接口 GE 1/0/3 为 Trunk 模式，允许 VLAN 200 通过，配置 VLAN 转换规则。

```
SwitchA(config)#interface ge 1/0/3
SwitchA(config-ge-1/0/3)#port link-type trunk
SwitchA(config-ge-1/0/3)#port trunk allow-pass vlan 100
SwitchA(config-ge-1/0/3)#vlan-mapping vlan 200 map-vlan 2008
```

检查结果

通过 **show vlan-mapping both interface** 命令查看 1:1 VLAN 转换配置。

```
SwitchA#show vlan-mapping interface
 Interface         MatchVlan(vlan/8021p)       MapVlan(vlan/8021p)
-------------------------------------------------------------------------
--
 ge-1/0/2           inner: -                    inner: -/-
                   outer: 100/-                outer: 1000/-
 ge-1/0/3           inner: -                    inner: -/-
                   outer: 200/-                outer: 2008/-
-------------------------------------------------------------------------
--
```

2.6 MRP/MVRP

2.6.1 简介

MRP（Multiple Registration Protocol，多属性注册协议）作为一个属性注册协议的载体，可以用来传递属性信息。MVRP（Multiple VLAN Registration Protocol，多 VLAN 注册协议）是 MRP 的一种应用，用于在设备间发布并学习 VLAN 配置信息。通过 MVRP，局域网中的设备可以自动同步 VLAN 信息，极大地减少了网络管理员的 VLAN 配置工作。

MRP 消息

MRP 消息主要包括 Join 消息、New 消息、Leave 消息和 LeaveAll 消息

- Join 消息：当一个 MRP 实体配置了某些属性，需要对端实体来注册自己的属性信息时，它会向对端实体发送 Join 消息。当一个 MRP 实体收到来自对端实体的 Join 消息时，它会注册该 Join 消息中的属性，并向本设备的其他实体传播该 Join 消息，其他实体收到传播的 Join 消息后，向其对端实体发送 Join 消息。
- New 消息：New 消息的作用和 Join 消息比较类似，都是用于对属性的声明。不同的是，New 消息主要用于 MSTP（Multiple Spanning Tree Protocol，多生成树协议）拓扑变化的情况。
- Leave 消息：当一个 MRP 实体注销了某些属性，需要对端实体进行同步注销时，它会向对端实体发送 Leave 消息。当一个 MRP 实体收到来自对端实体的 Leave 消息时，它会注销该 Leave 消息中的属性，并向本设备的其他实体传播该 Leave 消息，其他实体收到传播的 Leave 消息后，根据该 Leave 消息中的属性在本设备上的状态，决定是否向其对端实体发送该 Leave 消息（比如该 Leave 消息中的属性为某 VLAN，若该 VLAN 为动态 VLAN，且本设备上无实体注册该 VLAN，则在设备上删除该 VLAN，并向对端实体发送该 Leave 消息；若该 VLAN 为静态 VLAN，则不向对端实体发送该 Leave 消息）。
- LeaveAll 消息：每个 MRP 实体启动时都会启动各自的 LeaveAll 定时器，当该定时器超时后，MRP 实体就会向对端实体发送 LeaveAll 消息。当一个 MRP 实体收发 LeaveAll 消息时，它会启动 Leave 定时器，同时根据自身的属性状态决定是否发送 Join 消息要求对端实体重新注册某属性。该实体在 Leave 定时器超时前，重

新注册收到的来自对端实体的 Join 消息中的属性；在 Leave 定时器超时后，注销所有未重新注册的属性信息，从而周期性地清除网络中的垃圾属性。

Leave 消息、LeaveAll 消息通过与 Join 消息配合，确保属性的注销或重新注册。通过消息交互，所有待注册的属性信息可以传播到同一局域网内使能 GARP 功能的所有设备上。

MRP 定时器

MRP 消息发送的时间间隔通过定时器来实现，MRP 定义了四种定时器，用于控制 MRP 消息的发送周期。

- Periodic 定时器：每个 MRP 实体启动时都会启动各自的 Periodic 定时器，来控制 MRP 消息的周期发送。该定时器超时前，实体收集需要发送的 MRP 消息，在该定时器超时后，将所有待发送的 MRP 消息封装成尽可能少的报文发送出去，这样减少了报文发送数量。随后再重新启动 Periodic 定时器，开始新一轮的循环。
- Join 定时器：用来控制 Join 消息的发送。为了保证消息能够可靠地发送到对端实体，MRP 实体在发送 Join 消息时，将启动 Join 定时器。如果在该定时器超时前收到了来自对端实体的 JoinIn 消息，且该 JoinIn 消息中的属性与发出的 Join 消息中的属性一致，便不再重发该 Join 消息，否则在该定时器超时后，当 Periodic 定时器也超时，它将重发一次该 Join 消息。
- Leave 定时器：用来控制属性的注销。当 MRP 实体收到来自对端实体的 Leave 消息（或收发 LeaveAll 消息）时，将启动 Leave 定时器。如果在该定时器超时前，收到来自对端实体的 Join 消息，且该 Join 消息中的属性与收到的 Leave 消息中的属性一致（或与收发的 LeaveAll 消息中的某些属性一致），则这些属性不会在本实体被注销，其他属性则会在该定时器超时后被注销。
- LeaveAll 定时器：每个 MRP 实体启动时都会启动各自的 LeaveAll 定时器，当该定时器超时后，该实体就会向对端实体发送 LeaveAll 消息，随后再重新启动 LeaveAll 定时器，开始新一轮的循环，对端实体在收到 LeaveAll 消息后也重新启动 LeaveAll 定时器。

MVRP

MVRP（Multiple VLAN Registration Protocol，多 VLAN 注册协议）是 MRP 的一种应用，它基于 MRP 的工作机制，维护交换机中的 VLAN 动态注册信息，并传播该信息到其它交换机。

所有支持 MVRP 特性的交换机能够接收来自其它交换机的 VLAN 注册信息，并动态更新本地的 VLAN 注册信息。而且所有支持 MVRP 特性的交换机能够将本地的 VLAN 注册信息向其它交换机传播，以便使同一交换网内所有支持 MVRP 特性的设备 VLAN 信息达成一致。MVRP 传播的 VLAN 注册信息既包括本地手工配置的静态注册信息，也包括来自其它交换机的动态注册信息。

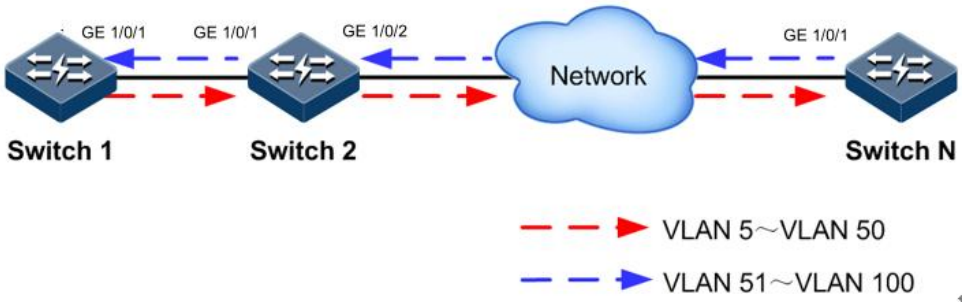
MVRP 有三种注册模式：

- 正常模式（Normal）：允许动态注册、注销 VLAN，传播动态、静态 VLAN 信息。
- 固定模式（Fixed）：禁止动态注册、注销 VLAN，只传播静态 VLAN 信息，不传播动态 VLAN 信息，只允许静态 VLAN 通过，即只对其他 MVRP 成员传播静态 VLAN 信息。

- 禁止模式（Forbidden）：禁止动态注册、注销 VLAN，禁止静态 VLAN 在接口上的创建，同时删除接口上除 VLAN 1 外的所有 VLAN，只允许缺省 VLAN（即 VLAN 1）通过，只对其他 MVRP 成员传播缺省 VLAN 的信息。

存在多台设备的网络中，如图 2-11 所示，如果想在每台设备上配置 VLAN 信息，并允许指定 VLAN 的报文通过是比较繁琐的。采用 MVRP 来动态注册和传播指定 VLAN，可以帮助网管人员提高工作效率，提高准确性。

图 2-11 MVRP 原理示意图



在图 2-11 中，Switch 1 的 GE 1/0/1，Switch 2 的 GE 1/0/1 和 GE 1/0/2，Switch N 的 GE 1/0/1 均为 Trunk 接口。在 Switch 1 中创建 VLAN 5～VLAN 50，则会按照红色连接线方向在接收接口上动态注册该 VLAN，直到传播到 Switch N 交换机。在 Switch N 上创建 VLAN 51～VLAN 100，则沿蓝色连接线注册该 VLAN 信息，使每个交换机可以完全处理 VLAN 5～VLAN 100 的报文。

2.6.2 配置准备

场景

利用 MVRP 机制，一个 MVRP 成员上的配置信息会迅速传播到整个局域网中所有使能 MVRP 功能的设备上。

通过 MVRP 功能配置的 Join 定时器、Leave 定时器和 LeaveAll 定时器的值，将应用于所有在同一网络内运行的 MVRP 应用。

前提

无

2.6.3 缺省配置

MVRP 缺省配置

设备上 MVRP 的缺省配置如下

功能	缺省值
全局 MVRP 功能状态	禁止

功能	缺省值
接口 MVRP 功能状态	禁止
MVRP 注册模式	Normal

2.6.4 配置 MVRP 基本功能

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**mvrp start**	使能全局 MVRP 功能。
3	JX(config)#**interface** *interface-type interface-number*	进入物理接口配置模式。
4	JX(config-ge-1/0/1)#**port link-type trunk**	配置接口为 Trunk 模式。
5	JX(config-ge-1/0/1)#**mvrp registration** { **fixed** | **forbidden** | **normal** }	配置 MVRP 注册模式。
6	JX(config-ge-1/0/1)#**mvrp enable**	使能接口 MVRP 功能。

注意

- 接口必须先配置为 Trunk 模式才能使能接口 MVRP 功能。
- 不建议用户在聚合组成员接口上使能 MVRP 功能。

2.6.5 检查配置

配置完成后，请在设备上执行以下命令检查配置结果。

序号	检查项	说明
1	JX#**show mvrp config**	查看 MVRP 的配置信息。
2	JX#**show mvrp**	查看 MVRP 的接口状态信息。
3	JX#**show mvrp interface** *interface-type interface-number* **vlan** *vlan-id*	查看 MVRP 的定时器信息。

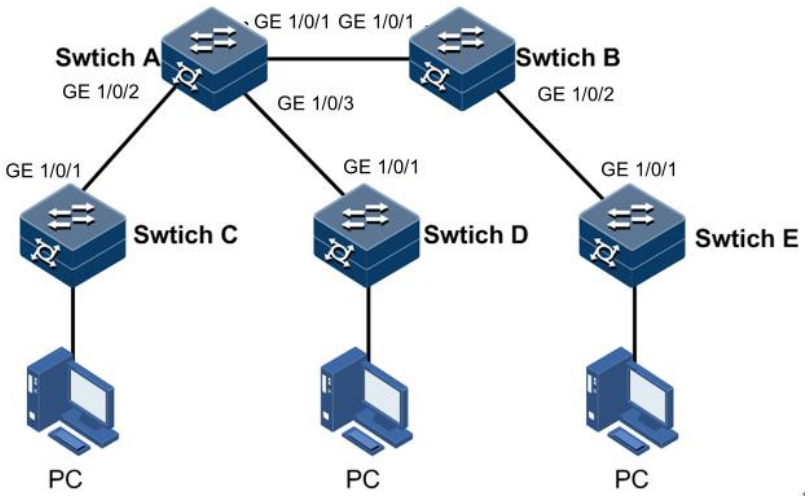
2.6.6 配置 MVRP 应用示例

组网需求

为了在交换机之间动态注册和更新 VLAN 信息，如图 2-12 所示，需要在各交换机上配置 MVRP 功能，来实现各交换机设备之间 VLAN 信息的动态注册和注销。具体要求如下：

- 在 Switch A 和 Switch C 中配置静态 VLAN 5～VLAN 10。
- Switch D 中配置静态 VLAN 15～VLAN 20。
- Switch E 中配置静态 VLAN 25～VLAN 30。
- 将所有与其他交换机相连的接口设置为 Trunk 模式，然后使能接口的 MVRP 功能。
- 配置各接口 MVRP 的 Join 定时器、Leave 定时器和 LeaveAll 定时器分别为 3000、15000 和 20000。

图 2-12 MVRP 应用组网示意图



配置步骤

步骤 1 创建 VLAN 并使能全局 MVRP 功能。

配置 Switch A。

```
JX#hostname SwitchA
SwitchA#config
SwitchA(config)#vlan 5-10
SwitchA(config)#mvrp start
```

配置 Switch B。

```
JX#hostname SwitchB
SwitchB#config
SwitchB(config)#mvrp start
```

配置 Switch C。

```
JX#hostname SwitchC
SwitchC#config
SwitchC(config)#vlan 5-10
SwitchC(config)#mvrp start
```

配置 Switch D。

```
JX#hostname SwitchD
SwitchD#config
SwitchD(config)#vlan 15-20
SwitchD(config)#mvrp start
```

配置 Switch E。

```
JX#hostname SwitchE
SwitchE#config
SwitchE(config)#vlan 25-30
SwitchE(config)#mvrp start
```

步骤 2 分别配置 Switch A 的 GE 1/0/1、GE 1/0/2、GE 1/0/3 接口，Switch B 的 GE 1/0/1、GE 1/0/2，Switch C 的 GE 1/0/1，Switch D 的 GE 1/0/1 接口为 Trunk 模式，并使能接口的 MVRP 功能。以配置 Switch A 的 GE 1/0/1 接口为例，其余接口配置步骤相同。

```
SwitchA(config)#interface ge 1/0/1
SwitchA(config-ge-1/0/1)#port link-type trunk
SwitchA(config-ge-1/0/1)#port trunk allow-pass vlan all
SwitchA(config-ge-1/0/1)#mvrp enable
SwitchA(config-ge-1/0/1)#exit
```

步骤 3 分别配置 Switch A 的 GE 1/0/1、GE 1/0/2、GE 1/0/3 接口、Switch B 的 GE 1/0/1、GE 1/0/2，Switch C 的 GE 1/0/1，Switch D 的 GE 1/0/1 接口的 GARP 定时器时间。以配置 Switch A 为例，其余接口配置步骤相同。

配置 Switch A。

```
SwitchA(config)#interface ge 1/0/1
SwitchA(config-ge-1/0/1)#mvrp timer join 3000
SwitchA(config-ge-1/0/1)#mvrp timer leave 15000
SwitchA(config-ge-1/0/1)#mvrp timer leave-all 20000
```

检查结果

通过 **show mvrp config** 命令查看设备上接口的 MVRP 配置信息，以 Switch A 为例。

```
SwitchA#show mvrp config

 Version              : MVRP_V13.10.00.00

 Compliance-GVRP       : disable

 Interface        JoinTime(ms)    LeaveTime(ms)   LeaveAllTime(ms)
PeriodicTime(ms)  Mode     State

--------------------------------------------------------------------------
--------------------------------

GE 1/0/1         3000          15000         20000          --
normal    enable
```

通过 **show vlan** 命令查看设备上的 VLAN 信息，以 Switch A 为例。

```
SwitchA#show vlan
NOTE:
    S: supervlan       P: pvlan        N: normal


 Vlan  Type       Ports('M': member , '-': not member)
--------------------------------------------------------------------------------
-------
 1    N/static    untagged: ge-1/0/1<->ge-1/0/16          MMMMMMMM
MMMMMMMM
                         ge-1/0/17<->ge-1/0/32          MMMMMMMM  MMMMMMMM
                         ge-1/0/33<->ge-1/0/48          MMMMMMMM  MMMMMMMM
                         10ge-1/0/49<->10ge-1/0/54      MMMMMM
 2    N/mvrp      tagged:   ge-1/0/1<->ge-1/0/16          M------- --------
 3    N/mvrp      tagged:   ge-1/0/1<->ge-1/0/16          M------- --------
 4    N/mvrp      tagged:   ge-1/0/1<->ge-1/0/16          M------- --------
 5    N/mvrp      tagged:   ge-1/0/1<->ge-1/0/16          M------- --------
 6    N/mvrp      tagged:   ge-1/0/1<->ge-1/0/16          M------- --------
 7    N/mvrp      tagged:   ge-1/0/1<->ge-1/0/16          M------- --------
 8    N/mvrp      tagged:   ge-1/0/1<->ge-1/0/16          M------- --------
 9    N/mvrp      tagged:   ge-1/0/1<->ge-1/0/16          M------- --------
 10   N/mvrp      tagged:   ge-1/0/1<->ge-1/0/16          M------- --------
 20   N/mvrp      tagged:   ge-1/0/1<->ge-1/0/16          M------- --------
--------------------------------------------------------------------------------
-------
  Total: 11   Static: 1   Dynamic: 10
```

2.7 VTP

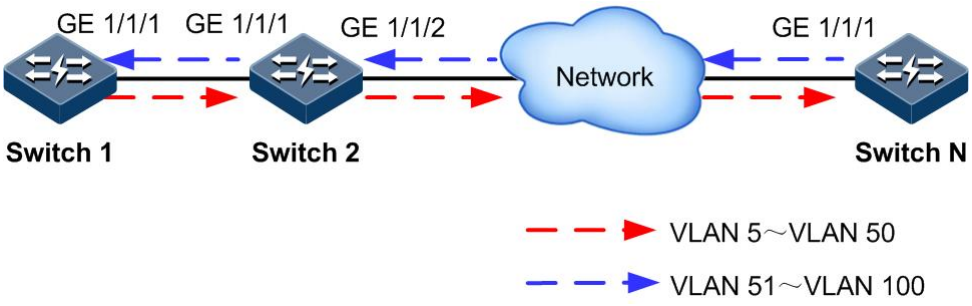
2.7.1 简介

VTP 协议（VLAN TRUNK PROTOCOL） 减少了在交换网络中的配置工作。当在 vtp server 上配置新的 vlan 时，这些 vlan 配置会通过交换机在同域中散播。这就减少了在各台交换机上都必须配置相同 vlan 的操作。

相关概念

- VTP 域：也称为 VLAN 管理域，由一个以上共享 VTP 域名的相互接连的交换机组成。要使用 VTP，就必须为每台交换机指定 VTP 域名．VTP 信息只能在 VTP 域内保持。Vlan 信息只在相同域中进行同步。
- 服务器(server)模式：该模式提供 VTP 消息：包括 VLAN ID 和名字信息，学习相同域名的 VTP 消息，转发相同域名的 VTP 消息，可以添加、删除和更改 VLAN。设置成功后进行 vlan 的配置，包括添加、删除和更改 VLAN。配置成功后，server 会发出汇总通告，通知该 VTP 域内的交换机 vlan 信息更改情况。然后发送一个或多个子集通告．添加、删除、修改某个 VLAN，改变 VLAN 的名称都会触发子集通告。如果有多个 VLAN，为了通告所有的信息，可能需要发送多个子集通告。Server 模式重启后，会自动发汇总通告和子集通告，用于 vtp 域内信息的同步。
- 客户机(client)模式：该模式请求 VTP 消息，学习相同域名的 VTP 消息，转发相同域名的 VTP 消息，可以配置静态 vlan，但配置的静态 vlan 不会通告出去，配置静态 vlan 也不会修改配置修订号。将 client 上的静态 vlan 删除后，client 的配置修订号会重置为 0，同时向 server 发送请求通告，用于同步 vlan 信息。将一台交换机配置成 client 模式时，该交换机会向 server 发送通告请求，之后收到汇总通告和子集通告，将自身配置和 vtp 域内配置统一。
- 透明传输(transparent)模式：该模式中的交换机不参与 VTP。当交换机处于透明模式时，它不通告其 VLAN 配置信息。而且，它的 VLAN 数据库更新与收到的通告也不保持同步．但它可以创建和删除本地的 VLAN．不过，这些 VLAN 的变更不会传播到其他任何交换机上。该模式转发 VTP 包。当 VTP 从 client 以及 server 切换到 transparent 模式时，配置修订号会置 0。透明模式的交换机在转发 VTP 消息之前要检查 VTP 消息中的域名和版本，只有两者合本机相同才转发。
- 修订号：VTP 通告信息中带有配置修订号，配置修订号的大小代表着 VLAN 配置信息的新旧程度。高版本号代表更新的 VLAN 配置信息。只要客户机接收到一个有更高配置修订号的更新，它都用该 VTP 更新中的 vlan 信息覆盖当前的 vlan 信息。每当服务器上修改了 VLAN 的配置(修改包括创建、删除 VLAN、更改 VLAN 的名称)，其配置修订版本号就会增加，然后用新的版本号向域中通告。

VTP 原理示意图



在 Switch 1 中创建 VLAN 5～VLAN 50，则会按照红色连接线方向在接收接口上动态注册该 VLAN，直到传播到 Switch N 交换机。

2.7.2 配置准备

场景

利用第 2 层中继帧，在一组交换机之间进行 VLAN 通信．VTP 从一个中心控制点开始，维护整个企业网上 VLAN 信息，确认配置的一致性。

前提

- 连接接口并配置接口的物理参数，使接口的物理层状态为 Up；
- 将接口加入 VLAN。
- 如果接口配置有环协议，请确保接口没有被环协议阻塞

2.7.3 缺省配置

VTP 缺省配置

设备上 MVRP 的缺省配置如下

功能	缺省值
全局 VTP 功能状态	禁止
VTP 域默认模式	Server
VTP 版本	1

2.7.4 配置 VTP 基本功能

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**configure**	进入全局配置模式。
2	JX(config)#**vtp domain** *name*	创建 VTP 域。
3	JX(config)#**password** *password*	配置 VTP 域密码
4	JX(config-vtp-domain-test)#**mode client**	设置 VTP 域的模式为客户机模式 client。
5	JX(config-ge-1/0/*)#**join vtp domain** *name*	接口加入 vtp 域。

2.7.5 检查配置

配置完成后，请在设备上执行以下命令检查配置结果。

序号	检查项	说明
1	JX#show vtp domain	查看 VTP 的域配置信息。
2	JX#show vtp interface	查看 VTP 的接口状态信息。

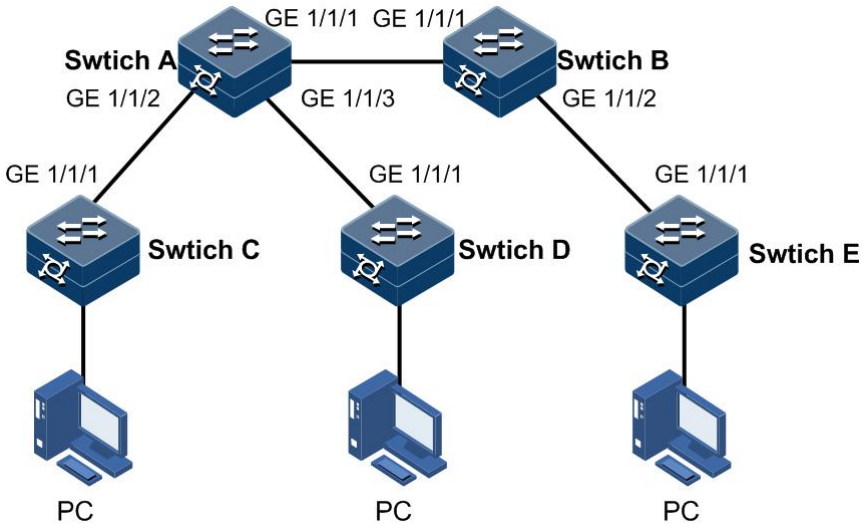
2.7.6 配置 VTP 应用示例

组网需求

为了在交换机之间动态注册和更新 VLAN 信息，如图 2-12 所示，需要在各交换机上配置 VTP 功能，来实现各交换机设备之间 VLAN 信息的动态注册和注销。具体要求如下：

- Switch A 作为 vtp 的 server 工作，配置静态 VLAN 5～VLAN 100。
- 其他 switch 作为 vtp 的 client，接收并创建 VLAN 5～VLAN 100。

图 2-13 VTP 应用组网示意图



配置步骤

步骤 1 创建 VTP 域并配置各自的工作模式。

配置 Switch A。

```
JX#hostname SwitchA
SwitchA#config
SwitchA(config)#vtp domain A
```

配置 Switch B。

```
JX#hostname SwitchB
SwitchB#config
SwitchB(config)#vtp domain A
SwitchB(config-vtp-domain-A)#mode client
```

其他交换机同 Switch B。

步骤 2 分别配置 Switch A 的 GE 1/0/1、GE 1/0/2、GE 1/0/3 接口，Switch B 的 GE 1/0/1、GE 1/0/2，Switch C 的 GE 1/0/1，Switch D 的 GE 1/0/1 接口加入 vtp 域。以配置 Switch A 的 GE 1/0/1 接口为例，其余接口配置步骤相同。

```
SwitchA(config)#interface ge 1/0/1
SwitchA(config-ge-1/0/1)#join vtp domain A
SwitchA(config-ge-1/0/1)#port hybrid vlan 5-100 tagged
SwitchA(config-ge-1/0/1)#exit
```

步骤 3 在 Swicth A 上创建 vlan 5-100

```
SwitchA(config)#vlan 5-100
```

检查结果

通过 **show vtp domain** 命令查看设备上的 VTP 域配置信息，以 Switch B 为例。

```
SwitchB#show vtp domain
 Domain: 1
  Name:                 A
  Version:               1
  Pruning:               Disable
  Password:              --
  Mode:                 client
  Configure Revision Num   2
  Vlan:                 1,5-100
```

通过 **show vtp interface** 命令查看设备上的 VTP 接口配置信息，以 Switch B 为例。

```
SwitchB#show vtp interface
  Interface      State    Domain     VlanList
--------------------------------------------------------------------------
  ge-1/0/1       enable   A          2-4094
  ge-1/0/2       enable   A          2-4094
```

通过 **show vlan 命令查看接口加入 vlan 的情况**

```
SwitchB(config)#show vlan
NOTE:
    S: supervlan       P: pvlan        N: normal

 Vlan  Type       Ports('M': member , '-': not member)
```

```
-------------------------------------------------------------------------
-------
 1    N/static   untagged: ge-1/0/1<->ge-1/0/16          MMMMMMMM
MMMMMMMM
                          ge-1/0/17<->ge-1/0/32         MMMMMMMM MMMMMMMM
                          ge-1/0/33<->ge-1/0/48         MMMMMMMM MMMMMMMM
                          10ge-1/0/49<->10ge-1/0/54     MMMMMM
 5    N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 6    N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 7    N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 8    N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 9    N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 10   N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 11   N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 12   N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 13   N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 14   N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 15   N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 16   N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 17   N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 18   N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 19   N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 20   N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 21   N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 22   N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 23   N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 24   N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 25   N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 26   N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 27   N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 28   N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 29   N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 30   N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 31   N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 32   N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 33   N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 34   N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 35   N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 36   N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 37   N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 38   N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 39   N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 40   N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 41   N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 42   N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 43   N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 44   N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 45   N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 46   N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 47   N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 48   N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 49   N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 50   N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 51   N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 52   N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
```

```
 53   N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 54   N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 55   N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 56   N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 57   N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 58   N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 59   N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 60   N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 61   N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 62   N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 63   N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 64   N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 65   N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 66   N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 67   N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 68   N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 69   N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 70   N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 71   N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 72   N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 73   N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 74   N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 75   N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 76   N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 77   N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 78   N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 79   N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 80   N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 81   N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 82   N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 83   N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 84   N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 85   N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 86   N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 87   N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 88   N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 89   N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 90   N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 91   N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 92   N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 93   N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 94   N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 95   N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 96   N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 97   N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 98   N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 99   N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
 100  N/vtp      tagged:   ge-1/0/1<->ge-1/0/16          MM------ --------
------------------------------------------------------------------------
-------
 Total: 97   Static: 1   Dynamic: 96
```

3 IP 业务

本章介绍 IP 业务特性的基本原理和配置过程，并提供相关的配置案例。

- IP 基础配置
- LOOPBACK 接口
- ARP
- NDP
- 静态路由
- 策略路由
- BGP
- OSPFV2
- OSPFV3
- IS-IS
- RIPng
- ECMP

3.1 IP 基础配置

3.1.1 简介

IP 接口是基于 VLAN 的虚拟接口。一般应用于需要对设备进行网管或多台设备之间需要进行路由连通的场合。

设备支持管理 VLAN 报文为双 TAG，能够发送和处理双 TAG 报文。

3.1.2 配置准备

场景

为每一个 VLAN 接口、SNMP 或 LOOPBACK 接口配置 IP 地址。

前提

为 VLAN 接口配置 IP 地址前，需创建 VLAN 并激活。

3.1.3 VLAN 接口的缺省配置

设备上 VLAN 接口的缺省配置如下。

功能	缺省值
管理 VLAN 内层 TPID	0x8100
管理 VLAN 内层 VLAN	1
SNMP 接口 IP 地址	192.168.0.1

3.1.4 配置 VLAN 接口 IPv4 地址

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**vlan** *vlan-id*	创建 VLAN。
3	JX(config)#**interface vlan** *vlan-id*	进入 VLAN 接口配置模式。
4	JX(config-vlanif-*)#**ip address** *ip-address* [*ip-mask*]	配置 VLAN 接口的主 IP 地址。 可以使用 **no ip address** *ip-address* 命令删除主 IP 地址配置。
5	JX(config-vlanif-*)#**ip address** *ip-address* [*ip-mask*] **sub**	配置接口的从 IP 地址。 可以使用 **no ip address** *ip-address* 命令删除从 IP 地址配置。

3.1.5 配置 VLAN 接口 IPv6 地址

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**vlan** *vlan-id*	创建 VLAN。
3	JX(config)#**interface vlan** *vlan-id*	进入 VLAN 接口配置模式。

步骤	配置	说明
4	JX(config-vlanif-*)#**ipv6 enable**	配置 VLAN 接口 IPv6 使能。
5	JX(config-vlanif-*)#**ipv6 address** *ipv6-address*/*prefix-length* JX(config-vlanif-*)#**ipv6 address** *ipv6-address* **link-local**	配置 VLAN 接口的 IPv6 地址。

3.1.6 检查配置

配置完成后，请在设备上执行以下命令检查配置结果。

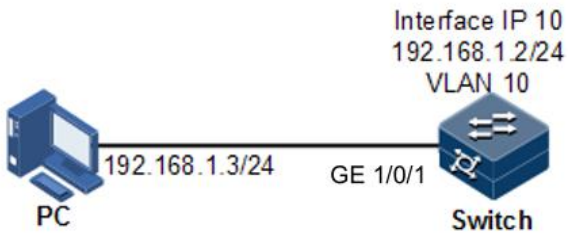
序号	检查项	说明
1	JX#**show ip interface**	查看 VLAN 接口的 IP 地址配置信息。
2	JX#**show ipv6 interface**	查看 VLAN 接口的 IPv6 地址配置信息。

3.1.7 配置 VLAN 接口 IP 地址实现和主机互通示例

组网需求

配置交换机设备 VLAN 接口，使图 3-1 中的主机和设备之间能够相互 Ping 通。

图 3-1 配置 VLAN 接口组网示意图



配置步骤

步骤 1 创建 VLAN，并将接口加入 VLAN。

```
JX#config
JX(config)#vlan 10
JX(config)#interface ge 1/0/1
JX(config-ge-1/0/1)#port hybrid pvid 10
```

```
JX(config-ge-1/0/1)#port hybrid vlan 10 untagged
JX(config-ge-1/0/1)#quit
```

步骤 2 在交换机设备上创建三层接口，配置三层接口的 IP 地址并关联 VLAN。

```
JX(config)#interface vlan 10
JX(config-vlan10)#ip address 192.168.1.2 255.255.255.0
```

检查结果

通过 **show vlan** 命令查看 VLAN 和物理接口的绑定关系是否正确。

```
JX#show vlan 10

vlan-10 information:
------------------------------------------------------------
  Description                :
  Admin state                : up
  Operation state             : down
  Vlan type                  : normal
  Vlan status                : static
  Unknown multicast state      : forward
  Unknown unicast state        : forward
  IPv4 address total number     : 0
  IPv6 address total number     : 0

  Ports :
    Interface                 Tagged
    ------------------------------------
    ge-1/0/1                  Untag
------------------------------------------------------------
```

通过 **show ip interface** 命令查看三层接口配置是否正确。

```
JX#show ip interface
 Total number: 2
 Interface         State(a/o) Addr/Prefix       Role     Type      Vpn-
instance
------------------------------------------------------------------------------
---------------------------
 loopback-0         up/up     127.0.0.1/8       primary   auto      N/A
 vlan-10           up/up    192.168.1.2/24    primary   static    N/A
------------------------------------------------------------------------------
---------------------------
```

通过 **ping** 命令查看设备和 PC 之间是否能够互通。

```
JX#ping 192.168.1.3
PING 192.168.1.3: 64 data bytes
Reply from 192.168.1.3: bytes=64 time=0ms TTL=64 icmp_seq=1
Reply from 192.168.1.3: bytes=64 time=0ms TTL=64 icmp_seq=2
Reply from 192.168.1.3: bytes=64 time=0ms TTL=64 icmp_seq=3
Reply from 192.168.1.3: bytes=64 time=0ms TTL=64 icmp_seq=4
Reply from 192.168.1.3: bytes=64 time=0ms TTL=64 icmp_seq=5
PING Statistics for 192.168.1.3
```

```
5 packets transmitted, 5 packets received, 0% packet loss
round-trip (ms)  min/avg/max = 0/0/0
```

3.2 LOOPBACK 接口

3.2.1 简介

LOOPBACK 接口是一种虚拟接口，可以分为两类：

- 由系统自动创建的 LOOPBACK 接口。IP 地址固定为 127.0.0.1，用来接收所有发送给本机的数据包。不通过路由协议对外发布。
- 由用户创建的 LOOPBACK 接口。在不影响物理接口配置的情况下，配置一个带有指定 IP 地址的本地接口，并且接口状态永远是 up 状态，能够被路由协议发布出去。

LOOPBACK 接口状态不受物理接口 Up/Down 的影响，只要保证设备运行正常，该 LOOPBACK 接口就不会 Down 掉。因此，LOOPBACK 接口地址常被用来标示物理设备本身，作为设备的管理地址。

3.2.2 配置准备

场景

使用 LOOPBACK 接口 IP 地址对设备进行 Telnet 登录，可以保证 Telnet 操作不会因接口的物理状态改变为 Down 掉，如果 PC 需要 ping 通 LOOPBACK 接口地址，需要 PC 端设置相对应的静态路由表项。LOOPBACK 接口还常被用来作为动态路由协议如 OSPF 等协议的 Router ID，作为设备的唯一标识。

前提

无

3.2.3 LOOPBACK 接口的缺省配置

无

3.2.4 配置 LOOPBACK 接口 IP 地址

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**interface loopback** *loopback-number*	进入 LOOPBACK 接口配置模式。
3	JX(config-loopback-*)#**ip address** *ip-address* [*ip-mask*] [**sub**]	配置 LOOPBACK 接口的 IP 地址。
4	JX(config-loopback-*)#**ipv6 enable**	配置 LOOPBACK 接口 IPv6 使能。

步骤	配置	说明
4	JX(config-loopback*)#**ipv6 address** *ipv6-address/prefix-length* JX(config-loopback-*)#**ipv6 address** *ipv6-address* **link-local**	配置 LOOPBACK 接口的 IPV6 地址。

3.2.5 检查配置

配置完成后，请在设备上执行以下命令检查配置结果。

序号	检查项	说明
1	JX#**show interface loopback** [*loopback-number*]	查看 LOOPBACK 接口配置信息。

3.3 ARP

3.3.1 简介

在 TCP/IP 网络环境中，每台主机都被分配了一个 32 位的 IP 地址，这种互联网地址是在网际范围内标识主机的一种逻辑地址。为了让报文在物理链路中传送，必须知道目的主机的物理地址。这就需要把 IP 地址通过映射关系转换成物理地址。在以太网环境中采用的物理地址是 48 位 MAC 地址，为了正确地向目的主机传送报文，必须把目的主机的 32 位 IP 地址转换成为 48 位以太网的地址。为此产生了 ARP（Address Resolution Protocol，地址解析协议），该协议主要用于 IP 地址到 MAC 地址的解析，建立 IP 地址和 MAC 地址的映射关系。

ARP 地址映射表项包括以下两种类型：

- 静态表项：静态表项是将 IP 地址和 MAC 地址进行静态绑定，用于防止 ARP 动态学习欺骗。
  - 静态 ARP 地址表项需要手动添加，手动删除。
  - 静态 ARP 地址不老化。
- 动态表项：设备通过 ARP 协议自动学习到的 MAC 地址。
  - 动态表项生成由交换机自动完成，不需要手动配置，可以调整动态 ARP 的一些参数。
  - 如果不使用，到达老化时间会老化。

3.3.2 配置准备

场景

IP 地址和 MAC 地址的映射关系保存在 ARP 地址映射表中。

一般情况下，ARP 地址映射表项由设备动态维护，设备按照 ARP 协议自动寻找 IP 地址和 MAC 地址之间的映射关系，无需用户关注。只有在防止 ARP 动态学习欺骗，需要添加静态 ARP 地址映射表项时，才需要对设备进行手动配置。

前提

无

3.3.3 ARP 的缺省配置

设备上 ARP 的缺省配置如下。

功能	缺省值
静态 ARP 表项	无
动态 ARP 老化时间	1200 秒

3.3.4 配置静态 ARP 表项

注意

- 静态添加的 ARP 表项的 IP 地址必须属于交换机三层接口所属的 IP 网段。
- 静态 ARP 表项，需要手动添加和手动删除。

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**arp static** *ip-address mac-address* (ge|10ge|25ge|40ge|100ge|400ge) *interface-number*	配置静态 ARP 表项。

3.3.5 配置动态 ARP 表项

请在需要配置的设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
3	JX(config)#**arp aging-time** *(aging-time | default)*	配置动态 ARP 老化时间。

3.3.6 检查配置

配置完成后，请在设备上执行以下命令检查配置结果。

序号	检查项	说明
1	JX#**show arp**	查看 ARP 地址映射表中的表项信息。

3.3.7 维护

用户可以通过以下命令，维护设备 ARP 特性的运行情况和配置情况。

命令	描述
JX(config)#**flush arp** [*all* / *dynamic*/ *static*]	清空 ARP 地址映射表中表项。

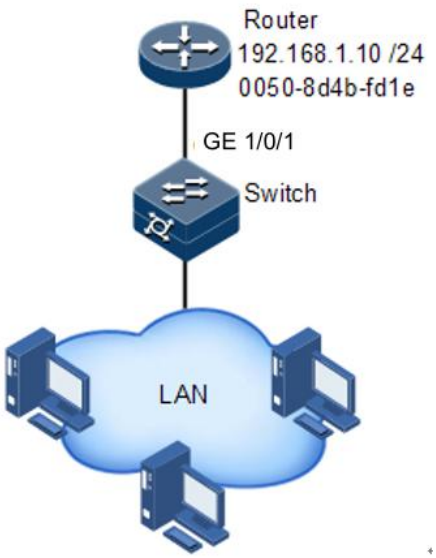
3.3.8 配置 ARP 示例

组网需求

如图 3-2 所示，交换机设备连接主机，通过接口 GE 1/0/1 连接上游的 Router。Router 的 IP 地址为 192.168.1.10/24，MAC 地址为 0050.8d4b.fd1e。

为了增加交换机和 Router 通信的安全性，需要在交换机上配置相应的静态 ARP 表项。

图 3-2 配置 ARP 组网示意图



配置步骤

创建 VLAN3。

```
JX#configure
JX(config)#vlan 3
```

将接口 GE1/0/1 加入 VLAN3 中。

```
JX(config)#interface ge 1/0/1
JX(config-ge-1/0/1)#port hybrid pvid 3
JX(config-ge-1/0/1)#port hybrid vlan 3 untagged
JX(config-ge-1/0/1)#quit
```

创建接口 VLANIF3。

```
JX(config)#interface vlan 3
```

配置接口 VLANIF3 的 IP 地址。

```
JX(config-vlanif-3)#ip address 192.168.1.1/24
JX(config-vlanif-3)#quit
```

配置增加一条 ARP 静态表项。

```
JX#config
JX(config)#arp static 192.168.1.10 00:50:8d:4b:fd:1e ge 1/0/1
```

检查结果

通过 **show arp** 命令查看 ARP 地址映射表中的所有表项信息是否正确。

```
JX#show arp
Arp aging time: 1200(s)
```

```
Arp entry types: D-Dynamic, S-Static, I-Interface, DH-Dhcp, B-Bgp, INV-
Invalid

IP-addr    Mac-addr   Type Aging  Vlan(O/I)  Interface         Vpn-
instance

------------------------------------------------------------------------------
---------------------------------

192.168.1.1    0002:5600:0001 I    -      -/-          vlan-3
N/A

192.168.1.10    0050:8d4b:fd1e  S    -      3/-          ge-1/0/1
N/A

------------------------------------------------------------------------------
---------------------------------

Total: 2       Dynamic: 0       Static: 1       Bgp: 0       Other: 1
```

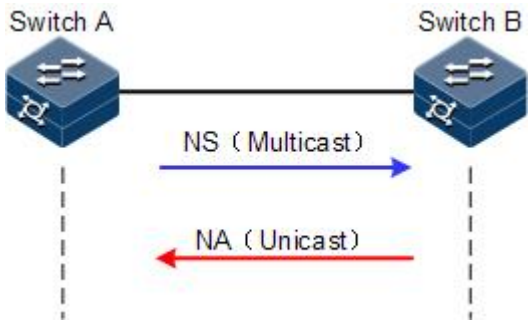
3.4 NDP

3.4.1 简介

NDP（Neighbor Discovery Protocol，邻居发现协议）通过在同一链路上的 IPv6 设备上使用邻居发现机制，来发现彼此的存在、确定彼此的 MAC 地址并维护邻居设备信息。

NDP 通过邻居请求消息 NS（Neighbor Solicitation）和邻居通告消息 NA（Neighbor Advertisement）获取同一链路上邻居设备的链路层地址，即 MAC 地址。

图 3-3 NDP 地址解析原理示意图



如图 3-3 所示，以 Switch A 为例，Switch A 要获取 Switch B 的链路层地址。具体报文处理过程如下：

1. Switch A 通过组播方式发送 NS 消息。其中 NS 消息的源地址是 Switch AVLAN 接口的 IPv6 地址，目的地址是 Switch B 的被请求节点组播地址，消息内容中还包含了 Switch A 的链路层 MAC 地址。

2. Switch B 收到 NS 消息后，判断报文的目的地址是否为自己的 IPv6 地址对应的被请求设备组播地址。如果是，则 Switch B 可以学习到 Switch A 的链路层地址，并以单播方式发送 NA 消息，其中包含了自身的链路层地址。

3. Switch A 收到 Switch B 发送的 NA 消息，就可以获得 Switch B 的链路层地址。

IPv6 邻居发现协议通过使用 ICMPv6 消息，还可以实现验证邻居是否可达、重复地址检测、路由设备发现/前缀发现、地址自动配置和重定向等功能。

3.4.2 配置准备

场景

IPv6 邻居发现协议不但实现了 IPv4 中的 ARP 协议、ICMP 重定向和 ICMP 设备发现等功能，还提供了邻居可达性检测功能。

前提

在配置 NDP 功能之前，需完成以下任务：

- 连接接口并配置接口的物理参数，使接口的物理层状态为 Up。
- 配置 VLAN 接口的 IPv6 地址。

3.4.3 NDP 的缺省配置

设备上 NDP 的缺省配置如下。

功能	缺省值
动态 NDP 老化时间	1200 秒

3.4.4 配置静态邻居表项

将邻居节点的 IPv6 地址解析为链路层地址，可以通过邻居请求消息 NS 及邻居通告消息 NA 来动态实现，也可以通过手工配置静态邻居表项来实现。

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config-vlanif-*)#**ipv6 neighbor-static** *ipv6-address mac-address* { **ge** | **10ge** } *interface-number*	配置静态邻居表项信息。

3.4.5 配置动态 NDP 老化时间

邻居信息表项中的表项并非永远有效，每一条记录都有一个生存周期，到达生存周期仍得不到刷新的记录将从邻居信息表项中删除，这个生存周期被称作老化时间。

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**ipv6 nd lifetime** *(aging-time | default)*	配置动态 NDP 老化时间。

3.4.6 检查配置

配置完成后，请在设备上执行以下命令检查配置结果。

序号	检查项	说明
1	JX#**show ipv6 neighbor**	查看所有 NDP 邻居信息。

3.4.7 维护

用户可以通过以下命令维护 NDP 特性。

命令	描述
JX(config)#**flush ipv6 neighbor** [*all* / *dynamic*| *static*]	清除所有 IPv6 邻居信息。

3.5 静态路由

3.5.1 简介

路由是将报文穿过网络传递到目的地的行为，在传递过程中采用路由表进行转发。不同 VLAN 间的设备进行通讯，或同一 VLAN 跨越不同网络进行通信，需要使用路由功能。

缺省路由

缺省路由是一种特殊的静态路由，只有在路由表中没有找到匹配的路由表项时才会被使用。在路由表中，缺省路由以到达网络 IP 地址 0.0.0.0、掩码为 0.0.0.0 的路由形式出现。可通过命令 **show ip route** 查看当前是否配置了缺省路由。如果设备没有配置缺省路由且报文的目的 IP 地址不在路由表中，那么设备在丢弃该报文的同时，会向报文发送端返回一个 ICMP 报文，报告该目的地址或网络不可达。

静态路由

静态路由是需要用户手动配置的路由，对系统要求低，适用于拓扑结构简单并且稳定的小型网络。缺点是不能自动适应网络拓扑的变化，需要人工干预。

3.5.2 配置准备

场景

对于拓扑结构比较简单的网络，可以配置静态路由。静态路需要由用户手动配置，通过配置静态路由可以建立一个互通的网络。

前提

正确配置 VLAN 接口的 IP 地址。

3.5.3 配置静态路由

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**ip route-static** *ip-address mask-address nexthop-address* {**preference** *preferencevalue*}	配置 IPv4 静态路由。
	JX(config)#**ipv6 route-static** *ipv6-address mask-length ipv6-nexthop-address* {**preference** *preference-value*}	配置 IPv6 静态路由。

3.5.4 检查配置

配置完成后，请在设备上执行以下命令检查配置结果。

序号	检查项	说明
1	JX#**show ip route**	查看路由表信息。
	JX#**show ipv6 route**	

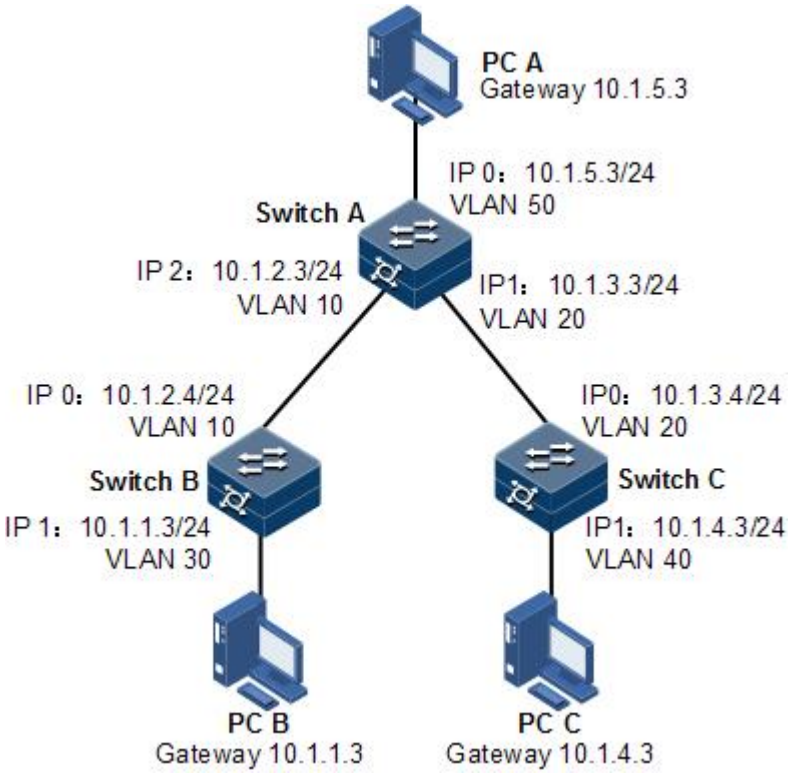
序号	检查项	说明
2	JX#**show { ip | ipv6 } route statistics**	查看路由统计信息。

3.5.5 配置静态路由示例

组网需求

配置静态路由，使图 3-4 中的任意两台主机或交换机设备之间能够相互 Ping 通。

图 3-4 配置静态路由组网示意图



配置步骤

步骤 1 配置各设备的 IP 地址。具体配置略。

步骤 2 在 Switch A 上配置静态路由。

```
JX#hostname SwitchA
SwitchA#config
SwitchA(config)#ip route-static 10.1.1.0 255.255.255.0 10.1.2.4
SwitchA(config)#ip route-static 10.1.4.0 255.255.255.0 10.1.3.4
```

步骤 3 在 Switch B 上配置缺省网关。

```
JX#hostname SwitchB
SwitchB#config
SwitchB(config)#ip route-static 0.0.0.0 0.0.0.0 10.1.2.3
```

步骤 4 在 Switch C 上配置缺省网关。

```
JX#hostname SwitchC
SwitchC#config
SwitchC(config)#ip route-static 0.0.0.0 0.0.0.0 10.1.3.3
```

步骤 5 在主机 A 上配置默认网关为 10.1.5.3，具体配置略。

步骤 6 在主机 B 上配置默认网关为 10.1.1.3，具体配置略。

步骤 7 在主机 C 上配置默认网关为 10.1.4.3，具体配置略。

检查结果

通过 **ping** 命令查看所有设备之间是否均能两两互通。

```
SwitchA#ping 10.1.1.3
PING 10.1.1.3: 64 data bytes
Reply from 10.1.1.3: bytes=64 time=0ms TTL=64 icmp_seq=1
Reply from 10.1.1.3: bytes=64 time=0ms TTL=64 icmp_seq=2
Reply from 10.1.1.3: bytes=64 time=0ms TTL=64 icmp_seq=3
Reply from 10.1.1.3: bytes=64 time=0ms TTL=64 icmp_seq=4
Reply from 10.1.1.3: bytes=64 time=0ms TTL=64 icmp_seq=5
PING Statistics for 10.1.1.3
5 packets transmitted, 5 packets received, 0% packet loss
round-trip (ms)  min/avg/max = 0/0/0
```

3.6 策略路由

3.6.1 简介

概述

传统上，普通的报文转发是依据报文的目的地址查询转发表来实现的，当遇到需要根据源 IP 来控制报文转发，根据报文的长度来控制报文转发或根据报文的其他属性来控制报文转发时，就需要一种新的路由机制来控制，也就是策略路由来控制。

所谓策略路由，顾名思义，即是根据一定的策略进行报文转发，因此策略路由是一种比目的路由更灵活的路由机制。在路由器转发一个数据报文时，首先根据配置的规则对报文进行过滤，匹配成功则按照一定的转发策略进行报文转发。这种规则可以是基于标准和扩展访问控制列表，也可以基于报文的长度；而转发策略则是控制报文按照指定的策略路由表进行转发，也可以修改报文的 IP 优先字段。因此，策略路由是对传统 IP 路由机制的有效增强。

策略路由能满足基于源 IP 地址、目的 IP 址、协议字段，甚至于 TCP、UDP 的源、目的端口等多种组合进行选路。简单点来说，只要 IP standard/extended ACL 能设置的，都可以做为策略路由的匹配规则进行转发。

策略路由（Policy Route）在决定一个 IP 包的下一跳转发地址或是下一跳缺省 IP 地址时，不是简单的根据目的 IP 地址决定，而是综合考虑多种因素来决定。如可以根据 DSCP（差分服务代码点）字段、源和目的端口号，源 IP 地址等来为数据包选择路径。策略路由可以在一定程度上实现流量工程，使不同服务质量的流或者不同性质的数据（语音、FTP）走不同的路径。

基于策略的路由为网络管理者提供了比传统路由协议对报文的转发和存储更强的控制能力。传统上，路由器用从路由协议派生出来的路由表，根据目的地址进行报文的转发。基于策略的路由比传统路由能力更强，使用更灵活，它使网络管理者不仅能够根据目的地址而且能够根据协议类型、报文大小、应用或 I P 源地址来选择转发路径。策略可以定义为通过多路由器的负载平衡或根据总流量在各线上进行报文转发的服务质量（QoS）。

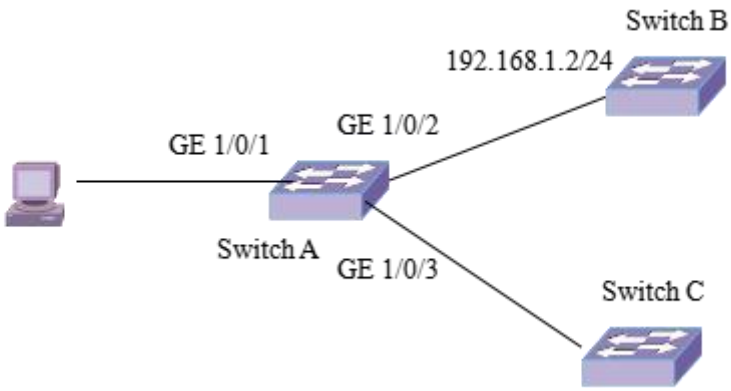
策略路由功能的实现是依靠芯片的支持，策略路由功能是通过命令行或其它配置界面，将软件表项转换为硬件表项存储到芯片上去，当流量通过时，芯片会按照策略路由硬件表来过滤报。

3.6.2 配置基于 ACL 的策略路由示例

组网需求

如下图所示，定义一条名为 aaa 的策略路由，所有从以太网接口 GE 1/0/1 接收的 IP 报文通过接口 GE 1/0/2 发送，下一跳 IP 是 192.168.1.2，其它报文仍然按照查找路由表的方式转发。

图 3-5 配置策略路由组网示意图



配置步骤

步骤 1 定义访问控制列表，ACL filter 1 匹配 IP 报文。

```
switch(config)#acl-ipv4 1001
switch(configure-acl-ipv4-1001)#rule 1 src-ip any dst-ip any
switch(configure-acl-ipv4-1001)#rule 1 action permit
```

步骤 2 定义策略的规则和动作。

```
switch(config)#policy-route aaa permit node 1
switch(config-policy-route-aaa-1)#if-match acl-ipv4 1001
switch(config-policy-route-aaa-1)#apply ip-address next-hop 192.168.1.2
switch(config-policy-route-aaa-1)#quit
```

步骤 3 在接口上使能策略。

```
switch(config)#inter ge 1/0/1
switch(config-ge-1/0/1)#policy-route bind-policy aaa
```

检查结果

```
switch(config-ge-1/0/1)#show ip policy-route aaa
policy-route : aaa
Node 1 : permit
if-match acl-ipv4 1001
apply ip-address next-hop 192.168.1.2
 enabled port : ge-1/0/1
```

3.7 BGP

3.7.1 简介

功能

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 的基本概念

自治系统 AS（Autonomous System）：AS 是指在一个实体管辖下的拥有相同选路策略的 IP 网络。BGP 网络中的每个 AS 都被分配一个唯一的 AS 号，用于区分不同的 AS。AS 号分为 2 字节 AS 号和 4 字节 AS 号，其中 2 字节 AS 号的范围为 1 至 65535，4 字节 AS 号的范围为 1 至 4294967295。支持 4 字节 AS 号的设备能够与支持 2 字节 AS 号的设备兼容。

BGP 分类：BGP 按照运行方式分为 EBGP（External/Exterior BGP）和 IBGP（Internal/Interior BGP）。EBGP：运行于不同 AS 之间的 BGP 称为 EBGP。为了防止 AS 间产生环路，当 BGP 设备接收 EBGP 对等体发送的路由时，会将带有本地 AS 号的路由丢弃。IBGP：运行于同一 AS 内部的 BGP 称为 IBGP。为了防止 AS 内产生环路，BGP 设备不将从 IBGP 对等体学到的路由通告给其他 IBGP 对等体，并与所有 IBGP 对等体建立全连接。

BGP 报文交互中的角色：BGP 报文交互中分为 Speaker 和 Peer 两种角色。Speaker：发送 BGP 报文的设备称为 BGP 发言者（Speaker），它接收或产生新的报文信息，并发布给其它 BGP Speaker。Peer：相互交换报文的 Speaker 之间互称对等体（Peer）。

BGP 的路由器号（Router ID）：BGP 的 Router ID 是一个用于标识 BGP 设备的 32 位值，通常是 IPv4 地址的形式，在 BGP 会话建立时发送的 Open 报文中携带。对等体之间建立 BGP 会话时，每个 BGP 设备都必须有唯一的 Router ID，否则对等体之间不能建立 BGP 连接

BGP4 对等体

BGP 邻居又称为对等体，分为两种。如果两个交换 BGP 报文的对等体属于不同的自治系统，那么这两个对等体就是 EBGP 对等体。如果两个交换 BGP 报文的对等体属于同一个自治系统，那么这两个对等体就是 IBGP 对等体。一个 AS 内的不同边界路由器之间也要建立 BGP 连接，只有这样才能实现路由信息在整个 AS 内的传递。

IBGP 对等体之间不一定是物理上直连的但必须保证逻辑上全连接。EBGP 对等体之间在绝大多数情况下是有物理上的直连链路的但是如果实在无法实现也可以配置逻辑链接。

BGP 把从 EBGP 获得的路由向它所有的 BGP 对等体通告，包括 IBGP 和 EBGP，而把从 IBGP 获得的路由不向它的 IBGP 对等体通告，向 EBGP 通告时要保证 IGP 同 BGP 同步。同步是指 BGP 一直要等到 IGP 在本 AS 中传播了同一条路由后，再给其它各 AS 通告这条路由。也就是说在通告给其它 AS 一条路由时先要保证本 AS 内部的路由器要知道该路由。

BGP4 路由通告

一条路由在一般情况下是从 AS 内部产生的，它由某种内部路由协议发现和计算传递到自治系统的边界，由自治系统边界路由器(ASBR)通过 EBGP 连接传播到其它自治系统中。

路由在传播过程中可能会经过若干个自治系统，这些自治系统称为过渡自治系统。若这个自治系统有多个边界路由器，这些路由器之间运行 IBGP 来交换路由信息。这时内部的路由器并不需要知道这些外部路由，它们只需要在边界路由器之间维护 IP 连通性。路由到达自治系统边界后，若内部路由器需要知道这些外部路由，ASBR 可以将路由引入内部路由协议。外部路由的数量是很大的，通常会超出内部路由器的处理能力，因此引入外部路由时一般需要过滤或聚合以减少路由的数量，极端的情况是使用默认路由。

BGP4 报文

BGP 对等体间通过以下 5 种报文进行交互，其中 Keepalive 报文为周期性发送，其余报文为触发式发送：

Open 报文：用于建立 BGP 对等体连接。Open 消息是 BGP 邻居使用的 TCP 连接建立之后的第一个报文，其内容包括当前的协议版本，自治系统，路由器标识符，以及一些可选参数。如果对方对报文中的某些参数不能达成一致，则无法建立 BGP 邻居。

Keepalive 报文：用于保持 BGP 连接。一旦双方对 Open 报文的内容达成一致，则开始周期性发送 KeepAlive 报文，此报文用于检测邻居的状态，一定时间内没有收到邻居发送的 KeepAlive 报文，则认为邻居发生故障。

Update 报文：用于在对等体之间交换路由信息。Update 报文用于承载路由信息，包括路由的各种属性，BGP 使用该报文向邻居通告路由信息。

Notification 报文：用于中断 BGP 连接。一旦 BGP 运行过程中发生了差错，就会发送 Notification 报文，报文中指明了差错的原因。

Route-refresh 报文：用于在改变路由策略后请求对等体重新发送路由信息。只有支持路由刷新（Route-refresh）能力的 BGP 设备会发送和响应此报文。

BGP4 路由属性

路由属性是对路由的特定描述，所有的 BGP 路由属性都可以分为 4 类，公认必须遵循（Well-known mandatory）：所有 BGP 设备都可以识别此类属性，且必须存在于 Update 报文中。如果缺少这类属性，路由信息就会出错；公认任意（Well-known discretionary）：所有 BGP 设备都可以识别此类属性，但不要求必须存在于 Update 报文中，即就算缺少这类属性，路由信息也不会出错；可选过渡（Optional transitive）：在 AS 之间具有可传递性的属性。BGP 设备可以不支持此属性，但它仍然会接收这类属性，并传递给其他对等体；可选非过渡（Optional non-transitive）：BGP 设备可以不识别此类属性，如果 BGP 设备不识别此类属性，则会被忽略该属性，且不会通告给其他对等体。

几种常用的 BGP 路由属性：

- Origin 属性（公认必须遵循）

ORIGIN 属性定义了路由信息的来源，标记一条 BGP 路由是怎么生成的。它有以下三种类型：

IGP：优先级最高，表示路由产生于本 AS 内。

EGP：优先级次之，表示路由通过 EGP 学到。

Incomplete：优先级最低，表示路由的来源无法确定。例如，从其它路由协议引入的路由信息。

- AS_Path 属性（公认必须遵循）

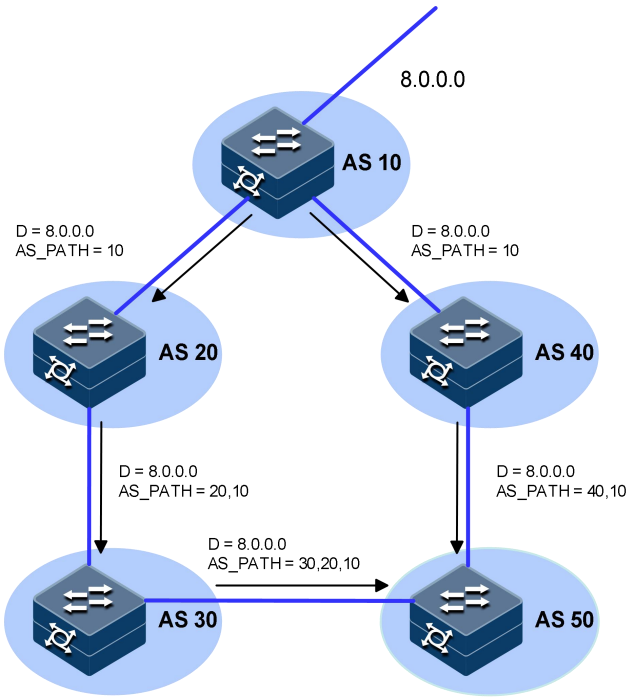
AS_PATH 属性记录了某条路由从本地到目的地址所要经过的所有 AS 号。当 BGP 路由器将一条路由通告到其他 AS 时，会把本地 AS 号添加在 AS_PATH 列表中。收到此路由的 BGP 路由器根据 AS_PATH 属性就可以知道到达目的地址所要经过的 AS。

AS_PATH 属性有以下两种类型：

AS_SEQUENCE：AS 号按照一定的顺序排列。如下图所示，离本地 AS 最近的相邻 AS 号排在前面，其他 AS 号按顺序依次排列。

AS_SET：AS 号只是经过的 AS 的简单罗列，没有顺序要求。

图 3-6 AS_PATH 属性



- AS_PATH 属性具有如下用途：

避免路由环路的形成：缺省情况下，如果 BGP 路由器接收到的路由的 AS_PATH 属性中已经包含了本地的 AS 号，则 BGP 路由器认为出现路由环路，不会接受该路由。

影响路由的选择：在其他因素相同的情况下，BGP 会优先选择路径较短的路由。比如在上图中，AS 50 中的 BGP 路由器会选择经过 AS 40 的路径作为到目的地址 8.0.0.0 的最优路由。用户可以使用路由策略来人为地增加 AS 路径的长度，以便更为灵活地控制 BGP 路径的选择。

对路由进行过滤：通过配置 AS 路径过滤列表，可以针对 AS_PATH 属性中所包含的 AS 号来对路由进行过滤。

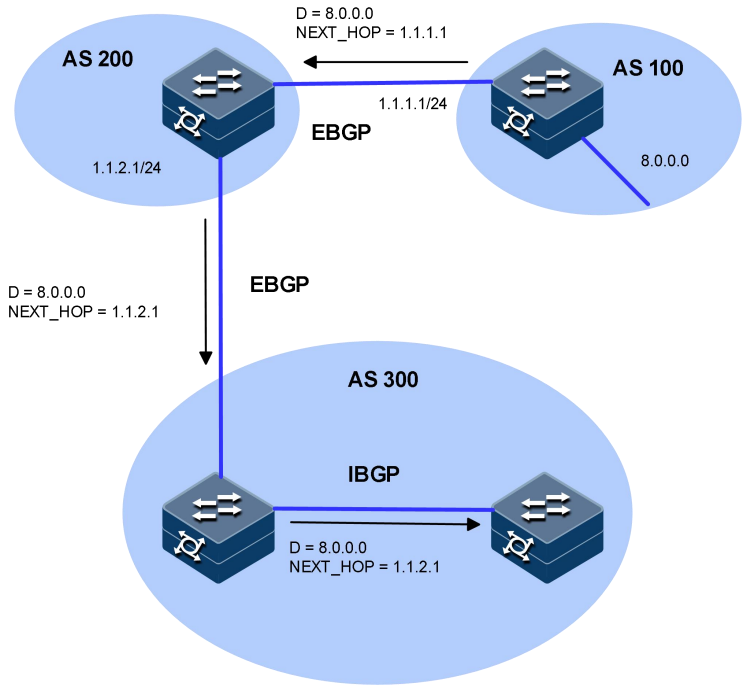
- Next_Hop 属性（公认必须遵循）

BGP 的 NEXT_HOP 属性取值不一定是邻居路由器的 IP 地址。如下图所示，NEXT_HOP 属性取值情况分为几种：

  - BGP 发言者把自己产生的路由发给所有邻居时，将该路由信息的 NEXT_HOP 属性设置为自己与对端连接的接口地址；
  - BGP 发言者把接收到的路由发送给 EBGP 对等体时，将该路由信息的 NEXT_HOP 属性设置为自己与对端连接的接口地址；

– BGP 发言者把从 EBGP 邻居得到的路由发给 IBGP 邻居时，并不改变该路由信息的 NEXT_HOP 属性。

图 3-7 NEXT_HOP 属性

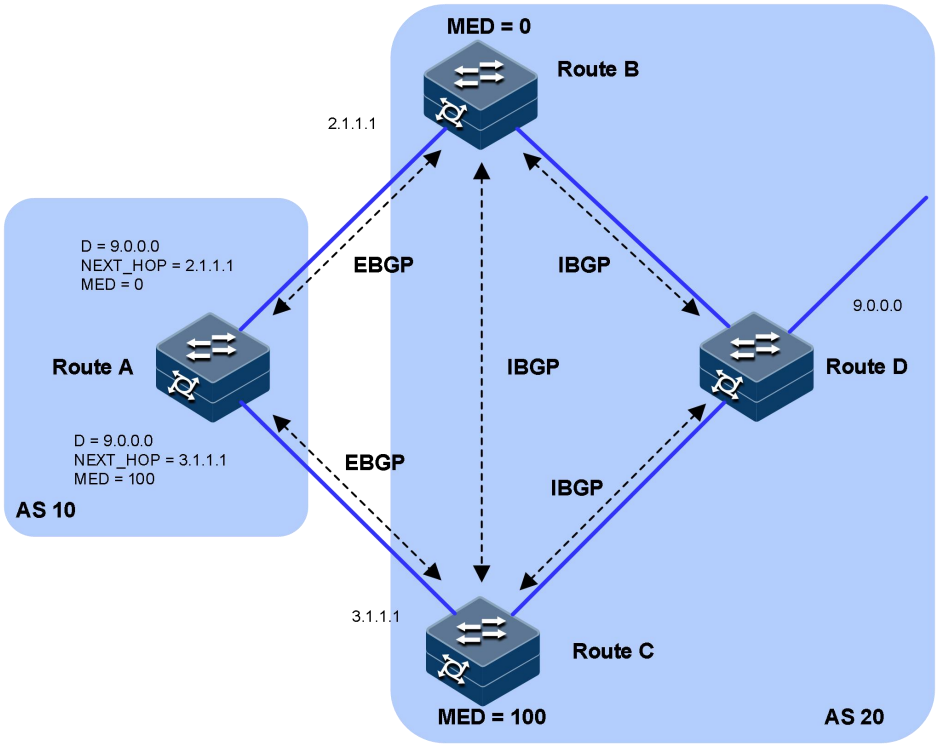


- MED（Multi-Exit Discriminator）属性（可选非过渡）

MED 属性仅在相邻两个 AS 之间交换，收到此属性的 AS 不会再将其通告给其它 AS。

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 BGP 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。如下图所示，从 AS 10 到 AS 20 的流量将选择 Router B 作为入口。通常情况下，BGP 只比较来自同一个 AS 的路由的 MED 属性值。

图 3-8 MED 属性

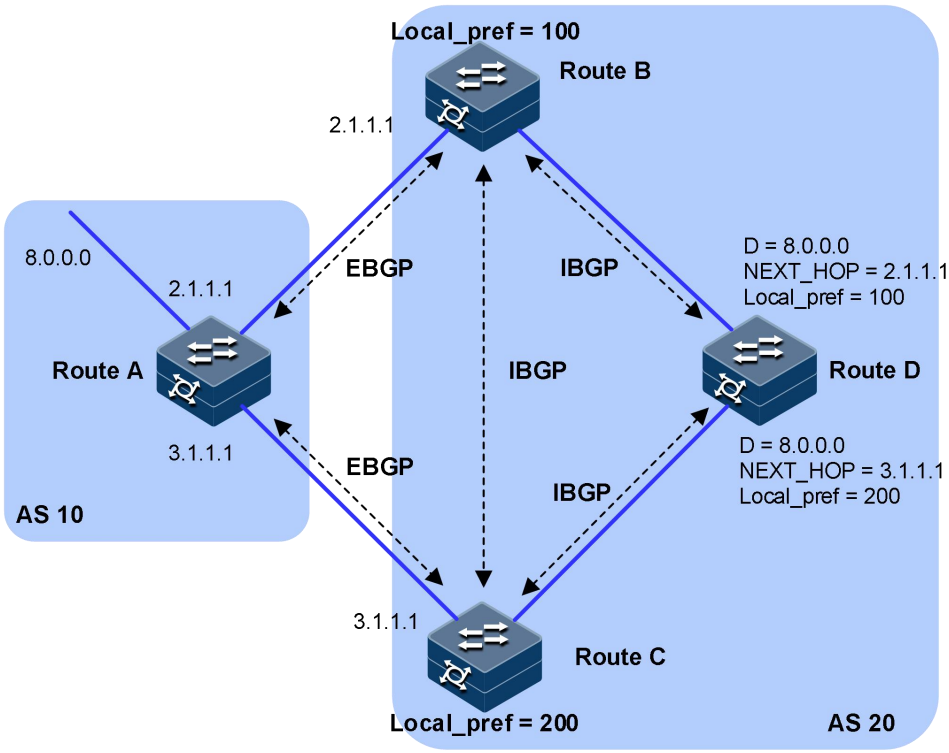


- Local_Pref 属性（公认任意）

LOCAL_PREF 属性仅在 IBGP 对等体之间交换，不通告给其他 AS。它表明 BGP 路由器的优先级。

LOCAL_PREF 属性用于判断流量离开 AS 时的最佳路由。当 BGP 路由器通过不同的 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。如下图所示，从 AS 20 到 AS 10 的流量将选择 Router C 作为出口。

图 3-9 LOCAL_PREF 属性



- 团体属性（可选过渡）

BGP 将具有相同特征的路由归为一组，称为一个团体，通过在路由中携带团体属性标识路由所属的团体。团体没有物理上的边界，不同 AS 的路由可以属于同一个团体。

根据需要，一条路由可以携带一个或多个团体属性值（每个团体属性值用一个四字节的整数表示）。接收到该路由的路由器可以通过比较团体属性值对路由作出适当的处理（比如决定是否发布该路由、在什么范围发布等），而不需要匹配复杂的过滤规则，从而简化路由策略的应用和降低维护管理的难度。

- 公认的团体属性有：
  - INTERNET：缺省情况下，所有的路由都属于 INTERNET 团体。具有此属性的路由可以被通告给所有的 BGP 对等体。
  - NO_EXPORT：具有此属性的路由在收到后，不能被发布到本地 AS 之外。如果使用了联盟，则不能被发布到联盟之外，但可以发布给联盟中的其他子 AS。
  - NO_ADVERTISE：具有此属性的路由被接收后，不能被通告给任何其他的 BGP 对等体。
  - NO_EXPORT_SUBCONFED：具有此属性的路由被接收后，不能被发布到本地 AS 之外，也不能发布到联盟中的其他子 AS。

除了公认的团体属性外，用户还可以使用团体属性列表自定义团体属性，以便更为灵活地控制路由策略。

BGP4 选择路由的策略

优选本地优先级（Local_Pref）最高的路由；

优选聚合路由（聚合路由优先级高于非聚合路由）；

优选 AS 路径（AS_Path）最短的路由；

比较 Origin 属性，依次选择 Origin 类型为 IGP、EGP、Incomplete 的路由；

优选 MED 值最低的路由；

优选从 EBGP 学来的路由（EBGP 路由优先级高于 IBGP 路由）；

优选 AS 内部 IGP 的 Metric 最低的路由；

优选 Router ID 最小的交换机发布的路由；

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。

BGP4 发布路由的策略

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 发言者只把自己使用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（在不使能 BGP 与 IGP 同步特性的情况下）；

连接一旦建立，BGP 发言者将把自己所有 BGP 路由发布给新对等体。

BGP4 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（Routes 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。

BGP4 路由衰减

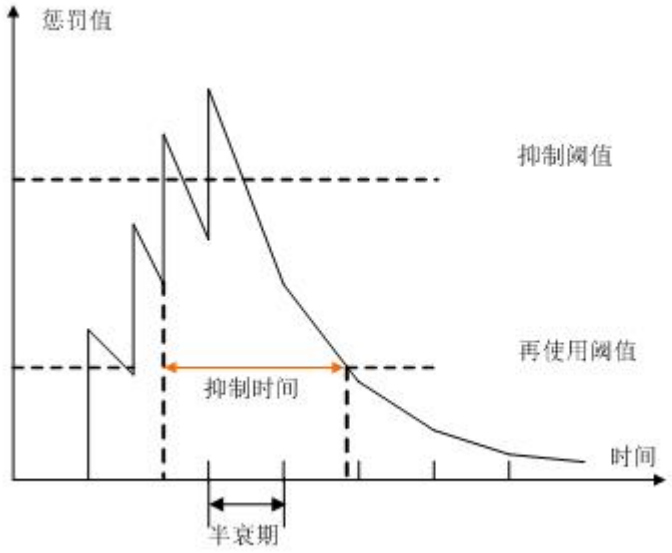
路由发生变化时，路由协议会向邻居发布路由更新，收到路由更新的路由器需要重新计算路由并修改路由表。如果发生路由振荡，即路由不稳定，路由表中的某条路由反复消失和重现，则会消耗大量的带宽资源和 CPU 资源，严重时会影响到网络的正常工作。

在多数情况下，BGP 协议都应用于复杂的网络环境中，路由变化十分频繁。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 衰减使用惩罚值来衡量一条路由的稳定性，惩罚值越高说明路由越不稳定。路由每次从可达状态变为不可达状态，或者可达路由的属性每次发生变化时，BGP 给此路由增加一定的惩罚值（系统固定为 1000，不可修改）。当惩罚值超过抑制阈值时，此路由被抑制，不参与路由选择。惩罚值达到设置的上限后，不再继续增加。

发生振荡的路由如果没有再次振荡，则路由的惩罚值会逐渐减少。每经过一段时间，惩罚值便会减少一半，这个时间称为半衰期（Half-life）。当惩罚值低于再使用阈值时，此路由变为可用路由，参与路由选择。

图 3-10 BGP 路由反射器示意图



BGP4 的团体

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。

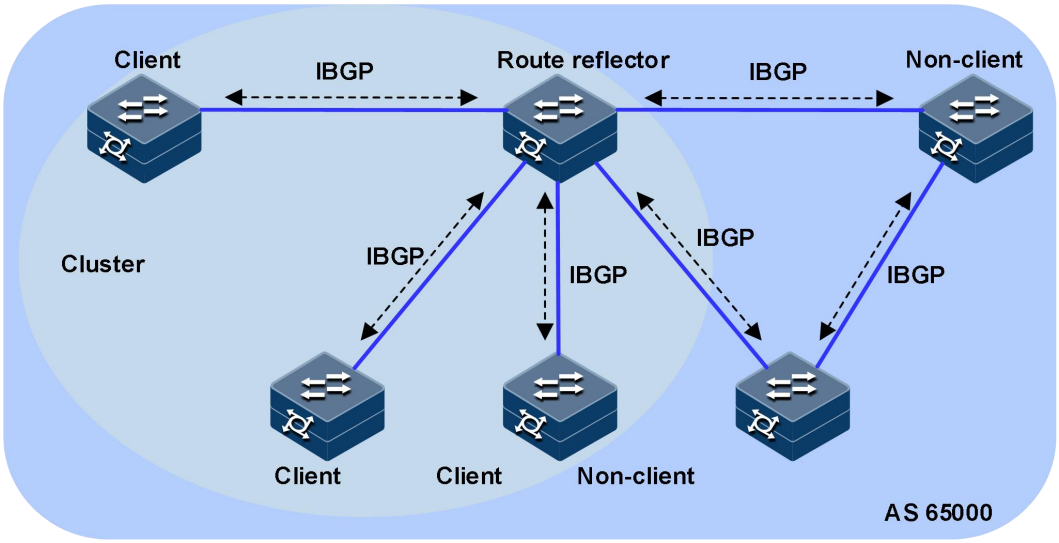
BGP4 的路由反射器

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。假设在一个 AS 内部有 n 台交换机，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，其中一台交换机作为路由反射器 RR（Route Reflector），其它交换机作为客户机（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，而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。

图 3-11 BGP 路由反射器示意图



BGP4 的联盟

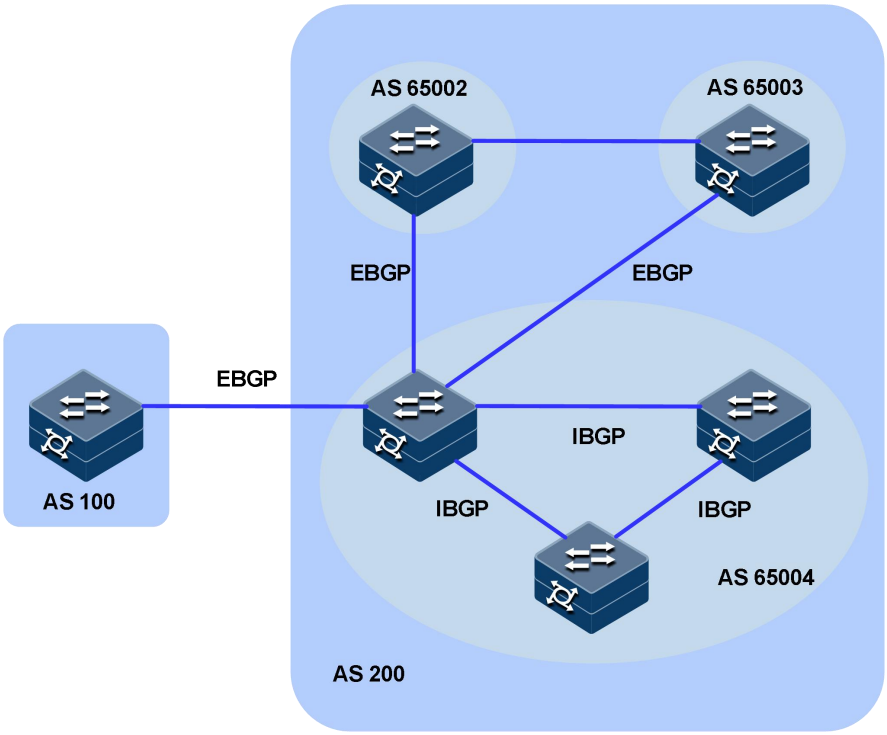
联盟（Confederation）是处理 AS 内部的 IBGP 网络连接激增的另一种方法，它将一个自治系统划分为若干个子自治系统，每个子自治系统内部的 IBGP 对等体建立全连接关系，子自治系统之间建立 EBGP 连接关系。

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，要求交换机重新进行配置，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。

图 3-12 BGP 联盟示意图



BGP4 的 MP-BGP 概述

传统的 BGP-4 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 可以为多种网络层协议传递路由信息，如 IPv4 组播、IPv6 单播、IPv6 组播、VPNv4 等。

MP-BGP 前向兼容，即支持 BGP 扩展的交换机与不支持 BGP 扩展的交换机可以互通。

MP-BGP 的扩展属性

MP-BGP 使用的报文中，与 IPv4 相关的三条信息都由 Update 报文携带，这三条信息分别是：NLRI、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的信息，不把它们传递给其它邻居。

地址族

MP-BGP 采用地址族（Address Family）和子地址族（Subsequent Address Family）来区分 MP_REACH_NLRI 属性、MP_UNREACH_NLRI 属性中携带路由信息所属的网络层协议。例如，如果 MP_REACH_NLRI 属性中 AFI（Address Family Identifier，地址族标识符）为 2、SAFI（Subsequent Address Family Identifier，子地址族标识符）为 1，则表示该属性中携带的是 IPv6 单播路由信息。关于地址族的一些取值可以参考 RFC1700（Assigned Numbers）。MP-BGP 扩展应用，包括对 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。

BFD for BGP 特性

在 IPv4 中使用 BFD（Bidirectional Forwarding Detection）为 BGP 协议提供更快速的链路故障检测。

BFD 能够快速检测到 BGP 对等体间的链路故障，并报告给 BGP 协议，从而实现 BGP 路由的快速收敛。

BFD 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。

3.7.2 配置准备

场景

BGP 是为取代最初的 EGP 而设计的另一种外部网关协议。不同于最初的 EGP，BGP 能够进行路由优选、避免路由环路、更高效率的传递路由和维护大量的路由信息。

前提

相邻节点的网络层可达。

3.7.3 缺省配置

设备上 BGP 的缺省配置如下。

功能	缺省值
BGP 全局功能状态	禁用
Keepalive 消息发送间隔	30s

功能	缺省值
对等体邻接关系保持时间	90s

3.7.4 配置 BGP 基本功能

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**bgp** *as-number*	启动 BGP，指定本地 AS 编号，并进入 BGP 视图。
3	JX(config-bgp)#**router-id** *ipv4-address*	配置 BGP 的 Router ID。
4	JX(config-bgp)#**neighbor** { *ipv4-address* | *ipv6-address* } **remote-as** *as-number*	创建 BGP 对等体。
5	JX(config-bgp)#**neighbor** *ipv4-address* **update-source** *ipv4-address* JX(config-bgp)#**neighbor** *ipv6-address* **update-source** *ipv6-address*	配置发起连接时使用的源地址。
6	JX (config-bgp)#**neighbor** { *ipv4-address* | *ipv6-address* } **password** { **cipher** *cipher-password* | **plain** *plain-password* }	配置 BGP 对等体在建立 TCP 连接时对 BGP 消息进行 MD5 认证。
7	JX(config-bgp)#**neighbor** { *ipv4-address* | *ipv6-address* } **ebgp-multihop**	配置 BGP 邻居支持多跳 EBGP。
8	JX(config-bgp)#**ipv4-family** { **unicast** | **vpn-instance** *vpn-instance-name* }	进入 BGP 的 IPv4 单播/BGP-VPN 实例地址族视图。
9	JX(config-bgp)#**ipv6-family** { **unicast** | **vpn-instance** *vpn-instance-name* }	进入 BGP 的 IPv6 单播/BGP-VPN 实例地址族视图。
10	JX(config-bgp-af-*)#**neighbor** { *ipv4-address* | *ipv6-address* } **enable**	在地址族视图下使能与指定对等体之间交换相关的路由信息。
11	JX(config-bgp-af-*)#**network** { *ipv4-address/mask-length* | *ipv6-address/mask-length* }	配置 BGP 发布的本地网络路由，即将本地路由表中的路由以静态方式加入到 BGP 路由表中，并发布给对等体。

步骤	配置	说明
12	JX(config-bgp-af-*)#**import-route** { **connected** | **isis** [*process-id*] | **ospf** [*process-id*] | **rip** [*process-id*] | **static** } [**med** *med* | **route-policy** *route-policy-name*]	引入其它协议路由信息。

3.7.5 简化 IBGP 网络连接

请在设备上进行以下配置。

步骤	配置	说明
1	JX(config-bgp-af-*)#**neighbor** { *ipv4-address* | *ipv6-address* } **route-reflector-client**	配置将本机作为路由反射器，并将对等体作为路由反射器的客户。
2	JX(config-bgp-af-*)#**cluster-id** *cluster-id*	配置路由反射器的集群 ID。
3	JX(config-bgp)#**confederation identifier** *as-number*	配置联盟 ID。
4	JX(config-bgp)#**confederation peer-as** *as-number-list*	配置属于同一个联盟的各子自治系统号。

3.7.6 配置 BGP 路由选路和负载分担

请在设备上进行以下配置。

步骤	配置	说明
1	JX(config-bgp-af-*)#**preference** { **external** { *preference* | **default** } | **internal** { *preference* | **default** } | **local** { *preference* | **default** }}*	配置外部、内部、本地路由的路由优先级。
2	JX(config-bgp-af-*)#**neighbor** { *ipv4-address* | *ipv6-address* } **next-hop-local**	设置向 IBGP 对等体通告路由时，把下一跳属性设为自身的 IP 地址。
3	JX(config-bgp-af-*)#**neighbor** { *ipv4-address* | *ipv6-address* } **preference** { *preference* | **default** }	为从对等体接收的路由分配首选值。
4	JX(config-bgp-af-*)#**default local-preference** { *local-preference* | **default** }	配置 BGP 的缺省本地优先级。

步骤	配置	说明
5	JX(config-bgp)#**neighbor** { *ipv4-address* | *ipv6-address* } **allow-as-loop** { *number* | **default** }	配置对于从对等体接收的路由，允许本地 AS 号在接收路由的 AS_PATH 属性中出现，并配置允许出现的次数。
6	JX(config-bgp-af-*)#**default local-med** { *local-med* | **default** }	配置 BGP 路由的缺省 MED 值。
7	JX(config-bgp-af-*)#**community** { *community-number* | **noadvertise** | **noexport** } { **additive** | **replace** | **none** }	配置 BGP 团体属性。
8	JX(config-bgp-af-*)#**neighbor** { *ipv4-address* | *ipv6-address* } **advertise-community**	配置将团体属性发布给对等体。

3.7.7 控制 BGP 路由的发布与接收

请在设备上进行以下配置。

步骤	配置	说明
1	JX(config-bgp-af-*)#**summary automatic enable**	使能自动聚合功能。(支持 BGP-IPv4 单播地址族视图、BGP-VPN 实例 IPv4 地址族视图)
2	JX(config-bgp-af-*)#**summary** *ipv4-address/prefix-length* [**aggregated-only**]* JX(config-bgp-af-*)#**ipv6 summary** *ipv6-address/prefix-length* [**aggregated-only**]*	在 BGP 路由表中创建一条聚合条目。
3	JX(config-bgp-af-*)#**filter-policy import route-policy** *route-policy-name*	配置 BGP 全局入过滤策略。
4	JX(config-bgp-af-*)#**filter-policy export** [**static** | **connected** | **rip** | **ospf** | **isis**] **route-policy** *route-policy-name*	配置 BGP 全局出过滤策略。

3.7.8 控制调整 BGP 网络的收敛速度

请在设备上进行以下配置。

步骤	配置	说明
1	JX(config-bgp)#**neighbor** { *ipv4-address* | *ipv6-address* } **connect-retry-interval** { *seconds* | **default** }	配置对等体的连接重传定时器。
2	JX(config-bgp)#**neighbor** { *ipv4-address* | *ipv6-address* } **keeplive-timer** { *keeplive-time* | **default** } **hold-timer** { *hold-time* | **default** }	配置本地路由器与指定对等体之间 BGP 会话的存活时间间隔和保持时间。
3	JX(config-bgp)#**neighbor** { *ipv4-address* | *ipv6-address* } **advertisement-interval** { *seconds* | **default** }	配置更新报文定时器。
4	JX(config-bgp-af-*)#**dampening half-life-reach** { *half-life-time* | **default** } **reuse** { *reuse* | **default** } **suppress** { *suppress* | **default** } **ceiling** { *max* | **default** }	配置路由抑制功能相关参数。
5	JX(config-bgp-af-*)#**dampening enable**	使能路由抑制功能。

3.7.9 配置 BGP 可靠性

请在设备上进行以下配置。

步骤	配置	说明
1	JX(config-bgp)#**graceful-restart enable**	使能 BGP 协议的 GR 能力。
2	JX(config-bgp)#**graceful-restart timer restart** { *timer* | **default** }	配置对端等待重建 BGP 会话的时间。
3	JX(config-bgp)#**graceful-restart timer selection-deferral** { *timer* | **default** }	配置本端等待 End-Of-RIB 标记的时间。
4	JX(config-bgp)#**neighbor** { *ipv4-address* | *ipv6-address* } **bfd enable**	配置为对等体创建 BFD 会话。

3.7.10 检查配置

配置完成后，请在设备上执行以下命令检查配置结果。

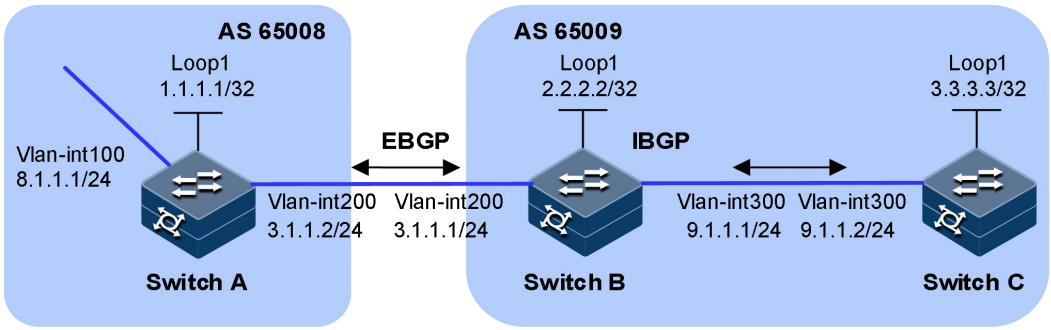
序号	检查项	说明
1	JX#**show bgp config**	查看 BGP 配置信息。
2	JX#**show bgp neighbor [ip-address** *ipv4-address* | **ipv6-address** *ipv6-address*]	显示 BGP 对等体的状态。
3	JX#**show bgp ip-route** JX#**show bgp ipv6-route**	显示 BGP 的路由表。

3.7.11 配置 BGP 的基本功能示例

组网需求

如下图所示，所有交换机均运行 BGP 协议。要求 Switch A 和 Switch B 之间建立 EBGP 连接，Switch B 和 Switch C 之间建立 IBGP 连接，使得 Switch C 能够访问 Switch A 直连的 8.1.1.0/24 网段。

图 3-13 BGP 基本配置组网图



配置步骤

步骤 1 配置各接口的 IP 地址。

步骤 2 配置 IBGP 连接。

在 AS 65009 内部，使用 OSPF 协议，保证 Switch B 到 Switch C 的 Loopback 接口路由可达， Switch C 到 Switch B 的 Loopback 接口路由可达。

```
SwitchB#configure
SwitchB(config)#ospf 1
SwitchB(config-ospf-1)#router-id 2.2.2.2
SwitchB(config-ospf-1)#network 2.2.2.2 255.255.255.255 area 0
SwitchB(config-ospf-1)#network 9.1.1.0 255.255.255.0 area 0
SwitchB(config-ospf-1)#end

SwitchC#configure
SwitchC(config)#ospf 1
SwitchC(config-ospf-1)#router-id 3.3.3.3
SwitchC(config-ospf-1)#network 3.3.3.3 255.255.255.255 area 0
```

```
SwitchC(config-ospf-1)#network 9.1.1.0 255.255.255.0 area 0
SwitchC(config-ospf-1)#end
```

为了防止端口状态不稳定引起路由震荡，本举例使用 Loopback 接口来创建 IBGP 对等体。

```
SwitchB#configure
SwitchB(config)#bgp 65009
SwitchB(config-bgp)#neighbor 3.3.3.3 remote-as 65009
SwitchB(config-bgp)#neighbor 3.3.3.3 update-source 2.2.2.2
SwitchB(config-bgp)#end

SwitchC#configure
SwitchC(config)#bgp 65009
SwitchC(config-bgp)#neighbor 2.2.2.2 remote-as 65009
SwitchC(config-bgp)#neighbor 2.2.2.2 update-source 3.3.3.3
SwitchC(config-bgp)#end
```

步骤 3 配置 EBGP 连接。

EBGP 邻居关系的两台路由器（通常属于两个不同运营商），处于不同的 AS 域，对端的 Loopback 接口一般路由不可达，所以一般使用直连地址建立 BGP 邻居。

因为要求 Switch C 能够访问 Switch A 直连的 8.1.1.0/24 网段，所以，建立 EBGP 连接后，需要将 8.1.1.0/24 网段路由通告到 BGP 路由表中。

```
SwitchA#configure
SwitchA(config)#bgp 65008
SwitchA(config-bgp)#neighbor 3.1.1.1 remote-as 65009
SwitchA(config-bgp)#ipv4-family unicast
SwitchA(config-bgp-af-ipv4)#network 8.1.1.0/24
SwitchA(config-bgp-af-ipv4)#end

SwitchB#configure
SwitchB(config)#bgp 65009
SwitchB(config-bgp)#neighbor 3.1.1.2 remote-as 65008
SwitchB(config-bgp)#exit
SwitchB(config)#show bgp neighbor

 BGP local router ID :2.2.2.2
 Local AS number :65009
 Total number of neighbors :2
 Neighbors in established state:2
 Neighbor             Ver AS      MsgIn    MsgOut   Up/Down
State/Change    VpnInstance
 3.1.1.2              4   65008    65       64       00:26:44 Established/1
N/A
 3.3.3.3              4   65009    78       78       00:32:19 Established/1
N/A
```

可以看出，Switch B 与 Switch C、Switch B 与 Switch A 之间的 BGP 连接均已建立。

查看各 Switch 的 BGP 路由表：

```
SwitchA#configure
SwitchA(config)#show bgp ip-route
DestAddr/Prefixlen     Neighbor             Nexthop              Protocol
Med        LocalPrf    Origin    Vpn-Instance   As-Path
------------------------------------------------------------------------------
------------------------------------------------------------
8.1.1.0/24             0.0.0.0              0.0.0.0              other    0
0          IGP       public
------------------------------------------------------------------------------
------------------------------------------------------------
 Total:1

SwitchB#configure
SwitchB(config)#show bgp ip-route
DestAddr/Prefixlen     Neighbor             Nexthop              Protocol
Med        LocalPrf    Origin    Vpn-Instance   As-Path
------------------------------------------------------------------------------
------------------------------------------------------------
8.1.1.0/24             3.1.1.2              3.1.1.2              bgp      0
0          IGP       public        AS_SEQUENCE 65008
------------------------------------------------------------------------------
------------------------------------------------------------
 Total:1

SwitchC#configure
SwitchC(config)#show bgp ip-route
DestAddr/Prefixlen     Neighbor             Nexthop              Protocol
Med        LocalPrf    Origin    Vpn-Instance   As-Path
------------------------------------------------------------------------------
------------------------------------------------------------
8.1.1.0/24             2.2.2.2              3.1.1.2              bgp      0
100        IGP       public        AS_SEQUENCE 65008
------------------------------------------------------------------------------
------------------------------------------------------------
 Total:1
```

从路由表可以看出，Switch A 没有学到 AS 65009 内部的任何路由，Switch C 虽然学到了 AS 65008 中的 8.1.1.0 的路由，但因为下一跳 3.1.1.2 不可达，所以也不是有效路由。

步骤 4 配置 EBGP 连接。

在 Switch B 上配置 BGP 引入直连路由，以便 Switch A 能够获取到网段 9.1.1.0/24 的路由，Switch C 能够获取到网段 3.1.1.0/24 的路由。

```
SwitchB#configure
SwitchB(config)#bgp
SwitchB(config-bgp)#ipv4-family unicast
SwitchB(config-bgp-af-ipv4)#import-route connected
SwitchB(config-bgp-af-ipv4)#end

查看各 SwitchA 和 SwitchC 的 BGP 路由表
SwitchA#configure
SwitchA(config)#show bgp ip-route
```

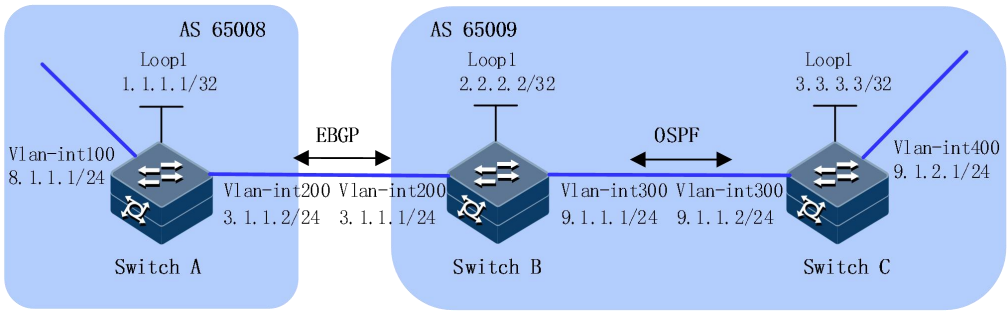
```
DestAddr/Prefixlen     Neighbor             Nexthop              Protocol
Med        LocalPrf    Origin    Vpn-Instance   As-Path
------------------------------------------------------------------------------
------------------------------------------------------------
2.2.2.2/32             3.1.1.1              3.1.1.1              bgp      0
0          Incomplete public        AS_SEQUENCE 65009
3.1.1.0/24             3.1.1.1              3.1.1.1              bgp      0
0          Incomplete public        AS_SEQUENCE 65009
8.1.1.0/24             0.0.0.0              0.0.0.0              other    0
0          IGP       public
9.1.1.0/24             3.1.1.1              3.1.1.1              bgp      0
0          Incomplete public        AS_SEQUENCE 65009
------------------------------------------------------------------------------
------------------------------------------------------------
  Total:4

SwitchC#configure
SwitchC(config)#show bgp ip-route
DestAddr/Prefixlen     Neighbor             Nexthop              Protocol
Med        LocalPrf    Origin    Vpn-Instance   As-Path
------------------------------------------------------------------------------
------------------------------------------------------------
2.2.2.2/32             2.2.2.2              2.2.2.2              bgp      0
100        Incomplete public
3.1.1.0/24             2.2.2.2              2.2.2.2              bgp      0
100        Incomplete public
8.1.1.0/24             2.2.2.2              3.1.1.2              bgp      0
100        IGP       public        AS_SEQUENCE 65008
9.1.1.0/24             2.2.2.2              2.2.2.2              bgp      0
100        Incomplete public
------------------------------------------------------------------------------
------------------------------------------------------------
  Total:4
```

3.7.12 配置 BGP 和 IGP 交互功能示例

组网需求

如下图所示，公司 A 的所有设备在 AS 65008 内，公司 B 的所有设备在 AS 65009 内，AS 65008 和 AS 65009 通过设备 Switch A 和 Switch B 相连。现要求实现 Switch A 能够访问 AS 65009 内的网段 9.1.2.0/24，Switch C 能够访问 AS 65008 内的网段 8.1.1.0/24。

图 3-14 BGP 与 IGP 交互配置组网图



配置步骤

步骤 1 配置各接口的 IP 地址。

步骤 2 配置 OSPF。

在 AS 65009 内配置 OSPF，使得 Switch B 能获取到到 9.1.2.0/24 网段的路由。

```
SwitchB#configure
SwitchB(config)#ospf 1
SwitchB(config-ospf-1)#router-id 2.2.2.2
SwitchB(config-ospf-1)#network 2.2.2.2 255.255.255.255 area 0
SwitchB(config-ospf-1)#network 9.1.1.0 255.255.255.0 area 0
SwitchB(config-ospf-1)#end

SwitchC#configure
SwitchC(config)#ospf 1
SwitchC(config-ospf-1)#router-id 3.3.3.3
SwitchC(config-ospf-1)#network 9.1.1.0 255.255.255.0 area 0
SwitchC(config-ospf-1)#import-route connect
SwitchC(config-ospf-1)#end
```

步骤 3 配置 EBGP 连接。

配置 EBGP 连接，并在 Switch A 上将 8.1.1.0/24 网段通告到 BGP 路由表中，以便 Switch B 获取到网段 8.1.1.0/24 的路由。

```
SwitchA#configure
SwitchA(config)#bgp 65008
SwitchA(config-bgp)#neighbor 3.1.1.1 remote-as 65009
SwitchA(config-bgp)#ipv4-family unicast
SwitchA(config-bgp-af-ipv4)#network 8.1.1.0/24
SwitchA(config-bgp-af-ipv4)#end

SwitchB#configure
SwitchB(config)#bgp 65009
SwitchB(config-bgp)#neighbor 3.1.1.2 remote-as 65008
SwitchB(config-bgp)#end
```

步骤 4 配置 BGP 与 IGP 交互

在 Switch B 上配置 BGP 引入 OSPF 路由，以便 Switch A 能够获取到到 9.1.2.0/24 网段的路由。

在 Switch B 上配置 OSPF 引入 BGP 路由，以便 Switch C 能够获取到到 8.1.1.0/24 网段的路由。

```
SwitchB#configure
SwitchB(config)#bgp
SwitchB(config-bgp)#ipv4-family unicast
SwitchB(config-bgp-af-ipv4)#import-route ospf 1
SwitchB(config-bgp-af-ipv4)#end
SwitchB#configure
SwitchB(config)#ospf 1
SwitchB(config-ospf-1)#import-route bgp
SwitchB(config-ospf-1)#end
```

查看 SwitchA BGP 路由表

```
SwitchA#configure
SwitchA(config)#show bgp ip-route
DestAddr/Prefixlen     Neighbor            Nexthop              Protocol
Med        LocalPrf    Origin    Vpn-Instance   As-Path
------------------------------------------------------------------------------
------------------------------------------------------------
2.2.2.2/32             3.1.1.1             3.1.1.1              bgp      1
0          Incomplete public       AS_SEQUENCE 65009
3.3.3.3/32             3.1.1.1             3.1.1.1              bgp      1
0          Incomplete public       AS_SEQUENCE 65009
8.1.1.0/24             0.0.0.0             0.0.0.0              other    0
0          IGP       public
9.1.1.0/24             3.1.1.1             3.1.1.1              bgp      1
0          Incomplete public       AS_SEQUENCE 65009
9.1.2.0/24             3.1.1.1             3.1.1.1              bgp      1
0          Incomplete public       AS_SEQUENCE 65009
------------------------------------------------------------------------------
------------------------------------------------------------
 Total:5
```

查看 SwitchC OSPF 路由表

```
SwitchC#configure
SwitchC(config)#show ospf route
 OSPF Process 1
 RoutType  Prefix            PathType  Cost    Cost2      NextHop
BackupNextHop    AreaId  Time
 ASBR     2.2.2.2/32         INTRA     1       0          9.1.1.1
0.0.0.0          0       0:01:33
 Network   2.2.2.2/32         INTRA     2       0          9.1.1.1
0.0.0.0          N/A     0:01:33
 Network   8.1.1.0/24         ASE2      1       1          9.1.1.1
0.0.0.0          N/A     0:01:33
 Network   9.1.1.0/24         INTRA     1       0          9.1.1.2
0.0.0.0          N/A     0:01:38
 Route :
 ABR      ASBR      Network   Intra     Inter     External
```

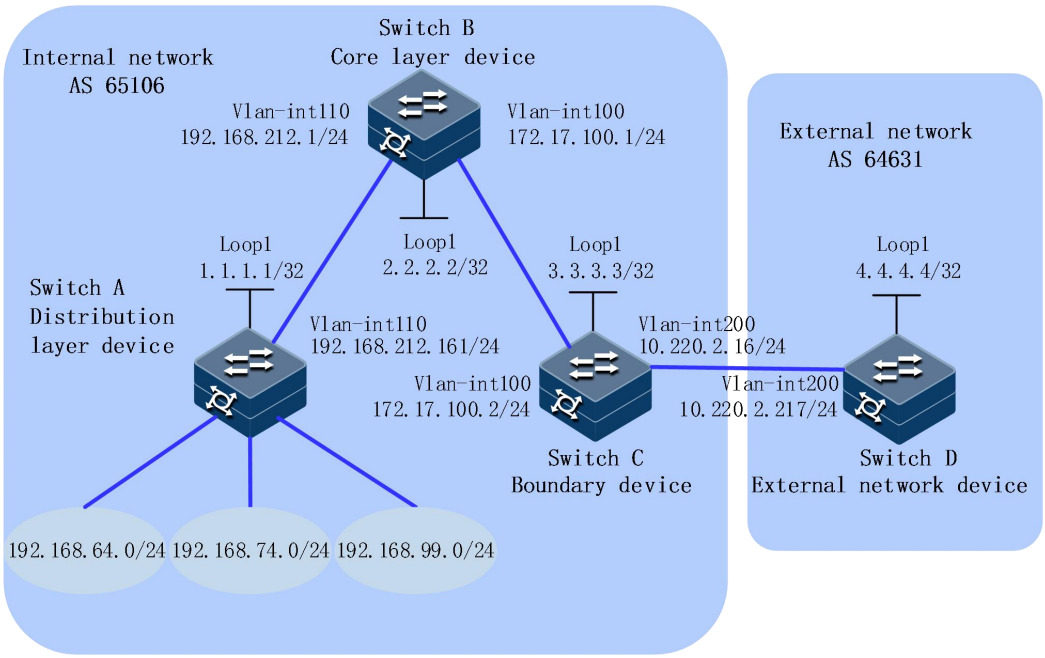
```
0       1       3       2       0       1

Path :
ABR     ASBR    Network  Intra   Inter   External
0       1       3       2       0       1
```

3.7.13 配置 BGP 路由聚合功能示例

组网需求

如下图所示，通过在边界设备 Switch C 和外部网络设备 Switch D 之间建立 EBGP 连接，实现公司内部网络与外部网络的互通。在公司内部，核心层设备 Switch B 与汇聚层设备 Switch A 之间配置静态路由，Switch B 与 Switch C 之间配置 OSPF，并在 OSPF 路由中引入静态路由，以实现公司内部网络的互通。公司内部网络包括三个网段：192.168.64.0/24、192.168.74.0/24 和 192.168.99.0/24。在 Switch C 上配置路由聚合，将这三个网段的路由聚合为一条路由，以减少通过 BGP 发布的路由数量。

图 3-15 BGP 路由聚合组网图



配置步骤

步骤 1 配置各接口的 IP 地址。

步骤 2 在 Switch A 和 Switch B 之间配置静态路由。

```
SwitchA#configure
SwitchA(config)#ip route-static 0.0.0.0 0.0.0.0 192.168.212.1

SwitchB#configure
SwitchB(config)#ip route-static 192.168.64.0 255.255.255.0
192.168.212.161
```

```
SwitchB(config)#ip route-static 192.168.74.0 255.255.255.0
192.168.212.161
SwitchB(config)#ip route-static 192.168.99.0 255.255.255.0
192.168.212.161
```

步骤 3 在 Switch B 和 Switch C 之间配置 OSPF，并引入静态路由。

```
SwitchB#configure
SwitchB(config)#ospf 1
SwitchB(config-ospf-1)#network 172.17.100.0 255.255.255.0 area 0
SwitchB(config-ospf-1)#import-route static
SwitchB(config-ospf-1)#end

SwitchC#configure
SwitchC(config)#ospf 1
SwitchC(config-ospf-1)#network 172.17.100.0 255.255.255.0 area 0
SwitchC(config-ospf-1)#network 10.220.2.0 255.255.255.0 area 0
SwitchC(config-ospf-1)#end
```

在 Switch C 上查看路由表信息，可以看到 Switch C 通过 OSPF 学习到了到达 192.168.64.0/24、192.168.74.0/24 和 192.168.99.0/24 网段的路由。

```
SwitchC#configure
SwitchC(config-ospf-1)#show ospf route
 OSPF Process 1
 RoutType  Prefix           PathType  Cost   Cost2      NextHop
BackupNextHop    AreaId  Time
 ASBR     2.2.2.2/32        INTRA    1      0         172.17.100.1
0.0.0.0         0      0:00:17
 Network   10.220.2.0/24      INTRA    1      0         10.220.2.16
0.0.0.0         N/A    0:01:01
 Network   172.17.100.0/24    INTRA    1      0         172.17.100.2
0.0.0.0         N/A    0:00:20
 Network   192.168.64.0/24    ASE2     1      1         172.17.100.1
0.0.0.0         N/A    0:00:17
 Network   192.168.74.0/24    ASE2     1      1         172.17.100.1
0.0.0.0         N/A    0:00:17
 Network   192.168.99.0/24    ASE2     1      1         172.17.100.1
0.0.0.0         N/A    0:00:17
 Route :
 ABR      ASBR     Network   Intra    Inter    External
 0        1        5         2        0        3

 Path :
 ABR      ASBR     Network   Intra    Inter    External
 0        1        5         2        0        3
```

步骤 4 在 Switch C 和 Switch D 之间配置 BGP，并引入 OSPF 路由。

```
SwitchC#configure
SwitchC(config)#bgp 65106
SwitchC(config-bgp)#router-id 3.3.3.3
SwitchC(config-bgp)#neighbor 10.220.2.217 remote-as 64631
SwitchC(config-bgp)#ipv4-family unicast
```

```
SwitchC(config-bgp-af-ipv4)#import-route ospf
SwitchC(config-bgp-af-ipv4)#end

SwitchD#configure
SwitchD(config)#bgp 64631
SwitchD(config-bgp)#router-id 4.4.4.4
SwitchD(config-bgp)#neighbor 10.220.2.16 remote-as 65106
SwitchD(config-bgp)#end
```

在 Switch D 上查看路由表信息，可以看到 Switch D 通过 BGP 学习到了到达 192.168.64.0/24、192.168.74.0/24 和 192.168.99.0/24 三个网段的路由。在 Switch D 上可以 ping 通 192.168.64.0/24、192.168.74.0/24 和 192.168.99.0/24 网段内的主机。

```
SwitchD#configure
SwitchD(config-bgp)#show ip route
 Max Routing Number Limit: 131072
 Max Destination Number Limit: 771
 Routing Tables:
 Destination      Gateway         Pre/Cost     Proto  Interface
Mpls     Status    Vpn-Instance
------------------------------------------------------------------------------
-------------------------------------
 4.4.4.4/32       4.4.4.4           0/0        local  loopback-1         no
Up       N/A
 10.220.2.0/24    10.220.2.217       0/0        local  vlan-200           no
Up       N/A
 10.220.2.217/32  10.220.2.217       0/0        local  vlan-200           no
Up       N/A
 127.0.0.1/32     127.0.0.1         0/0        local  loopback-0         no
Up       N/A
 172.17.100.0/24  10.220.2.16      255/1        bgp    vlan-200           no
Up       N/A
 192.168.64.0/24  10.220.2.16      255/1        bgp    vlan-200           no
Up       N/A
 192.168.74.0/24  10.220.2.16      255/1        bgp    vlan-200           no
Up       N/A
 192.168.99.0/24  10.220.2.16      255/1        bgp    vlan-200           no
Up       N/A
------------------------------------------------------------------------------
-------------------------------------
 Total: 8         Static: 0         Down: 0         Destination: 8
```

步骤 5 在 Switch C 上配置路由聚合，将路由 192.168.64.0/24、192.168.74.0/24 和 192.168.99.0/24 聚合为 192.168.64.0/18，并抑制发布具体路由。

```
SwitchC#configure
SwitchC(config)#bgp
SwitchC(config-bgp)#ipv4-family unicast
SwitchC(config-bgp-af-ipv4)#summary 192.168.64.0/18 aggregated-only
SwitchC(config-bgp-af-ipv4)#end
```

在 Switch D 上查看路由表信息，可以看到 Switch D 上到达公司内部三个网络的路由聚合为一条路由 192.168.64.0/18。在 Switch D 上可以 ping 通 192.168.64.0/24、192.168.74.0/24 和 192.168.99.0/24 网段内的主机。

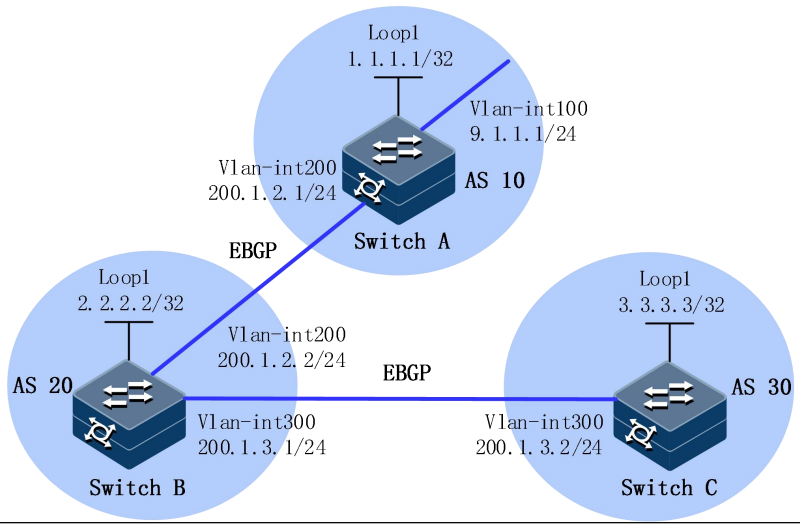
```
SwitchD#configure
SwitchD(config-bgp)#show ip route
 Max Routing Number Limit: 131072
 Max Destination Number Limit: 771
 Routing Tables:
 Destination      Gateway         Pre/Cost     Proto  Interface
Mpls     Status    Vpn-Instance
------------------------------------------------------------------------------
-------------------------------------
 4.4.4.4/32       4.4.4.4           0/0        local  loopback-1         no
Up       N/A
 10.220.2.0/24    10.220.2.217       0/0        local  vlan-200           no
Up       N/A
 10.220.2.217/32  10.220.2.217       0/0        local  vlan-200           no
Up       N/A
 127.0.0.1/32     127.0.0.1         0/0        local  loopback-0         no
Up       N/A
 172.17.100.0/24  10.220.2.16       255/1       bgp    vlan-200           no
Up       N/A
 192.168.64.0/18  10.220.2.16       255/0       bgp    vlan-200           no
Up       N/A
------------------------------------------------------------------------------
-------------------------------------
 Total: 6         Static: 0         Down: 0         Destination: 6
```

3.7.14 配置 BGP 团体功能示例

组网需求

如下图所示，Switch B 分别与 Switch A、Switch C 之间建立 EBGP 连接。通过在 Switch A 上配置 NO_EXPORT 团体属性，使得 AS 10 发布到 AS 20 中的路由，不会再被 AS 20 发布到其他 AS。

图 3-16 BGP 团体组网图



配置步骤

步骤 1 配置各接口的 IP 地址。

步骤 2 配置 EBGP。

```
SwitchA#configure
SwitchA(config)#bgp 10
SwitchA(config-bgp)#router-id 1.1.1.1
SwitchA(config-bgp)#neighbor 200.1.2.2 remote-as 20
SwitchA(config-bgp)#ipv4-family unicast
SwitchA(config-bgp-af-ipv4)#network 9.1.1.0/24
SwitchA(config-bgp-af-ipv4)#end

SwitchB#configure
SwitchB(config)#bgp 20
SwitchB(config-bgp)#router-id 2.2.2.2
SwitchB(config-bgp)#neighbor 200.1.2.1 remote-as 10
SwitchB(config-bgp)#neighbor 200.1.3.2 remote-as 30
SwitchB(config-bgp)#end

SwitchC#configure
SwitchC(config)#bgp 30
SwitchC(config-bgp)#router-id 3.3.3.3
SwitchC(config-bgp)#neighbor 200.1.3.1 remote-as 20
```

查看 Switch B、Switch C 的路由表。

```
SwitchB#configure
SwitchB(config)#show bgp ip-route
DestAddr/Prefixlen     Neighbor            Nexthop             Protocol
Med        LocalPrf    Origin    Vpn-Instance   As-Path
------------------------------------------------------------------------------
------------------------------------------------------
9.1.1.0/24             200.1.2.1           200.1.2.1           bgp      0
0          IGP       public        AS_SEQUENCE 10
------------------------------------------------------------------------------
------------------------------------------------------
  Total:1

SwitchC#configure
SwitchC(config)#show bgp ip-route
DestAddr/Prefixlen     Neighbor            Nexthop             Protocol
Med        LocalPrf    Origin    Vpn-Instance   As-Path
------------------------------------------------------------------------------
------------------------------------------------------
9.1.1.0/24             200.1.3.1           200.1.3.1           bgp      0
0          IGP       public        AS_SEQUENCE 20,10
------------------------------------------------------------------------------
------------------------------------------------------
  Total:1
```

可以看出，Switch C 从 Switch B 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。

```
SwitchA#configure
SwitchA(config)#bgp
SwitchA(config-bgp)#ipv4-family unicast
SwitchA(config-bgp-af-ipv4)#community noexport additive
SwitchA(config-bgp-af-ipv4)#neighbor 200.1.2.2 advertise-community
```

查看 Switch B、Switch C 的路由表。

```
SwitchB#configure
SwitchB(config)#show bgp ip-route destination-address 9.1.1.0/24
 BGP routing table entry information

 NetWork               :9.1.1.0/24
 Peer                 :200.1.2.1
 NextHop               :200.1.2.1
 Protocol              :bgp
 Med                  :0
 Local Preference       :0
 As-Path               :AS_SEQUENCE 10
 Origin                :IGP
 Best                 :True
 InLabel               :0
 OutLabel              :0
 AtomicAggr            :0
 HWState               :1
 HWUpdateTime           :0h:2m:58s
 SummaryRoute           :0
 SummaryFilter          :0
 WaitDelete            :0
 BestSysroute           :1
 MPLSUpdate            :0
 ImportFiltered         :0
 IGPSyncWait           :0
 SourceVRF             :0
 Damp Supressed         :N
 DirectNexthop          :200.1.2.1
 Less Priority Reason    :none
 Community             :ffffff01
 VPN-Instance           :public
```

```
SwitchC#configure
SwitchC(config)#show bgp ip-route
```

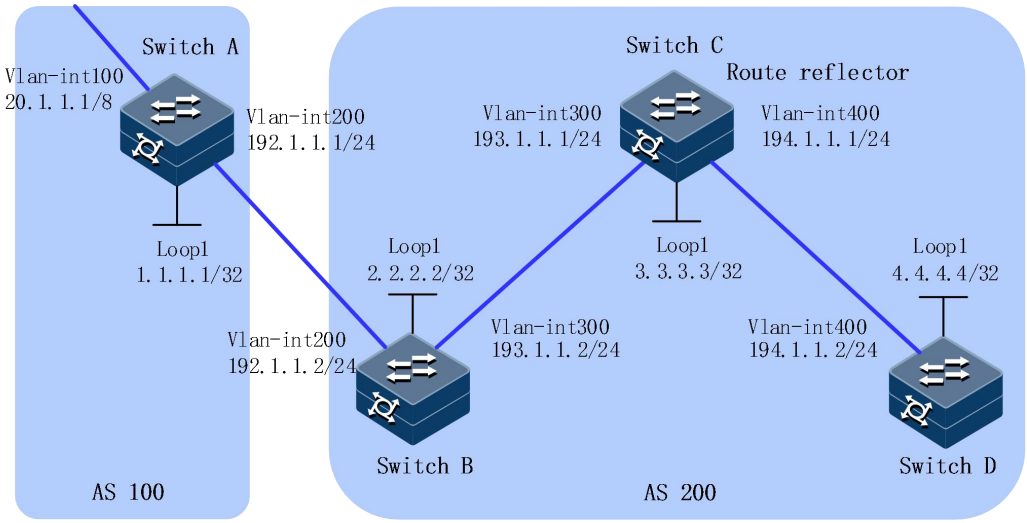
在 Switch B 的 BGP 路由表中可以看到配置的团体属性，Switch B 不会通过 BGP 将到达目的地址 9.1.1.0/24 的路由发布出去。

3.7.15 配置 BGP 路由反射器功能示例

组网需求

如下图所示，所有交换机运行 BGP 协议，Switch A 与 Switch B 建立 EBGP 连接，Switch C 与 Switch B 和 Switch D 之间建立 IBGP 连接。Switch C 作为路由反射器，Switch B 和 Switch D 为 Switch C 的客户机。Switch D 能够通过 Switch C 学到路由 20.0.0.0/8。

图 3-17 配置 BGP 路由反射器的组网图



配置步骤

步骤 1 配置各接口的 IP 地址。

步骤 2 在 AS 200 内配置 OSPF。

```
SwitchB#configure
SwitchB(config)#ospf 1
SwitchB(config-ospf-1)#router-id 2.2.2.2
SwitchB(config-ospf-1)#network 193.1.1.0 255.255.255.0 area 0
SwitchB(config-ospf-1)#network 2.2.2.2 255.255.255.255 area 0
SwitchB(config-ospf-1)#end

SwitchC#configure
SwitchC(config)#ospf 1
SwitchC(config-ospf-1)#router-id 3.3.3.3
SwitchC(config-ospf-1)#network 193.1.1.0 255.255.255.0 area 0
SwitchC(config-ospf-1)#network 194.1.1.0 255.255.255.0 area 0
SwitchC(config-ospf-1)#network 3.3.3.3 255.255.255.255 area 0

SwitchD#configure
SwitchD(config)#ospf 1
SwitchD(config-ospf-1)#router-id 4.4.4.4
SwitchD(config-ospf-1)#network 194.1.1.0 255.255.255.0 area 0
SwitchD(config-ospf-1)#network 4.4.4.4 255.255.255.255 area 0
SwitchD(config-ospf-1)#end
```

步骤 3 配置 BGP 连接。

```
SwitchA#configure
SwitchA(config)#bgp 100
SwitchA(config-bgp)#router-id 1.1.1.1
SwitchA(config-bgp)#neighbor 192.1.1.2 remote-as 200
SwitchA(config-bgp)#ipv4-family unicast
SwitchA(config-bgp-af-ipv4)#network 20.0.0.0/8
SwitchA(config-bgp-af-ipv4)#end

SwitchB#configure
SwitchB(config)#bgp 200
SwitchB(config-bgp)#router-id 2.2.2.2
SwitchB(config-bgp)#neighbor 192.1.1.1 remote-as 100
SwitchB(config-bgp)#neighbor 193.1.1.1 remote-as 200
SwitchB(config-bgp)#ipv4-family unicast
SwitchB(config-bgp-af-ipv4)#neighbor 193.1.1.1 next-hop-local
SwitchB(config-bgp-af-ipv4)#end

SwitchC#configure
SwitchC(config)#bgp 200
SwitchC(config-bgp)#router-id 3.3.3.3
SwitchC(config-bgp)#neighbor 193.1.1.2 remote-as 200
SwitchC(config-bgp)#neighbor 194.1.1.2 remote-as 200
SwitchC(config-bgp)#end

SwitchD#configure
SwitchD(config)#bgp 200
SwitchD(config-bgp)#router-id 4.4.4.4
SwitchD(config-bgp)#neighbor 194.1.1.1 remote-as 200
SwitchD(config-bgp)#end
```

步骤 4 配置路由反射器。

```
SwitchC#configure
SwitchC(config)#bgp
SwitchC(config)#ipv4-family unicast
SwitchC(config-bgp-af-ipv4)#neighbor 193.1.1.2 route-reflector-client
SwitchC(config-bgp-af-ipv4)#neighbor 194.1.1.2 route-reflector-client
SwitchC(config-bgp-af-ipv4)#end
```

查看 Switch B、Switch D 的 BGP 路由表。

```
SwitchB#configure
SwitchB(config)#show bgp ip-route
DestAddr/Prefixlen     Neighbor             Nexthop              Protocol
Med        LocalPrf    Origin    Vpn-Instance   As-Path
------------------------------------------------------------------------------
------------------------------------------------------
20.0.0.0/8            192.1.1.1            192.1.1.1            bgp      0
0          IGP       public       AS_SEQUENCE 100
------------------------------------------------------------------------------
------------------------------------------------------
```

```
 Total:1

SwitchD#configure
SwitchD(config)#show bgp ip-route
DestAddr/Prefixlen     Neighbor            Nexthop              Protocol
Med        LocalPrf    Origin     Vpn-Instance   As-Path
-------------------------------------------------------------------------------
-------------------------------------------------------
20.0.0.0/8             194.1.1.1           193.1.1.2            bgp      0
100        IGP         public         AS_SEQUENCE 100
-------------------------------------------------------------------------------
-------------------------------------------------------
 Total:1
```

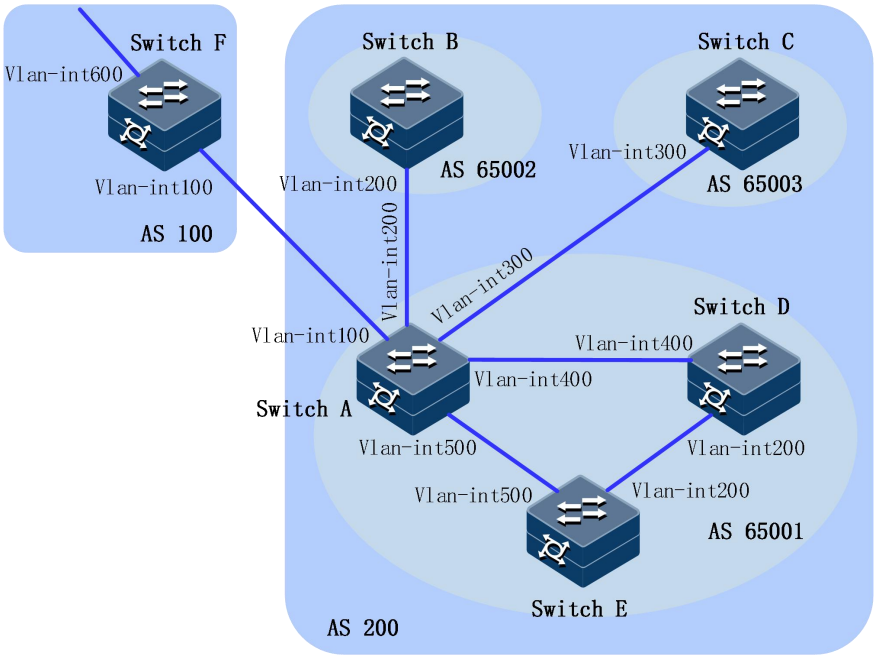
可以看出，Switch D 从 Switch C 已经学到了 20.0.0.0/8 路由。

3.7.16 配置 BGP 联盟功能示例

组网需求

如下图所示，AS 200 中有多台 BGP 交换机，为了减少 IBGP 的连接数，现将他们划分为 3 个子自治系统：AS 65001、AS 65002 和 AS 65003。其中 AS 65001 内的三台交换机建立 IBGP 全连接。

图 3-18 配置联盟组网图



设备	接口	IP 地址
Switch A	Vlan-int100	200.1.1.1/24

	Vlan-int200	10.1.1.1/24
	Vlan-int300	10.1.2.1/24
	Vlan-int400	10.1.3.1/24
	Vlan-int500	10.1.4.1/24
Switch B	Vlan-int200	10.1.1.2/24
Switch C	Vlan-int300	10.1.2.2/24
Switch D	Vlan-int200	10.1.5.1/24
	Vlan-int400	10.1.3.2/24
Switch E	Vlan-int200	10.1.5.2/24
	Vlan-int500	10.1.4.2/24
Switch F	Vlan-int100	200.1.1.2/24
	Vlan-int600	9.1.1.1/24

配置步骤

步骤 1 配置各接口的 IP 地址。

步骤 2 配置 BGP 联盟。

```
SwitchA#configure
SwitchA(config)#bgp 65001
SwitchA(config-bgp)#router-id 1.1.1.1
SwitchA(config-bgp)#confederation identifier 200
SwitchA(config-bgp)#confederation peer-as 65002,65003
SwitchA(config-bgp)#neighbor 10.1.1.2 remote-as 65002
SwitchA(config-bgp)#neighbor 10.1.2.2 remote-as 65003
SwitchA(config-bgp)#ipv4-family unicast
SwitchA(config-bgp-af-ipv4)#neighbor 10.1.1.2 next-hop-local
SwitchA(config-bgp-af-ipv4)#neighbor 10.1.2.2 next-hop-local
SwitchA(config-bgp-af-ipv4)#end

SwitchB#configure
SwitchB(config)#bgp 65002
SwitchB(config-bgp)#router-id 2.2.2.2
SwitchB(config-bgp)#confederation identifier 200
SwitchB(config-bgp)#confederation peer-as 65001,65003
SwitchB(config-bgp)#neighbor 10.1.1.1 remote-as 65001
SwitchB(config-bgp)#end

SwitchC#configure
SwitchC(config)#bgp 65003
SwitchC(config-bgp)#router-id 3.3.3.3
SwitchC(config-bgp)#confederation identifier 200
SwitchC(config-bgp)#confederation peer-as 65001,65002
```

```
SwitchC(config-bgp)#neighbor 10.1.2.1 remote-as 65001
SwitchC(config-bgp)#end
```

步骤 3 配置 AS 65001 内的 IBGP 连接。

```
SwitchA#configure
SwitchA(config)#bgp 65001
SwitchA(config-bgp)#neighbor 10.1.3.2 remote-as 65001
SwitchA(config-bgp)#neighbor 10.1.4.2 remote-as 65001
SwitchA(config-bgp)#ipv4-family unicast
SwitchA(config-bgp-af-ipv4)#neighbor 10.1.3.2 next-hop-local
SwitchA(config-bgp-af-ipv4)#neighbor 10.1.4.2 next-hop-local
SwitchA(config-bgp)#end

SwitchD#configure
SwitchD(config)#bgp 65001
SwitchD(config-bgp)#router-id 4.4.4.4
SwitchD(config-bgp)#confederation identifier 200
SwitchD(config-bgp)#neighbor 10.1.3.1 remote-as 65001
SwitchD(config-bgp)#neighbor 10.1.5.2 remote-as 65001
SwitchD(config-bgp)#end

SwitchE#configure
SwitchE(config)#bgp 65001
SwitchE(config-bgp)#router-id 5.5.5.5
SwitchE(config-bgp)#confederation identifier 200
SwitchE(config-bgp)#neighbor 10.1.4.1 remote-as 65001
SwitchE(config-bgp)#neighbor 10.1.5.1 remote-as 65001
SwitchE(config-bgp)#end
```

步骤 4 配置 AS 100 和 AS 200 之间的 EBGP 连接。

```
SwitchA#configure
SwitchA(config)#bgp 65001
SwitchA(config-bgp)#neighbor 200.1.1.2 remote-as 100
SwitchA(config-bgp)#end

SwitchF#configure
SwitchF(config)#bgp 100
SwitchF(config-bgp)#router-id 6.6.6.6
SwitchF(config-bgp)#neighbor 200.1.1.1 remote-as 200
SwitchF(config-bgp)#ipv4-family unicast
SwitchF(config-bgp-af-ipv4)#network 9.1.1.0/24
SwitchF(config-bgp-af-ipv4)#end
```

步骤 5 验证配置。

```
SwitchB#configure
SwitchB(config)#show bgp ip-route
DestAddr/Prefixlen     Neighbor            Nexthop             Protocol
Med        LocalPrf    Origin    Vpn-Instance   As-Path
------------------------------------------------------------------------------
------------------------------------------------------
9.1.1.0/24             10.1.1.1            10.1.1.1            bgp      0
100        IGP       public       AS_CONFED_SEQUENCE 65001
------------------------------------------------------------------------------
------------------------------------------------------
```

```
 Total:1

SwitchC#configure
SwitchC(config)#show bgp ip-route
DestAddr/Prefixlen     Neighbor            Nexthop             Protocol
Med        LocalPrf    Origin    Vpn-Instance   As-Path
------------------------------------------------------------------------------
------------------------------------------------------------
9.1.1.0/24             10.1.2.1            10.1.2.1            bgp      0
100        IGP       public        AS_CONFED_SEQUENCE 65001
------------------------------------------------------------------------------
------------------------------------------------------------
 Total:1

SwitchD#configure
SwitchD(config)#show bgp ip-route
DestAddr/Prefixlen     Neighbor            Nexthop             Protocol
Med        LocalPrf    Origin    Vpn-Instance   As-Path
------------------------------------------------------------------------------
------------------------------------------------------------
9.1.1.0/24             10.1.3.1            10.1.3.1            bgp      0
100        IGP       public        AS_SEQUENCE 100
------------------------------------------------------------------------------
------------------------------------------------------------
 Total:1
```

Switch F 只需要和 Switch A 建立 EBGP 连接，而不需要和 Switch B、Switch C 建立连接，同样可以通过联盟将路由信息传递给 Switch B 和 Switch C。

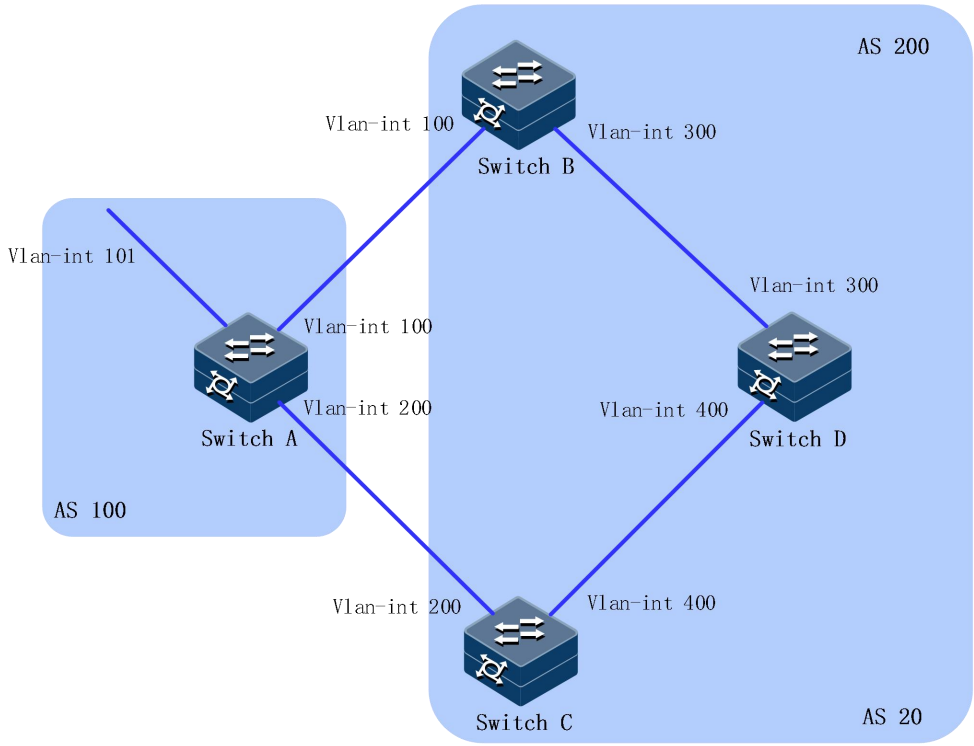
Switch B 和 Switch D 在同一个联盟里，但是属于不同的子自治系统，它们都是通过 Switch A 来获取外部路由信息，生成的 BGP 路由表项也是一致的，等效于在同一个自治系统内，但是又不需要物理上全连接。

3.7.17 配置 BGP 路径选择示例

组网需求

如下图所示，所有路由器都运行 BGP 协议。Switch A 与 Switch B 和 Switch C 之间运行 EBGP；Switch D 与 Switch B 和 Switch C 之间运行 IBGP。AS 200 中运行 OSPF 协议。配置路由策略，使得 Switch D 优选从 Switch C 学到的 1.0.0.0/8 路由。

图 3-19 配置 BGP 路径选择的组网图



设备	接口	IP 地址
Switch A	Vlan-int101	1.0.0.1/8
	Vlan-int100	192.1.1.1/24
	Vlan-int200	193.1.1.1/24
Switch B	Vlan-int100	192.1.1.2/24
	Vlan-int300	194.1.1.2/24
Switch C	Vlan-int400	195.1.1.2/24
	Vlan-int200	193.1.1.2/24
Switch D	Vlan-int400	195.1.1.1/24
	Vlan-int300	194.1.1.1/24

配置步骤

步骤 1 配置各接口的 IP 地址。

步骤 2 配置 Switch B、Switch C 和 Switch D 之间运行 OSPF 协议。

```
SwitchB#configure
SwitchB(config)#ospf 1
SwitchB(config-ospf-1)#network 192.1.1.0 255.255.255.0 area 0
SwitchB(config-ospf-1)#network 194.1.1.0 255.255.255.0 area 0
```

```
SwitchB(config-ospf-1)#end

SwitchC#configure
SwitchC(config)#ospf 1
SwitchC(config-ospf-1)#network 193.1.1.0 255.255.255.0 area 0
SwitchC(config-ospf-1)#network 195.1.1.0 255.255.255.0 area 0
SwitchC(config-ospf-1)#end

SwitchD#configure
SwitchD(config)#ospf 1
SwitchD(config-ospf-1)#network 194.1.1.0 255.255.255.0 area 0
SwitchD(config-ospf-1)#network 195.1.1.0 255.255.255.0 area 0
SwitchD(config-ospf-1)#end
```

步骤 3 配置 BGP 连接。

```
SwitchA#configure
SwitchA(config)#bgp 100
SwitchA(config-bgp)#neighbor 192.1.1.2 remote-as 200
SwitchA(config-bgp)#neighbor 193.1.1.2 remote-as 200
SwitchA(config-bgp)#ipv4-family unicast
SwitchA(config-bgp-af-ipv4)#network 1.0.0.0/8
SwitchA(config-bgp-af-ipv4)#end

SwitchB#configure
SwitchB(config)#bgp 200
SwitchB(config-bgp)#neighbor 192.1.1.1 remote-as 100
SwitchB(config-bgp)#neighbor 194.1.1.1 remote-as 200
SwitchB(config-bgp)#end

SwitchC#configure
SwitchC(config)#bgp 200
SwitchC(config-bgp)#neighbor 193.1.1.1 remote-as 100
SwitchC(config-bgp)#neighbor 195.1.1.1 remote-as 200
SwitchC(config-bgp)#end

SwitchD#configure
SwitchD(config)#bgp 200
SwitchD(config-bgp)#neighbor 194.1.1.2 remote-as 200
SwitchD(config-bgp)#neighbor 195.1.1.2 remote-as 200
SwitchD(config-bgp)#end
```

步骤 4 通过配置 1.0.0.0/8 路由的不同属性值，使得 Switch D 优选从 Switch C 学到的路由。。

方法一：在 Switch A 上对发布给对等体 192.1.1.2 的 1.0.0.0/8 路由配置较高的 MED 属性值，使得 Switch D 优选从 Switch C 学到的路由。

```
SwitchA#configure
SwitchA(config)#route-policy apply_med_100 permit node 10
SwitchA(config-route-policy-apply_med_100-10)#apply cost 100
SwitchA(config-route-policy-apply_med_100-10)#exit
SwitchA(config)#route-policy apply_med_50 permit node 10
SwitchA(config-route-policy-apply_med_50-10)#apply cost 50
SwitchA(config-route-policy-apply_med_50-10)#exit
SwitchA(config)#bgp
```

```
SwitchA(config-bgp)#ipv4-family unicast
SwitchA(config-bgp-af-ipv4)#neighbor 192.1.1.2 filter-policy export
route-policy apply_med_100
SwitchA(config-bgp-af-ipv4)#neighbor 193.1.1.2 filter-policy export
route-policy apply_med_50
SwitchA(config-bgp)#end
```

查看 Switch D 的 BGP 路由表。

```
SwitchD#configure
SwitchD(config)#show bgp ip-route destination-address 1.0.0.0/8
 BGP routing table entry information

 NetWork               :1.0.0.0/8
 Peer                 :194.1.1.2
 NextHop               :192.1.1.1
 Protocol              :bgp
 Med                  :100
 Local Preference       :100
 As-Path               :AS_SEQUENCE 100
 Origin                :IGP
 Best                 :False
 InLabel               :0
 OutLabel              :0
 AtomicAggr            :0
 HWState               :0
 HWUpdateTime          :0h:1m:43s
 SummaryRoute          :0
 SummaryFilter         :0
 WaitDelete            :0
 BestSysroute          :0
 MPLSUpdate            :0
 ImportFiltered        :0
 IGPSyncWait           :0
 SourceVRF             :0
 Damp Supressed        :N
 DirectNexthop         :194.1.1.2
 Less Priority Reason    :none
 VPN-Instance          :public
 BGP routing table entry information

 NetWork               :1.0.0.0/8
 Peer                 :195.1.1.2
 NextHop               :193.1.1.1
 Protocol              :bgp
 Med                  :50
 Local Preference       :100
 As-Path               :AS_SEQUENCE 100
 Origin                :IGP
 Best                 :True
 InLabel               :0
 OutLabel              :0
 AtomicAggr            :0
 HWState               :1
```

```
 HWUpdateTime          :0h:1m:13s
 SummaryRoute          :0
 SummaryFilter         :0
 WaitDelete           :0
 BestSysroute          :1
 MPLSUpdate           :0
 ImportFiltered        :0
 IGPSyncWait          :0
 SourceVRF            :0
 Damp Supressed        :N
 DirectNexthop         :195.1.1.2
 Less Priority Reason   :none
 VPN-Instance          :public
```

可以看到，Switch D 从 Switch C 学到 1.0.0.0/8 的路由是最优的。

方法二：在 Switch B 和 Switch C 上分别对 1.0.0.0/8 路由配置不同的本地优先级，使得 Switch D 优选从 Switch C 学到的路由。在 Switch C 上本地优先级为 200（缺省的本地优先级为 100）。

```
SwitchC#configure
SwitchC(config)#bgp
SwitchC(config-bgp)#ipv4-family unicast
SwitchC(config-bgp-af-ipv4)#default local-preference 200
SwitchC(config-bgp-af-ipv4)#end
```

查看 Switch D 的 BGP 路由表。

```
SwitchD#configure
SwitchD(config)#show bgp ip-route destination-address 1.0.0.0/8
 BGP routing table entry information

 NetWork              :1.0.0.0/8
 Peer                :194.1.1.2
 NextHop              :192.1.1.1
 Protocol             :bgp
 Med                 :0
 Local Preference      :100
 As-Path              :AS_SEQUENCE 100
 Origin               :IGP
 Best                :False
 InLabel              :0
 OutLabel             :0
 AtomicAggr           :0
 HWState             :0
 HWUpdateTime          :0h:0m:45s
 SummaryRoute          :0
 SummaryFilter         :0
 WaitDelete           :0
 BestSysroute          :0
 MPLSUpdate           :0
 ImportFiltered        :0
 IGPSyncWait          :0
 SourceVRF            :0
 Damp Supressed        :N
 DirectNexthop         :194.1.1.2
```

```
Less Priority Reason    :local preference
VPN-Instance           :public
BGP routing table entry information

NetWork               :1.0.0.0/8
Peer                 :195.1.1.2
NextHop               :193.1.1.1
Protocol              :bgp
Med                  :0
Local Preference       :200
As-Path               :AS_SEQUENCE 100
Origin               :IGP
Best                 :True
InLabel               :0
OutLabel              :0
AtomicAggr            :0
HWState              :1
HWUpdateTime          :0h:1m:35s
SummaryRoute          :0
SummaryFilter         :0
WaitDelete           :0
BestSysroute          :1
MPLSUpdate           :0
ImportFiltered        :0
IGPSyncWait          :0
SourceVRF            :0
Damp Supressed        :N
DirectNexthop         :195.1.1.2
Less Priority Reason    :none
VPN-Instance          :public
```

可以看到，Switch D 从 Switch C 学到 1.0.0.0/8 的路由是最优的。

3.8 OSPFV2

3.8.1 简介

功能

开放式最短路径优先 OSPF（Open Shortest Path First）是 IETF 组织开发的一个基于链路状态的内部网关协议（Interior Gateway Protocol）。

OSPF 具有以下特性：

- 适应范围广：支持各种规模的网络，最多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，使得自治系统中的其他节点能够快速同步这一变化。
- 无环路：OSPF 根据收集到的链路状态，用最短路径树算法计算路由，该算法保证了 OSPF 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，区域间传送的路由信息被进一步抽象，减少了占用的网络带宽。

- 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：支持基于接口的报文验证，保证报文交互的安全性。
- 组播发送：在能够发送组播的链路上，以组播地址发送协议报文，减少对其他设备的干扰。

OSPF 路由器 ID

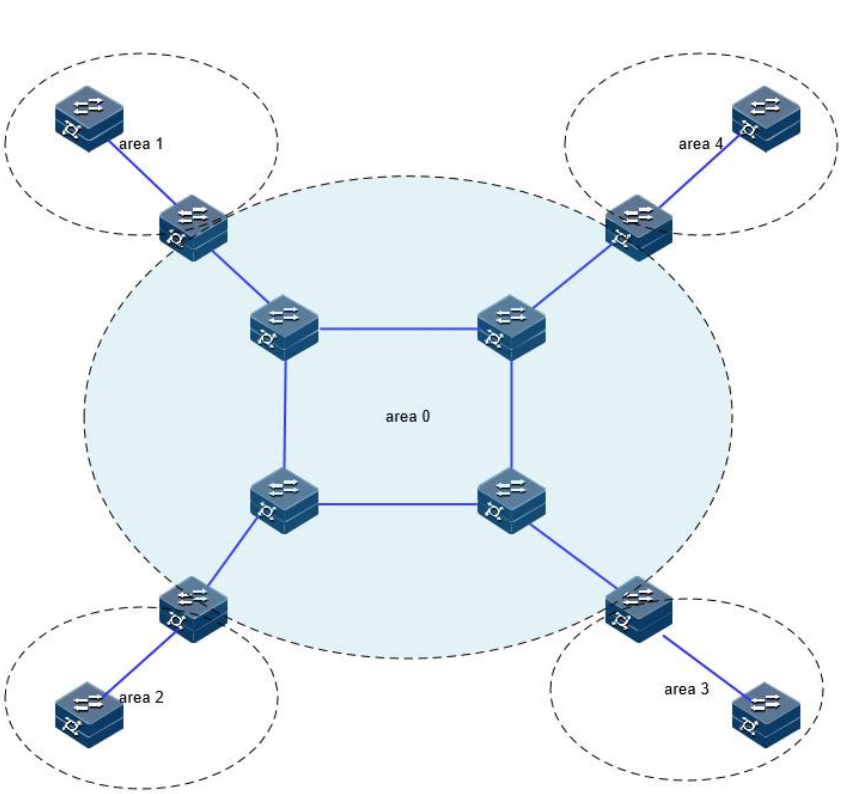
一台路由器如果要运行 OSPF 协议，必须存在路由器 ID。路由器 ID 是一个 32 比特无符号整数，是一台路由器在自治系统中的唯一标识。路由器的 ID 可以手工配置，也可以由系统自动产生，系统从当前接口的 IP 地址中自动选择最大的一个地址作为路由器的 ID 号。如果系统当前未配置任何 IP 地址，则无法启动 OSPF。

OSPF 区域划分

随着网络规模日益扩大，当一个大型网络中的设备都运行 OSPF 路由协议时，设备数量的增多会导致链路状态数据库 LSDB（Link-State Database）非常庞大，占用大量的存储空间，并使得运行 SPF 算法的复杂度增加，导致设备负担很重。在网络规模增大之后，拓扑结构发生变化的概率也增大，网络会经常处于“动荡”之中，造成网络中会有大量的 OSPF 协议报文在传递，降低了网络的带宽利用率。更为严重的是，每一次变化都会导致网络中所有的设备重新进行路由计算。

为了解决上述问题，OSPF 协议将自治系统划分成不同的区域（Area）。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。

图 3-20 OSPF 区域划分

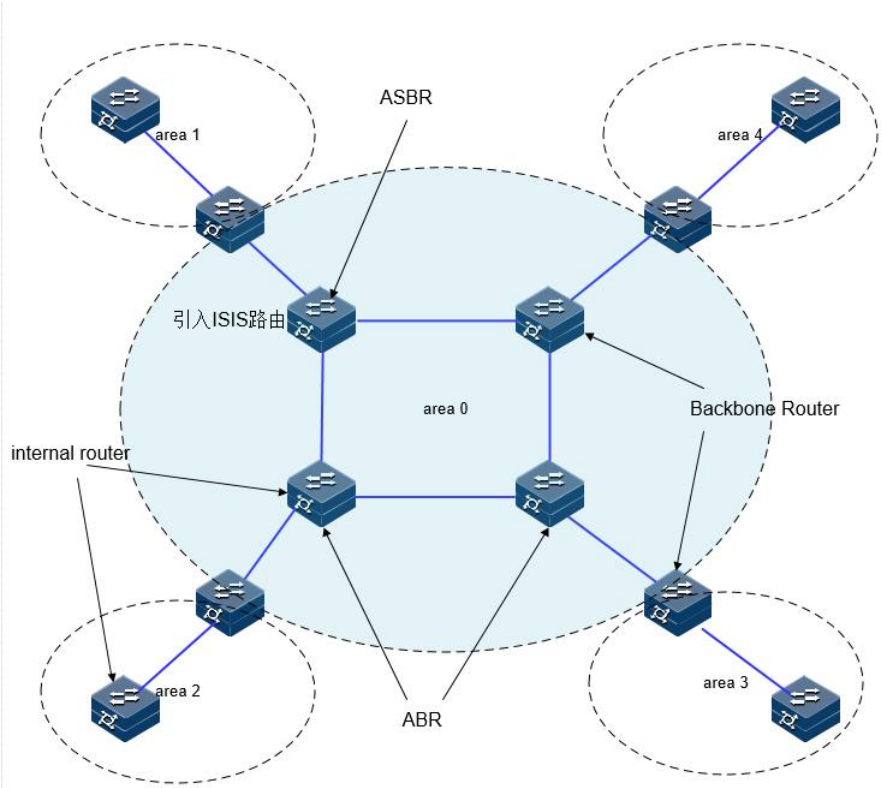


划分区域后，可以在区域边界路由器上进行路由聚合，减少通告到其他区域的 LSA 数量。另外，划分区域还可以使网络拓扑变化造成的影响最小化。

OSPF 的区域类型包括普通区域、Stub 区域、NSSA 区域。

OSPF 路由器类型

图 3-21 OSPF 路由器类型示例



- 区域内路由器（Internal Router）：该类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 Router）：该类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 Router）：该类设备至少有一个接口属于骨干区域。所有的 ABR 和位于 Area0 的内部设备都是骨干路由器。
- 自治系统边界路由器 ASBR（AS Boundary Router）：与其他 AS 交换路由信息的设备称为 ASBR。ASBR 并不一定位于 AS 的边界，它可能是区域内设备，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。

OSPF 协议报文

OSPF 有以下五种类型的协议报文：

- Hello 报文：周期性发送，用于发现和维持 OSPF 邻居关系。
- DD（Database Description Packet）报文：描述本地 LSDB 的摘要信息，用于两台路由器开始建立邻接时进行数据库同步。
- LSR 报文（Link State Request Packet）：向对方请求所需的 LSA。
- LSU 报文（Link State Update Packet）：向对方发送其所需要的 LSA。

- LSAck 报文（Link State Acknowledgment Packet）：用来对收到的 LSA 进行确认。

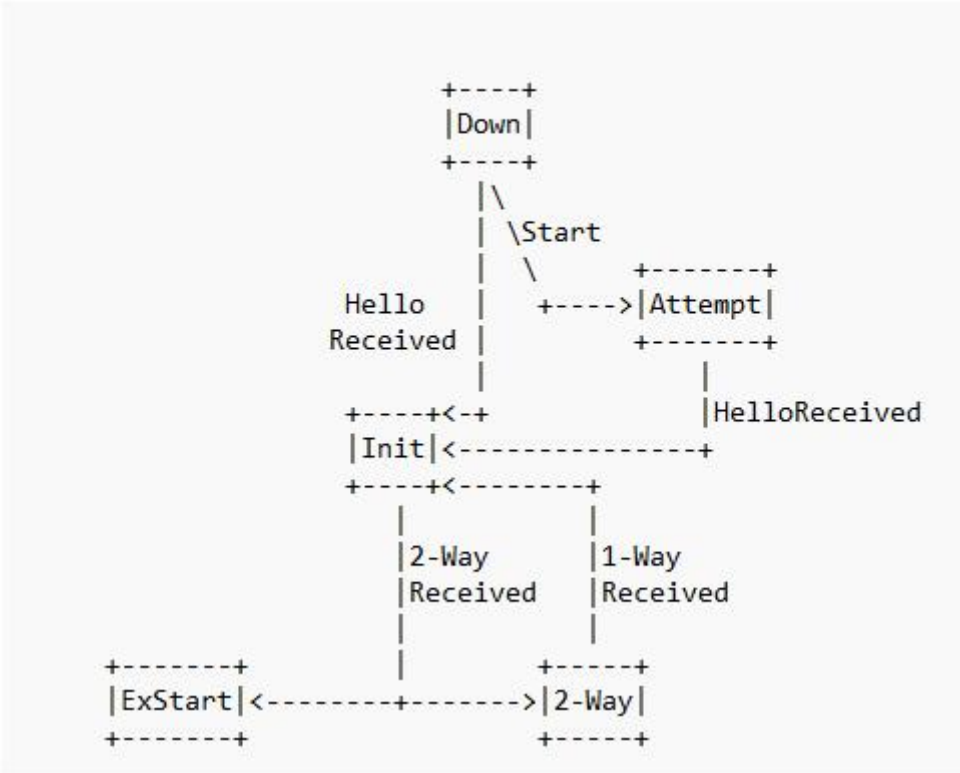
OSPF LSA 类型

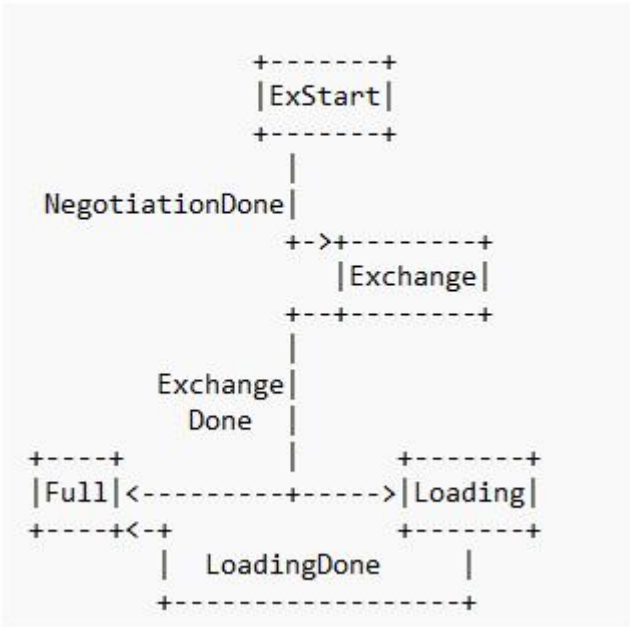
OSPF 中对路由信息的描述都是封装在 LSA 中发布出去，常用的 LSA 有以下类型：

- Router LSA（Type1）：每个路由器都会产生，描述了路由器的链路状态和开销，在所属的区域内传播。
- Network LSA（Type2）：由 DR 产生，描述本网段的链路状态，在所属的区域内传播。
- Network Sumary LSA（Type3）：由 ABR（Area Border Router）产生，描述区域内某个网段的路由，并通告给其他区域。
- ASBR Summary LSA（Type4）：由 ABR 产生，描述到 ASBR（Autonomous System Boundary Router）的路由，通告给相关区域。
- AS External LSA（Type5）：由 ASBR 产生，描述到 AS 外部的路由，通告到所有的区域（除了 Stub 区域和 NSSA（Not-So-Stubby Area）区域）。
- NSSA LSA（Type7）：由 ASBR 产生，描述到 AS 外部的路由，仅在 NSSA 区域内传播。
- Opaque LSA（Type9/Type10/Type11）：Opaque LSA 提供用于 OSPF 的扩展的通用机制。

OSPF 邻居状态机

图 3-22 OSPF 邻居状态机



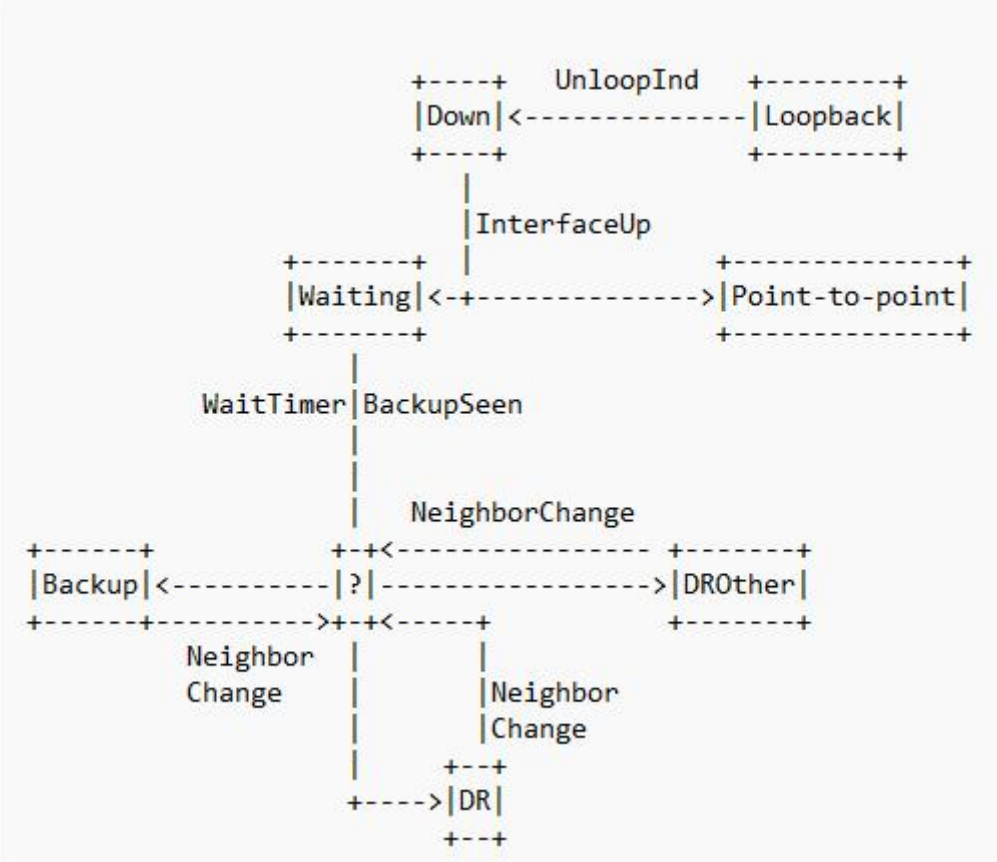


邻居状态机的含义：

- Down：邻居会话的初始阶段。表明没有在邻居失效时间间隔内收到来自邻居设备的 Hello 报文。
- Attempt：处于本状态时，定期向手工配置的邻居发送 Hello 报文。说明： Attempt 状态只适用于 NBMA 类型的接口。
- Init：本状态表示已经收到了邻居的 Hello 报文，但是对端并没有收到本端发送的 Hello 报文。
- 2-way：互为邻居。本状态表示双方互相收到了对端发送的 Hello 报文，建立了邻居关系。如果不形成邻接关系则邻居状态机就停留在此状态，否则进入 Exstart 状态。
- Exstart：协商主/从关系。建立主/从关系主要是为了保证在后续的 DD 报文交换中能够有序的发送。
- Exchange：交换 DD 报文。本端设备将本地的 LSDB 用 DD 报文来描述，并发给邻居设备。
- Loading：正在同步 LSDB。两端设备发送 LSR 报文向邻居请求对方的 LSA，同步 LSDB。
- Full：建立邻接。两端设备的 LSDB 已同步，本端设备和邻居设备建立了邻接状态。

OSPF 接口状态机

图 3-23 OSPF 接口状态机



接口状态机的含义：

- Down：接口状态的初始状态。
- Loopback：在这种状态下，路由器到网络的接口被环路回绕。
- Waiting：仅适用于广播和 NBMA 接口类型，在该阶段试图识别网络上的 DR 和 BDR。
- Point-to-point：仅适用于点对点，点到多点和虚连接的接口。在这个状态下，尝试和接口另一端路由器建立邻接关系。
- DR：在这种状态下，路由器成功所连网络的 DR，尝试与其他路由器建立邻接关系。
- Backup：在这种状态下，路由器成功所连网络的 BDR，尝试与其他路由器建立邻接关系。
- DROther：在这种状态下，路由器成功所连网络的 DRother，仅会和网络上的 DR 和 BDR 建立邻接关系。

OSPF 邻居和邻接

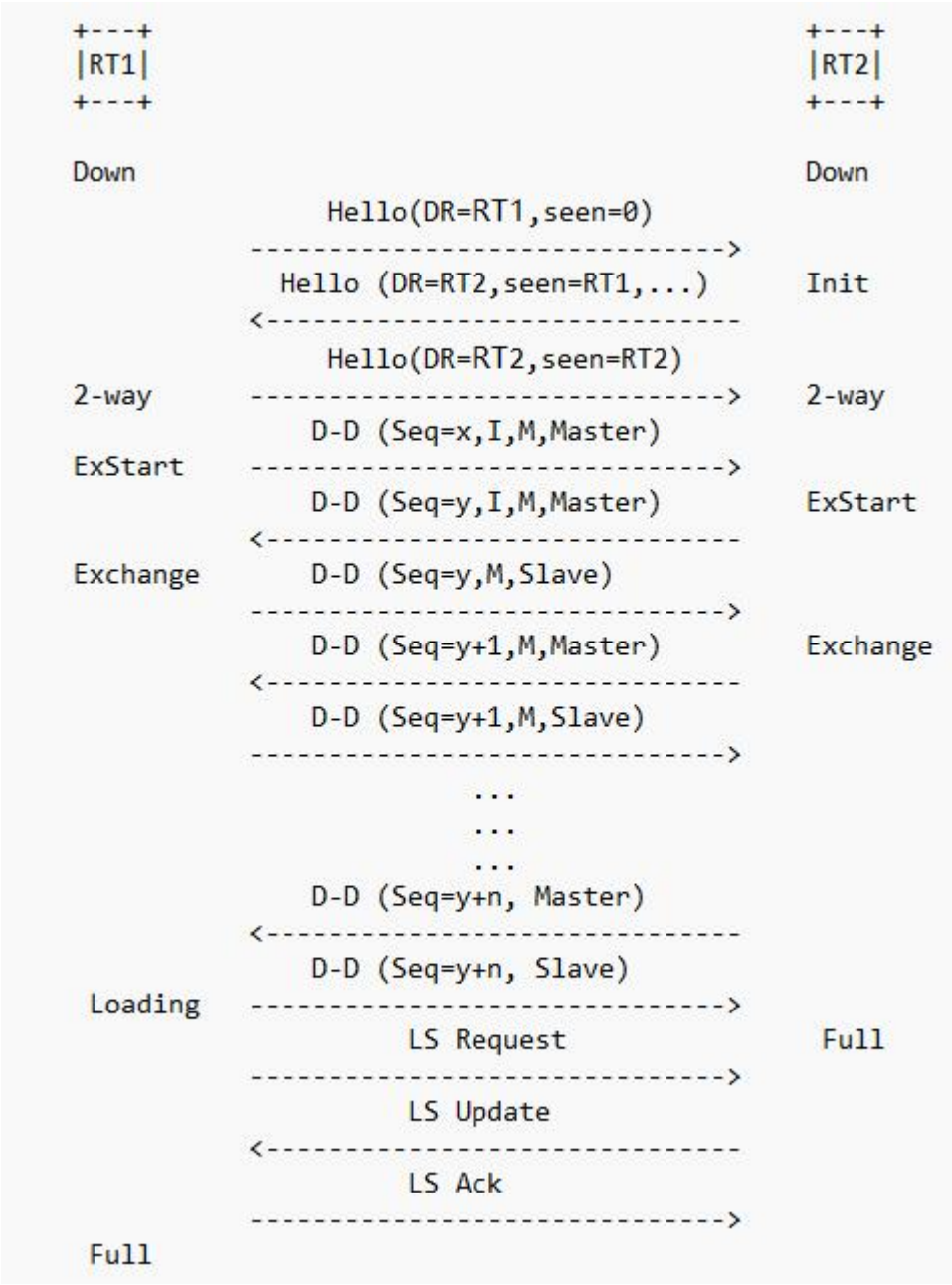
在 OSPF 中，邻居（Neighbors）和邻接（Adjacencies）是两个不同的概念。

邻居关系：OSPF 路由器启动后，会通过 OSPF 接口向外发送 Hello 报文。收到 Hello 报文的 OSPF 路由器会检查报文中所定义的一些参数，如果双方一致就会形成邻居关系。

邻接关系：形成邻居关系的双方不一定都能形成邻接关系，这要根据网络类型而定。只有当双方成功交换 DD 报文，并能交换 LSA 之后，才形成真正意义上的邻接关系。

在广播网络中建立邻接关系流程：

图 3-24 OSPF 广播网络建邻示意图



RT1 和 RT2 都连接到广播网络。假设 RT2 为网络的 DR（指定路由器），且 RT2 的 Router ID 高于 Router RT1。

（1）建立邻居关系，RT1 的一个连接到广播类型网络的接口上激活了 OSPF 协议，会开始发送 Hello 数据包。此时，RouterA 认为自己是 DR 路由器，但不确定邻居是哪台路由器。RT2 收到这个 hello 报文(将邻居状态机改为 Init 状态)，并在其下一个 hello 中数据包表明它本身就是 DR（指定路由器）。RT1 收到 RT2 发来的 Hello 报文。这反过来又导致 RT1 转到状态 2-way，开始启动邻接流程。

（2）主/从关系协商、DD 报文交换

RT1 首先发送一个 DD 报文，宣称自己是 Master（MS=1），并规定序列号 Seq=x。I=1 表示这是第一个 DD 报文，报文中并不包含 LSA 的摘要，只是为了协商主从关系。M=1 说明这不是最后一个报文。

RT2 在收到 RT1 的 DD 报文后，将 RT1 的邻居状态机改为 Exstart，并且回应一个 DD 报文。由于 RT2 的 Router ID 较大，所以在报文中 RT2 认为自己是 Master，并且重新规定了序列号 Seq=y。

RT1 收到报文后，同意了 RT2 为 Master，并将 RT2 的邻居状态机改为 Exchange。RT1 使用 RT2 的序列号 Seq=y 来发送新的 DD 报文，该报文开始正式地传送 LSA 的摘要，通过这种方式确认数据库中哪些 LSA 是需要更新的。在报文中 RT1 将 MS=0，说明自己是 Slave。

RT2 收到报文后，将 RT1 的邻居状态机改为 Exchange，并发送新的 DD 报文来描述自己的 LSA 摘要，此时 RT2 将报文的序列号改为 Seq=y+1。

上述过程持续进行，RT1 通过重复 RT2 的序列号来确认已收到 RT2 的报文。RT2 通过将序列号 Seq 加 1 来确认已收到 RouterA 的报文。当发送最后一个 DD 报文时，在报文中写上 M=0。

（3）LSDB 同步（LSA 请求、LSA 传输、LSA 应答

RT1 收到最后一个 DD 报文后，发现 RT2 的数据库中存在自己没有的 LSA，将邻居状态机改为 Loading 状态。

RT2 发现 RT1 的 LSA，RT2 都已经有了，不需要再请求，将 RT1 的邻居状态机改为 Full 状态。

RT1 发送 LSR 报文向 RT2 请求更新 LSA。RT2 用 LSU 报文来回应 RT1 的请求。RT1 收到后，发送 LSAck 报文确认。

上述过程持续到 RT1 中的 LSA 与 RT2 的 LSA 完全同步为止，此时 RT1 将 RT2 的邻居状态机改为 Full 状态。当路由器交换完 DD 报文并更新所有的 LSA 后，此时邻接关系建立完成。

OSPF 路由计算

OSPF 路由的计算过程可简单描述如下：

每台 OSPF 路由器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 路由器。

每台 OSPF 路由器都会收集其它路由器发来的 LSA，所有的 LSA 形成链路状态数据库 LSDB（Link State Database），LSDB 是对整个自治系统的网络拓扑结构的描述。

OSPF 路由器将 LSDB 转换成一张带权的有向图，这张图是对整个网络拓扑结构的真实反映。各 OSPF 路由器得到的有向图是完全相同的。

每台 OSPF 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 骨干区域

OSPF 划分区域之后，并非所有的区域都是平等的关系。其中有一个区域与众不同，通常被称为骨干区域，它的区域号（Area ID）是 0。

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。对此，OSPF 有以下规定：

所有非骨干区域必须与骨干区域保持连通。骨干区域自身也必须保持连通。但在实际应用中，可能会因为网络拓扑等限制，无法满足以上要求；这时可以通过配置 OSPF 虚连接满足要求。

OSPF 虚连接

虚连接指在两台 ABR 之间通过一个非骨干区域而建立的一条逻辑上的连接通道。虚连接相当于在两个 ABR 之间形成了一个点到点的连接。为虚连接两端提供一条非骨干区域内部路由的区域称为中转区域（TransitArea）。

虚连接有如下特点：

- 虚连接的两端必须是 ABR。
- 必须在两端同时配置虚连接，虚连接方能生效。
- 虚连接和物理接口一样可以配置接口参数，如发送 HELLO 报文间隔等。

两台 ABR 之间直接传递 OSPF 报文信息时，他们之间的 OSPF 路由器只起到转发报文的作用。由于协议报文的目的地址不是这些路由器，所以这些报文对于他们而言是透明的，只是当作普通的 IP 报文来转发。

OSPFStub 区域

Stub 区域的特点：

- Stub 区域的 ABR 不传播它们接收到的自治系统外部路由，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会大大减少。
- Stub 区域是一种可选的配置属性，并不是每个区域都符合配置的条件。通常来说，Stub 区域是位于自治系统边界，只有一个 ABR 的非骨干区域。
- 为保证到自治系统外的路由依旧可达，Stub 区域的 ABR 将生成一条缺省路由，并发布给 Stub 区域中的其他非 ABR 路由器。

配置 Stub 区域的注意事项：

- 骨干区域不能配置成 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必须都要配置 Stub 区域。
- Stub 区域内不能存在 ASBR，即自治系统外部的路由不能在本区域内传播。虚连接不能穿过 Stub 区域。

OSPFNSSA 区域

在 RFC1587 NSSA Option 中增加一类新的区域：NSSA 区域；同时增加一类新的 LSA：NSSA LSA（或称为 Type7 LSA）。

NSSA 区域其实是 Stub 区域的一个变形，它和 Stub 区域有许多相似的地方。两者的差别在于，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，同时又不会学习来自 OSPF 网络其它区域的外部路由。

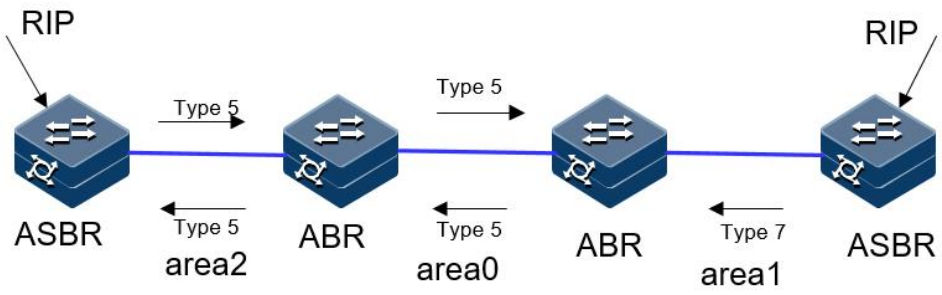
NSSA 区域的特点:

与 Stub 区域类似，NSSA 区域也不能配置虚连接。

与 Stub 区域类似，NSSA 区域也不允许 AS-External-LSA（即 Type5 LSA 注入，但可以允许 Type7 LSA 注入）。Type7 LSA 由 NSSA 区域的 ASBR 产生，在 NSSA 区域内传播。当 Type7 LSA 到达 NSSA 的 ABR 时，由 ABR 将 Type7 LSA 转换成 AS-ExternalLSA，传播到其他区域。

如图所示，运行 OSPF 协议的自治系统包括 3 个区域：区域 1、区域 2 和区域 0

图 3-25 OSPFNSSA 区域示意图



区域 1 被定义为 NSSA 区域。与区域 1、区域 2 相连的非 OSPF 网络运行 RIP 协议。区域 1 从 RIP 网络接收的 RIP 路由传播到 NSSA ASBR 后，由 NSSA ASBR 产生 Type7 LSA 在区域 1 内传播；当 Type7 LSA 到达 NSSA ABR 后，转换成 Type5 LSA 传播到区域 0 和区域 2。另一方面，区域 2 从 RIP 网络中接收的 RIP 路由通过区域 2 的 ASBR 产生 Type-5 LSA 在 OSPF 自治系统中传播。但由于区域 1 是 NSSA 区域，所以 Type-5LSA 不会到达区域 1。

OSPF 路由聚合

路由聚合是指：ABR 将具有相同前缀的路由信息聚合在一起后，形成一条路由发布到其它区域。

AS 被划分成不同的区域后，区域间可以通过路由聚合来减少路由信息，减小路由表的规模，提高路由器的运算速度。

OSPF 有两种路由聚合方式：

- ABR 聚合

ABR 向其它区域发送路由信息时，以网段为单位生成 Type3 LSA。如果该区域中存在一些连续的网段，则可以通过路由聚合的命令将这些连续的网段聚合成一个网段。这样 ABR 只发送一条聚合后的 LSA，所有属于命令指定的聚合网段范围的 LSA 将不会再被单独发送出去。

例如，区域 1 内有三条区域内路由 19.1.1.0/24，19.1.2.0/24，19.1.3.0/24，如果此时在 ABR 上配置了路由聚合，将三条路由聚合成一条 19.1.0.0/16，则 ABR 就只生成一条聚合后的 LSA，并发布给其他区域的路由器。

- ASBR 聚合

配置路由聚合后，如果本地设备是自治系统边界路由器 ASBR，将对引入的聚合地址范围内的 Type5 LSA 进行聚合。当配置了 NSSA 区域时，还要对引入的聚合地址范围内的 Type7 LSA 进行聚合。如果本地设备既是 ASBR 又是 ABR，则对由 Type7LSA 转化成的 Type5 LSA 进行聚合处理。

OSPF 路由类型

OSPF 将路由分为 4 级，按优先顺序分别是：区域内路由（Intra Area）；区域间路由（InterArea）；第一类外部路由（Type1 External）；第二类外部路由（Type2 External）。

AS 内部路由：AS 区域内和区域间路由描述的是 AS 内部的网络结构。缺省情况下，这两种路由的协议优先级为 10。

AS 外部路由：外部路由则描述了应该如何选择到 AS 以外目的地址的路由。OSPF 将引入的 AS 外部路由分为两类：Type1 和 Type2。缺省情况下，这两种路由的协议优先级为 150。

第一类外部路由：指接收的是 IGP 路由（例如静态路由和 RIP 路由）。由于这类路由的可信程度比较高，所以计算出的外部路由的开销与自治系统内部的路由开销是相同的，并且和 OSPF 自身路由的开销具有可比性；即到第一类外部路由的开销等于本路由器到相应的 ASBR 的开销+ASBR 到该路由目的地址的开销。

第二类外部路由：指接收的是 EGP 路由。由于这类路由的可信度比较低，所以 OSPF 协议认为从 ASBR 到自治系统之外的开销远远大于在自治系统之内到达 ASBR 的开销；所以计算路由开销时将主要考虑前者，即到第二类外部路由的开销=ASBR 到该路由目的地址的开销。如果两条路由计算出的开销值相等，再考虑本路由器到相应的 ASBR 的开销。

OSPF 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：OSPF 缺省认为网络类型是 Broadcast。在该类型的网络中，通常以组播形式（224.0.0.5 和 224.0.0.6）发送协议报文。
- NBMA（Non-Broadcast Multi-Access）类型：在该类型的网络中，以单播形式发送协议报文。
- 点到多点 P2MP（point-to-multipoint）类型：点到多点必须是由其他的网络类型强制更改的。常用做法是将非全连通的 NBMA 改为点到多点的网络。在该类型的网络中，以组播形式（224.0.0.5）发送协议报文。
- 点到点 P2P（point-to-point）类型：在该类型的网络中，以组播形式（224.0.0.5）发送协议报文。

OSPFDR 和 BDR

在广播网和 NBMA 网络中，任意两台路由器之间都要传递路由信息。如果网络中有 n 台路由器，则需要建立 nx(n-1)/2 个邻接关系。这使得任何一台路由器的路由变化都会导致多次传递，浪费了带宽资源。

为解决这一问题，OSPF 协议定义了 DR（Designated Router）、BDR（Backup Designated Router）和除 DR 和 BDR 之外的路由器（DROther）。

- DR：所有路由器都只将信息发送给 DR，由 DR 将网络链路状态广播出去。
- BDR：如果 DR 由于某种故障而失效，则网络中的路由器必须重新选举 DR，并与新的 DR 同步。这需要较长的时间，在这段时间内，路由的计算是不正确的。为了能够缩短这个过程，OSPF 提出了 BDR（BackupDesignatedRouter）的概念。BDR 实际上是对 DR 的一个备份，在选举 DR 的同时也选举出 BDR，BDR 也和本网段内的所有路由器建立邻接关系并交换路由信息。当 DR 失效后，BDR 会立即成为 DR。由于不需要重新选举，并且邻接关系事先已建立，所以这个过程是非常短暂的。当然这时还需要再重新选举出一个新的 BDR，虽然一样需要较长的时间，但并不会影响路由的计算。
- DROther：除 DR 和 BDR 之外的路由器（DROther）之间将不再建立邻接关系，也不再交换任何路由信息。这样就减少了广播网和 NBMA 网络上各路由器之间邻接关系的数量。

DR/BDR 选举过程：

- DR 和 BDR 不是人为指定的，而是由本网段中所有的路由器共同选举出来的。路由器接口的 DR 优先级决定了该接口在选举 DR、BDR 时所具有的资格。本网段内 DR 优先级大于 0 的路由器都可作为“候选人”。选举过程如下：
- 每台路由器将自己选出的 DR 写入 Hello 报文中，发给网段上的每台路由器。
- 如果处于同一网段的两台路由器同时宣布自己是 DR，DR 优先级高者胜出。如果优先级相等，则 RouterID 大者胜出。如果一台路由器的优先级为 0，则它不会被选举为 DR 或 BDR。

DR/BDR 选举特点：

- 只有在广播或 NBMA 类型接口时才会选举 DR，在点到点或点到多点类型的接口上不需要选举 DR。
- DR 是指某个网段中概念，是针对路由器的接口而言的。某台路由器在一个接口上可能是 DR，在另一个接口上有可能是 BDR，或者是 DROther。
- 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。
- DR 不一定就是 DR 优先级最大的路由器；同理，BDR 也不一定就是 DR 优先级第二大的路由器。

OSPF 报文认证

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。

路由器支持两种验证方式：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。

3.8.2 配置准备

场景

完成配置 OSPF 的基本功能后，即可以实现通过 OSPF 协议构建三层网络

前提

配置接口的 IP 地址，使各相邻节点网络层可达。

3.8.3 缺省配置

设备上 OSPF 的缺省配置如下。

功能	缺省值
OSPF 特性	不使能
Hello 报文发送间隔	10s
邻居失效时间	40s
计算接口开销的带宽参考值	100Mbit/s

3.8.4 配置 OSPF 基本功能

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)# **ospf** *process-id* [**vpn-instance** *NAME*]	启动 OSPF 进程，进入 OSPF 视图 process-id 缺省值为 1 vpn-instance 表示 vpn 实例，若指定则此 OSPF 进程属于 VPN 实例，若未指定则属于公网实例。
3	JX(config-ospf-1)#**router-id** *ipv4-address*	配置 OSPF 进程的 ID 号，需要保证 router id 全网唯一，不配置的情况下，系统会从当前接口的 IP 地址中选择一个作为 router id。

步骤	配置	说明
4	（1）在实例模式配置 JX(config-ospf-1)#**network** *network-address network-mask* **area** *area-id* （2）在接口下配置 JX (config-vlanif-1)# **ospf** *process-id* **area** *area-id*	接口使能 OSPF 并创建区域，有两种方式 （1）配置 OSPF 区域以及区域所包含的网段 接口的 IP 地址满足网段要求时，接口上便可正常运行 OSPF 协议。 （2）配置接口所运行的实例和区域。 两者配置矛盾时，以接口配置为主。
5	JX(config)# **interface** *interface-type interface-number*	进入接口模式，下面以 vlan 口 为例。
6	JX (config-vlanif-1)# **ospf if-type** { **broadcast** | **nbma** | **p2p** | **p2multip** }	配置接口的网络类型。（可选）
7	JX(config)# **ospf** *process-id* [**vpn-instance** *NAME*]	进入 OSPF 视图
8	JX(config-ospf-1)# **peer** *ipv4-address* **priority** *priority*	配置 NBMA 网络的邻居（可选）
9	JX(config-ospf-1)# **maximum load-balancing** *number*	配置 OSPF 最大等价路由条数（可选）
10	JX(config-ospf-1)# **rfc1583 compatible** { **enable** | **disable** }	配置兼容 RFC1583 的外部路由选择规则（可选）
11	JX(config-ospf-1)# **preference** *preference-value* { **route-policy** *NAME* }	配置 OSPF 域内路由和域间路由的优先级
12	JX(config-ospf-1)# **preference ase** *preference-value* { **route-policy** *NAME* }	配置 OSPF 外部路由的优先级

3.8.5 配置 OSPF STUB 区域

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)# **ospf** *process-id* [**vpn-instance** *NAME*]	进入 OSPF 视图。

步骤	配置	说明
3	JX(config-ospf-1)# **area** *area-id* **stub** [**no-summary-lsa**]	配置区域为 stub 区域，区域不能为骨干区域。 **no-summary** 表示禁止 ABR 向 Stub 区域内发送 Summary LSA。只有在 ABR 上配置 **stub** 命令时，可选参数 **no-summary** 才能对该区域起作用。

3.8.6 配置 OSPF NSSA 区域

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)# **ospf** *process-id* [**vpn-instance** *NAME*]	进入 OSPF 视图。
3	JX(config-ospf-1)# **area** *area-id* **nssa** [**no-summary-lsa**]	配置区域为 stub 区域，区域不能为骨干区域。 **no-summary** 表示禁止 ABR 向 nssa 区域内发送 Summary LSA。
4	JX(config-ospf-1)# **area** *area-id* **nssa default-cost** *cost*	配置 ABR 发送到 NSSA 区域的 Type3 LSA 的缺省路由的开销(可选)。
5	JX(config-ospf-1)# **area** *area-id* **nssa translator** { **always** | **candidate** }	candidate 状态时，NSSA 区域会根据规则自动选择一个 ABR 作为转换器，将 Type7 LSA 转换为 Type5 LSA。通过在 ABR 上配置 translator always，可以将某一个 ABR 指定为转换器。

3.8.7 配置 OSPF 引入外部路由

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**ospf** *process-id* [**vpn-instance** *NAME*]	进入 OSPF 视图。

步骤	配置	说明
3	JX(config-ospf-1)#**import-route** { **connect** | **static** | **rip** | **bgp** | **isis** | **ospf** } [**cost** { **cost** | **inherit** }|**route-policy** *NAME*]*	引入其它协议的路由信息。 cost 参数为 OSPF 引入的外部路由的缺省度量值，inherit 表示引入路由的开销值为路由自带的 cost 值。 如果没有指定开销值，则使用 default route-attribute 命令设置的缺省开销值。route-policy 参数用于绑定路由策略，在路由引入时进行过滤。

3.8.8 配置 OSPF 路由聚合

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)# **ospf** *process-id* [**vpn-instance** *NAME*]	进入 OSPF 视图。
3	JX(config-ospf-1)# **area** *area-id* **abr-summary** {*dest-address/dest-masklen* | *dest-address/dest-mask*}	配置 OSPF 的 ABR 路由聚合
4	JX(config-ospf-1)# **area** *area-id* **nssa summary** {*dest-address/dest-masklen* | *dest-address/dest-mask*}	配置在本路由器进行 type-7LSA 到 type-5LSA 转换时的汇聚功能。
5	JX(config-ospf-1)# **asbr-summary** *dest-address/dest-masklen*	配置 OSPF 的 ASBR 路由聚合。
6	JX(config-ospf-1)# **asbr-summary** { **connect** | **static** | **rip** | **bgp** | **isis** | **ospf** } *dest-address/dest-masklen*	配置 OSPF 针对某个协议引入的路由进行 ASBR 路由聚合
7	JX(config-ospf-1)# **asbr-summary** { **rip** | **isis** | **ospf** } *process-id dest-address/dest-masklen*	配置 OSPF 针对某个协议某个实例引入的路由进行 ASBR 路由聚合

3.8.9 配置 OSPF 虚连接

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。

步骤	配置	说明
2	JX(config)# **ospf** *process-id* [**vpn-instance** *NAME*]	进入 OSPF 视图。
3	JX(config-ospf-1)# **area** *area-id* **virtual-link** *ipv4-address*	创建并配置虚连接。
4	JX(config-ospf-1)# **area** *area-id* **virtual-link** *ipv4-address* **hello-interval** *value*	配置虚连接的 hello 包发送间隔（可选）。
5	JX(config-ospf-1)# **area** *area-id* **virtual-link** *ipv4-address* **dead-interval** *value*	配置虚连接的邻居失效时间（可选）。
6	JX(config-ospf-1)# **area** *area-id* **virtual-link** *ipv4-address* **retransmit-interval** *value*	配置虚连接的 LSA 重传时间间隔（可选）。
7	JX(config-ospf-1)# **area** *area-id* **virtual-link** *ipv4-address* **transmit-delay** *value*	配置虚连接延迟发送 LSA 的时间间隔（可选）。

3.8.10 配置 OSPF 认证功能

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)# **interface** *interface-type interface-number*	进入接口模式
3	JX (config-vlanif-1)# **ospf authentication md5** *key-id* { **cipher** | **plain** } *KEY*	配置 OSPF 的接口认证，模式为 MD5 验证。
4	JX (config-vlanif-1)# **ospf authentication simple** { **cipher** | **plain** } *KEY*	配置 OSPF 的接口认证，模式为简单验证。
5	JX(config)# **ospf** *process-id* { **vpn-instance** *NAME* }	进入 OSPF 视图。
6	JX(config-ospf-1)# **area** *area-id* **authentication md5** *key-id* { **cipher** | **plain** } *KEY*	配置 OSPF 的区域认证，模式为 MD5 验证。
7	JX(config-ospf-1)# **area** *area-id* **authentication simple** { **cipher** | **plain** } *KEY*	配置 OSPF 的区域认证，模式为简单验证。

步骤	配置	说明
8	JX(config-ospf-1)# **area** *area-id* **virtual-link** *ipv4-address* **authentication md5** *key-id* { **cipher** | **plain** } *KEY*	配置 OSPF 虚连接报文认证，模式为 MD5 验证。
9	JX(config-ospf-1)# **area** *area-id* **virtual-link** *ipv4-address* **authentication simple** { **cipher** | **plain** } *KEY*	配置 OSPF 虚连接报文认证，模式为简单验证。

3.8.11 检查配置

配置完成后，请在设备上执行以下命令检查配置结果。

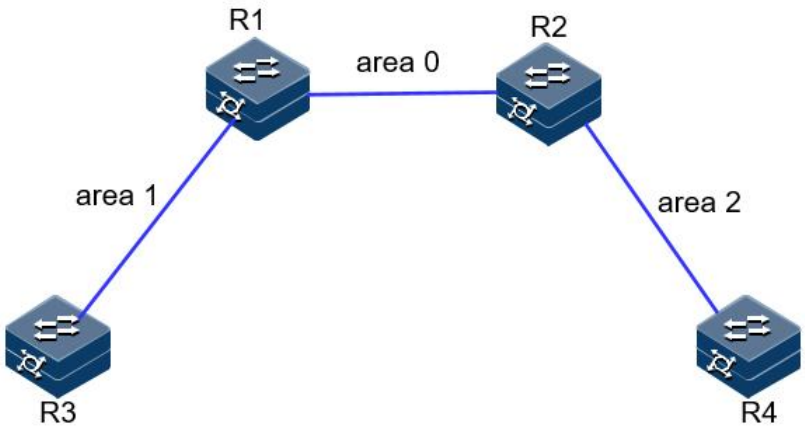
步骤	配置	说明
1	JX#**show ospf interface**	查看 OSPF 接口。
2	JX# **show ospf process** *process-id*	查看 OSPF 实例。
3	JX# **show ospf area**	查看 OSPF 区域。
4	JX# **show ospf neighbor**	查看 OSPF 邻居，对端进行对应的配置之后，可以建立邻居。
5	JX# **show ospf route**	查看 OSPF 路由表。
6	JX# **show ospf database**	查看 OSPF 数据库。
7	JX# **show ospf import-route**	查看 OSPF 引入的路由信息。
8	JX# **show ospf virtual-link**	查看 OSPF 虚连接信息。
9	JX#**show ospf interface** *interface-type interface-number*	查看接口的具体信息。
10	JX# **show ospf area area-id** *area-id*	查看区域的具体信息。
11	JX# **show ospf information**	查看 OSPF 的全局信息。

3.8.12 配置 OSPF 基本功能示例

组网需求

如下图所示，所有的 router 都运行 OSPF，并将整个自治系统划分为 3 个区域，其中 Router 1 和 router 2 为 ABR 来转发区域之间的路由。配置完成后，每台 router 都应学到自治系统内的到所有网段的路由。

图 3-26 OSPF 基本配置组网图



配置步骤

步骤 1 配置各接口的 IP 地址。

R1 的两个接口地址：1.1.1.1/24 和 3.1.1.1/24

R2 的两个接口地址：1.1.1.2/24 和 4.1.1.2/24

R3 的一个接口地址：3.1.1.3/24

R4 的一个接口地址：4.1.1.4/24

步骤 2 配置 OSPF。

R1：

```
JX-R1#configure
JX-R1(config)#ospf 1
JX-R1(config-ospf-1)#router-id 1.1.1.1
JX-R1(config-ospf-1)#network 1.1.1.0 255.255.255.0 area 0
JX-R1(config-ospf-1)#network 3.1.1.0 255.255.255.0 area 1
JX-R1(config-ospf-1)#
```

R2：

```
JX-R2#configure
JX-R2(config)#ospf 1
JX-R2(config-ospf-1)#router-id 2.2.2.2
JX-R2(config-ospf-1)#network 1.1.1.0 255.255.255.0 area 0
JX-R2(config-ospf-1)#network 4.1.1.0 255.255.255.0 area 2
JX-R2(config-ospf-1)#
```

R3：

```
JX-R3#configure
JX-R3(config)#ospf 1
JX-R3(config-ospf-1)#router-id 3.3.3.3
JX-R3(config-ospf-1)#network 3.1.1.0 255.255.255.0 area 1
JX-R3(config-ospf-1)#
```

R4：

```
JX-R4#configure
```

```
JX-R4(config)#ospf 1
JX-R4(config-ospf-1)#router-id 4.4.4.4
JX-R4(config-ospf-1)#network 4.1.1.0 255.255.255.0 area 2
JX-R4(config-ospf-1)#
```

步骤 3 验证配置结果。

使用 **show ospf neighbor** 查看邻居建立结果：

R1:

```
JX-R1(config)#show ospf neighbor

 OSPF Process 1
 IpAddress  NeighborID  Priority  State   Aging   UpTime        Interface
-------------------------------------------------------------------------------
---------------------------
 1.1.1.2    2.2.2.2      1     full       38     0:00:32       vlan-10
 3.1.1.3    3.3.3.3      1     full       35     0:05:04       vlan-13
-------------------------------------------------------------------------------
---------------------------
  Down:0   Attempt:0  Init:0   TwoWay:0   ExchangeStart:0   Exchange:0
  Loading:0    Full:2
```

R2：

```
JX-R2(config)#show ospf neighbor
OSPF Process 1
 IpAddress  NeighborID  Priority  State   Aging    UpTime     Interface
-------------------------------------------------------------------------------
---------------------------
 1.1.1.1     1.1.1.1       1      full     39     0:01:41     vlan-10
 4.1.1.4     4.4.4.4       1      full     39     0:01:40     vlan-24
-------------------------------------------------------------------------------
---------------------------
Down:0   Attempt:0  Init:0   TwoWay:0   ExchangeStart:0   Exchange:0
Loading:0    Full:2
```

通过 **show ospf database** 查看数据库

```
JX-R1(config)#show ospf database
Database of OSPF Process 1

                    Router LSA (area 0)
 LinkId          ADV Router       Age     Seq#        CheckSum    Len
 1.1.1.1         1.1.1.1          291     0x80000002   0xe05f      36
 2.2.2.2         2.2.2.2          292     0x80000002   0xa294      36

                    Network LSA (area 0)
 LinkId          ADV Router       Age     Seq#        CheckSum    Len
 1.1.1.2         2.2.2.2          292     0x80000001   0x83bd      32

                    SummaryNetwork LSA (area 0)
 LinkId          ADV Router       Age     Seq#        CheckSum    Len
```

```
3.1.1.0         1.1.1.1         563     0x80000002   0x173b      28
4.1.1.0         2.2.2.2         291     0x80000002   0xeb61      28

                    Router LSA (area 1)
LinkId          ADV Router      Age     Seq#         CheckSum    Len
1.1.1.1         1.1.1.1         563     0x80000002   0xfa42      36
3.3.3.3         3.3.3.3         563     0x80000002   0x7bb0      36

                    Network LSA (area 1)
LinkId          ADV Router      Age     Seq#         CheckSum    Len
3.1.1.1         1.1.1.1         563     0x80000001   0xd36c      32

                    SummaryNetwork LSA (area 1)
LinkId          ADV Router      Age     Seq#         CheckSum    Len
1.1.1.0         1.1.1.1         292     0x80000002   0x3123      28
4.1.1.0         1.1.1.1         290     0x80000001   0x163b      28
```

通过 **show ospf route** 查看路由表

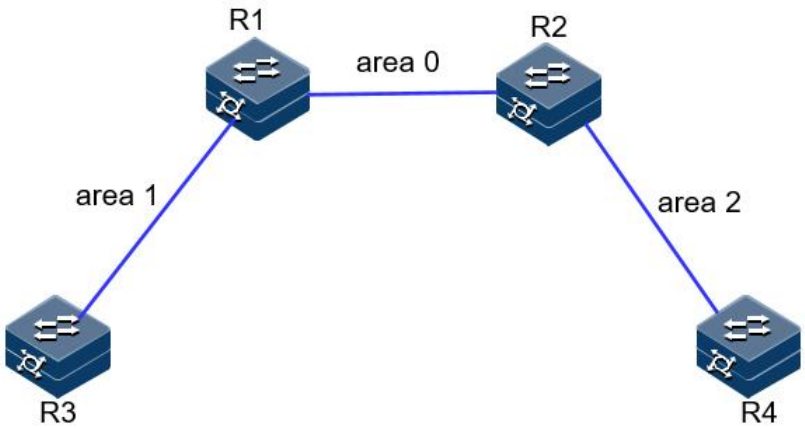
```
JX-R1(config)#show ospf route
 OSPF Process 1
 RoutType  Prefix            PathType  Cost    Cost2      NextHop
BackupNextHop    AreaId  Time
 ABR      2.2.2.2/32         INTRA     1       0          1.1.1.2
0.0.0.0          0      0:06:38
 Network   1.1.1.0/24         INTRA     1       0          1.1.1.1
0.0.0.0          N/A    0:06:40
 Network   3.1.1.0/24         INTRA     1       0          3.1.1.1
0.0.0.0          N/A    0:11:12
 Network   4.1.1.0/24         INTER     2       0          1.1.1.2
0.0.0.0          N/A    0:06:38
 Route :
 ABR      ASBR     Network   Intra     Inter     External
 1       0        3         2         1         0

 Path :
 ABR      ASBR     Network   Intra     Inter     External
 1       0        3         2         1         0
```

3.8.13 配置 OSPF STUB 区域示例

组网需求

如下图所示，所有的 router 都运行 OSPF，并将整个自治系统划分为 3 个区域，其中 R1 和 R2 为 ABR 来转发区域之间的路由。

图 3-27 OSPF STUB 区域示例组网图



配置步骤

步骤 1 配置同 OSPF 基本功能配置。

步骤 2 配置 area 1 为 stub 区域。

R1：

```
JX-R1#configure
JX-R1(config)#ospf 1
JX-R1(config-ospf-1)#area 1 stub
```

R3：

```
JX-R3#configure
JX-R3(config)#ospf 1
JX-R3(config-ospf-1)#area 1 stub
```

步骤 3 配置 R4 引入 100.1.1.1 的五类 LSA

```
JX-R4(config)#ip route-static 100.1.1.1 255.255.255.255 4.1.1.5
JX-R4(config)#ospf 1
JX-R4(config-ospf-1)#import-route static
```

步骤 4 验证配置结果

当 R3 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。

```
JX-R3(config)#show ospf route
 OSPF Process 1
 RoutType  Prefix            PathType  Cost    Cost2      NextHop
BackupNextHop    AreaId  Time
 ABR      1.1.1.1/32         INTRA     1       0          3.1.1.1
0.0.0.0          1       0:00:05
 ASBR     4.4.4.4/32         INTER     3       0          3.1.1.1
0.0.0.0          0       0:00:00
 Network  1.1.1.0/24         INTER     2       0          3.1.1.1
0.0.0.0          N/A     0:00:05
 Network  3.1.1.0/24         INTRA     1       0          3.1.1.3
0.0.0.0          N/A     0:00:07
```

```
 Network   4.1.1.0/24        INTER    3      0         3.1.1.1
0.0.0.0         N/A    0:00:05
 Network   100.1.1.1/32      ASE2     3      1         3.1.1.1
0.0.0.0         N/A    0:00:05
 Route :
 ABR      ASBR     Network   Intra    Inter    External
 1        1        4         1        2        1

 Path :
 ABR      ASBR     Network   Intra    Inter    External
 1        1        4         1        2        1
```

当 R3 所在区域配置为 Stub 区域时，已经看不到 AS 外部的路由，取而代之的是一条通往区域外部的缺省路由。

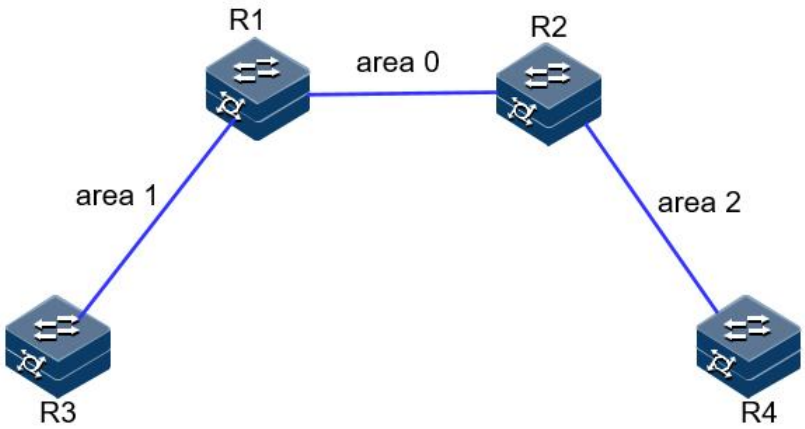
```
JX-R3(config)#show ospf route
 OSPF Process 1
 RoutType  Prefix           PathType  Cost    Cost2      NextHop
BackupNextHop    AreaId  Time
 ABR      1.1.1.1/32        INTRA    1      0         3.1.1.1
0.0.0.0         1      0:00:07
 Network   0.0.0.0/0         INTER    2      0         3.1.1.1
0.0.0.0         N/A    0:00:07
 Network   1.1.1.0/24        INTER    2      0         3.1.1.1
0.0.0.0         N/A    0:00:07
 Network   3.1.1.0/24        INTRA    1      0         3.1.1.3
0.0.0.0         N/A    0:00:12
 Network   4.1.1.0/24        INTER    3      0         3.1.1.1
0.0.0.0         N/A    0:00:07
 Route :
 ABR      ASBR     Network   Intra    Inter    External
 1        0        4         1        3        0

 Path :
 ABR      ASBR     Network   Intra    Inter    External
 1        0        4         1        3        0
```

3.8.14 配置 OSPF NSSA 区域示例

组网需求

如下图所示，所有的 router 都运行 OSPF，并将整个自治系统划分为 3 个区域，其中 R1 和 R2 为 ABR 来转发区域之间的路由。

图 3-28 OSPF NSSA 区域示例组网图



配置步骤

步骤 1 配置同 OSPF 基本功能配置。

步骤 2 配置 area 1 为 NSSA 区域。

R1：

```
JX-R1#configure
JX-R1(config)#ospf 1
JX-R1(config-ospf-1)#area 1 nssa
```

R3：

```
JX-R3#configure
JX-R3(config)#ospf 1
JX-R3(config-ospf-1)#area 1 nssa
```

步骤 3 配置 R4 引入 100.1.1.1 的五类 LSA

```
JX-R4(config)#ip route-static 100.1.1.1 255.255.255.255 4.1.1.5
JX-R4(config)#ospf 1
JX-R4(config-ospf-1)#import-route static
```

步骤 4 验证配置结果

nssa 区域的数据库比正常区域的数据库多一个缺省 NSSA 类型 LSA，R3 上没有 100.1.1.1 的外部 LSA

```
JX-R3(config)#show ospf database
Database of OSPF Process 1

                    Router LSA (area 1)
 LinkId          ADV Router       Age     Seq#        CheckSum     Len
 1.1.1.1         1.1.1.1          1763    0x8000001b  0x6aad       36
 3.3.3.3         3.3.3.3          1762    0x8000001b  0xe424       36

                    Network LSA (area 1)
 LinkId          ADV Router       Age     Seq#        CheckSum     Len
 3.1.1.3         3.3.3.3          1762    0x8000001a  0xb85c       32
```

```
                    SummaryNetwork LSA (area 1)
 LinkId          ADV Router      Age     Seq#       CheckSum    Len
 1.1.1.0         1.1.1.1         6       0x8000001b  0x86ac     28
 4.1.1.0         1.1.1.1         6       0x8000001b  0x69c5     28

                    NSSA LSA (area 1)
 LinkId          ADV Router      Age     Seq#       CheckSum    Len
 0.0.0.0         1.1.1.1         6       0x8000001b  0x9c17     36
```

步骤 5 配置 R3 引入 200.1.1.0 的静态路由

```
JX-R3(config)#ip route-static 200.1.1.0 255.255.255.0 3.1.1.1
JX-R3(config)#ospf 1
JX-R3(config-ospf-1)#import-route static
```

步骤 6 验证配置结果

R3 上存在 200.1.1.0 的五类 LSA 和 NSSA LSA

```
JX-R3(config)#show ospf database
Database of OSPF Process 1

                    Router LSA (area 1)
 LinkId          ADV Router      Age     Seq#       CheckSum    Len
 1.1.1.1         1.1.1.1         286     0x8000001c  0x68ae     36
 3.3.3.3         3.3.3.3         194     0x8000001d  0xe61e     36

                    Network LSA (area 1)
 LinkId          ADV Router      Age     Seq#       CheckSum    Len
 3.1.1.3         3.3.3.3         286     0x8000001b  0xb65d     32

                    SummaryNetwork LSA (area 1)
 LinkId          ADV Router      Age     Seq#       CheckSum    Len
 1.1.1.0         1.1.1.1         330     0x8000001b  0x86ac     28
 4.1.1.0         1.1.1.1         330     0x8000001b  0x69c5     28

                    NSSA LSA (area 1)
 LinkId          ADV Router      Age     Seq#       CheckSum    Len
 0.0.0.0         1.1.1.1         330     0x8000001b  0x9c17     36
 200.1.1.0        3.3.3.3         0       0x80000001  0x1bd1     36

                    ASExternal LSA
 LinkId          ADV Router      Age     Seq#       CheckSum    Len
 200.1.1.0        3.3.3.3         0       0x80000001  0x9163     36
```

查看 R4 的路由表和数据库，存在 200.1.1.0 路由和 LSA。

```
JX-R4(config)#show ospf route
 OSPF Process 1
 RoutType  Prefix           PathType  Cost   Cost2      NextHop
BackupNextHop    AreaId  Time
 ABR      2.2.2.2/32        INTRA    1      0         4.1.1.2
0.0.0.0         2      0:03:46
 ASBR     1.1.1.1/32        INTER    2      0         4.1.1.2
0.0.0.0         0      0:00:00
 ASBR     3.3.3.3/32        INTER    3      0         4.1.1.2
0.0.0.0         0      0:00:00
```

```
 Network   1.1.1.0/24        INTER    2      0         4.1.1.2
0.0.0.0         N/A    13:59:11
 Network   3.1.1.0/24        INTER    3      0         4.1.1.2
0.0.0.0         N/A    13:59:11
 Network   4.1.1.0/24        INTRA    1      0         4.1.1.4
0.0.0.0         N/A    13:59:13
 Network   200.1.1.0/24       ASE2     3      1         4.1.1.2
0.0.0.0         N/A    0:03:46
 Route :
 ABR      ASBR     Network   Intra    Inter    External
 1        2        4         1        2        1

 Path :
 ABR      ASBR     Network   Intra    Inter    External
 1        2        4         1        2        1

JX-R4(config)#show ospf database
Database of OSPF Process 1

                    Router LSA (area 2)
 LinkId          ADV Router       Age     Seq#        CheckSum    Len
 2.2.2.2         2.2.2.2          1782    0x8000001d  0xb65d      36
 4.4.4.4         4.4.4.4          1403    0x8000001e  0x3bc4      36

                    Network LSA (area 2)
 LinkId          ADV Router       Age     Seq#        CheckSum    Len
 4.1.1.4         4.4.4.4          1782    0x8000001c  0x4cc0      32

                    SummaryNetwork LSA (area 2)
 LinkId          ADV Router       Age     Seq#        CheckSum    Len
 1.1.1.0         2.2.2.2          1783    0x8000001e  0xda59      28
 3.1.1.0         2.2.2.2          1781    0x8000001c  0xce64      28

                    SummaryASBR LSA (area 2)
 LinkId          ADV Router       Age     Seq#        CheckSum    Len
 1.1.1.1         2.2.2.2          589     0x8000001b  0xc86c      28
 3.3.3.3         2.2.2.2          454     0x80000001  0xaa9b      28

                    ASExternal LSA
 LinkId          ADV Router       Age     Seq#        CheckSum    Len
 100.1.1.1        4.4.4.4         1400    0x8000001b  0x91a3      36
 200.1.1.0        1.1.1.1         262     0x80000001  0xcd2f      36
```

3.8.15 配置 OSPF 引入外部路由示例

组网需求

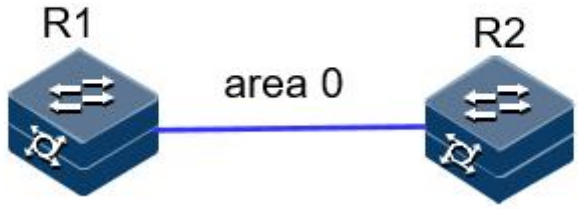
如下图所示， 2 个 router 都运行 OSPF，并将有个都配置为区域 0。假定 R1 需要向 OSPF 导入外部路由，R1 存在静态路由 100.1.1.0/24，本地路由 200.1.1.1/32 但是对外部路由有如下要求：

- 接受所有直连路由，并采用默认配置；

- 接收所有静态路由，并为路由配置开销 2000；

配置完成后，每台设备都应学到自治系统内的到所有网段的路由。。

图 3-29 OSPF 引入外部路由组网图



配置步骤

步骤 1 配置各接口的 IP 地址。

R1 的一个接口地址：1.1.1.1/24

R2 的一个接口地址：1.1.1.2/24

步骤 2 配置 OSPF。

R1：

```
JX-R1#configure
JX-R1(config)#ospf 1
JX-R1(config-ospf-1)#router-id 1.1.1.1
JX-R1(config-ospf-1)#network 1.1.1.0 255.255.255.0 area 0
JX-R1(config-ospf-1)#
```

R2：

```
JX-R2#configure
JX-R2(config)#ospf 1
JX-R2(config-ospf-1)#router-id 2.2.2.2
JX-R2(config-ospf-1)#network 1.1.1.0 255.255.255.0 area 0
JX-R2(config-ospf-1)#
```

步骤 3 配置重分配。

```
JX-R1#configure
JX-R1(config)#ospf 1
JX-R1(config-ospf-1)#import-route connect
JX-R1(config-ospf-1)#import-route static cost 2000
```

步骤 4 检查配置结果。

```
JX-R1(config)#show ospf import-route
  Import Route of OSPF(1)
 dest             mask             nexthop          cost    proto   pid
 1.1.1.0          255.255.255.0      1.1.1.1          1      connect 0
 100.1.1.0        255.255.255.0      1.1.1.3          60     static  0
 200.1.1.1        255.255.255.255    200.1.1.1        1      connect 0
```

```
JX-R1(config)#show ospf database ls-type as-external-lsa
Database of OSPF Process 1

                  ASExternal LSA
 LinkId          ADV Router       Age     Seq#        CheckSum    Len
 1.1.1.0         1.1.1.1          324     0x80000001  0xdce9      36
 100.1.1.0       1.1.1.1          21      0x80000001  0x4a40      36
 200.1.1.1       1.1.1.1          285     0x80000001  0x7b86      36
```

在 R2 查看路由表：

```
JX-R2(config)#show ospf route
 OSPF Process 1
 RoutType  Prefix            PathType  Cost   Cost2      NextHop
BackupNextHop    AreaId  Time
 ASBR     1.1.1.1/32         INTRA    1      0          1.1.1.1
0.0.0.0          0      0:00:38
 Network   1.1.1.0/24         INTRA    1      0          1.1.1.2
0.0.0.0          N/A    0:09:33
 Network   100.1.1.0/24       ASE2     1      2000       1.1.1.3
0.0.0.0          N/A    0:00:38
 Network   200.1.1.1/32       ASE2     1      1          1.1.1.1
0.0.0.0          N/A    0:05:02
 Route :
 ABR      ASBR     Network   Intra    Inter    External
 0        1        3         1        0        2

 Path :
 ABR      ASBR     Network   Intra    Inter    External
 0        1        3         1        0        2
```

3.8.16 配置 OSPF 路由聚合示例

组网需求

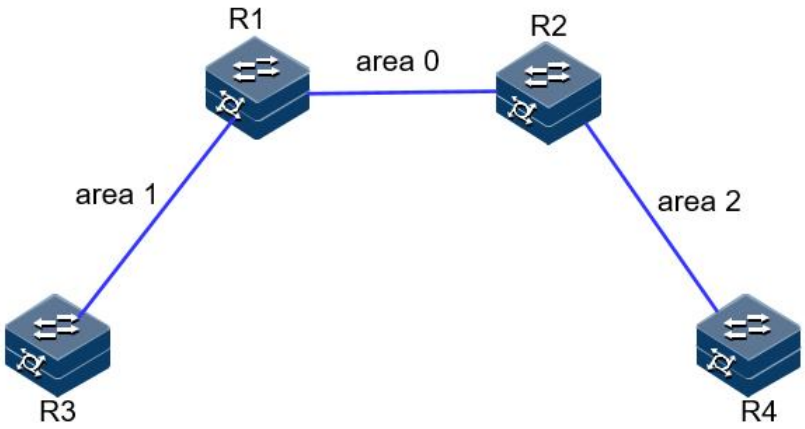
如下图所示，网络要求：区域 2 中存 10.1.1.1/32,10.1.2.1/32,20.1.1.1/32,20.1.2.1/32 的区域内路由，希望将 10.1.1.1/32 和 10.1.2.1/32 聚合为 10.1.0.0/16 通告；而希望 20.1.1.1/32 和 20.1.2.1/32 不导入其他区域。

R4 具有 30.1.1.1/32 和 30.1.2.1/32 的外部路由，希望将此路由通告给其他区域。

区域 1 的设备能力较差，不能接受大量外部路由，但是 R3 具有 40.1.1.1/32 和 40.1.1.2/32 的外部路由，希望将此路由通告给其他区域。

根据上述要求，我们可以为区域 2 配置聚合条目和过滤条目，为区域 1 配置 NSSA 属性。

图 3-30 OSPF 路由聚合组网图



配置步骤

步骤 1 配置同 OSPF 基本功能配置。

步骤 2 配置 area 1 为 nssa 区域。

```
R1：
JX-R1#configure
JX-R1(config)#ospf 1
JX-R1(config-ospf-1)#area 1 nssa

R3：
JX-R3#configure
JX-R3(config)#ospf 1
JX-R3(config-ospf-1)#area 1 nssa
```

步骤 3 查看聚合前的路由条目和数据库。

```
JX-R2(config)#show ospf route
 OSPF Process 1
 RoutType  Prefix            PathType  Cost    Cost2      NextHop
BackupNextHop    AreaId  Time
 ABR      1.1.1.1/32         INTRA     1       0          1.1.1.1
0.0.0.0          0       0:02:22
 ASBR     1.1.1.1/32         INTRA     1       0          1.1.1.1
0.0.0.0          0       0:02:22
 ASBR     3.3.3.3/32         INTER     2       0          1.1.1.1
0.0.0.0          0       0:02:22
 ASBR     4.4.4.4/32         INTRA     1       0          4.1.1.4
0.0.0.0          0       0:00:00
 Network  1.1.1.0/24         INTRA     1       0          1.1.1.2
0.0.0.0          N/A     14:59:45
 Network  3.1.1.0/24         INTER     2       0          1.1.1.1
0.0.0.0          N/A     14:59:44
 Network  4.1.1.0/24         INTRA     1       0          4.1.1.2
0.0.0.0          N/A     14:59:44
 Network  10.1.1.1/32        INTRA     2       0          4.1.1.4
0.0.0.0          N/A     0:05:20
```

```
 Network   10.1.2.1/32        INTRA    2      0         4.1.1.4
0.0.0.0         N/A    0:05:15
 Network   20.1.1.1/32        INTRA    2      0         4.1.1.4
0.0.0.0         N/A    0:05:07
 Network   20.1.2.1/32        INTRA    2      0         4.1.1.4
0.0.0.0         N/A    0:05:06
 Network   30.1.1.1/32        ASE2     1      1         4.1.1.3
0.0.0.0         N/A    0:04:07
 Network   30.1.2.1/32        ASE2     1      1         4.1.1.3
0.0.0.0         N/A    0:04:03
 Network   40.1.1.1/32        ASE2     2      1         1.1.1.1
0.0.0.0         N/A    0:02:24
 Network   40.1.1.2/32        ASE2     2      1         1.1.1.1
0.0.0.0         N/A    0:02:22
 Route :
 ABR      ASBR     Network   Intra    Inter    External
 1        3        11        6        1        4

 Path :
 ABR      ASBR     Network   Intra    Inter    External
 1        3        11        6        1        4
```

步骤 4 在 R2 配置 ABR 聚合，将 10.1.1.1/32 和 10.1.2.1/32 聚合为 10.1.0.0/16 通告，20.1.1.1/32 和 20.1.2.1/32 汇聚之后不通告。

```
JX-R2#configure
JX-R2(config)#ospf 1
JX-R2(config-ospf-1)#area 2 abr-summary 10.1.0.0/16 advertise
JX-R2(config-ospf-1)#area 2 abr-summary 20.1.0.0/16 no-advertise
```

步骤 5 验证配置结果

R1 的路由表中仅存在聚合后的 10.1.0.0/16，无 10.1.1.1/32、10.1.2.1/32、20.1.1.1/32 和 20.1.2.1/32。

```
JX-R1(config)#show ospf route
OSPF Process 1
 RoutType  Prefix            PathType  Cost    Cost2      NextHop
BackupNextHop    AreaId  Time
 ABR      2.2.2.2/32         INTRA    1      0         1.1.1.2
0.0.0.0         0      0:00:11
 ASBR     4.4.4.4/32         INTER    2      0         1.1.1.2
0.0.0.0         0      0:00:11
 ASBR     3.3.3.3/32         INTRA    1      0         3.1.1.3
0.0.0.0         0      0:00:00
 Network   1.1.1.0/24        INTRA    1      0         1.1.1.1
0.0.0.0         N/A    15:09:50
 Network   3.1.1.0/24        INTRA    1      0         3.1.1.1
0.0.0.0         N/A    14:19:11
 Network   4.1.1.0/24        INTER    2      0         1.1.1.2
0.0.0.0         N/A    15:09:48
 Network   10.1.0.0/16       INTER    3      0         1.1.1.2
0.0.0.0         N/A    0:06:36
```

```
 Network   30.1.1.1/32       ASE2     2      1         1.1.1.2
0.0.0.0         N/A    0:14:12
 Network   30.1.2.1/32       ASE2     2      1         1.1.1.2
0.0.0.0         N/A    0:14:08
 Network   40.1.1.1/32       ASE2     1      1         3.1.1.2
0.0.0.0         N/A    0:00:14
 Network   40.1.1.2/32       ASE2     1      1         3.1.1.2
0.0.0.0         N/A    0:00:11
 Route :
 ABR      ASBR     Network   Intra    Inter    External
 1        2        8         2        2        4

 Path :
 ABR      ASBR     Network   Intra    Inter    External
 1        2        8         2        2        4
```

步骤 6 在 R4 上配置 asbr 聚合，将 30.1.1.1/32 和 30.1.2.1/32 聚合为 30.1.0.0/16

```
JX-R4#configure
JX-R4(config)#ospf 1
JX-R4(config-ospf-1)#asbr-summary 30.1.0.0/16
```

步骤 7 验证配置结果

查看 R2 路由表和数据库，聚合成功

```
JX-R2(config)#show ospf route
 OSPF Process 1
 RoutType  Prefix           PathType  Cost   Cost2      NextHop
BackupNextHop    AreaId  Time
 ABR      1.1.1.1/32        INTRA    1      0         1.1.1.1
0.0.0.0         0      0:01:33
 ASBR     1.1.1.1/32        INTRA    1      0         1.1.1.1
0.0.0.0         0      0:01:33
 ASBR     3.3.3.3/32        INTER    2      0         1.1.1.1
0.0.0.0         0      0:01:33
 ASBR     4.4.4.4/32        INTRA    1      0         4.1.1.4
0.0.0.0         0      0:00:00
 Network   1.1.1.0/24        INTRA    1      0         1.1.1.2
0.0.0.0         N/A    15:14:02
 Network   3.1.1.0/24        INTER    2      0         1.1.1.1
0.0.0.0         N/A    15:14:01
 Network   4.1.1.0/24        INTRA    1      0         4.1.1.2
0.0.0.0         N/A    15:14:01
 Network   10.1.1.1/32       INTRA    2      0         4.1.1.4
0.0.0.0         N/A    0:19:37
 Network   10.1.2.1/32       INTRA    2      0         4.1.1.4
0.0.0.0         N/A    0:19:32
 Network   20.1.1.1/32       INTRA    2      0         4.1.1.4
0.0.0.0         N/A    0:19:24
 Network   20.1.2.1/32       INTRA    2      0         4.1.1.4
0.0.0.0         N/A    0:19:23
 Network   30.1.0.0/16       ASE2     1      1         4.1.1.4
0.0.0.0         N/A    0:01:34
 Network   40.1.1.1/32       ASE2     2      1         1.1.1.1
0.0.0.0         N/A    0:04:26
```

```
 Network   40.1.1.2/32        ASE2     2      1         1.1.1.1
0.0.0.0          N/A    0:04:23
 Route :
 ABR      ASBR     Network   Intra    Inter    External
 1        3        10        6        1        3

 Path :
 ABR      ASBR     Network   Intra    Inter    External
 1        3        10        6        1        3

JX-R2(config)#show ospf database ls-type as-external-lsa
Database of OSPF Process 1

                    ASExternal LSA
 LinkId          ADV Router      Age     Seq#       CheckSum    Len
 30.1.0.0        4.4.4.4         110     0x80000001  0xe0c1      36
 40.1.1.1        1.1.1.1         283     0x80000001  0xf9a1      36
 40.1.1.2        1.1.1.1         279     0x80000001  0xefaa      36
```

步骤 8 在 R1 上配置 nssa 聚合，将 R3 产生的 NSSA LSA 转换时进行聚合。

```
JX-R1#configure
JX-R1(config)#ospf 1
JX-R1(config-ospf-1)#area 1 nssa summary 40.1.1.0/24 advertise
```

步骤 9 检查配置结果

R1 生成的 external-lsa 进行了聚合

```
JX-R2(config)#show ospf route
 OSPF Process 1
 RoutType  Prefix            PathType  Cost   Cost2      NextHop
BackupNextHop    AreaId  Time
 ABR      1.1.1.1/32         INTRA     1      0          1.1.1.1
0.0.0.0          0      0:00:54
 ASBR     1.1.1.1/32         INTRA     1      0          1.1.1.1
0.0.0.0          0      0:00:54
 ASBR     3.3.3.3/32         INTER     2      0          1.1.1.1
0.0.0.0          0      0:00:54
 ASBR     4.4.4.4/32         INTRA     1      0          4.1.1.4
0.0.0.0          0      0:00:00
 Network   1.1.1.0/24        INTRA     1      0          1.1.1.2
0.0.0.0          N/A    15:18:01
 Network   3.1.1.0/24        INTER     2      0          1.1.1.1
0.0.0.0          N/A    15:18:00
 Network   4.1.1.0/24        INTRA     1      0          4.1.1.2
0.0.0.0          N/A    15:18:00
 Network   10.1.1.1/32       INTRA     2      0          4.1.1.4
0.0.0.0          N/A    0:23:36
 Network   10.1.2.1/32       INTRA     2      0          4.1.1.4
0.0.0.0          N/A    0:23:31
 Network   20.1.1.1/32       INTRA     2      0          4.1.1.4
0.0.0.0          N/A    0:23:23
 Network   20.1.2.1/32       INTRA     2      0          4.1.1.4
0.0.0.0          N/A    0:23:22
```

```
 Network   30.1.0.0/16        ASE2      1      1          4.1.1.4
0.0.0.0          N/A     0:05:33
 Network   40.1.1.0/24        ASE2      1      2          1.1.1.1
0.0.0.0          N/A     0:00:54
 Route :
 ABR      ASBR     Network   Intra     Inter     External
 1        3        9         6         1         2

 Path :
 ABR      ASBR     Network   Intra     Inter     External
 1        3        9         6         1         2
JX-R2(config)#show ospf database ls-type as-external-lsa
Database of OSPF Process 1

                    ASExternal LSA
 LinkId          ADV Router      Age      Seq#         CheckSum     Len
 30.1.0.0        4.4.4.4         337      0x80000001   0xe0c1       36
 40.1.1.0        1.1.1.1         58       0x80000001   0xb7ea       36
```

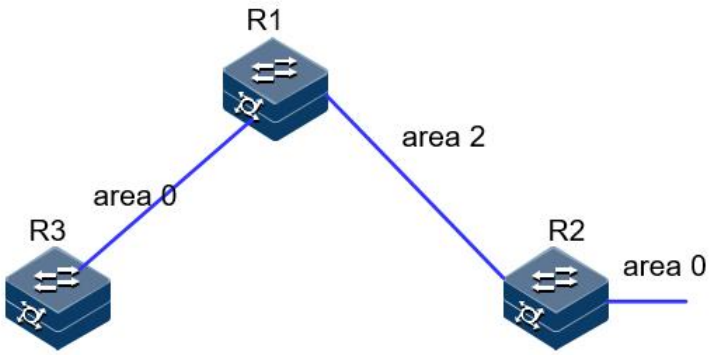
3.8.17 配置 OSPF 虚连接示例

组网需求

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。

如下图所示，R3 位于区域 0；R1 连接区域 0 和区域 2，而 R2 连接区域 0 和区域 2，此时区域 0 被分割，区域 0 内无法学习到区域 2 的内部路由；区域 2 也无法学习到区域 0 的内部路由和其他区域的路由。此时需要在 R1 和 R4 间配置虚链路，连接 0 域。

图 3-31 配置 OSPF 虚连接的组网图



配置步骤

步骤 1 配置各接口的 IP 地址。

R1 的两个接口地址：1.1.1.1/24 和 3.1.1.1/24

R2 的一个接口地址：1.1.1.2/24 和一个环回口地址 2.2.2.2/32

R3 的一个接口地址：3.1.1.3/24 和一个环回口地址 3.3.3.3/32

步骤 2 配置 OSPF。

R1：

```
JX-R1#configure
JX-R1(config)#ospf 1
JX-R1(config-ospf-1)#router-id 1.1.1.1
JX-R1(config-ospf-1)#network 1.1.1.0 255.255.255.0 area 2
JX-R1(config-ospf-1)#network 3.1.1.0 255.255.255.0 area 0
JX-R1(config-ospf-1)#
```

R2：

```
JX-R2#configure
JX-R2(config)#ospf 1
JX-R2(config-ospf-1)#router-id 2.2.2.2
JX-R2(config-ospf-1)#network 1.1.1.0 255.255.255.0 area 2
JX-R2(config-ospf-1)#network 2.2.2.2 255.255.255.255 area 0
JX-R2(config-ospf-1)#
```

R3：

```
JX-R3#configure
JX-R3(config)#ospf 1
JX-R3(config-ospf-1)#router-id 3.3.3.3
JX-R3(config-ospf-1)#network 3.1.1.0 255.255.255.0 area 0
JX-R3(config-ospf-1)#network 3.3.3.3 255.255.255.255 area 0
JX-R3(config-ospf-1)#
```

步骤 3 检查配置结果。

```
JX-R1#show ospf neighbor

 OSPF Process 1
 IpAddress         NeighborID        Priority  State         Aging
UpTime       Interface
------------------------------------------------------------------------------
---------------------------
 1.1.1.2           2.2.2.2           1       full          37     0:06:10
vlan-10
 3.1.1.3           3.3.3.3           1       full          30     0:04:57
vlan-13
------------------------------------------------------------------------------
---------------------------
  Down:0       Attempt:0       Init:0       TwoWay:0       ExchangeStart:0
Exchange:0
  Loading:0       Full:2
```

R2 上无法学习到 3.3.3.3 的路由，R1、R3 也无法学习到 2.2.2.2 的路由

```
JX-R2(config)#show ospf route
 OSPF Process 1
 RoutType  Prefix            PathType  Cost   Cost2      NextHop
BackupNextHop    AreaId  Time
```

```
 ABR      1.1.1.1/32         INTRA    1      0         1.1.1.1
0.0.0.0          2      0:07:41
 Network   1.1.1.0/24         INTRA    1      0         1.1.1.2
0.0.0.0          N/A    0:09:38
 Network   2.2.2.2/32         INTRA    1      0         2.2.2.2
0.0.0.0          N/A    0:09:20
 Route :
 ABR      ASBR     Network   Intra    Inter    External
 1        0        2         2        0        0

 Path :
 ABR      ASBR     Network   Intra    Inter    External
 1        0        2         2        0        0

JX-R1#show ospf route
 OSPF Process 1
 RoutType  Prefix            PathType  Cost    Cost2      NextHop
BackupNextHop    AreaId  Time
 ABR      2.2.2.2/32         INTRA    1      0         1.1.1.2
0.0.0.0          2      0:08:38
 Network   1.1.1.0/24         INTRA    1      0         1.1.1.1
0.0.0.0          N/A    0:09:55
 Network   3.1.1.0/24         INTRA    1      0         3.1.1.1
0.0.0.0          N/A    0:08:42
 Network   3.3.3.3/32         INTRA    2      0         3.1.1.3
0.0.0.0          N/A    0:08:38
 Route :
 ABR      ASBR     Network   Intra    Inter    External
 1        0        3         3        0        0

 Path :
 ABR      ASBR     Network   Intra    Inter    External
 1        0        3         3        0        0
```

步骤 4 配置虚连接。

R1：

```
JX-R1#configure
JX-R1(config)#ospf 1
JX-R1(config-ospf-1)#area 2 virtual-link 2.2.2.2
```

R2：

```
JX-R2#configure
JX-R2(config)#ospf 1
JX-R2(config-ospf-1)#area 2 virtual-link 1.1.1.1
```

步骤 5 检查配置结果。

虚连接邻居正常建立

```
JX-R1(config)#show ospf virtual-link
 OSPF Process 1
 Area Id                 :2
 Enable Status            :Enable
```

```
 State                    :pointToPoint
 Event Change              :3
 Hello Interval            :10
 Retransmit Interval        :5
 Transmit Delay            :1
 Dead Interval             :40
 Auth Type                 :0(None)
 Neighbor Address           :1.1.1.2
 Neighbor Router Id         :2.2.2.2
 Neighbor State            :full
 Neighbor Event Count       :5
 Neighbor Retransmit Queue Length:0
 Neighbor Options           :0x2
```

路由表学习正常

```
JX-R1(config)#show ospf route
 OSPF Process 1
 RoutType  Prefix            PathType  Cost   Cost2      NextHop
BackupNextHop    AreaId  Time
 ABR      2.2.2.2/32         INTRA    1      0         1.1.1.2
0.0.0.0          2      0:01:45
 Network   1.1.1.0/24         INTRA    1      0         1.1.1.1
0.0.0.0          N/A    0:13:44
 Network   2.2.2.2/32         INTRA    2      0         1.1.1.2
0.0.0.0          N/A    0:01:45
 Network   3.1.1.0/24         INTRA    1      0         3.1.1.1
0.0.0.0          N/A    0:12:31
 Network   3.3.3.3/32         INTRA    2      0         3.1.1.3
0.0.0.0          N/A    0:12:27
 Route :
 ABR      ASBR     Network   Intra    Inter    External
 2        0        4         4        0        0

 Path :
 ABR      ASBR     Network   Intra    Inter    External
 1        0        4         4        0        0

JX-R2(config)#show ospf route
 OSPF Process 1
 RoutType  Prefix            PathType  Cost   Cost2      NextHop
BackupNextHop    AreaId  Time
 ABR      1.1.1.1/32         INTRA    1      0         1.1.1.1
0.0.0.0          2      0:02:23
 Network   1.1.1.0/24         INTRA    1      0         1.1.1.2
0.0.0.0          N/A    0:14:58
 Network   2.2.2.2/32         INTRA    1      0         2.2.2.2
0.0.0.0          N/A    0:14:40
 Network   3.1.1.0/24         INTRA    2      0         1.1.1.1
0.0.0.0          N/A    0:02:23
 Network   3.3.3.3/32         INTRA    3      0         1.1.1.1
0.0.0.0          N/A    0:02:23
 Route :
 ABR      ASBR     Network   Intra    Inter    External
```

```
2       0       4       4       0       0

Path :
ABR     ASBR    Network  Intra   Inter   External
1       0       4       4       0       0
```

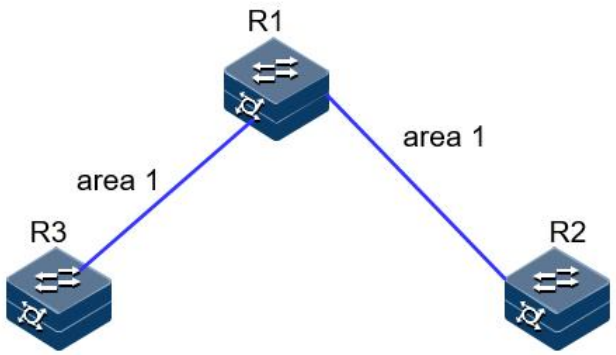
3.8.18 配置 OSPF 认证示例

组网需求

如下图所示，R1，R2,R3 均位于区域 0；R2 和 R3 之间建立虚连接，配置要求：

- R1 与 R2 间采用简单密码认证，密码为 test。
- R2 与 R3 建立虚链路，采用 MD5 认证，密码为 aaa,ID 为 100。
- R1 与 R3 采用 MD5 认证，密码为 ccc，ID 为 110。

图 3-32 配置 OSPF 虚连接的组网图



配置步骤

步骤 1 配置各接口的 IP 地址。

- R1 的两个接口地址：1.1.1.1/24 和 3.1.1.1/24
- R2 的一个接口地址：1.1.1.2/24
- R3 的一个接口地址：3.1.1.3/24

步骤 2 配置 OSPF。

R1：

```
JX-R1#configure
JX-R1(config)#ospf 1
JX-R1(config-ospf-1)#router-id 1.1.1.1
JX-R1(config-ospf-1)#network 1.1.1.0 255.255.255.0 area 1
JX-R1(config-ospf-1)#network 3.1.1.0 255.255.255.0 area 1
JX-R1(config-ospf-1)#
```

R2：

```
JX-R2#configure
JX-R2(config)#ospf 1
```

```
JX-R2(config-ospf-1)#router-id 2.2.2.2
JX-R2(config-ospf-1)#network 1.1.1.0 255.255.255.0 area 1
JX-R2(config-ospf-1)#area 1 virtual-link 3.3.3.3
JX-R2(config-ospf-1)#
```

R3：

```
JX-R3#configure
JX-R3(config)#ospf 1
JX-R3(config-ospf-1)#router-id 3.3.3.3
JX-R3(config-ospf-1)#network 3.1.1.0 255.255.255.0 area 1
JX-R3(config-ospf-1)#area 1 virtual-link 2.2.2.2
JX-R3(config-ospf-1)#
```

步骤 3 检查配置结果。

```
JX-R1#show ospf neighbor
 OSPF Process 1
 IpAddress         NeighborID        Priority  State         Aging
UpTime       Interface
------------------------------------------------------------------------------
---------------------------
 1.1.1.2           2.2.2.2           1       full          36     0:00:14
vlan-12
 3.1.1.3           3.3.3.3           1       full          38     0:01:12
vlan-13
------------------------------------------------------------------------------
---------------------------
  Down:0      Attempt:0      Init:0      TwoWay:0      ExchangeStart:0
Exchange:0
  Loading:0      Full:2

 JX-R2(config)#show ospf virtual-link
 OSPF Process 1
 Area Id                 :1
 Enable Status           :Enable
 State                   :pointToPoint
 Event Change            :3
 Hello Interval          :10
 Retransmit Interval     :5
 Transmit Delay          :1
 Dead Interval           :40
 Auth Type               :0(None)
 Neighbor Address        :3.1.1.3
 Neighbor Router Id      :3.3.3.3
 Neighbor State          :full
 Neighbor Event Count    :5
 Neighbor Retransmit Queue Length:0
 Neighbor Options        :0x2
```

步骤 4 配置认证，R1R2 和 R1R3 之间的认证进入接口配置。

R1：

```
JX-R1(config)#interface vlan 12
JX-R1(config-vlanif-12)#ospf authentication simple plain test
```

```
JX-R1(config-vlanif-12)#exit
JX-R1(config)#interface vlan 13
JX-R1(config-vlanif-13)#ospf authentication md5 110 plain ccc
JX-R1(config-vlanif-13)#exit
```

R2：

```
JX-R2(config)#interface vlan 12
JX-R2(config-vlanif-12)#ospf authentication simple plain test
JX-R2(config-vlanif-12)#exit
JX-R2(config)#ospf 1
JX-R2(config-ospf-1)#area 1 virtual-link 3.3.3.3 authentication md5 100
plain aaa
```

R3：

```
JX-R3(config)#interface vlan 13
JX-R3(config-vlanif-13)#ospf authentication md5 110 plain ccc
JX-R3(config-vlanif-13)#exit
JX-R3(config)#ospf 1
JX-R3(config-ospf-1)#area 1 virtual-link 2.2.2.2 authentication md5 100
plain aaa
```

步骤 5 检查配置结果。

邻居正常建立

```
JX-R1(config)#show ospf neighbor

 OSPF Process 1
 IpAddress         NeighborID        Priority  State         Aging
UpTime       Interface
------------------------------------------------------------------------
---------------------------
 1.1.1.2           2.2.2.2           1       full          39     0:00:02
vlan-12
 3.1.1.3           3.3.3.3           1       full          39     0:00:12
vlan-13
------------------------------------------------------------------------
---------------------------
  Down:0      Attempt:0      Init:0      TwoWay:0      ExchangeStart:0
Exchange:0
  Loading:0      Full:2

JX-R2(config)#show ospf virtual-link
 OSPF Process 1
 Area Id                 :1
 Enable Status            :Enable
 State                   :pointToPoint
 Event Change             :15
 Hello Interval           :10
 Retransmit Interval       :5
 Transmit Delay           :1
 Dead Interval            :40
 Auth Type               :2(MD5 Key)
 MD5 Key ID               :100
 MD5 Key                 :aaa
 Neighbor Address          :3.1.1.3
```

```
Neighbor Router Id         :3.3.3.3
Neighbor State            :full
Neighbor Event Count       :5
Neighbor Retransmit Queue Length:0
Neighbor Options          :0x2
```

3.9 OSPFV3

3.9.1 简介

功能

OSPFv3 是运行于 IPv6 的 OSPF 路由协议（RFC2740），它在 OSPFv2 基础上进行了增强，是一个独立的路由协议。OSPFv3 在 Hello 报文、状态机、LSDB、洪泛机制和路由计算等方面的工作原理和 OSPFv2 保持一致。OSPFv3 协议把自治系统划分成逻辑意义上的一个或多个区域，通过 LSA（Link State Advertisement）的形式发布路由。OSPFv3 依靠在 OSPFv3 区域内各设备间交互 OSPFv3 报文来达到路由信息的统一。OSPFv3 报文封装在 IPv6 报文内，可以采用单播和组播的形式发送。

OSPFV3 和 OSPFV2 的比较

OSPFv3 基于链路，而不是网段。IPv6 是基于链路而不是网段的。这样，在配置 OSPFv3 时，不需要考虑是否配置在同一网段，只要在同一链路，就可以不配置 IPv6 全局地址而直接建立联系。

OSPFv3 上移除了 IP 地址的意义。这样做的目的是为了使“拓扑与地址分离。OSPFv3 可以不依赖 IPv6 全局地址的配置来计算出 OSPFv3 的拓扑结构。

OSPFv3 的报文及 LSA 格式发生改变。OSPFv3 报文不包含 IP 地址；OSPFv3 的 Router LSA 和 Network LSA 里不包含 IP 地址。IP 地址部分由新增的两类 LSA（Link LSA 和 Intra Area Prefix LSA）宣告。

OSPFv3 利用 IPv6 链路本地地址，IPv6 使用链路本地（Link-local）地址在同一链路上发现邻居及自动配置等。OSPFv3 是运行在 IPv6 上的路由协议，同样使用链路本地地址来维持邻居，同步 LSA 数据库。

OSPFv3 移除所有认证字段，OSPFv3 的认证直接使用 IPv6 的认证及安全处理，不再需要其自身来完成认证，使用协议时只需关注协议本身即可。

新增两种 LSA, Link LSA：用于路由器宣告各个链路上对应的链路本地地址及其所配置的 IPv6 全局地址，仅在链路内洪泛。Intra Area Prefix LSA：用于向其他宣告本路由器或本网络（广播网及 NBMA）的 IPv6 全局地址信息，在区域内洪泛。

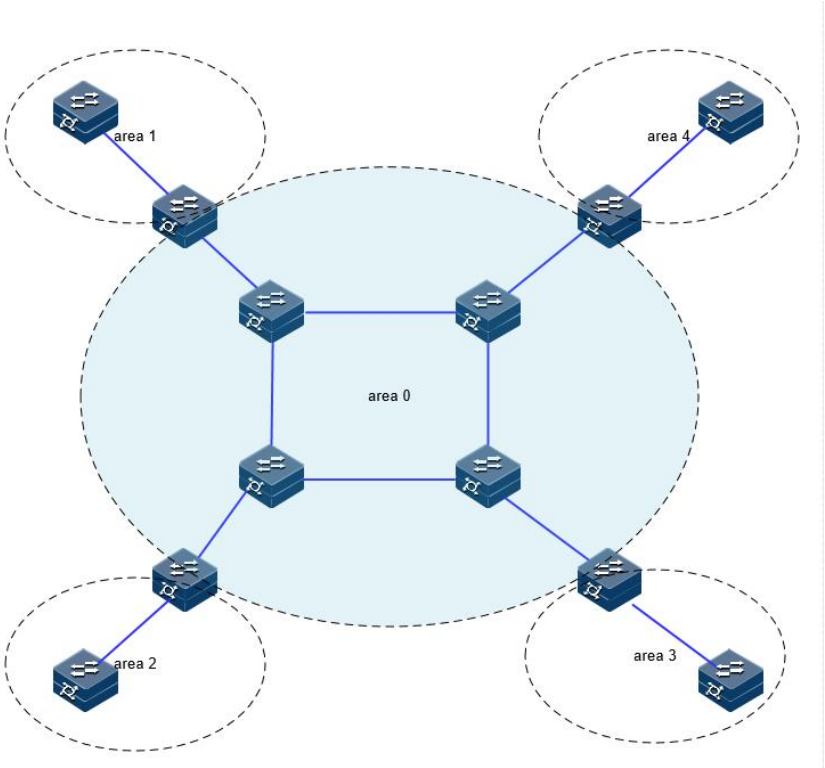
OSPFV3 区域划分

随着网络规模日益扩大，当一个大型网络中的设备都运行 OSPFv3 路由协议时，设备数量的增多会导致链路状态数据库 LSDB（Link-State Database）非常庞大，占用大量的存储空间，并使得运行 SPF 算法的复杂度增加，导致设备负担很重。在网络规模增

大之后，拓扑结构发生变化的概率也增大，网络会经常处于“动荡”之中，造成网络中会有大量的 OSPFV3 协议报文在传递，降低了网络的带宽利用率。更为严重的是，每一次变化都会导致网络中所有的设备重新进行路由计算。

为了解决上述问题，OSPFv3 协议将自治系统划分成不同的区域（Area）。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 OSPFv3 的接口必须指明属于哪一个区域。

图 3-33 OSPFV3 区域划分

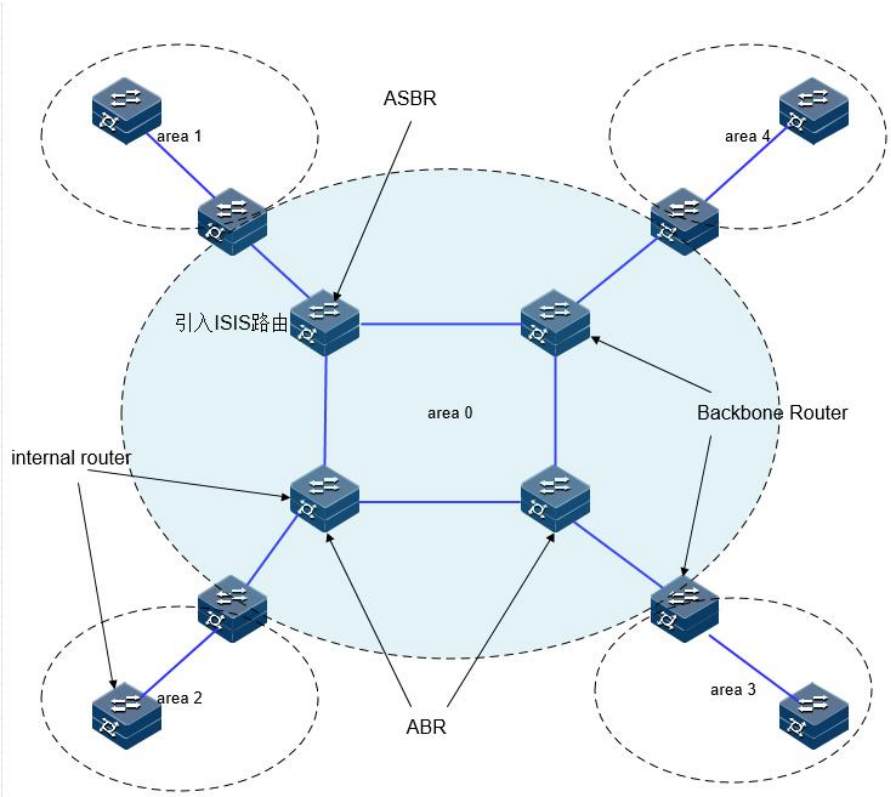


划分区域后，可以在区域边界路由器上进行路由聚合，减少通告到其他区域的 LSA 数量。另外，划分区域还可以使网络拓扑变化造成的影响最小化。

OSPFV3 的区域类型包括普通区域、Stub 区域、NSSA 区域。

OSPFV3 路由器类型

图 3-34 OSPFV3 路由器类型示例



区域内路由器（Internal Router）：该类设备的所有接口都属于同一个 OSPFV3 区域。

区域边界路由器 ABR（Area Border Router）：该类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 Router）：该类设备至少有一个接口属于骨干区域。所有的 ABR 和位于 Area0 的内部设备都是骨干路由器。

自治系统边界路由器 ASBR（AS Boundary Router）：与其他 AS 交换路由信息的设备称为 ASBR。ASBR 并不一定位于 AS 的边界，它可能是区域内设备，也可能是 ABR。只要一台 OSPFV3 设备引入了外部路由的信息，它就成为 ASBR。

OSPFV3 协议报文

OSPFV3 有以下五种类型的协议报文：

- Hello 报文：周期性发送，用于发现和维持 OSPFV3 邻居关系。
- DD（Database Description Packet）报文：描述本地 LSDB 的摘要信息，用于两台路由器开始建立邻接时进行数据库同步。
- LSR 报文（Link State Request Packet）：向对方请求所需的 LSA。

- LSU 报文（Link State Update Packet）：向对方发送其所需要的 LSA。
- LSAck 报文（Link State Acknowledgment Packet）：用来对收到的 LSA 进行确认。

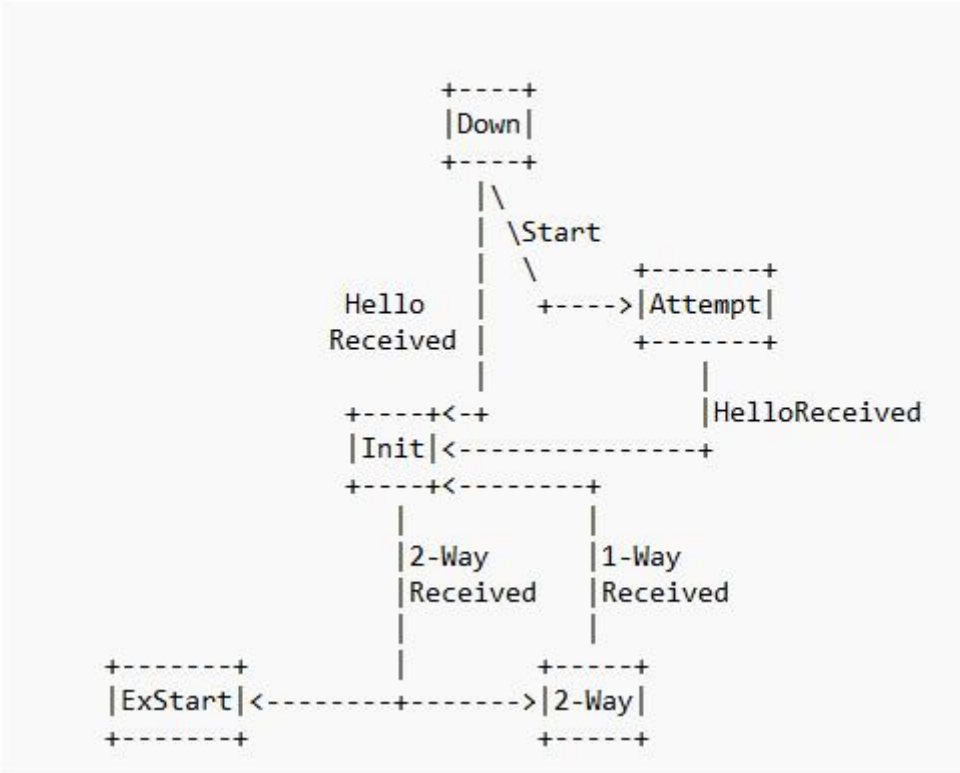
OSPFV3 LSA 类型

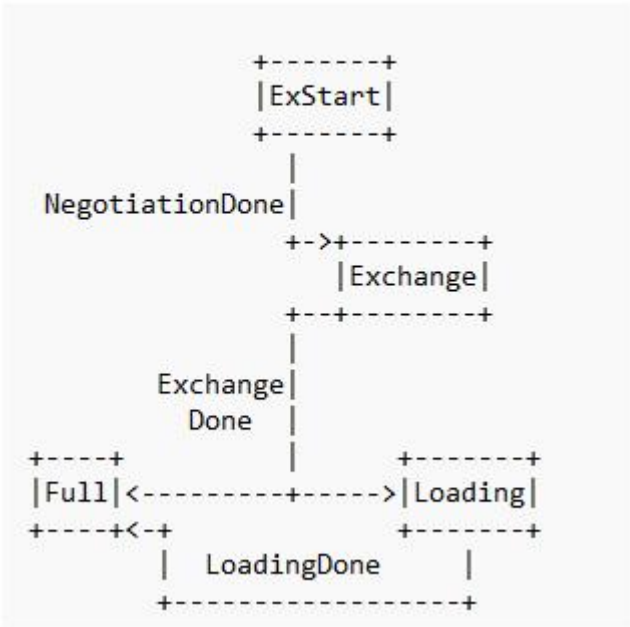
OSPFV3 中对路由信息的描述都是封装在 LSA 中发布出去，常用的 LSA 有以下类型：

- Router LSA（Type1）：每个路由器都会产生，描述了路由器的链路状态和开销，在所属的区域内传播。
- Network LSA（Type2）：由 DR 产生，描述本链路的链路状态，在所属的区域内传播。
- Inter-Area-Prefix-LSA（Type3）：由 ABR 产生，描述区域内某个网段的路由，并通告给其他区域。
- Inter-Area-Router-LSA（Type4）：由 ABR 产生，描述到 ASBR 的路由，通告给除 ASBR 所在区域的其他相关区域。。
- AS-External-LSA（Type5）：由 ASBR 产生，描述到 AS 外部的路由，通告到所有的区域（除了 Stub 区域和 NSSA 区域）。
- NSSA LSA（Type7）：由 ASBR 产生，描述到 AS 外部的路由，仅在 NSSA 区域内传播。
- Link-LSA（Type8）：每个设备都会为每个链路产生一个 Link-LSA，描述到此 Link 上的 link-local 地址、IPv6 前缀地址，并提供将会在 Network-LSA 中设置的链路选项，它仅在此链路内传播。
- Intra-Area-Prefix-LSA（Type9）：每个设备及 DR 都会产生一个或多个此类 LSA，在所属的区域内传播。设备产生的此类 LSA，描述与 Route-LSA 相关联的 IPv6 前缀地址。DR 产生的此类 LSA，描述与 Network-LSA 相关联的 IPv6 前缀地址。

OSPFV3 邻居状态机

图 3-35 OSPFV3 邻居状态机





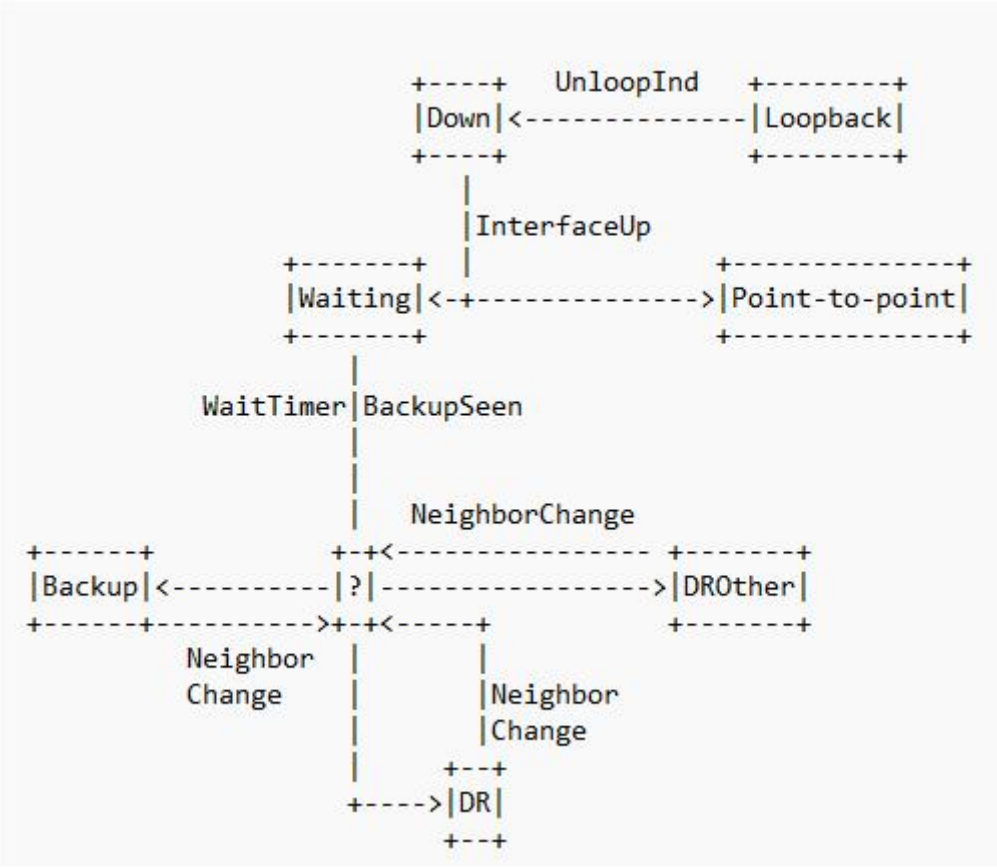
邻居状态机的含义：

- Down：邻居会话的初始阶段。表明没有在邻居失效时间间隔内收到来自邻居设备的 Hello 报文。
- Attempt：处于本状态时，定期向手工配置的邻居发送 Hello 报文。说明： Attempt 状态只适用于 NBMA 类型的接口。

- Init：本状态表示已经收到了邻居的 Hello 报文，但是对端并没有收到本端发送的 Hello 报文。
- 2-way：互为邻居。本状态表示双方互相收到了对端发送的 Hello 报文，建立了邻居关系。如果不形成邻接关系则邻居状态机就停留在此状态，否则进入 Exstart 状态。
- Exstart：协商主/从关系。建立主/从关系主要是为了保证在后续的 DD 报文交换中能够有序的发送。
- Exchange：交换 DD 报文。本端设备将本地的 LSDB 用 DD 报文来描述，并发给邻居设备。
- Loading：正在同步 LSDB。两端设备发送 LSR 报文向邻居请求对方的 LSA，同步 LSDB。
- Full：建立邻接。两端设备的 LSDB 已同步，本端设备和邻居设备建立了邻接状态。

OSPFV3 接口状态机

图 3-36 OSPFV3 接口状态机



接口状态机的含义：

- Down：接口状态的初始状态。
- Loopback：在这种状态下，路由器到网络的接口被环路回绕。
- Waiting：仅适用于广播和 NBMA 接口类型，在该阶段试图识别网络上的 DR 和 BDR。

- Point-to-point：仅适用于点对点，点到多点和虚连接的接口。在这个状态下，尝试和接口另一端路由器建立邻接关系。
- DR：在这种状态下，路由器成功所连网络的 DR，尝试与其他路由器建立邻接关系。
- Backup：在这种状态下，路由器成功所连网络的 BDR，尝试与其他路由器建立邻接关系。
- DROther：在这种状态下，路由器成功所连网络的 DRother，仅会和网络上的 DR 和 BDR 建立邻接关系。

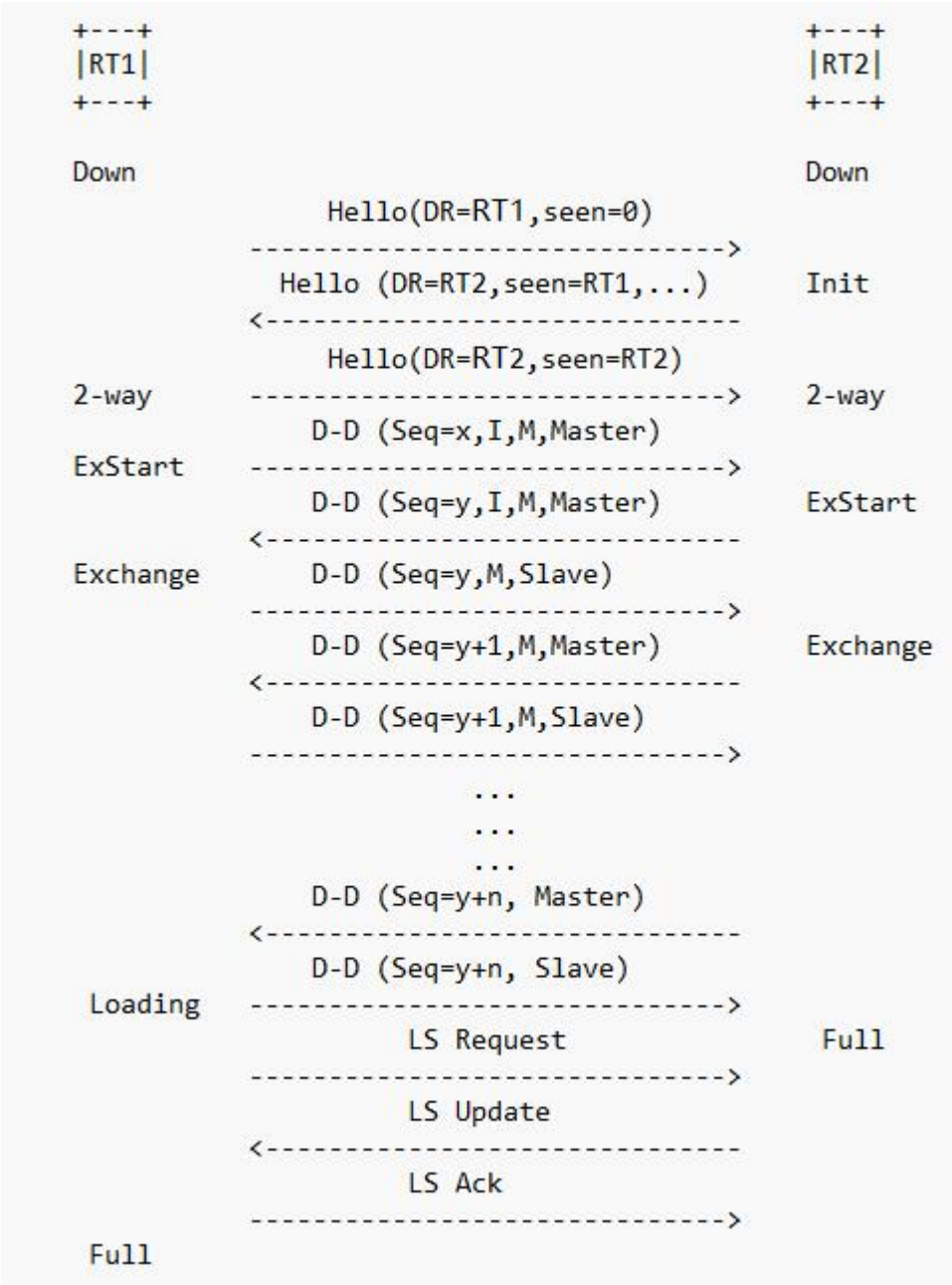
OSPFV3 邻居和邻接

在 OSPFV3 中，邻居（Neighbors）和邻接（Adjacencies）是两个不同的概念。

- 邻居关系：OSPFV3 路由器启动后，会通过 OSPFV3 接口向外发送 Hello 报文。收到 Hello 报文的 OSPFV3 路由器会检查报文中所定义的一些参数，如果双方一致就会形成邻居关系。
- 邻接关系：形成邻居关系的双方不一定都能形成邻接关系，这要根据网络类型而定。只有当双方成功交换 DD 报文，并能交换 LSA 之后，才形成真正意义上的邻接关系。

在广播网络中建立邻接关系流程：

图 3-37 OSPFV3 广播网络建邻示意图



RT1 和 RT2 都连接到广播网络。假设 RT2 为网络的 DR（指定路由器），且 RT2 的 Router ID 高于 Router RT1。

- 建立邻居关系，RT1 的一个连接到广播类型网络的接口上激活了 OSPFV3 协议，会开始发送 Hello 数据包。此时，RouterA 认为自己是 DR 路由器，但不确定邻居是哪台路由器。RT2 收到这个 hello 报文(将邻居状态机改为 Init 状态)，并在其下一个 hello 中数据包表明它本身就是 DR（指定路由器）。RT1 收到 RT2 发来的 Hello 报文。这反过来又导致 RT1 转到状态 2-way，开始启动邻接流程。
- 主/从关系协商、DD 报文交换

- RT1 首先发送一个 DD 报文，宣称自己是 Master（MS=1），并规定序列号 Seq=x。I=1 表示这是第一个 DD 报文，报文中并不包含 LSA 的摘要，只是为了协商主从关系。M=1 说明这不是最后一个报文。
- RT2 在收到 RT1 的 DD 报文后，将 RT1 的邻居状态机改为 Exstart，并且回应一个 DD 报文。由于 RT2 的 Router ID 较大，所以在报文中 RT2 认为自己是 Master，并且重新规定了序列号 Seq=y。
- RT1 收到报文后，同意了 RT2 为 Master，并将 RT2 的邻居状态机改为 Exchange。RT1 使用 RT2 的序列号 Seq=y 来发送新的 DD 报文，该报文开始正式地传送 LSA 的摘要，通过这种方式确认数据库中哪些 LSA 是需要更新的。在报文中 RT1 将 MS=0，说明自己是 Slave。
- RT2 收到报文后，将 RT1 的邻居状态机改为 Exchange，并发送新的 DD 报文来描述自己的 LSA 摘要，此时 RT2 将报文的序列号改为 Seq=y+1。

上述过程持续进行，RT1 通过重复 RT2 的序列号来确认已收到 RT2 的报文。RT2 通过将序列号 Seq 加 1 来确认已收到 RouterA 的报文。当发送最后一个 DD 报文时，在报文中写上 M=0。

LSDB 同步（LSA 请求、LSA 传输、LSA 应答

- RT1 收到最后一个 DD 报文后，发现 RT2 的数据库中存在自己没有的 LSA，将邻居状态机改为 Loading 状态。
- RT2 发现 RT1 的 LSA，RT2 都已经有了，不需要再请求，将 RT1 的邻居状态机改为 Full 状态。
- RT1 发送 LSR 报文向 RT2 请求更新 LSA。RT2 用 LSU 报文来回应 RT1 的请求。RT1 收到后，发送 LSAck 报文确认。

上述过程持续到 RT1 中的 LSA 与 RT2 的 LSA 完全同步为止，此时 RT1 将 RT2 的邻居状态机改为 Full 状态。当路由器交换完 DD 报文并更新所有的 LSA 后，此时邻接关系建立完成。

OSPFV3 路由计算

OSPFV3 路由的计算过程可简单描述如下：

- 每台 OSPFv3 路由器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 OSPFv3 路由器。
- 每台 OSPFv3 路由器都会收集其它路由器发来的 LSA，所有的 LSA 形成链路状态数据库 LSDB（Link State Database），LSDB 是对整个自治系统的网络拓扑结构的描述。
- OSPFv3 路由器将 LSDB 转换成一张带权的有向图，这张图是对整个网络拓扑结构的真实反映。各 OSPFv3 路由器得到的有向图是完全相同的。
- 每台 OSPFv3 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。

OSPFV3 骨干区域

OSPFV3 划分区域之后，并非所有的区域都是平等的关系。其中有一个区域与众不同，通常被称为骨干区域，它的区域号（Area ID）是 0。

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。对此，OSPFV3 有以下规定：

所有非骨干区域必须与骨干区域保持连通。骨干区域自身也必须保持连通。但在实际应用中，可能会因为网络拓扑等限制，无法满足以上要求；这时可以通过配置 OSPFV3 虚连接满足要求。

OSPFV3 虚连接

虚连接指在两台 ABR 之间通过一个非骨干区域而建立的一条逻辑上的连接通道。虚连接相当于在两个 ABR 之间形成了一个点到点的连接。为虚连接两端提供一条非骨干区域内部路由的区域称为中转区域（Transit Area）。

虚连接有如下特点：

- 虚连接的两端必须是 ABR。
- 必须在两端同时配置虚连接，虚连接方能生效。
- 虚连接和物理接口一样可以配置接口参数，如发送 HELLO 报文间隔等。
- 两台 ABR 之间直接传递 OSPFV3 报文信息时，他们之间的 OSPFV3 路由器只起到转发报文的作用。由于协议报文的目的地址不是这些路由器，所以这些报文对于他们而言是透明的，只是当作普通的 IP 报文来转发。

OSPFV3 Stub 区域

Stub 区域的特点：

- Stub 区域的 ABR 不传播它们接收到的自治系统外部路由，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会大大减少。
- Stub 区域是一种可选的配置属性，并不是每个区域都符合配置的条件。通常来说，Stub 区域是位于自治系统边界，只有一个 ABR 的非骨干区域。
- 为保证到自治系统外的路由依旧可达，Stub 区域的 ABR 将生成一条缺省路由，并发布给 Stub 区域中的其他非 ABR 路由器。

配置 Stub 区域的注意事项：

- 骨干区域不能配置成 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必须都要配置 Stub 区域。
- Stub 区域内不能存在 ASBR，即自治系统外部的路由不能在本区域内传播。虚连接不能穿过 Stub 区域。

OSPFV3 NSSA 区域

在 RFC1587 NSSA Option 中增加一类新的区域：NSSA 区域；同时增加一类新的 LSA：NSSA LSA（或称为 Type7 LSA）。

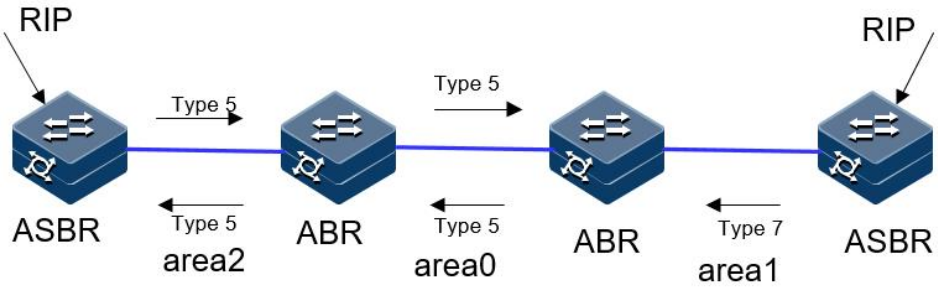
NSSA 区域其实是 Stub 区域的一个变形，它和 Stub 区域有许多相似的地方。两者的差别在于，NSSA 区域能够将自治域外部路由引入并传播到整个 OSPFV3 自治域中，同时又不会学习来自 OSPFV3 网络其它区域的外部路由。

NSSA 区域的特点:

- 与 Stub 区域类似，NSSA 区域也不能配置虚连接。
- 与 Stub 区域类似，NSSA 区域也不允许 AS-External-LSA（即 Type5 LSA 注入，但可以允许 Type7 LSA 注入）。Type7 LSA 由 NSSA 区域的 ASBR 产生，在 NSSA 区域内传播。当 Type7 LSA 到达 NSSA 的 ABR 时，由 ABR 将 Type7 LSA 转换成 AS-External LSA，传播到其他区域。

如图所示，运行 OSPFV3 协议的自治系统包括 3 个区域：区域 1、区域 2 和区域 0

图 3-38 OSPFV3 NSSA 区域示意图



区域 1 被定义为 NSSA 区域。与区域 1、区域 2 相连的非 OSPFV3 网络运行 RIP 协议。区域 1 从 RIP 网络接收的 RIP 路由传播到 NSSA ASBR 后，由 NSSA ASBR 产生 Type7 LSA 在区域 1 内传播；当 Type7 LSA 到达 NSSA ABR 后，转换成 Type5 LSA 传播到区域 0 和区域 2。另一方面，区域 2 从 RIP 网络中接收的 RIP 路由通过区域 2 的 ASBR 产生 Type-5LSA 在 OSPFV3 自治系统中传播。但由于区域 1 是 NSSA 区域，所以 Type-5 LSA 不会到达区域 1。

OSPFV3 路由聚合

路由聚合是指：ABR 将具有相同前缀的路由信息聚合在一起后，形成一条路由发布到其它区域。

AS 被划分成不同的区域后，区域间可以通过路由聚合来减少路由信息，减小路由表的规模，提高路由器的运算速度。

OSPFV3 有两种路由聚合方式：

- ABR 聚合

ABR 向其它区域发送路由信息时，以网段为单位生成 Type3 LSA。如果该区域中存在一些连续的网段，则可以通过路由聚合的命令将这些连续的网段聚合成一个网段。这样 ABR 只发送一条聚合后的 LSA，所有属于命令指定的聚合网段范围的 LSA 将不会再被单独发送出去。

例如，区域 1 内有三条区域内路由 19.1.1.0/24，19.1.2.0/24，19.1.3.0/24，如果此时在 ABR 上配置了路由聚合，将三条路由聚合成一条 19.1.0.0/16，则 ABR 就只生成一条聚合后的 LSA，并发布给其他区域的路由器。

- ASBR 聚合

配置路由聚合后，如果本地设备是自治系统边界路由器 ASBR，将对引入的聚合地址范围内的 Type5 LSA 进行聚合。当配置了 NSSA 区域时，还要对引入的聚合地址范围内的 Type7 LSA 进行聚合。如果本地设备既是 ASBR 又是 ABR，则对由 Type7 LSA 转化成的 Type5 LSA 进行聚合处理。

OSPFV3 路由类型

OSPFV3 将路由分为 4 级，按优先顺序分别是：区域内路由（Intra Area）； 区域间路由（Inter Area）；第一类外部路由（Type1 External）； 第二类外部路由（Type2 External）。

- AS 内部路由：AS 区域内和区域间路由描述的是 AS 内部的网络结构。缺省情况下，这两种路由的协议优先级为 10。
- AS 外部路由：外部路由则描述了应该如何选择到 AS 以外目的地址的路由。OSPFV3 将引入的 AS 外部路由分为两类：Type1 和 Type2。缺省情况下，这两种路由的协议优先级为 150。
- 第一类外部路由：指接收的是 IGP 路由（例如静态路由和 RIP 路由）。由于这类路由的可信程度比较高，所以计算出的外部路由的开销与自治系统内部的路由开销是相同的，并且和 OSPFV3 自身路由的开销具有可比性；即到第一类外部路由的开销等于本路由器到相应的 ASBR 的开销+ASBR 到该路由目的地址的开销。
- 第二类外部路由：指接收的是 EGP 路由。由于这类路由的可信度比较低，所以 OSPFV3 协议认为从 ASBR 到自治系统之外的开销远远大于在自治系统之内到达 ASBR 的开销；所以计算路由开销时将主要考虑前者，即到第二类外部路由的开销=ASBR 到该路由目的地址的开销。如果两条路由计算出的开销值相等，再考虑本路由器到相应的 ASBR 的开销。

OSPFV3 DR 和 BDR

在广播网和 NBMA 网络中，任意两台路由器之间都要传递路由信息。如果网络中有 n 台路由器，则需要建立 nx(n-1)/2 个邻接关系。这使得任何一台路由器的路由变化都会导致多次传递，浪费了带宽资源。

为解决这一问题，OSPFV3 协议定义了 DR（Designated Router）、BDR（Backup Designated Router）和除 DR 和 BDR 之外的路由器（DR Other）。

- DR：所有路由器都只将信息发送给 DR，由 DR 将网络链路状态广播出去。
- BDR：如果 DR 由于某种故障而失效，则网络中的路由器必须重新选举 DR，并与新的 DR 同步。这需要较长的时间，在这段时间内，路由的计算是不正确的。为了能够缩短这个过程，OSPFV3 提出了 BDR（Backup Designated Router）的概念。BDR 实际上是对 DR 的一个备份，在选举 DR 的同时也选举出 BDR，BDR 也和本网段内的所有路由器建立邻接关系并交换路由信息。当 DR 失效后，BDR 会立即成为 DR。由于不需要重新选举，并且邻接关系事先已建立，所以这个过程是非常短暂的。当然这时还需要再重新选举出一个新的 BDR，虽然一样需要较长的时间，但并不会影响路由的计算。

- DR Other：除 DR 和 BDR 之外的路由器（DR Other）之间将不再建立邻接关系，也不再交换任何路由信息。这样就减少了广播网和 NBMA 网络上各路由器之间邻接关系的数量。

DR/BDR 选举过程：

- DR 和 BDR 不是人为指定的，而是由本网段中所有的路由器共同选举出来的。路由器接口的 DR 优先级决定了该接口在选举 DR、BDR 时所具有的资格。本网段内 DR 优先级大于 0 的路由器都可作为“候选人”。选举过程如下：
- 每台路由器将自己选出的 DR 写入 Hello 报文中，发给网段上的每台路由器。
- 如果处于同一网段的两台路由器同时宣布自己是 DR，DR 优先级高者胜出。如果优先级相等，则 Router ID 大者胜出。如果一台路由器的优先级为 0，则它不会被选举为 DR 或 BDR。

DR/BDR 选举特点：

- 只有在广播或 NBMA 类型接口时才会选举 DR，在点到点或点到多点类型的接口上不需要选举 DR。
- DR 是指某个网段中概念，是针对路由器的接口而言的。某台路由器在一个接口上可能是 DR，在另一个接口上有可能是 BDR，或者是 DR Other。
- 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。
- DR 不一定就是 DR 优先级最大的路由器；同理，BDR 也不一定就是 DR 优先级第二大的路由器。

3.9.2 配置准备

场景

完成通过配置 OSPFv3 基本功能可以组建基本的 OSPFv3 网络。

前提

使能 ipv6 能力，使各相邻节点网络层可达。

3.9.3 缺省配置

设备上 OSPFV3 的缺省配置如下。

功能	缺省值
OSPFV3 特性	不使能
Hello 报文发送间隔	10s
邻居失效时间	40s
计算接口开销的带宽参考值	100Mbit/s

3.9.4 配置 OSPFV3 基本功能

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)# **ospfv3** *process-id* [**vpn-instance** *NAME*]	启动 OSPFV3 进程，进入 OSPFV3 视图 process-id 缺省值为 1 vpn-instance 表示 vpn 实例，若指定则此 OSPFV3 进程属于 VPN 实例，若未指定则属于公网实例。
3	JX(config-ospfv3-1)#**router-id** *ipv4-address*	配置 OSPFV3 进程的 ID 号，需要保证 router id 全网唯一，不配置的情况下，系统会从当前接口的 IP 地址中选择一个作为 router id。
4	JX(config)# **interface** *interface-type interface-number*	进入接口模式，本章以 vlan 口为例。
5	JX (config-vlanif-1)# **ospfv3** *process-id* **area** *area-id* [**instance** *imstance-id*]	接口使能 OSPFV3 并创建区域，配置接口所运行的实例和区域。
6	JX (config-vlanif-1)# **ospfv3 if-type** { **broadcast** | **p2p** } [**instance** *imstance-id*]	配置接口的网络类型。（可选）
7	JX(config)# **ospfv3** *process-id* [**vpn-instance** *NAME*]	进入 OSPFV3 视图
8	JX(config-ospfv3-1)# **maximum load-balancing** {*load-balancing-number* | **default**}	配置 OSPFV3 最大等价路由条数（可选）
9	JX(config-ospfv3-1)# **preference** *preference-value* [**route-policy** *NAME*]	配置 OSPFV3 域内路由和域间路由的优先级
10	JX(config-ospfv3-1)# **preference ase** *preference-value* [**route-policy** *NAME*]	配置 OSPFV3 外部路由的优先级

3.9.5 配置 OSPFV3 STUB 区域

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。

步骤	配置	说明
2	JX(config)# **ospfv3** *process-id* [**vpn-instance** *NAME*]	进入 OSPFV3 视图。
3	JX(config-ospfv3-1)# **area** *area-id* **stub [no-summary-lsa**]	配置区域为 stub 区域，区域不能为骨干区域。 **no-summary** 表示禁止 ABR 向 Stub 区域内发送 Summary LSA。只有在 ABR 上配置 **stub** 命令时，可选参数 **no-summary** 才能对该区域起作用。

3.9.6 配置 OSPFV3 NSSA 区域

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)# **ospfv3** *process-id* [**vpn-instance** *NAME*]	进入 OSPFV3 视图。
3	JX(config-ospfv3-1)# **area** *area-id* **nssa [no-summary-lsa**]	配置区域为 stub 区域，区域不能为骨干区域。 **no-summary** 表示禁止 ABR 向 nssa 区域内发送 Summary LSA。
4	JX(config-ospfv3-1)# **area** *area-id* **nssa translator** { **always** | **candidate** }	candidate 状态时，NSSA 区域会根据规则自动选择一个 ABR 作为转换器，将 Type7 LSA 转换为 Type5 LSA。通过在 ABR 上配置 translator always，可以将某一个 ABR 指定为转换器。

3.9.7 配置 OSPFV3 引入外部路由

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)# **ospfv3** *process-id* [**vpn-instance** *NAME*]	进入 OSPFV3 视图。

步骤	配置	说明
3	JX(config-ospfv3-1)# **import-route** { **connect** | **static** | **rip** | **bgp** | **isis** | **ospf** } { **cost** { *cost*-value | **inherit** } | **route-policy** *NAME* }*	引入其它协议的路由信息。 cost 参数为 OSPFV3 引入的外部路由的缺省度量值，inherit 表示引入路由的开销值为路由自带的 cost 值。如果没有指定开销值，则使用 default route-attribute 命令设置的缺省开销值。 route-policy 参数用于绑定路由策略，在路由引入时进行过滤。

3.9.8 配置 OSPFV3 路由聚合

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)# **ospfv3** *process-id* [**vpn-instance** *NAME*]	进入 OSPFV3 视图。
3	JX(config-ospfv3-1)# **area** *area-id* **abr-summary** *dst-address/dst-maskLen* { **advertise** | **no-advertise** } [**cost** { *cost-value* | **inherit-minimum** }]	配置 OSPFV3 的 ABR 路由聚合
4	JX(config-ospfv3-1)# **area** *area-id* **nssa summary** *dst-address/dst-maskLen* { **advertise** | **no-advertise** } [**cost** { *cost-value* | **inherit-minimum** }]	配置在本路由器进行 type-7LSA 到 type-5LSA 转换时的汇聚功能。
5	JX(config-ospfv3-1)# **asbr-summary** *dst-address/dst-maskLen*	配置 OSPFV3 的 ASBR 路由聚合。
6	JX(config-ospfv3-1)# **asbr-summary** { **connect** | **static** | **rip** | **bgp** | **isis** | **ospf** } *dst-address/dst-maskLen* [**advertise** { **enable** | **disable** } | **translate** { **enable** | **disable** } | **cost** { *cost* | **default** } | **type** { *cost-type* | **default** }]*	配置 OSPFV3 针对某个协议引入的路由进行 ASBR 路由聚合

3.9.9 配置 OSPFV3 虚连接

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。

步骤	配置	说明
2	JX(config)# **ospfv3** *process-id* [**vpn-instance** *NAME*]	进入 OSPFV3 视图。
3	JX(config-ospfv3-1)# **area** *area-id* **virtual-link** *ipv4-address*	创建并配置虚连接。
4	JX(config-ospfv3-1)# **area** *area-id* **virtual-link** *ipv4-address* **hello-interval** { *hello-interval-time* | **default** }	配置虚连接的 hello 包发送间隔（可选）。
5	JX(config-ospfv3-1)# **area** *area-id* **virtual-link** *ipv4-address* **dead-interval** { *dead-interval-time* | **default** }	配置虚连接的邻居失效时间（可选）。
6	JX(config-ospfv3-1)# **area** *area-id* **virtual-link** *ipv4-address* **retransmit-interval** { *retransmit-interval-time* | **default** }	配置虚连接的 LSA 重传时间间隔（可选）。
7	JX(config-ospfv3-1)# **area** *area-id* **virtual-link** *neighbor-address* **transmit-delay** { *transmit-interval-time* | **default** }	配置虚连接延迟发送 LSA 的时间间隔（可选）。

3.9.10 检查配置

配置完成后，请在设备上执行以下命令检查配置结果。

步骤	配置	说明
1	JX#**show ospfv3 interface**	查看 OSPFV3 接口。
2	JX# **show ospfv3 process** *process-id*	查看 OSPFV3 实例。
3	JX# **show ospfv3 area**	查看 OSPFV3 区域。
4	JX# **show ospfv3 neighbor**	查看 OSPFV3 邻居，对端进行对应的配置之后，可以建立邻居。
5	JX# **show ospfv3 route**	查看 OSPFV3 路由表。
6	JX# **show ospfv3 database**	查看 OSPFV3 数据库。
7	JX# **show ospfv3 import-route**	查看 OSPFV3 引入的路由信息。
8	JX# **show ospfv3 virtual-link**	查看 OSPFV3 虚连接信息。
9	JX#**show ospfv3 interface** *interface-type interface-number*	查看接口的具体信息。

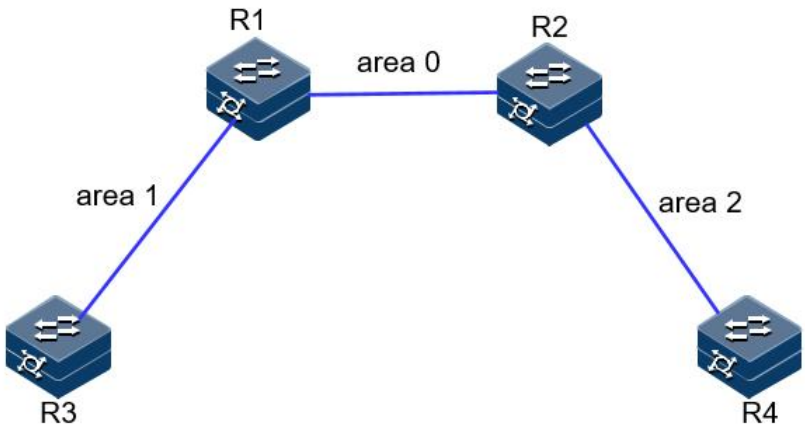
步骤	配置	说明
10	JX# **show ospfv3 area area-id** *area-id*	查看区域的具体信息。
11	JX# **show ospfv3 information**	查看 OSPFV3 的全局信息。

3.9.11 配置 OSPFV3 基本功能示例

组网需求

如下图所示，所有的 router 都运行 OSPFV3，并将整个自治系统划分为 3 个区域，其中 R1 和 R2 为 ABR 来转发区域之间的路由。配置完成后，每台 router 都应学到自治系统内的到所有网段的路由。

图 3-39 OSPFV3 基本配置组网图



配置步骤

步骤 1 配置各接口的 IPV6 地址和 IPV6 能力。

R1 和 R2 建邻接口使用 vlan12,R1 和 R3 建邻接口使用 vlan 13,,R2 和 R4 建邻接口使用 vlan24。

步骤 2 配置 OSPFV3。

R1：

```
JX-R1#configure
JX-R1(config)#ospfv3 1
JX-R1(config-ospfv3-1)#router-id 1.1.1.1
JX-R1(config-ospfv3-1)#exit
JX-R1(config)#interface vlan 13
JX-R1(config-vlanif-13)#ospfv3 1 area 1
JX-R1(config-vlanif-13)#exit
JX-R1(config)#interface vlan 12
JX-R1(config-vlanif-12)#ospfv3 1 area 0
```

```
JX-R1(config-vlanif-12)#exit
```

R2：

```
JX-R2#configure
JX-R2(config)#ospfv3 1
JX-R2(config-ospfv3-1)#router-id 2.2.2.2
JX-R2(config-ospfv3-1)#exit
JX-R2(config)#interface vlan 12
JX-R2(config-vlanif-12)#ospfv3 1 area 0
JX-R2(config-vlanif-12)#exit
JX-R2(config)#interface vlan 24
JX-R2(config-vlanif-24)#ospfv3 1 area 2
JX-R2(config-vlanif-24)#exit
```

R3：

```
JX-R3#configure
JX-R3(config)#ospfv3 1
JX-R3(config-ospfv3-1)#router-id 3.3.3.3
JX-R3(config-ospfv3-1)#exit
JX-R3(config)#interface vlan 13
JX-R3(config-vlanif-13)#ospfv3 1 area 1
JX-R3(config-vlanif-13)#exit
```

R4：

```
JX-R4#configure
JX-R4(config)#ospfv3 1
JX-R4(config-ospfv3-1)#router-id 4.4.4.4
JX-R4(config-ospfv3-1)#exit
JX-R4(config)#interface vlan 24
JX-R4(config-vlanif-24)#ospfv3 1 area 2
JX-R4(config-vlanif-24)#exit
```

步骤 3 验证配置结果。

（1）使用 show ospfv3 neighbor 查看邻居建立结果：

R1:

```
JX-R1(config)#show ospfv3 neighbor

 Ospfv3 Process 1
 NeighborId        Priority  State    Interface     Instance  Aging
UpTime   IpAddress
-------------------------------------------------------------------------
---------------------------------------
 2.2.2.2           1       Full     vlan-12       0        36       1:20:40
fe80::f2f1:f2ff:fef3:201
 3.3.3.3           1       Full     vlan-13       0        32       1:13:54
fe80::f2f1:f2ff:fef3:301
-------------------------------------------------------------------------
---------------------------------------
  Down:0       Attempt:0       Init:0       TwoWay:0       ExchangeStart:0
Exchange:0
  Loading:0       Full:2
```

R2：

```
JX-R2(config)#show ospfv3 neighbor

 Ospfv3 Process 1
 NeighborId        Priority  State    Interface     Instance  Aging
UpTime   IpAddress
-------------------------------------------------------------------------
-----------------------------------
 1.1.1.1           1       Full     vlan-12       0        38       1:27:13
fe80::f2f1:f2ff:fef3:101
 4.4.4.4           1       Full     vlan-24       0        28       0:10:27
fe80::f2f1:f2ff:fef3:101
-------------------------------------------------------------------------
-----------------------------------
  Down:0       Attempt:0       Init:0       TwoWay:0       ExchangeStart:0
Exchange:0
  Loading:0       Full:2
```

（2）通过 show ospfv3 database 查看数据库

```
JX-R1(config)#show ospfv3 database
Database of OSPFv3 Process 1

          Router Link State (Area 0.0.0.0)
 LinkId          ADV Router       Age     Seq#       CheckSum    Len
 0.0.0.0         1.1.1.1          77      0x80000002  0x1660      40
 0.0.0.0         2.2.2.2          78      0x80000009  0xe981      40

          Network Link State (Area 0.0.0.0)
 LinkId          ADV Router       Age     Seq#       CheckSum    Len
 64.0.0.13       2.2.2.2          78      0x80000001  0xa423      32

         Inter Area Prefix Link State (Area 0.0.0.0)
 LinkId          ADV Router       Age     Seq#       CheckSum    Len
Prefix
 0.0.0.2         1.1.1.1          118     0x80000001  0x4c8d      36
13::/64
 0.0.0.2         2.2.2.2          409     0x80000004  0xf4cc      36
24::/64

         Intra Area Prefix Link State (Area 0.0.0.0)
 LinkId          ADV Router       Age     Seq#       CheckSum    Len
 0.0.3.232       2.2.2.2          78      0x80000001  0x83e6      44

         Link(Type-8) State(interface vlan-12 Area 0.0.0.0)
 LinkId          ADV Router       Age     Seq#       CheckSum    Len
 64.0.0.13       1.1.1.1          118     0x80000001  0x4d41      76
 64.0.0.13       2.2.2.2          448     0x80000004  0x780d      76

          Router Link State (Area 0.0.0.1)
 LinkId          ADV Router       Age     Seq#       CheckSum    Len
 0.0.0.0         1.1.1.1          119     0x80000005  0x82ea      40
 0.0.0.0         3.3.3.3          120     0x80000009  0x3b27      40

          Network Link State (Area 0.0.0.1)
```

```
 LinkId          ADV Router      Age     Seq#       CheckSum    Len
 64.0.0.14       3.3.3.3         120     0x80000001  0x9e20      32

         Inter Area Prefix Link State (Area 0.0.0.1)
 LinkId          ADV Router      Age     Seq#       CheckSum    Len
Prefix
 0.0.0.2         1.1.1.1         118     0x80000001  0x409a      36
12::/64
 0.0.0.3         1.1.1.1         72      0x80000001  0x15b1      36
24::/64

         Intra Area Prefix Link State (Area 0.0.0.1)
 LinkId          ADV Router      Age     Seq#       CheckSum    Len
 0.0.3.232       3.3.3.3         120     0x80000001  0xb5aa      44

         Link(Type-8) State(interface vlan-13 Area 0.0.0.1)
 LinkId          ADV Router      Age     Seq#       CheckSum    Len
 64.0.0.14       1.1.1.1         122     0x80000001  0x8ffb      76
 64.0.0.14       3.3.3.3         36      0x80000004  0xeb90      76
```

（3）通过 show ospfv3 route 查看路由表

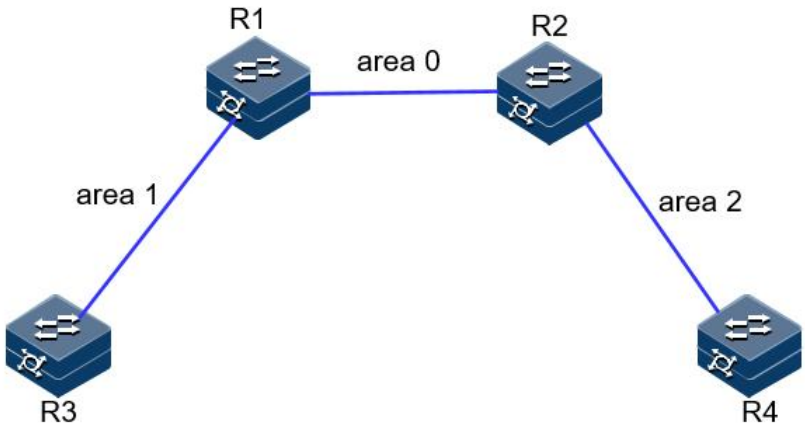
```
JX-R1(config)#show ospfv3 route
 Ospfv3 Process 1
 RoutType  Prefix              AreaId        PathType    Cost    Cost2
NextHopIf   NextHopNbr                  Backu
pNextHop
 ABR      2.2.2.2/32           0.0.0.0       INTRA       1       0
vlan-12     fe80::f2f1:f2ff:fef3:201       ::
 PREFIX    12::/64             0.0.0.0       INTRA       1       0
vlan-12     12::                        ::
 PREFIX    13::/64             0.0.0.1       INTRA       1       0
vlan-13     13::                       ::
 PREFIX    24::/64             0.0.0.0       INTER       2       0
vlan-12     fe80::f2f1:f2ff:fef3:201       ::
 Route:
 Prefix(Intra  Inter  External)ABR      ASBR     External
 3     2     0      0       1       0

 Path:
 Prefix(Intra  Inter  External)ABR      ASBR     External
 3     2     0      0       1       0
```

3.9.12 配置 OSPFV3 STUB 区域示例

组网需求

如下图所示，所有的 router 都运行 OSPFV3，并将整个自治系统划分为 3 个区域，其中 R1 和 R2 为 ABR 来转发区域之间的路由。

图 3-40 OSPFV3 STUB 区域示例组网图



配置步骤

步骤 1 配置同 OSPFV3 基本功能配置。

步骤 2 配置 area 1 为 stub 区域。

R1：

```
JX-R1#configure
JX-R1(config)#ospfv3 1
JX-R1(config-ospfv3-1)#area 1 stub
```

R3：

```
JX-R3#configure
JX-R3(config)#ospfv3 1
JX-R3(config-ospfv3-1)#area 1 stub
```

步骤 3 配置 R4 引入 100.1.1.1 的五类 LSA

```
JX-R4(config)#ipv6 route-static 100::1 128 1::3
JX-R4(config)#ospfv3 1
JX-R4(config-ospfv3-1)#import-route static
```

步骤 4 验证配置结果

当 R3 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。

```
JX-R3(config)#show ospfv3 route
 Ospfv3 Process 1
 RoutType  Prefix              AreaId        PathType    Cost    Cost2
NextHopIf   NextHopNbr                  Backu
pNextHop
 ABR      1.1.1.1/32           0.0.0.1       INTRA       1       0
vlan-13     fe80::f2f1:f2ff:fef3:101       ::
 ASBR     4.4.4.4/32           0.0.0.1       INTER       3       0
vlan-13     fe80::f2f1:f2ff:fef3:101       ::
 PREFIX   12::/64              0.0.0.1       INTER       2       0
vlan-13     fe80::f2f1:f2ff:fef3:101       ::
 PREFIX   13::/64              0.0.0.1       INTRA       1       0
vlan-13     13::                          ::
```

```
 PREFIX    24::/64              0.0.0.1        INTER      3      0
vlan-13     fe80::f2f1:f2ff:fef3:101       ::
 PREFIX    100::1/128            n/a           EXTERNAL_2  3      1
vlan-13     fe80::f2f1:f2ff:fef3:101       ::
 Route:
 Prefix(Intra  Inter  External)ABR      ASBR     External
 4     1     0      1       1       1

 Path:
 Prefix(Intra  Inter  External)ABR      ASBR     External
 4     1     0      1       1       1
```

当 R3 所在区域配置为 Stub 区域时，已经看不到 AS 外部的路由，取而代之的是一条通往区域外部的缺省路由。

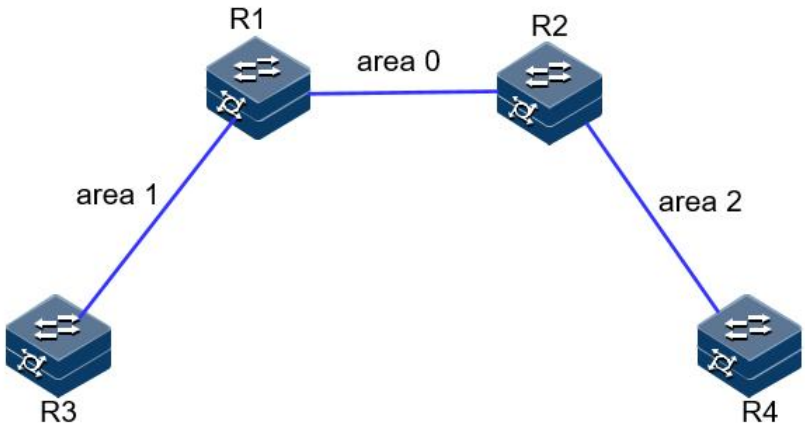
```
JX-R3(config)#show ospfv3 route
 Ospfv3 Process 1
 RoutType  Prefix               AreaId         PathType    Cost    Cost2
NextHopIf   NextHopNbr                   Backu
pNextHop
 ABR      1.1.1.1/32            0.0.0.1        INTRA      1      0
vlan-13     fe80::f2f1:f2ff:fef3:101       ::
 PREFIX    ::/0                 0.0.0.1        INTER      2      0
vlan-13     fe80::f2f1:f2ff:fef3:101       ::
 PREFIX    12::/64              0.0.0.1        INTER      2      0
vlan-13     fe80::f2f1:f2ff:fef3:101       ::
 PREFIX    13::/64              0.0.0.1        INTRA      1      0
vlan-13     13::                          ::
 PREFIX    24::/64              0.0.0.1        INTER      3      0
vlan-13     fe80::f2f1:f2ff:fef3:101       ::
 Route:
 Prefix(Intra  Inter  External)ABR      ASBR     External
 4     1     0      0       1       0

 Path:
 Prefix(Intra  Inter  External)ABR      ASBR     External
 4     1     0      0       1       0
```

3.9.13 配置 OSPFV3 NSSA 区域示例

组网需求

如下图所示，所有的 router 都运行 OSPFV3，并将整个自治系统划分为 3 个区域，其中 R1 和 R2 为 ABR 来转发区域之间的路由。

图 3-41 OSPFV3 NSSA 区域示例组网图



配置步骤

步骤 1 配置同 OSPFV3 基本功能配置。

步骤 2 配置 area 1 为 NSSA 区域。

R1：

```
JX-R1#configure
JX-R1(config)#ospfv3 1
JX-R1(config-ospfv3-1)#area 1 nssa
```

R3：

```
JX-R3#configure
JX-R3(config)#ospfv3 1
JX-R3(config-ospfv3-1)#area 1 nssa
```

步骤 3 配置 R4 引入 100.1.1.1 的五类 LSA

```
JX-R4(config)#ipv6 route-static 100::1 128 1::3
JX-R4(config)#ospfv3 1
JX-R4(config-ospfv3-1)#import-route static
```

步骤 4 验证配置结果

nssa 区域的数据库比正常区域的数据库多一个缺省 NSSA 类型 LSA，R3 上没有 100::1 的外部 LSA

```
JX-R3(config)#show ospfv3 database
Database of OSPFv3 Process 1

          Router Link State (Area 0.0.0.1)
 LinkId          ADV Router      Age     Seq#        CheckSum    Len
 0.0.0.0         1.1.1.1         5       0x80000004  0xa64       40
 0.0.0.0         3.3.3.3         4       0x80000002  0xc8a2      40

          Network Link State (Area 0.0.0.1)
 LinkId          ADV Router      Age     Seq#        CheckSum    Len
 64.0.0.14       1.1.1.1         5       0x80000001  0x1fa1      32

          Inter Area Prefix Link State (Area 0.0.0.1)
```

```
 LinkId          ADV Router      Age     Seq#       CheckSum    Len
Prefix
 0.0.0.8         1.1.1.1         195     0x80000001  0x4d0      36
12::/64
 0.0.0.9         1.1.1.1         195     0x80000001  0xd8e7     36
24::/64

          Intra Area Prefix Link State (Area 0.0.0.1)
 LinkId          ADV Router      Age     Seq#       CheckSum    Len
 0.0.3.232       1.1.1.1         5       0x80000001  0x8de2     44

          Nssa Link State (Area 0.0.0.1)
 LinkId          ADV Router      Age     Seq#       CheckSum    Len
Prefix
 0.0.0.1         1.1.1.1         195     0x80000001  0x4de7
28     ::/0

          Link(Type-8) State(interface vlan-13 Area 0.0.0.1)
 LinkId          ADV Router      Age     Seq#       CheckSum    Len
 64.0.0.14       1.1.1.1         195     0x80000001  0xb3d1     76
 64.0.0.14       3.3.3.3         11      0x80000001  0x1663     76
```

步骤 5 配置 R3 引入 200::1 的静态路由

```
JX-R3(config)#ipv6 route-static 200::1 128 13::5
JX-R3(config)#ospfv3 1
JX-R3(config-ospfv3-1)#import-route static
```

步骤 6 验证配置结果

R3 上存在 200::1 的五类 LSA 和 NSSA LSA

```
JX-R3(config)#show ospfv3 database
Database of OSPFv3 Process 1

           Router Link State (Area 0.0.0.1)
 LinkId          ADV Router      Age     Seq#       CheckSum    Len
 0.0.0.0         1.1.1.1         106     0x80000004  0xa64      40
 0.0.0.0         3.3.3.3         30      0x80000003  0xcc9b     40

           Network Link State (Area 0.0.0.1)
 LinkId          ADV Router      Age     Seq#       CheckSum    Len
 64.0.0.14       1.1.1.1         106     0x80000001  0x1fa1     32

          Inter Area Prefix Link State (Area 0.0.0.1)
 LinkId          ADV Router      Age     Seq#       CheckSum    Len
Prefix
 0.0.0.8         1.1.1.1         296     0x80000001  0x4d0      36
12::/64
 0.0.0.9         1.1.1.1         296     0x80000001  0xd8e7     36
24::/64

          Intra Area Prefix Link State (Area 0.0.0.1)
 LinkId          ADV Router      Age     Seq#       CheckSum    Len
 0.0.3.232       1.1.1.1         106     0x80000001  0x8de2     44
```

```
          Nssa Link State (Area 0.0.0.1)
 LinkId          ADV Router       Age     Seq#        CheckSum    Len
Prefix
 0.0.0.1         1.1.1.1          296     0x80000001   0x4de7
28     ::/0
 0.0.0.2         3.3.3.3          28      0x80000001   0xe6ad      44
200::1/128

          Link(Type-8) State(interface vlan-13 Area 0.0.0.1)
 LinkId          ADV Router       Age     Seq#        CheckSum    Len
 64.0.0.14        1.1.1.1         296     0x80000001   0xb3d1      76
 64.0.0.14        3.3.3.3         112     0x80000001   0x1663      76

           AS External Link State
 LinkId          ADV Router       Age     Seq#        CheckSum    Len
Prefix
 0.0.0.2         3.3.3.3          28      0x80000001   0x2155      44
200::1/128
```

查看 R4 的路由表和数据库，存在 200.1.1.0 路由和 LSA。

```
JX-R4(config)#show ospfv3 database
Database of OSPFv3 Process 1

           Router Link State (Area 0.0.0.2)
 LinkId          ADV Router       Age     Seq#        CheckSum    Len
 0.0.0.0         2.2.2.2          1222    0x80000005   0x73e3      40
 0.0.0.0         4.4.4.4          1187    0x80000004   0x3c13      40

           Network Link State (Area 0.0.0.2)
 LinkId          ADV Router       Age     Seq#        CheckSum    Len
 64.0.0.25        2.2.2.2         1221    0x80000002   0xc0ed      32

          Inter Area Prefix Link State (Area 0.0.0.2)
 LinkId          ADV Router       Age     Seq#        CheckSum    Len
Prefix
 0.0.0.4         2.2.2.2          189     0x80000004   0x8c9       36
12::/64
 0.0.0.6         2.2.2.2          154     0x80000005   0x4c8       36
13::/64

          Inter Area Router Link State (Area 0.0.0.2)
 LinkId          ADV Router       Age     Seq#        CheckSum    Len
Prefix
 0.0.0.1         2.2.2.2          337     0x80000001    0xa0f        32
1.1.1.1
 0.0.0.2         2.2.2.2          69      0x80000001    0x62ae       32
3.3.3.3

          Intra Area Prefix Link State (Area 0.0.0.2)
 LinkId          ADV Router       Age     Seq#        CheckSum    Len
 0.0.3.232        2.2.2.2         1221    0x80000002   0x63e7      44

          Link(Type-8) State(interface vlan-24 Area 0.0.0.2)
```

```
 LinkId          ADV Router      Age     Seq#       CheckSum    Len
 64.0.0.25       2.2.2.2         430     0x80000005  0x5bf8      76
 64.0.0.25       4.4.4.4         1260    0x80000002  0xd57b      76

          AS External Link State
 LinkId          ADV Router      Age     Seq#       CheckSum    Len
Prefix
 0.0.0.2         1.1.1.1         68      0x80000001  0x5d21      44
200::1/128
 0.0.0.2         4.4.4.4         1185    0x80000001  0xf77b      44
100::1/128

JX-R4(config)#show ospfv3 route
 Ospfv3 Process 1
 RoutType  Prefix              AreaId        PathType   Cost   Cost2
NextHopIf   NextHopNbr                  Backu
pNextHop
 ABR      2.2.2.2/32           0.0.0.2       INTRA      1      0
vlan-24    fe80::f2f1:f2ff:fef3:201      ::
 ASBR     1.1.1.1/32           0.0.0.2       INTER      2      0
vlan-24    fe80::f2f1:f2ff:fef3:201      ::
 ASBR     3.3.3.3/32           0.0.0.2       INTER      3      0
vlan-24    fe80::f2f1:f2ff:fef3:201      ::
 PREFIX   12::/64              0.0.0.2       INTER      2      0
vlan-24    fe80::f2f1:f2ff:fef3:201      ::
 PREFIX   13::/64              0.0.0.2       INTER      3      0
vlan-24    fe80::f2f1:f2ff:fef3:201      ::
 PREFIX   24::/64              0.0.0.2       INTRA      1      0
vlan-24    24::                       ::
 PREFIX   200::1/128           n/a           EXTERNAL_2 2      1
vlan-24    fe80::f2f1:f2ff:fef3:201      ::
 Route:
 Prefix(Intra  Inter  External)ABR      ASBR     External
 4     1     0      1       1       2

 Path:
 Prefix(Intra  Inter  External)ABR      ASBR     External
 4     1     0      1       1       2
```

3.9.14 配置 OSPFV3 引入外部路由示例

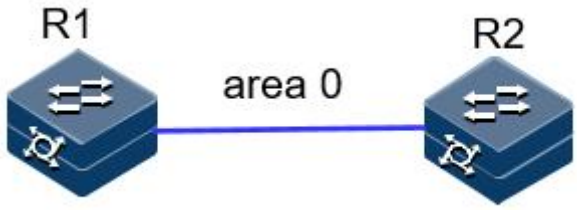
组网需求

如下图所示， 2 个 router 都运行 OSPFV3，并将有个都配置为区域 0。假定 R1 需要向 OSPF 导入外部路由，R1 存在静态路由 30::1/128，本地路由 40::1/128 但是对外部路由有如下要求：

- 接受所有直连路由，并采用默认配置；
- 接收所有静态路由，并为路由配置开销 2000；

配置完成后，每台设备都应学到自治系统内的到所有网段的路由。。

图 3-42 OSPV3F 引入外部路由组网图



配置步骤

步骤 1 配置各接口的 IP 地址。

R1 的一个接口地址：1.1.1.1/24

R2 的一个接口地址：1.1.1.2/24

步骤 2 配置 OSPF。

R1：

```
JX-R1#configure
JX-R1(config)#ospfv3 1
JX-R1(config-ospfv3-1)#router-id 1.1.1.1
JX-R1(config-ospfv3-1)#exit
JX-R1(config)#interface vlan 12
JX-R1(config-vlanif-12)#ospfv3 1 area 0
JX-R1(config-vlanif-12)#exit
```

R2：

```
JX-R2#configure
JX-R2(config)#ospfv3 1
JX-R2(config-ospfv3-1)#router-id 2.2.2.2
JX-R2(config-ospfv3-1)#exit
JX-R2(config)#interface vlan 12
JX-R2(config-vlanif-12)#ospfv3 1 area 0
```

步骤 3 配置重分配。

```
JX-R1#configure
JX-R1(config)#ospfv3 1
JX-R1(config-ospfv3-1)#import-route connect
JX-R1(config-ospfv3-1)#import-route static cost 2000
```

步骤 4 检查配置结果。

```
JX-R1(config)#show ospfv3 import-route
  Import Route of OSPF(1)
 dest                      cost    proto   pid
 1::/64                     1      connect 0
 30::1/128                  60      static  0
 40::1/128                  1       connect 0
JX-R1(config)#show ospfv3 database ls-type external
Database of OSPFv3 Process 1
```

```
          AS External Link State
 LinkId          ADV Router       Age     Seq#       CheckSum    Len
Prefix
 0.0.0.2         1.1.1.1          35      0x80000001  0x8bed      44
30::1/128
 0.0.0.3         1.1.1.1          32      0x80000001  0x5d6a      36
1::/64
 0.0.0.4         1.1.1.1          32      0x80000001  0x3608      44
40::1/128
```

在 R2 查看路由表：

```
JX-R2(config)#show ospfv3 route
 Ospfv3 Process 1
 RoutType  Prefix               AreaId         PathType    Cost    Cost2
NextHopIf   NextHopNbr                   Backu
pNextHop
 ASBR     1.1.1.1/32            0.0.0.0        INTRA       1       0
vlan-12     fe80::f2f1:f2ff:fef3:101       ::
 PREFIX    1::/64                n/a           EXTERNAL_2  1       1
vlan-12     fe80::f2f1:f2ff:fef3:101       ::
 PREFIX    12::/64               0.0.0.0        INTRA       1       0
vlan-12     12::                          ::
 PREFIX    30::1/128             n/a            EXTERNAL_2  1       2000
vlan-12     fe80::f2f1:f2ff:fef3:101       ::
 PREFIX    40::1/128             n/a            EXTERNAL_2  1       1
vlan-12     fe80::f2f1:f2ff:fef3:101       ::
 Route:
 Prefix(Intra  Inter  External)ABR      ASBR     External
 4     1     0      3        0        1

 Path:
 Prefix(Intra  Inter  External)ABR      ASBR     External
 4     1     0      3        0        1
```

3.9.15 配置 OSPFV3 路由聚合示例

组网需求

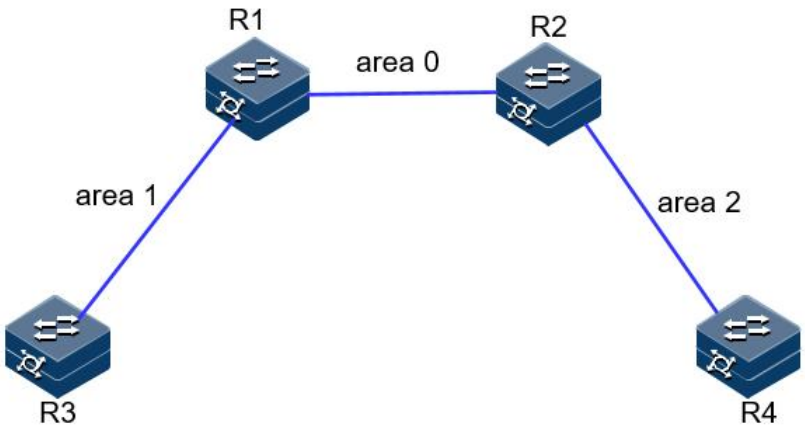
如下图所示，网络要求：区域 2 中存 10::1/128, 10::2/128, 20::1:1/112,20::2:1/112 的区域内路由，希望将 10::1/128, 10::2/128 聚合为 10::0/112 通告；而希望 20::1:1/112 和 20::2:1/112 不导入其他区域。

R4 具有 30::1/128 和 30::2/128 的外部路由，希望将此路由通告给其他区域。

区域 1 的设备能力较差，不能接受大量外部路由，但是 R3 具有 200::1/128 和 200::2/128 的外部路由，希望将此路由通告给其他区域。

根据上述要求，我们可以为区域 2 配置聚合条目和过滤条目，为区域 1 配置 NSSA 属性。

图 3-43 OSPFV3 路由聚合组网图



配置步骤

步骤 1 配置同 OSPFV3 基本功能配置。

步骤 2 配置 area 1 为 nssa 区域。

```
R1：
JX-R1#configure
JX-R1(config)#ospfv3 1
JX-R1(config-ospfv3-1)#area 1 nssa

R3：
JX-R3#configure
JX-R3(config)#ospfv3 1
JX-R3(config-ospfv3-1)#area 1 nssa
```

步骤 3 查看聚合前的路由条目和数据库。

```
JX-R2(config)#show ospfv3 route
 Ospfv3 Process 1
 RoutType  Prefix              AreaId        PathType    Cost    Cost2
NextHopIf   NextHopNbr                  Backu
pNextHop
 ABR      1.1.1.1/32           0.0.0.0       INTRA       1       0
vlan-12     fe80::f2f1:f2ff:fef3:101       ::
 ASBR     1.1.1.1/32           0.0.0.0       INTRA       1       0
vlan-12     fe80::f2f1:f2ff:fef3:101       ::
 ASBR     3.3.3.3/32           0.0.0.0       INTER       2       0
vlan-12     fe80::f2f1:f2ff:fef3:101       ::
 ASBR     4.4.4.4/32           0.0.0.2       INTRA       1       0
vlan-24     fe80::f2f1:f2ff:fef3:101       ::
 PREFIX   10::1/128            0.0.0.2       INTRA       2       0
vlan-24     fe80::f2f1:f2ff:fef3:101       ::
 PREFIX   10::2/128            0.0.0.2       INTRA       2       0
vlan-24     fe80::f2f1:f2ff:fef3:101       ::
 PREFIX   12::/64              0.0.0.0       INTRA       1       0
vlan-12     12::                        ::
```

```
 PREFIX   13::/64              0.0.0.0       INTER      2      0
vlan-12    fe80::f2f1:f2ff:fef3:101       ::
 PREFIX   20::1:0/112           0.0.0.2       INTRA      2      0
vlan-24    fe80::f2f1:f2ff:fef3:101       ::
 PREFIX   20::2:0/112           0.0.0.2       INTRA      2      0
vlan-24    fe80::f2f1:f2ff:fef3:101       ::
 PREFIX   24::/64              0.0.0.2       INTRA      1      0
vlan-24    24::                          ::
 PREFIX   30::1/128             n/a           EXTERNAL_2  1      1
vlan-24    fe80::f2f1:f2ff:fef3:101       ::
 PREFIX   30::2/128             n/a           EXTERNAL_2  1      1
vlan-24    fe80::f2f1:f2ff:fef3:101       ::
 PREFIX   200::1/128            n/a           EXTERNAL_2  1      1
vlan-12    fe80::f2f1:f2ff:fef3:101       ::
 PREFIX   200::2/128            n/a           EXTERNAL_2  1      1
vlan-12    fe80::f2f1:f2ff:fef3:101       ::
 Route:
 Prefix(Intra  Inter  External)ABR      ASBR     External
 11    6     0      4        1        3

 Path:
 Prefix(Intra  Inter  External)ABR      ASBR     External
 11    6     0      4        1        3
```

步骤 4 在 R2 配置 ABR 聚合，将 10::1/128 和 10::2/128 聚合为 10::0/112 通告，20::1:1/112 和 20::2:1/112 汇聚之后不通告。

```
JX-R2#configure
JX-R2(config)#ospfv3 1
JX-R2(config-ospfv3-1)#area 2 abr-summary 10::0/112 advertise
JX-R2(config-ospfv3-1)#area 2 abr-summary 20::0/98 not-advertise
```

步骤 5 验证配置结果

R1 的路由表中仅存在聚合后的 10::/112，无 10::1/128、10::2/128、20::1:1/112 和 20::2:1/112。

```
JX-R1(config)#show ospfv3 route
 Ospfv3 Process 1
 RoutType  Prefix              AreaId        PathType   Cost   Cost2
NextHopIf   NextHopNbr                  Backu
pNextHop
 ABR      2.2.2.2/32           0.0.0.0       INTRA      1      0
vlan-12    fe80::f2f1:f2ff:fef3:201       ::
 ASBR     4.4.4.4/32           0.0.0.0       INTER      2      0
vlan-12    fe80::f2f1:f2ff:fef3:201       ::
 ASBR     3.3.3.3/32           0.0.0.1       INTRA      1      0
vlan-13    fe80::f2f1:f2ff:fef3:301       ::
 PREFIX   10::/112             0.0.0.0       INTER      3      0
vlan-12    fe80::f2f1:f2ff:fef3:201       ::
 PREFIX   12::/64              0.0.0.0       INTRA      1      0
vlan-12    12::                          ::
 PREFIX   13::/64              0.0.0.1       INTRA      1      0
vlan-13    13::                          ::
```

```
 PREFIX    24::/64              0.0.0.0        INTER      2      0
vlan-12     fe80::f2f1:f2ff:fef3:201       ::
 PREFIX    30::1/128            n/a            EXTERNAL_2  2      1
vlan-12     fe80::f2f1:f2ff:fef3:201       ::
 PREFIX    30::2/128            n/a            EXTERNAL_2  2      1
vlan-12     fe80::f2f1:f2ff:fef3:201       ::
 PREFIX    200::1/128           n/a            EXTERNAL_2  1      1
vlan-13     fe80::f2f1:f2ff:fef3:301       ::
 PREFIX    200::2/128           n/a            EXTERNAL_2  1      1
vlan-13     fe80::f2f1:f2ff:fef3:301       ::
 Route:
 Prefix(Intra  Inter  External)ABR      ASBR     External
 8     2     0      4        1        2

 Path:
 Prefix(Intra  Inter  External)ABR      ASBR     External
 8     2     0      4        1        2
```

步骤 6 在 R4 上配置 asbr 聚合，将 30::1/128 和 30::2/128 聚合为 30::0/112

```
JX-R4#configure
JX-R4(config)#ospfv3 1
JX-R4(config-ospfv3-1)#asbr-summary 30::0/112
```

步骤 7 验证配置结果

查看 R2 路由表和数据库，聚合成功

```
JX-R2(config)#show ospfv3 database ls-type external
Database of OSPFv3 Process 1

           AS External Link State
 LinkId          ADV Router       Age     Seq#        CheckSum    Len
Prefix
 0.0.0.2         1.1.1.1          1590    0x80000001  0x5d21      44
200::1/128
 0.0.0.3         1.1.1.1          488     0x80000001  0x6d0f      44
200::2/128
 0.0.0.5         4.4.4.4          23      0x80000001  0x86cb      44
30::/112

JX-R2(config)#show ospfv3 route
 Ospfv3 Process 1
 RoutType  Prefix               AreaId         PathType    Cost    Cost2
NextHopIf   NextHopNbr                    Backu
pNextHop
 ABR      1.1.1.1/32           0.0.0.0        INTRA       1      0
vlan-12     fe80::f2f1:f2ff:fef3:101       ::
 ASBR     1.1.1.1/32           0.0.0.0        INTRA       1      0
vlan-12     fe80::f2f1:f2ff:fef3:101       ::
 ASBR     3.3.3.3/32           0.0.0.0        INTER       2      0
vlan-12     fe80::f2f1:f2ff:fef3:101       ::
 ASBR     4.4.4.4/32           0.0.0.2        INTRA       1      0
vlan-24     fe80::f2f1:f2ff:fef3:101       ::
 PREFIX    10::1/128            0.0.0.2        INTRA       2      0
vlan-24     fe80::f2f1:f2ff:fef3:101       ::
```

```
 PREFIX   10::2/128            0.0.0.2       INTRA      2      0
vlan-24    fe80::f2f1:f2ff:fef3:101      ::
 PREFIX   12::/64              0.0.0.0       INTRA      1      0
vlan-12    12::                        ::
 PREFIX   13::/64              0.0.0.0       INTER      2      0
vlan-12    fe80::f2f1:f2ff:fef3:101      ::
 PREFIX   20::1:0/112           0.0.0.2       INTRA      2      0
vlan-24    fe80::f2f1:f2ff:fef3:101      ::
 PREFIX   20::2:0/112           0.0.0.2       INTRA      2      0
vlan-24    fe80::f2f1:f2ff:fef3:101      ::
 PREFIX   24::/64              0.0.0.2       INTRA      1      0
vlan-24    24::                        ::
 PREFIX   30::/112             n/a           EXTERNAL_2  1      1
vlan-24    fe80::f2f1:f2ff:fef3:101      ::
 PREFIX   200::1/128            n/a           EXTERNAL_2  1      1
vlan-12    fe80::f2f1:f2ff:fef3:101      ::
 PREFIX   200::2/128            n/a           EXTERNAL_2  1      1
vlan-12    fe80::f2f1:f2ff:fef3:101      ::
 Route:
 Prefix(Intra  Inter  External)ABR      ASBR     External
 10    6     0      3        1        3

 Path:
 Prefix(Intra  Inter  External)ABR      ASBR     External
 10    6     0      3        1        3
```

步骤 8 在 R1 上配置 nssa 聚合，将 R3 产生的 NSSA LSA 转换时进行聚合。

```
JX-R1#configure
JX-R1(config)#ospfv3 1
JX-R1(config-ospfv3-1)#area 1 nssa summary 200::/112 advertise
```

步骤 9 检查配置结果

R1 生成的 external-lsa 进行了聚合

```
JX-R2(config)#show ospfv3 route
 Ospfv3 Process 1
 RoutType  Prefix               AreaId        PathType    Cost    Cost2
NextHopIf   NextHopNbr                  Backu
pNextHop
 ABR      1.1.1.1/32            0.0.0.0       INTRA      1      0
vlan-12     fe80::f2f1:f2ff:fef3:101      ::
 ASBR     1.1.1.1/32            0.0.0.0       INTRA      1      0
vlan-12     fe80::f2f1:f2ff:fef3:101      ::
 ASBR     3.3.3.3/32            0.0.0.0       INTER      2      0
vlan-12     fe80::f2f1:f2ff:fef3:101      ::
 ASBR     4.4.4.4/32            0.0.0.2       INTRA      1      0
vlan-24     fe80::f2f1:f2ff:fef3:101      ::
 PREFIX   10::1/128             0.0.0.2       INTRA      2      0
vlan-24     fe80::f2f1:f2ff:fef3:101      ::
 PREFIX   10::2/128             0.0.0.2       INTRA      2      0
vlan-24     fe80::f2f1:f2ff:fef3:101      ::
```

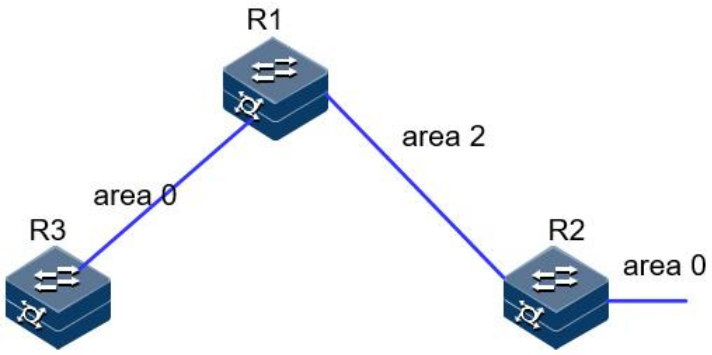
```
 PREFIX    12::/64              0.0.0.0        INTRA      1      0
vlan-12    12::                          ::
 PREFIX    13::/64              0.0.0.0        INTER      2      0
vlan-12    fe80::f2f1:f2ff:fef3:101       ::
 PREFIX    20::1:0/112           0.0.0.2        INTRA      2      0
vlan-24    fe80::f2f1:f2ff:fef3:101       ::
 PREFIX    20::2:0/112           0.0.0.2        INTRA      2      0
vlan-24    fe80::f2f1:f2ff:fef3:101       ::
 PREFIX    24::/64              0.0.0.2        INTRA      1      0
vlan-24    24::                          ::
 PREFIX    30::/112             n/a            EXTERNAL_2  1      1
vlan-24    fe80::f2f1:f2ff:fef3:101       ::
 PREFIX    200::/112            n/a            EXTERNAL_1  3      0
vlan-12    fe80::f2f1:f2ff:fef3:101       ::
 Route:
 Prefix(Intra  Inter  External)ABR      ASBR     External
 9     6     0      2        1        3

 Path:
 Prefix(Intra  Inter  External)ABR      ASBR     External
 9     6     0      2        1        3

JX-R2(config)#show ospfv3 database ls-type external
Database of OSPFv3 Process 1

          AS External Link State
 LinkId          ADV Router       Age      Seq#        CheckSum    Len
Prefix
 0.0.0.4         1.1.1.1          22      0x80000001   0xb8d7      44
200::/112
 0.0.0.5         4.4.4.4          180     0x80000001   0x86cb      44
30::/112
```

3.9.16 配置 OSPFV3 虚连接示例

组网需求

如下图所示，R3 位于区域 0；R1 连接区域 0 和区域 2，而 R2 连接区域 0 和区域 2，此时区域 0 被分割，区域 0 内无法学习到区域 2 的内部路由；区域 2 也无法学习到区域 0 的内部路由和其他区域的路由。此时需要在 R1 和 R4 间配置虚链路，连接 0 域。

图 3-44 配置 OSPFV3 虚连接的组网图



配置步骤

步骤 1 配置各接口的 IP 地址。

R1 和 R2 建邻接口使用 vlan12,R1 和 R3 建邻接口使用 vlan 13。R2 加一个环回口 1,地址 22::2/128，R3 增加环回口 1，地址 33::3/128。

步骤 2 配置 OSPF。

R1：

```
JX-R1#configure
JX-R1(config)#ospfv3 1
JX-R1(config-ospfv3-1)#router-id 1.1.1.1
JX-R1(config-ospfv3-1)#exit
JX-R1(config)#interface vlan 13
JX-R1(config-vlanif-13)#ospfv3 1 area 0
JX-R1(config-vlanif-13)#exit
JX-R1(config)#interface vlan 12
JX-R1(config-vlanif-12)#ospfv3 1 area 2
JX-R1(config-vlanif-12)#exit
```

R2：

```
JX-R2#configure
JX-R2(config)#ospfv3 1
JX-R2(config-ospfv3-1)#router-id 2.2.2.2
JX-R2(config-ospfv3-1)#exit
JX-R2(config)#interface vlan 12
JX-R2(config-vlanif-12)#ospfv3 1 area 2
JX-R2(config-vlanif-12)#exit
JX-R2(config)#interface loopback 1
JX-R2(config-loopback-1)#ospfv3 1 area 0
```

R3：

```
JX-R3#configure
JX-R3(config)#ospfv3 1
JX-R3(config-ospfv3-1)#router-id 3.3.3.3
JX-R3(config-ospfv3-1)#exit
JX-R3(config)#interface vlan 13
```

```
JX-R3(config-vlanif-13)#ospfv3 1 area 0
JX-R3(config-vlanif-13)#exit
JX-R3(config)#interface loopback 1
JX-R3(config-loopback-1)#ospfv3 1 area 0
```

步骤 3 检查配置结果。

```
JX-R1(config)#show ospfv3 neighbor

 Ospfv3 Process 1
 NeighborId        Priority  State    Interface     Instance  Aging
UpTime   IpAddress
--------------------------------------------------------------------------
-----------------------------------
 2.2.2.2           1       Full     vlan-12       0        34       0:01:52
fe80::f2f1:f2ff:fef3:201
 3.3.3.3           1       Full     vlan-13       0        29       0:01:40
fe80::f2f1:f2ff:fef3:301
--------------------------------------------------------------------------
-----------------------------------
  Down:0       Attempt:0       Init:0       TwoWay:0       ExchangeStart:0
Exchange:0
  Loading:0       Full:2
```

R2 上无法学习到 33::3 的路由，R1、R3 也无法学习到 22::2 的路由

```
JX-R1(config)#show ospfv3 route
 Ospfv3 Process 1
 RoutType  Prefix               AreaId         PathType    Cost    Cost2
NextHopIf   NextHopNbr                   Backu
pNextHop
 ABR      2.2.2.2/32            0.0.0.2        INTRA       1       0
vlan-12     fe80::f2f1:f2ff:fef3:201       ::
 PREFIX    12::/64               0.0.0.2        INTRA       1       0
vlan-12     12::                          ::
 PREFIX    13::/64               0.0.0.0        INTRA       1       0
vlan-13     13::                          ::
 PREFIX    33::3/128              0.0.0.0        INTRA       2       0
vlan-13     fe80::f2f1:f2ff:fef3:301       ::
 Route:
 Prefix(Intra  Inter  External)ABR      ASBR     External
 3     3     0      0       1       0

 Path:
 Prefix(Intra  Inter  External)ABR      ASBR     External
 3     3     0      0       1       0

JX-R2(config)#show ospfv3 route
 Ospfv3 Process 1
 RoutType  Prefix               AreaId         PathType    Cost    Cost2
NextHopIf   NextHopNbr                   Backu
pNextHop
 ABR      1.1.1.1/32            0.0.0.2        INTRA       1       0
vlan-12     fe80::f2f1:f2ff:fef3:101       ::
 PREFIX    12::/64               0.0.0.2        INTRA       1       0
vlan-12     12::                          ::
```

```
 PREFIX   22::2/128            0.0.0.0       INTRA      1      0
loopback-1  22::2                      ::
 Route:
 Prefix(Intra  Inter  External)ABR      ASBR     External
 2     2     0      0       1       0

 Path:
 Prefix(Intra  Inter  External)ABR      ASBR     External
 2     2     0      0       1       0
```

步骤 4 配置虚连接。

R1：

```
JX-R1#configure
JX-R1(config)#ospfv3 1
JX-R1(config-ospfv3-1)#area 2 virtual-link 2.2.2.2
```

R2：

```
JX-R2#configure
JX-R2(config)#ospfv3 1
JX-R2(config-ospfv3-1)#area 2 virtual-link 1.1.1.1
```

步骤 5 检查配置结果。

虚连接邻居正常建立

```
JX-R1(config)#show ospfv3 virtual-link
 Ospfv3 Process 1
 Area Id                 :0.0.0.2
 Neighbor Id              :2.2.2.2
 Local Interface           :2.2.2.2
 State                   :pointToPoint
 Event Change             :1
 Hello Interval           :10(s)
 Retransmit Interval       :5(s)
 Transmit Delay           :1(s)
 Dead Interval            :60(s)
 Link Lsa Count           :0
 Link LSA CksumSum         :0
 Neighbor Prefix          :12::2
 Neighbor State           :Full
 Neighbor Event Change     :4
```

路由表学习正常

```
JX-R1(config)#show ospfv3 route
 Ospfv3 Process 1
 RoutType  Prefix              AreaId        PathType   Cost   Cost2
NextHopIf   NextHopNbr                  Backu
pNextHop
 ABR      2.2.2.2/32           0.0.0.2       INTRA      1      0
vlan-12    fe80::f2f1:f2ff:fef3:201      ::
 PREFIX   12::/64              0.0.0.2       INTRA      1      0
vlan-12    12::                       ::
```

```
 PREFIX   12::1/128            0.0.0.2       INTRA      0      0
vlan-12    12::1                      ::
 PREFIX   12::2/128            0.0.0.2       INTRA      1      0
vlan-12    fe80::f2f1:f2ff:fef3:201      ::
 PREFIX   13::/64              0.0.0.0       INTRA      1      0
vlan-13    13::                       ::
 PREFIX   22::2/128            0.0.0.0       INTRA      2      0
vlan-12    fe80::f2f1:f2ff:fef3:201      ::
 PREFIX   33::3/128            0.0.0.0       INTRA      2      0
vlan-13    fe80::f2f1:f2ff:fef3:301      ::
 Route:
 Prefix(Intra  Inter  External)ABR      ASBR     External
 6     6     0      0       2        0

 Path:
 Prefix(Intra  Inter  External)ABR      ASBR     External
 6     6     0      0       1        0

JX-R2(config)#show ospfv3 route
 Ospfv3 Process 1
 RoutType  Prefix              AreaId        PathType    Cost    Cost2
NextHopIf   NextHopNbr                  Backu
pNextHop
 ABR      1.1.1.1/32           0.0.0.2       INTRA      1      0
vlan-12    fe80::f2f1:f2ff:fef3:101      ::
 PREFIX   12::/64              0.0.0.2       INTRA      1      0
vlan-12    12::                       ::
 PREFIX   12::1/128            0.0.0.2       INTRA      1      0
vlan-12    fe80::f2f1:f2ff:fef3:101      ::
 PREFIX   12::2/128            0.0.0.2       INTRA      0      0
vlan-12    12::2                      ::
 PREFIX   13::/64              0.0.0.0       INTRA      2      0
vlan-12    fe80::f2f1:f2ff:fef3:101      ::
 PREFIX   22::2/128            0.0.0.0       INTRA      1      0
loopback-1  22::2                     ::
 PREFIX   33::3/128            0.0.0.0       INTRA      3      0
vlan-12    fe80::f2f1:f2ff:fef3:101      ::
 Route:
 Prefix(Intra  Inter  External)ABR      ASBR     External
 6     6     0      0       2        0

 Path:
 Prefix(Intra  Inter  External)ABR      ASBR     External
 6     6     0      0       1        0
```

3.10 IS-IS

3.10.1 简介

基本概念

IS-IS (Intermediate System-to-Intermediate System intra-domain routing information exchange protocol, 中间系统到中间系统的域内路由信息交换协议)最初是国际标准化组织(International Organization for Standardization, ISO) 为它的无连接网络协议 (Connection Less Network Protocol, CLNP) 设计的 一种动态路由协议。 IS-IS 属于内部网关协议 (Interior Gateway Protocol, IGP 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 (Shortest Path First, SPF) 算法进行路由计算。IS-IS 可支持大中型网络，而且路由收敛速度快，因此可以作为除 OSPF 协议外的另一选择。20 世纪 90 年代中期，一些运营商选择 IS-IS 作为其网络的 IGP 路由协议，主要是因为集成 IS-IS 能够同时支持 CLNP 和 IP, 易于从早期的 OSI 网络平滑过渡到 IP 网络。

IS-IS 的几个基本术语如下：

- IS（Intermediate System），中间系统。相当于 TCP/IP 中的路由器，是 IS-IS 协议中生成路由和传播路由信息的基本单元。在下文中 IS 和路由器具有相同的含义。
- RD（Routing Domain），路由域。在一个路由域中一群 IS 通过相同的路由协议来交换路由信息。
- Area，区域，路由域的细分单元，IS-IS 允许将整个路由域分为多个区域。
- LSDB（Link State Database），链路状态数据库。所有的网络内连接状态组成了链路状态数据库，在每一个 IS 中都至少有一个 LSDB。IS 使用 SPF 算法，利用 LSDB 来生成自己的路由。
- LSP（Link State Protocol Data Unit），链路状态报文。在 IS-IS 中，每一个 IS 都会生成至少一个 LSP，这些 LSP 包含了本 IS 的所有链路状态信息。每个 IS 收集本区域内所有的 LSP 与自己本地生成的 LSP 构成自己的 LSDB。
- Level-1 路由， Level-1 路由存在于同一区域内的不同 IS 之间，所以又称为区域内路由。当 IS 要发送报文到另外一个 IS 时，查看报文中的目的地址，如果发现其位于区域内的不同子网，则 IS 会选择最优的路径进行转发；如果目的地址不在同一区域，则 IS 把数据转发到本区域内最近的 Level-1-2 路由器上，然后由 Level-1-2 路由器负责数据转发。
- Level-2 路由，Level-2 路由存在于同一路由域内的区域间，所以又称为区域间路由。当目的地址在不同区域时， IS 发送报文到最近的一个 Level-2 IS, 由 Level-2 IS 负责将其转发到另一个区域。
- Level-1 路由器 (Ll 路由器）， Level-1 路由器负责区域内的路由，并且只维护一个 Level-1 LSDB 。该 LSDB 包含本区域的路由信息 到区域外的报文转发给最近的 Level-1-2 路由器。
- Level-2 路由器(L2 路由器），Level-2 路由器负责区域间的路由，并且只维护一个 Level-2 LSDB 。该 LSDB 包含区域间的路由信息。所有 Level-2 路由器和 Level-1-2 路由器组成路由域的骨干网，负责在不同区域间通信，骨干网必须是物理连续的。

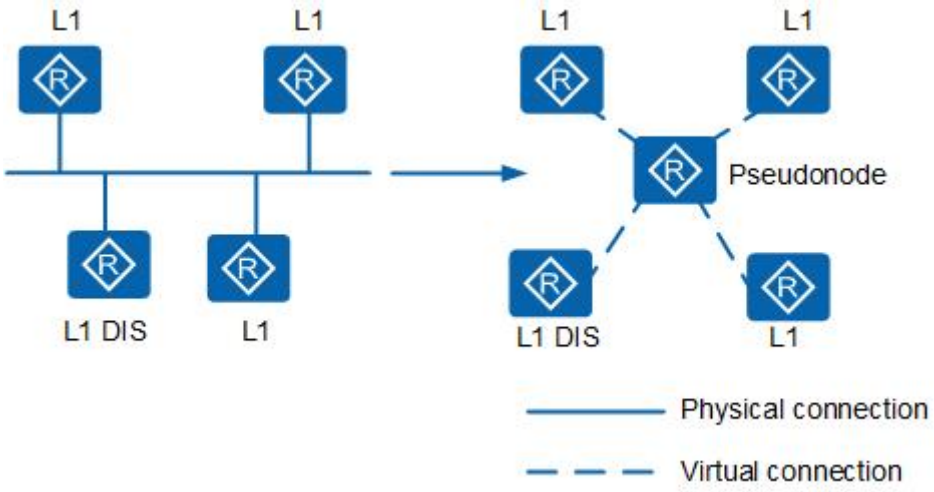
- Level-1-2 路由器（Ll/L2 路由器）。同时属于 Level-1 和 Level-2 的路由器称为 Level-1-2 路由器，Level-1-2 路由器维护两个 LSDB, Level-1 LSDB 用于区域内路由， Level-2 的 LSDB 用于区域间路由。

DIS 和伪节点

在广播网络中，IS-IS 需要在所有的路由器中选举一个路由器作为 DIS（Designated Intermediate System）。DIS 用来创建和更新伪节点（Pseudo nodes），并负责生成伪节点的链路状态协议数据单元 LSP（Link state Protocol Data Unit），用来描述这个网络上有哪些网络设备。

伪节点是用来模拟广播网络的一个虚拟节点，并非真实的路由器。在 IS-IS 中，伪节点用 DIS 的 System ID 和一个字节的 Circuit ID（非 0 值）标识。

- **伪节点示意图**



使用伪节点可以简化网络拓扑，使路由器产生的 LSP 长度较小。另外，当网络发生变化时，需要产生的 LSP 数量也会较少，减少 SPF 的资源消耗。

Level-1 和 Level-2 的 DIS 是分别选举的，用户可以为不同级别的 DIS 选举设置不同的优先级。DIS 优先级数值最大的被选为 DIS。如果优先级数值最大的路由器有多台，则其中 MAC 地址最大的路由器会被选中。不同级别的 DIS 可以是同一台路由器，也可以是不同的路由器。

IS-IS 协议中 DIS 与 OSPF 协议中 DR（Designated Router）的区别：

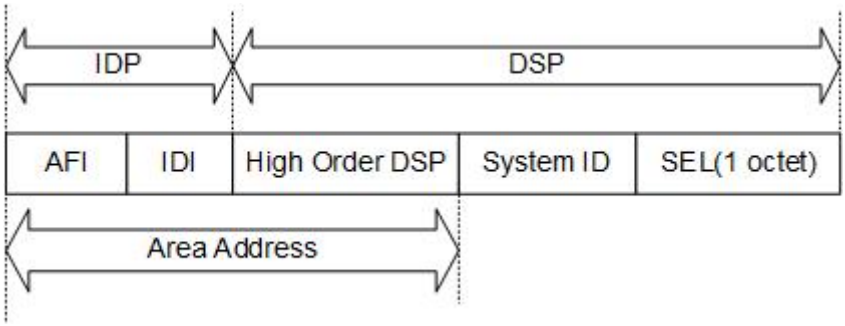
- 在 IS-IS 广播网中，优先级为 0 的路由器也参与 DIS 的选举，而在 OSPF 中优先级为 0 的路由器则不参与 DR 的选举。
- 在 IS-IS 广播网中，当有新的路由器加入，并符合成为 DIS 的条件时，这个路由器会被选中成为新的 DIS，原有的伪节点被删除。此更改会引起一组新的 LSP 泛洪。而在 OSPF 中，当一台新路由器加入后，即使它的 DR 优先级值最大，也不会立即成为该网段中的 DR。

- 在 IS-IS 广播网中，同一网段上的同一级别的路由器之间都会形成邻接关系，包括所有的非 DIS 路由器之间也会形成邻接关系。而在 OSPF 中，路由器只与 DR 和 BDR 建立邻接关系。

IS-IS 的地址结构

网络服务访问点 NSAP（Network Service Access Point）是 OSI 协议中用于定位资源的地址。NSAP 的地址结构如图 4 所示，它由 IDP（Initial Domain Part）和 DSP（Domain Specific Part）组成。IDP 和 DSP 的长度都是可变的，NSAP 总长最多是 20 个字节，最少 8 个字节。•IDP 相当于 IP 地址中的主网络号。它是由 ISO 规定，并由 AFI（Authority and Format Identifier）与 IDI（Initial Domain Identifier）两部分组成。AFI 表示地址分配机构和地址格式，IDI 用来标识域。DSP 相当于 IP 地址中的子网号和主机地址。它由 High Order DSP、System ID 和 SEL 三个部分组成。High Order DSP 用来分割区域，System ID 用来区分主机，SEL（NSAP Selector）用来指示服务类型。

- IS-IS 协议的地址结构示意图



- Area Address：IDP 和 DSP 中的 High Order DSP 一起，既能够标识路由域，也能够标识路由域中的区域，因此，它们一起被称为区域地址（Area Address），相当于 OSPF 中的区域编号。同一 Level-1 区域内的所有路由器必须具有相同的区域地址，Level-2 区域内的路由器可以具有不同的区域地址。一般情况下，一个路由器只需要配置一个区域地址，且同一区域中所有节点的区域地址都要相同。为了支持区域的平滑合并、分割及转换，在设备的实现中，一个 IS-IS 进程下最多可配置 3 个区域地址。
- System ID：System ID 用来在区域内唯一标识主机或路由器。在设备的实现中，它的长度固定为 48bit（6 字节）。

在实际应用中，一般使用 Router ID 与 System ID 进行对应。假设一台路由器使用接口 Loopback0 的 IP 地址 192.168.1.1 作为 Router ID，则它在 IS-IS 中使用的 System ID 可通过如下方法转换得到.

- 将 IP 地址 192.168.1.1 的每个十进制数都扩展为 3 位，不足 3 位的在前面补 0，得到 192.168.001.001
- 将扩展后的地址分为 3 部分，每部分由 4 位数字组成，得到 1921.6800.1001。重新组合的 1921.6800.1001 就是 System ID。
- 实际 System ID 的指定可以有不同的方法，但要保证能够唯一标识主机或路由器。
- SEL：SEL 的作用类似 IP 中的“协议标识符”，不同的传输协议对应不同的 SEL。在 IP 上 SEL 均为 00。

网络实体名称 NET（Network Entity Title）指的是设备本身的网络层信息，可以看作是一类特殊的 NSAP（SEL＝00）。NET 的长度与 NSAP 的相同，最多为 20 个字节，最少为 8 个字节。在路由器上配置 IS-IS 时，只需要考虑 NET 即可，NSAP 可不必去关注。例如有 NET 为：ab.cdef.1234.5678.9abc.00，则其中 Area Address 为 ab.cdef，System ID 为 1234.5678.9abc，SEL 为 00。

IS-IS 的报文类型

IS-IS 报文有以下几种类型：HELLO PDU（Protocol Data Unit）、LSP 和 SNP。

- Hello PDU

Hello 报文用于建立和维持邻居关系，也称为 IIH（IS-to-IS Hello PDUs）。其中，广播网中的 Level-1 IS-IS 使用 Level-1 LAN IIH；广播网中的 Level-2 IS-IS 使用 Level-2 LAN IIH；非广播网络中则使用 P2P IIH。它们的报文格式有所不同。P2P IIH 中相对于 LAN IIH 来说，多了一个表示本地链路 ID 的 Local Circuit ID 字段，缺少了表示广播网中 DIS 的优先级的 Priority 字段以及表示 DIS 和伪节点 System ID 的 LAN ID 字段。

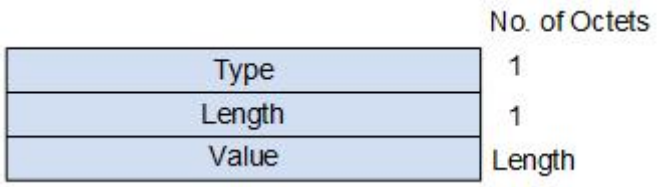
- LSP

链路状态报文 LSP（Link State PDUs）用于交换链路状态信息。LSP 分为两种：Level-1 LSP 和 Level-2 LSP。Level-1 LSP 由 Level-1 IS-IS 传送，Level-2 LSP 由 Level-2 IS-IS 传送，Level-1-2 IS-IS 则可传送以上两种 LSP。

LSP 报文中主要字段的解释如下：

- ATT 字段：当 Level-1-2 IS-IS 在 Level-1 区域内传送 Level-1 LSP 时，如果 Level-1 LSP 中设置了 ATT 位，则表示该区域中的 Level-1 IS-IS 可以通过此 Level-1-2 IS-IS 通往外部区域。
- OL（LSDB Overload）字段：过载标志位。设置了过载标志位的 LSP 虽然还会在网络中扩散，但是在计算通过过载路由器的路由时不会被采用。即对路由器设置过载位后，其它路由器在进行 SPF 计算时不会使用这台路由器做转发，只计算该节点上的直连路由。更多内容请参见《原理描述-IS-IS 过载位》。
- IS Type 字段：用来指明生成此 LSP 的 IS-IS 类型是 Level-1 还是 Level-2 IS-IS（01 表示 Level-1，11 表示 Level-2）。
- SNP：序列号报文 SNP（Sequence Number PDUs）通过描述全部或部分数据库中的 LSP 来同步各 LSDB（Link-State DataBase），从而维护 LSDB 的完整与同步。SNP 包括全序列号报文 CSNP（Complete SNP）和部分序列号报文 PSNP（Partial SNP），进一步又可分为 Level-1 CSNP、Level-2 CSNP、Level-1 PSNP 和 Level-2 PSNP。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。在广播网络上，CSNP 由 DIS 定期发送（缺省的发送周期为 10 秒）；在点到点链路上，CSNP 只在第一次建立邻接关系时发送。PSNP 只列举最近收到的一个或多个 LSP 的序号，它能够一次对多个 LSP 进行确认，当发现 LSDB 不同步时，也用 PSNP 来请求邻居发送新的 LSP。IS-IS 报文中的变长字段部分是多个 TLV（Type-Length-Value）三元组。其格式如图所示。TLV 也称为 CLV（Code-Length-Value）。

- 不同 PDU 类型所包含的 TLV 是不同的

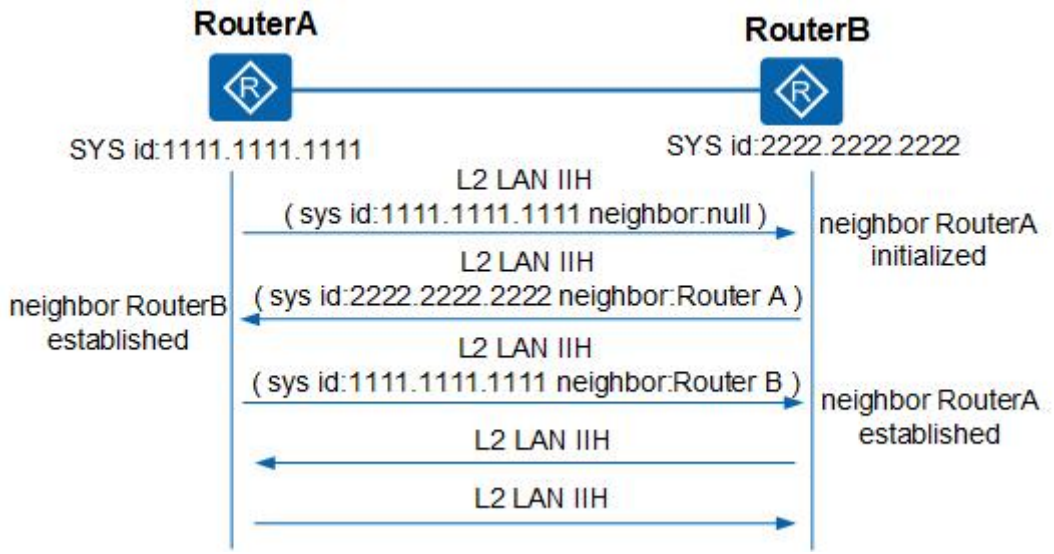


TLV Type	名称	所应用的 PDU 类型
1	Area Addresses	IIH、LSP
2	IS Neighbors（LSP）	LSP
4	Partition Designated Level2 IS	L2 LSP
6	IS Neighbors（MAC Address）	LAN IIH
7	IS Neighbors（SNPA Address）	LAN IIH
8	Padding	IIH
9	LSP Entries	SNP
10	Authentication Information	IIH、LSP、SNP
128	IP Internal Reachability Information	LSP
129	Protocols Supported	IIH、LSP
130	IP External Reachability Information	L2 LSP
131	Inter-Domain Routing Protocol Information	L2 LSP
132	IP Interface Address	IIH、LSP

IS-IS 邻居关系的建立

IS-IS 是一种链路状态路由协议，每一台路由器都会生成一个 LSP，它包含了该路由器所有使能 IS-IS 协议接口的链路状态信息。通过跟相邻设备建立 IS-IS 邻接关系，互相更新本地设备的 LSDB，可以使得 LSDB 与整个 IS-IS 网络的其他设备的 LSDB 实现同步。然后根据 LSDB 运用 SPF 算法计算出 IS-IS 路由。如果此 IS-IS 路由是到目的地址的最优路由，则此路由会下发到 IP 路由表中，并指导报文的转发。两台运行 IS-IS 的路由器在交互协议报文实现路由功能之前必须首先建立邻居关系。在不同类型的网络上，IS-IS 的邻居建立方式并不相同。广播链路邻居关系的建立

- **广播链路邻居关系建立过程**



- RouterA 广播发送 Level-2 LAN IIH，此报文中无邻居标识。
- RouterB 收到此报文后，将自己和 RouterA 的邻居状态标识为 Initial。然后，RouterB 再向 RouterA 回复 Level-2 LAN IIH，此报文中标识 RouterA 为 RouterB 的邻居。
- RouterA 收到此报文后，将自己与 RouterB 的邻居状态标识为 Up。然后 RouterA 再向 RouterB 发送一个标识 RouterB 为 RouterA 邻居的 Level-2 LAN IIH。
- RouterB 收到此报文后，将自己与 RouterA 的邻居状态标识为 Up。这样，两个路由器成功建立了邻居关系。

因为是广播网络，需要选举 DIS，所以在邻居关系建立后，路由器会等待两个 Hello 报文间隔，再进行 DIS 的选举。Hello 报文中包含 Priority 字段，Priority 值最大的将被选举为该广播网的 DIS。若优先级相同，接口 MAC 地址较大的被选举为 DIS。

- P2P 链路邻居关系的建立
- 在 P2P 链路上，邻居关系的建立不同于广播链路。分为两次握手机制和三次握手机制。
- 两次握手机制：只要路由器收到对端发来的 Hello 报文，就单方面宣布邻居为 Up 状态，建立邻居关系。
- 三次握手机制：此方式通过三次发送 P2P 的 IS-IS Hello PDU 最终建立起邻居关系，类似广播邻居关系的建立。

IS-IS 按如下原则建立邻居关系：

- 只有同一层次的相邻路由器才有可能成为邻居。
- 对于 Level-1 路由器来说，区域号必须一致。
- 链路两端 IS-IS 接口的网络类型必须一致。
- 链路两端 IS-IS 接口的地址必须处于同一网段。

IS-IS 的 LSP 交互过程

LSP 产生的原因

IS-IS 路由域内的所有路由器都会产生 LSP，以下事件会触发一个新的 LSP：

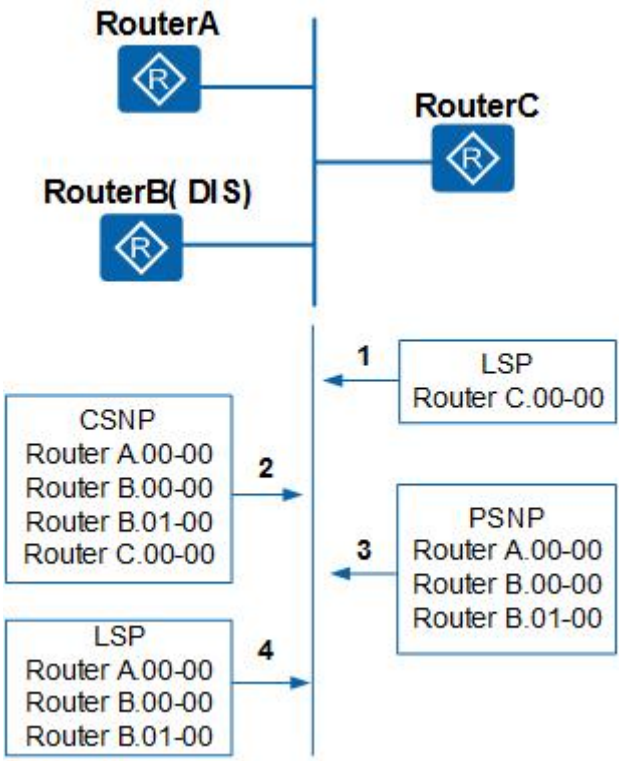
- 邻居 Up 或 Down
- IS-IS 相关接口 Up 或 Down
- 引入的 IP 路由发生变化
- 区域间的 IP 路由发生变化
- 接口被赋了新的 metric 值
- 周期性更新
- 收到邻居新的 LSP 的处理过程
- 将接收的新的 LSP 合入到自己的 LSDB 数据库中，并标记为 flooding。
- 发送新的 LSP 到除了收到该 LSP 的接口之外的接口。
- 邻居再扩散到其他邻居。

LSP 的“泛洪”

LSP 报文的“泛洪”（flooding）是指当一个路由器向相邻路由器通告自己的 LSP 后，相邻路由器再将同样的 LSP 报文传送到除发送该 LSP 的路由器外的其它邻居，并这样逐级将 LSP 传送到整个层次内所有路由器的一种方式。通过这种“泛洪”，整个层次内的每一个路由器就都可以拥有相同的 LSP 信息，并保持 LSDB 的同步。

每一个 LSP 都拥有一个标识自己的 4 字节的序列号。在路由器启动时所发送的第一个 LSP 报文中的序列号为 1，以后当需要生成新的 LSP 时，新 LSP 的序列号在前一个 LSP 序列号的基础上加 1。更高的序列号意味着更新的 LSP。

图 3-45 P2P 链路上 LSDB 数据库的同步过程



- 如图所示，新加入的路由器 RouterC 首先发送 Hello 报文，与该广播域中的路由器建立邻居关系。
- 建立邻居关系之后，RouterC 等待 LSP 刷新定时器超时，然后将自己的 LSP 发往组播地址（Level-1：01-80-C2-00-00-14；Level-2：01-80-C2-00-00-15）。这样网络上所有的邻居都将收到该 LSP。
- 该网段中的 DIS 会把收到 RouterC 的 LSP 加入到 LSDB 中，并等待 CSNP 报文定时器超时并发送 CSNP 报文，进行该网络内的 LSDB 同步。
- RouterC 收到 DIS 发来的 CSNP 报文，对比自己的 LSDB 数据库，然后向 DIS 发送 PSNP 报文请求自己没有的 LSP。
- DIS 收到该 PSNP 报文请求后向 RouterC 发送对应的 LSP 进行 LSDB 的同步。

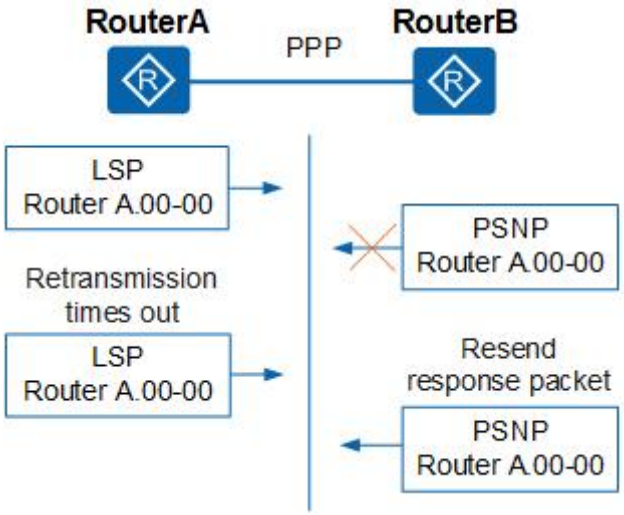
在上述过程中 DIS 的 LSDB 更新过程如下：

- DIS 接收到 LSP，在数据库中搜索对应的记录。若没有该 LSP，则将其加入数据库，并广播新数据库内容。
- 若收到的 LSP 序列号大于本地 LSP 的序列号，就替换为新报文，并广播新数据库内容；若收到的 LSP 序列号小本地 LSP 的序列号，就向入端接口发送本地 LSP 报文。
- 若两个序列号相等，则比较 Remaining Lifetime。若收到的 LSP 的 Remaining Lifetime 小于本地 LSP 的 Remaining Lifetime，就替换为新报文，并广播新数据库内容；若收到的 LSP 的 Remaining Lifetime 大于本地 LSP 的 Remaining Lifetime，就向入端接口发送本地 LSP 报文。
- 若两个序列号和 Remaining Lifetime 都相等，则比较 Checksum。若收到的 LSP 的 Checksum 大于本地 LSP 的 Checksum，就替换为新报文，并广播新数据库内容；

若收到的 LSP 的 Checksum 小于本地 LSP 的 Checksum，就向入端接口发送本地 LSP 报文。

- 若两个序列号、Remaining Lifetime 和 Checksum 都相等，则不转发该报文。

图 3-46 P2P 链路上 LSDB 数据库的同步过程



- RouterA 先与 RouterB 建立邻居关系。
- 建立邻居关系之后，RouterA 与 RouterB 会先发送 CSNP 给对端设备。如果对端的 LSDB 与 CSNP 没有同步，则发送 PSNP 请求索取相应的 LSP。
- 如图 3 所示假定 RouterB 向 RouterA 索取相应的 LSP。RouterA 发送 RouterB 请求的 LSP 的同时启动 LSP 重传定时器，并等待 RouterB 发送的 PSNP 作为收到 LSP 的确认。
- 如果在接口 LSP 重传定时器超时后，RouterA 还没有收到 RouterB 发送的 PSNP 报文作为应答，则重新发送该 LSP 直至收到 PSNP 报文。

在 P2P 链路中设备的 LSDB 更新过程如下：

- 若收到的 LSP 比本地的序列号更小，则直接给对方发送本地的 LSP，然后等待对方给自己一个 PSNP 报文作为确认；若收到的 LSP 比本地的序列号更大，则将这个新的 LSP 存入自己的 LSDB，再通过一个 PSNP 报文来确认收到此 LSP，最后再将这个新 LSP 发送给除了发送该 LSP 的邻居以外的邻居。
- 若收到的 LSP 序列号和本地相同，则比较 Remaining Lifetime，若收到 LSP 的 Remaining Lifetime 小于本地 LSP 的 Remaining Lifetime，则将收到的 LSP 存入 LSDB 中并发送 PSNP 报文来确认收到此 LSP，然后将该 LSP 发送给除了发送该 LSP 的邻居以外的邻居；若收到 LSP 的 Remaining Lifetime 大于本地 LSP 的 Remaining Lifetime，则直接给对方发送本地的 LSP，然后等待对方给自己一个 PSNP 报文作为确认。
- 若收到的 LSP 和本地 LSP 的序列号相同且 Remaining Lifetime 都不为 0，则比较 Checksum，若收到 LSP 的 Checksum 大于本地 LSP 的 Checksum，则将收到的 LSP 存入 LSDB 中并发送 PSNP 报文来确认收到此 LSP，然后将该 LSP 发送给除了发送该 LSP 的邻居以外的邻居；若收到 LSP 的 Checksum 小于本地 LSP 的

Checksum，则直接给对方发送本地的 LSP，然后等待对方给自己一个 PSNP 报文作为确认。

- 若收到的 LSP 和本地 LSP 的序列号、Remaining Lifetime 和 Checksum 都相同，则不转发该报文。

3.10.2 配置准备

场景

IS-IS 属于内部网关 IGP。用于自治系统内部。IS-IS 也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。

前提

相邻节点的网络层可达。

3.10.3 缺省配置

设备上 IS-IS 的缺省配置如下。

功能	缺省值
IS-IS 全局功能状态	禁用
DIS 优先级	64
设备 level 级别	level-1-2
发送 Hello 报文时间间隔	10s
LSP 刷新时间间隔	900s
LSP 最大有效时间	1200s

3.10.4 配置 IS-IS 基本功能

请在设备上进行以下配置。

步骤	配置	说明
1	`JX#config`	进入全局配置模式。
2	`JX(config)#isis` *`Process id`*	创建 IS-IS 进程并进入 IS-IS 视图
3	`JX(config-isis-1)#network-entity` *`net`*	在进入 IS-IS 视图之后，必须完成 IS-IS 进程的 NET 配置，IS-IS 协议才能真正启动

步骤	配置	说明
4	JX(config-isis-1)#**is-type** { **level-1** | **level-1-2** | **level-2** }	当 Level 级别为 Level-2 时，设备可以与同一或者不同区域的 Level-2 设备或者其它区域的 Level-1-2 设备形成邻居关系，并且只维护一个 Level-2 的 LSDB。 同时属于 Level-1 和 Level-2 的路由器称为 Level-1-2 路由器，它可以与同一区域的 Level-1 和 Level-1-2 路由器形成 Level-1 邻居关系，也可以与其他区域的 Level-2 和 Level-1-2 路由器形成 Level-2 的邻居关系。Level-1 路由器必须通过 Level-1-2 路由器才能连接至其他区域。Level-1-2 路由器维护两个 LSDB，Level-1 的 LSDB 用于区域内路由，Level-2 的 LSDB 用于区域间路由。
5	JX(config-vlanif-1)#**isis enable** [*instance-id*]	使能 IS-IS 接口
6	1. JX (config-vlanif-1)#**isis circuit-level** { **level-1** | **level-2** | **level-1-2** | **default** }	配置接口的 level 级别。
7	2. JX (config-vlanif-1)#**isis dis-priority** { *priority-value* | **default** } [**level-1** | **level-2**]	指定路由器的优先级数值
8	3. JX (config-vlanif-1)#**isis circuit-type** { **broadcast** | **p2p** }	配置 IS-IS 链路类型，默认链路类型为广播类型

3.10.5 配置 IS-IS 网络的安全性

请在设备上进行以下配置。

步骤	配置	说明
1	4. JX(config-isis-1)#**area-authentication-mode** { **simple** | **md5** } { **cipher** | **plain** } **password [snp-packet** { **authentication-avoid** | **send-only** } | **all-send-only]**	配置 IS-IS 区域认证密码，并设置 ISIS 区域按照预定的方式和密码验证收到的 Level-1 路由信息报文（LSP 和 SNP），也可以用来为发送的 Level-1 报文加上认证信息。
2	5. JX(config-isis-1)#**domain-authentication-mode** { **simple** | **md5** } { **cipher** | **plain** } **password [snp-packet** { **authentication-avoid** | **send-only** } | **all-send-only]**	配置 IS-IS 路由域认证密码，并设置 IS-IS 路由域按照预设的方式和密码验证收到的 Level-2 路由信息报文（LSP 和 SNP）。

步骤	配置	说明
3	6. JX(config-vlanif-1)#**isis authentication-mode** { **simple** | **md5** } { **cipher** | **plain** } **password** { **level-1** | **level-2** | **p2p** }	设置 IS-IS 接口以指定的方式和密码验证 Hello 报文，并在发送的 Hello 报文中添加认证信息。

3.10.6 配置 IS-IS 的选路

请在设备上进行以下配置。

步骤	配置	说明
1	7. JX(config-isis-1)#**preference** { *priority* | **default** }	配置 IS-IS 协议路由的优先级。
2	8. JX(config-isis-1)#**cost-style** { **narrow** | **wide** | **compatible** | **narrow-compatible** | **wide-compatible** | **default** } { **level-1** | **level-2** }	可以接收开销类型为 narrow 和 wide 的路由，但却只发送 narrow 的路由。
3	9. JX(config-isis-1)#**maximum load-balancing** { *load-balancing-number* | **default** }	设置形成负载分担的等价路由的最大条数。
4	**10.** JX(config-isis-1)#**import-route level-2 to level-1**	将 Level-2 区域的路由渗透到本地 Level-1 区域。缺省情况下，Level-2 区域的路由信息不渗透到 Level-1 区域。

3.10.7 配置 IS-IS 的路由信息的交互

请在设备上进行以下配置。

步骤	配置	说明
1	11. JX(config-isis-1)#**default-route-originate** { **none** | **level-1** | **level-2** | **level-1-2** }	设置 IS-IS 路由器生成缺省路由。
2	12. JX(config-isis-1)#**import-route** { **connect** | **static** | **rip** | **ospf** | **bgp** | **isis** } { **level-1** | **level-2** | **level-1-2** } [**cost** { **cost-value** | **default** | **inherit** } | **route-policy policy-name**]*	引入其它路由信息。
3	13. JX(config-isis-1)#**filter-policy import route-policy** *route-policy-name*	IS-IS 路由加入 IP 路由表时的过滤策略。

步骤	配置	说明
4	14. JX(config-isis-1)#**summary** *ip-address/maskLen* { **level-1** | **level-2** }	路由条目过多，会导致在转发数据时降低路由表查找速度，同时会增加管理复杂度，通过配置路由聚合，可以减小路由表的规模。

3.10.8 配置 IS-IS 的路由的收敛

请在设备上进行以下配置。

步骤	配置	说明
1	15. JX(config-vlanif-1)#**isis hello-interval** { *interval-value* | **default** } **[level-1 | level-2]**	设置 IS-IS 发送 hello 包时间间隔。
2	16. JX(config-vlanif-1)#**isis hello-multiplier**{ *multiple-value* | **default** } **[level-1 | level-2]**	配置 IS-IS 的邻居保持时间倍数。
3	17. JX(config-vlanif-1)#**isis csnp-interval** { *interval-value* | **default** } **[level-1 | level-2]**	配置指定层级的 csnp 报文发送间隔。
4	18. JX(config-vlanif-1)#**isis psnp-interval** { *interval-value* | **default** } **[level-1 | level-2]**	配置指定层级的 psnp 报文发送间隔。
5	19. JX(config-isis-1)#**lsp-max-lifetime** { *max-life-value* | **default** }	配置路由器的链路状态数据包的刷新时间间隔。
6	20. JX(config-isis-1)#**lsp-generation max-interval** { *max-interval* | **default** } **init-interval** { *int-interval* | **default** } **incr-interval** { *incr-interval* | **default** } { **level-1** | **level-2** }	设置指定 IS-IS 进程产生 LSP 的延迟时间。

3.10.9 配置 IS-IS 的可靠性

请在设备上进行以下配置。

步骤	配置	说明
1	21. JX(config-vlanif-1)#**isis bfd** { **enable** | **disable** }	使能或去使能 IS-IS 在指定接口下配置 BFD（即打开 BFD 开关），建立缺省参数值的 BFD 会话。

3.10.10 配置维护 IS-IS

请在设备上进行以下配置。

步骤	配置	说明
1	22. JX(config-isis-1)#**reset isis** { *isis-instance* | **all** }	重置所有或者单个 IS-IS 实例。

3.10.11 检查配置

配置完成后，请在设备上执行以下命令检查配置结果。

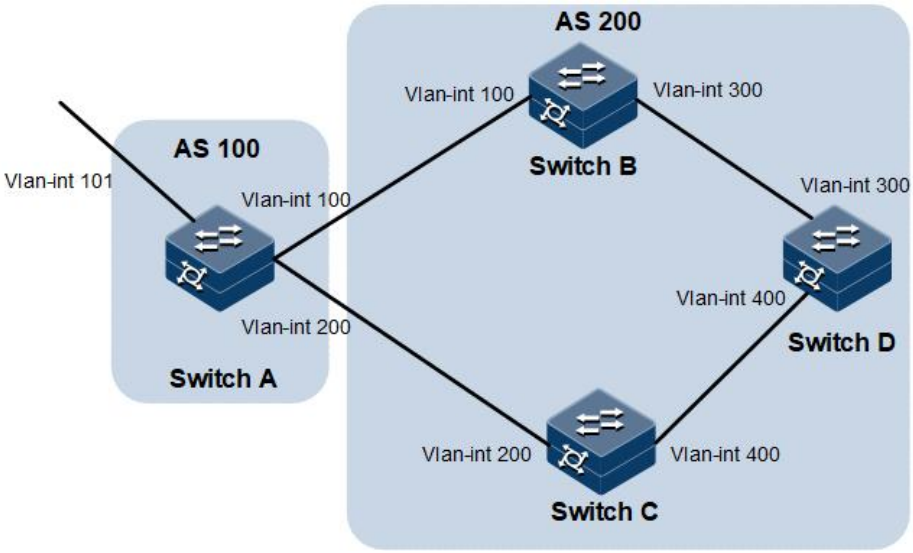
序号	检查项	说明
1	23. JX#**show isis config**	查看 IS-IS 配置信息。
2	24. JX#**show isis database**	显示显示链路状态数据库。
3	25. JX#**show isis interface**	显示 IS-IS 的接口信息或者其接口的详细信息。
4	26. JX#**show isis route**	显示 IS-IS 路由信息
5	27. JX#**show isis neighbor**	显示 IS-IS 的邻居信息。

3.10.12 配置 IS-IS 的基本功能示例

组网需求

如下图所示，Switch A、Switch B、Switch C 和 Switch D 属于同一自治系统，要求它们之间通过 IS-IS 协议达到 IP 网络互连的目的。

Switch A 和 Switch B 为 Level-1 交换机，Switch D 为 Level-2 交换机，Switch C 作为 Level-1-2 交换机将两个区域相连。Switch A、Switch B 和 Switch C 的区域号为 10，Switch D 的区域号为 20。



配置步骤

步骤 1 配置各接口的 IP 地址。

步骤 2 配置 IS-IS。

配置 Switch A:

```
SwitchA#configure
SwitchA(config)#isis 1
SwitchA(config-isis-1)#is-type level-1
SwitchA(config-isis-1)#network-entity 10.0000.0000.0001.00
SwitchA(config-isis-1)#quit
SwitchA(config)#int vlan 100
SwitchA(config-vlanif-100)#isis enable 1
SwitchA(config-vlanif-100)#quit
```

配置 Switch B:

```
SwitchB#configure
SwitchB(config)#isis 1
SwitchB(config-isis-1)#is-type level-1
SwitchB(config-isis-1)#network-entity 10.0000.0000.0002.00
SwitchB(config-isis-1)#quit
SwitchB(config)#int vlan 200
SwitchB(config-vlanif-200)#isis enable 1
SwitchB(config-vlanif-200)#quit
```

配置 Switch C:

```
SwitchC#configure
SwitchC(config)#isis 1
SwitchC(config-isis-1)#network-entity 10.0000.0000.0003.00
SwitchC(config-isis-1)#quit
SwitchC(config)#int vlan 100
```

```
SwitchC(config-vlanif-100)#isis enable 1
SwitchC(config-vlanif-100)#quit
SwitchC(config)#int vlan 200
SwitchC(config-vlanif-200)#isis enable 1
SwitchC(config-vlanif-200)#quit
SwitchC(config)#int vlan 300
SwitchC(config-vlanif-300)#isis enable 1
SwitchC(config-vlanif-300)#quit
```

配置 Switch D:

```
SwitchD#configure
SwitchD(config)#isis 1
SwitchD(config-isis-1)#is-type level-2
SwitchD(config-isis-1)#network-entity 20.0000.0000.0004.00
SwitchD(config-isis-1)#quit
SwitchD(config)#int vlan 300
SwitchD(config-vlanif-300)#isis enable 1
SwitchD(config-vlanif-300)#quit
```

检查结果

显示各交换机的 IS-IS database 信息，查看 LSP 是否完整

```
SwitchA(config)#show isis database

                  Database Information for IS-IS(1)

                    Level-1 Link State Database
The total number of isis 1 database is: 5
#    ID                  Level   Seq          Checksum  Lifetime  Length
ATT/P/OL
1    0000.0000.0001.00-00   1      3            52707    958      69
0/0/0
2    0000.0000.0001.01-00   1      1            8889     958      52
0/0/0
3    0000.0000.0002.00-00   1      3            7823     829      69
0/0/0
4    0000.0000.0002.01-00   1      1            13222    829      52
0/0/0
5    0000.0000.0003.00-00   1      7            24706    968      112
1/0/0

SwitchB(config)#show isis database

                  Database Information for IS-IS(1)

                    Level-1 Link State Database
The total number of isis 1 database is: 5
#    ID                  Level   Seq          Checksum  Lifetime  Length
ATT/P/OL
1    0000.0000.0001.00-00   1      3            52707    958      69
0/0/0
```

```
2    0000.0000.0001.01-00   1      1            8889     958      52
0/0/0
3    0000.0000.0002.00-00   1      3            7823     829      69
0/0/0
4    0000.0000.0002.01-00   1      1            13222    829      52
0/0/0
5    0000.0000.0003.00-00   1      7            24706    968      112
1/0/0
```

```
SwitchC(config)#show isis database

                  Database Information for IS-IS(1)

                    Level-1 Link State Database
The total number of isis 1 database is: 5
#    ID                  Level   Seq          Checksum  Lifetime  Length
ATT/P/OL
1    0000.0000.0001.00-00   1      3            52707    958      69
0/0/0
2    0000.0000.0001.01-00   1      1            8889     958      52
0/0/0
3    0000.0000.0002.00-00   1      3            7823     829      69
0/0/0
4    0000.0000.0002.01-00   1      1            13222    829      52
0/0/0
5    0000.0000.0003.00-00   1      7            24706    968      112
1/0/0

                    Level-2 Link State Database
The total number of isis 1 database is: 3
#    ID                  Level   Seq          Checksum  Lifetime  Length
ATT/P/OL
6    0000.0000.0003.00-00   2      5            42205    957      101
0/0/0
7    0000.0000.0003.03-00   2      1            18060    732      52
0/0/0
8    0000.0000.0004.00-00   2      5            20494    708      84
0/0/0
L1 Database Count :5
L2 Database Count :3
```

```
SwitchD(config)#show isis database

                  Database Information for IS-IS(1)

                    Level-2 Link State Database
The total number of isis 1 database is: 3
#    ID                  Level   Seq          Checksum  Lifetime  Length
ATT/P/OL
1    0000.0000.0003.00-00   2      5            42205    957      101
0/0/0
```

```
2    0000.0000.0003.03-00   2      1            18060    732      52
0/0/0
3    0000.0000.0004.00-00   2      5            20494    708      84
0/0/0
```

显示各交换机的 IS-IS 路由信息。Level-1 交换机的路由表中应该有一条缺省路由，且下一跳为 Level-1-2 交换机，Level-2 交换机的路由表中应该有所有 Level-1 和 Level-2 的路由。

```
SwitchA(config)#show isis route
ISIS process 1
Level    Dst                 Nexthop          MAC             Cost    Time
Level-1  0.0.0.0/0            10.1.1.1         f0f1:f2f3:0101   10
0:10:56
Level-1  10.1.1.0/24          10.1.1.2         0000:0000:0000   10
0:11:31
Level-1  10.1.2.0/24          10.1.1.1         f0f1:f2f3:0101   20
0:10:56
Level-1  192.168.0.0/24       10.1.1.1         f0f1:f2f3:0101   20
0:10:56

SwitchB(config)#show isis route
ISIS process 1
Level    Dst                 Nexthop          MAC             Cost    Time
Level-1  0.0.0.0/0            10.1.2.1         f0f1:f2f3:0101   10
0:13:47
Level-1  10.1.1.0/24          10.1.2.1         f0f1:f2f3:0101   20
0:11:49
Level-1  10.1.2.0/24          10.1.2.2         0000:0000:0000   10
0:14:04
Level-1  192.168.0.0/24       10.1.2.1         f0f1:f2f3:0101   20
0:13:47

SwitchC(config)#show isis route
ISIS process 1
Level    Dst                 Nexthop          MAC             Cost    Time
Level-1  10.1.1.0/24          10.1.1.1         0000:0000:0000   10
0:12:31
Level-1  10.1.2.0/24          10.1.2.1         0000:0000:0000   10
0:16:14
Level-1  192.168.0.0/24       192.168.0.1      0000:0000:0000   10
0:16:39
Level-2  172.16.0.0/16        192.168.0.2      00e0:fcd7:47d1   10
0:16:26
Level-2  10.1.1.0/24          10.1.1.1         0000:0000:0000   10
0:12:31
Level-2  10.1.2.0/24          10.1.2.1         0000:0000:0000   10
0:16:14
Level-2  192.168.0.0/24       192.168.0.1      0000:0000:0000   10
0:16:39

SwitchD(config)#show isis route
ISIS process 1
Level    Dst                 Nexthop          MAC             Cost    Time
```

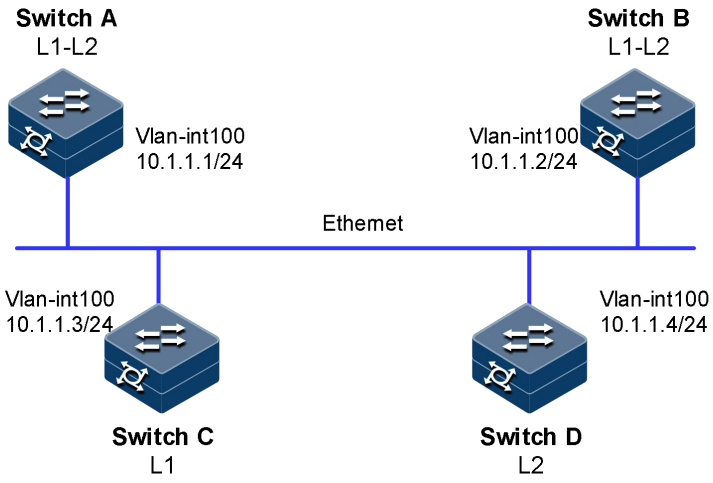
```
Level-2   172.16.0.0/16          172.16.0.1        00e0:fcd7:47d1    10
0:16:26
Level-2   10.1.1.0/24            192.168.0.1       0000:0000:0000    10
0:12:31
Level-2   10.1.2.0/24            192.168.0.1       0000:0000:0000    10
0:16:14
Level-2   192.168.0.0/24         192.168.0.2       0000:0000:0000    10
0:16:39
```

3.10.13 配置 IS-IS 的 DIS 选择示例

组网需求

如下图所示，Switch A、Switch B、Switch C 和 Switch D 都运行 IS-IS 路由协议以实现互连，它们属于同一区 域 10，网络类型为广播网（以太网）。

Switch A 和 Switch B 是 Level-1-2 交换机，Switch C 为 Level-1 交换机，Switch D 为 Level-2 交换机。要求通过改变接口的 DIS 优先级，将 Switch A 配置为 Level-1-2 的 DIS。



配置步骤

步骤 1 配置各接口的 IP 地址。

步骤 2 配置 IS-IS。

配置 Switch A:

```
SwitchA#configure
SwitchA(config)#isis 1
SwitchA(config-isis-1)#network-entity 10.0000.0000.0001.00
SwitchA(config-isis-1)#quit
SwitchA(config)#int vlan 100
SwitchA(config-vlanif-100)#isis enable 1
SwitchA(config-vlanif-100)#quit
```

配置 Switch B:

```
SwitchB#configure
SwitchB(config)#isis 1
SwitchB(config-isis-1)#network-entity 10.0000.0000.0002.00
SwitchB(config-isis-1)#quit
SwitchB(config)#int vlan 100
SwitchB(config-vlanif-100)#isis enable 1
SwitchB(config-vlanif-100)#quit
```

配置 Switch C,并配置该设备的 level1 的优先级为 127:

```
SwitchC#configure
SwitchC(config)#isis 1
SwitchC(config-isis-1)#network-entity 10.0000.0000.0003.00
SwitchC(config-isis-1)#quit
SwitchC(config)#int vlan 100
SwitchC(config-vlanif-100)#isis enable 1
SwitchC(config-vlanif-100)#isis dis-priority 127 level-1
SwitchC(config-vlanif-100)#quit
```

配置 Switch D,并配置该设备的 level2 的优先级为 127:

```
SwitchD#configure
SwitchD(config)#isis 1
SwitchD(config-isis-1)#network-entity 10.0000.0000.0004.00
SwitchD(config-isis-1)#quit
SwitchD(config)#int vlan 100
SwitchD(config-vlanif-100)#isis enable 1
SwitchC(config-vlanif-100)#isis dis-priority 127 level-2
SwitchD(config-vlanif-100)#quit
```

查看 Switch A 的 IS-IS 邻居信息。

```
SwitchA(config)#show isis neighbor

                Neighbor information for IS-IS(1)
The total number of isis(1) neighbor is:6
Interface    Level   ID             MAC            Priority  HoldTime
UpTime            State       Name
vlan-100     L1      0000.0000.0002   f0f1.f2f3.0201   64       15       0
days 0:01:11     up
vlan-100     L2      0000.0000.0002   f0f1.f2f3.0201   64       11       0
days 0:01:32     up
vlan-100     L1      0000.0000.0003   f0f1.f2f3.0301   64       29       0
days 0:01:12     up
vlan-100     L2      0000.0000.0003   f0f1.f2f3.0301   64       23       0
days 0:01:12     up
vlan-100     L1      0000.0000.0004   f0f1.f2f3.0401   64       23       0
days 0:00:23     up
vlan-100     L2      0000.0000.0004   f0f1.f2f3.0401   64       24       0
days 0:00:48     up
```

查看 Switch B 的 IS-IS 邻居信息。

```
SwitchB(config)#show isis neighbor

                Neighbor information for IS-IS(1)
The total number of isis(1) neighbor is:6
Interface    Level   ID             MAC             Priority  HoldTime
UpTime            State       Name
vlan-100     L2     0000.0000.0001   f0f1.f2f3.0101   64       19       0
days 0:00:48     up
vlan-100     L1     0000.0000.0001   f0f1.f2f3.0101   64       21       0
days 0:00:47     up
vlan-100     L1     0000.0000.0003   f0f1.f2f3.0301   64       21       0
days 0:00:30     up
vlan-100     L2     0000.0000.0003   f0f1.f2f3.0301   64       27       0
days 0:00:30     up
vlan-100     L1     0000.0000.0004   f0f1.f2f3.0401   64       27       0
days 0:00:08     up
vlan-100     L2     0000.0000.0004   f0f1.f2f3.0401   64       26       0
days 0:00:08     up
```

查看 Switch C 的 IS-IS 邻居信息。

```
SwitchC(config)#show isis neighbor

                Neighbor information for IS-IS(1)
The total number of isis(1) neighbor is:6
Interface    Level   ID             MAC             Priority  HoldTime
UpTime            State       Name
vlan-100     L2     0000.0000.0002   f0f1.f2f3.0201   64       26       0
days 0:00:36     up
vlan-100     L2     0000.0000.0001   f0f1.f2f3.0101   64       13       0
days 0:00:34     up
vlan-100     L1     0000.0000.0002   f0f1.f2f3.0201   64       21       0
days 0:00:35     up
vlan-100     L1     0000.0000.0004   f0f1.f2f3.0401   64       28       0
days 0:00:14     up
vlan-100     L2     0000.0000.0004   f0f1.f2f3.0401   64       27       0
days 0:00:09     up
vlan-100     L1     0000.0000.0001   f0f1.f2f3.0101   64       24       0
days 0:00:05     up
```

查看 Switch D 的 IS-IS 邻居信息。

```
SwitchD(config)#show isis neighbor

                Neighbor information for IS-IS(1)
The total number of isis(1) neighbor is:6
Interface    Level   ID             MAC             Priority  HoldTime
UpTime            State       Name
vlan-1       L1     0000.0000.0003   f0f1.f2f3.0301   64       22       0
days 0:00:25     up
```

```
vlan-1      L2     0000.0000.0003   f0f1.f2f3.0301   64       27       0
days 0:00:30     up
vlan-1      L2     0000.0000.0002   f0f1.f2f3.0201   64       25       0
days 0:00:37     up
vlan-1      L1     0000.0000.0002   f0f1.f2f3.0201   64       29       0
days 0:00:31     up
vlan-1      L2     0000.0000.0001   f0f1.f2f3.0101   64       19       0
days 0:00:19     up
vlan-1      L1     0000.0000.0001   f0f1.f2f3.0101   64       28       0
days 0:00:10     up
```

显示 Switch A 的 IS-IS 接口信息。

```
SwitchD(config)#show isis interface

               Interface information for IS-IS(1)
The total number of isis(1) interface is:1
Interface           CircID  Adminstate  Type       Level MeshEnable
SmallHello
vlan-1               1       on        broadcast   L1-2  N/A         false
```

查看显示交换机的 IS-IS database 信息 level1 的 dis 为 **0000.0000.0003.01-00,** level2 的 dis 为 **0000.0000.0004.01-00**

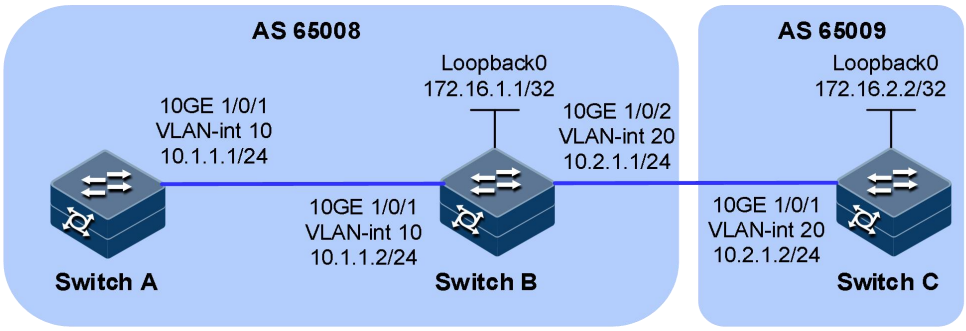
```
SIM-MPU(config)#show isis database

                  Database Information for IS-IS(1)

                    Level-1 Link State Database
The total number of isis 1 database is: 5
#    ID                  Level   Seq           Checksum  Lifetime  Length
ATT/P/OL
1    0000.0000.0001.00-00   1      4             1448     1147     69
0/0/0
2    0000.0000.0002.00-00   1      3             7312     1134     69
0/0/0
3    0000.0000.0003.00-00   1      2             13176    1133     69
0/0/0
4   0000.0000.0003.01-00  1    3       16247   1134   74   0/0/0
5    0000.0000.0004.00-00   1      3             18018    1143     69
0/0/0

                    Level-2 Link State Database
The total number of isis 1 database is: 5
#    ID                  Level   Seq           Checksum  Lifetime  Length
ATT/P/OL
6    0000.0000.0001.00-00   2      3             12668    1138     69
0/0/0
7    0000.0000.0002.00-00   2      3             18021    1137     69
0/0/0
8    0000.0000.0003.00-00   2      3             23374    1141     69
0/0/0
9    0000.0000.0004.00-00   2      2             29238    1144     69
0/0/0
```

```
10  0000.0000.0004.01-00  2    1       12423   1144   74   0/0/0
L1 Database Count :5
L2 Database Count :5
```

3.10.14 配置 IS-IS 引入外部路由

组网需求

如下图所示，SwitchA 和 SwitchB 同属一个自治系统，两者建立 IS-IS 邻居关系。SwitchB 与 SwitchC 之间建立 EBGP 连接。现要求各网段之间都能互相通信，并且在 AS65008 发送传递路由信息至 AS65009 时需要改变度量。



配置步骤

步骤 1 配置各接口的 IP 地址。

步骤 2 配置 IS-IS 和 BGP 相关配置。

配置 Switch A:

```
SwitchA#configure
SwitchA(config)#isis 1
SwitchA(config-isis-1)#network-entity 10.0000.0000.0001.00
SwitchA(config-isis-1)#quit
SwitchA(config)#int vlan 10
SwitchA(config-vlanif-10)#isis enable 1
SwitchA(config-vlanif-10)#quit
```

配置 Switch B:

```
SwitchB#configure
SwitchB(config)#isis 1
SwitchB(config-isis-1)#network-entity 10.0000.0000.0002.00
SwitchB(config-isis-1)#import-route bgp
SwitchB(config-isis-1)#quit
SwitchB(config)#int vlan 10
SwitchB(config-vlanif-10)#isis enable 1
SwitchB(config-vlanif-10)#quit
SwitchB(config)#int vlan 20
SwitchB(config-vlanif-20)#isis enable 1
SwitchB(config-vlanif-20)#quit
```

```
SwitchB(config)#bgp 65008
SwitchB(config-bgp)#router-id 172.16.1.1
SwitchB(config-bgp)#peer 10.2.1.2 as-number 65009
SwitchB(config-bgp)#ipv4-family unicast
SwitchB(config-bgp-af-ipv4)#neighbor 10.2.1.2 enable
SwitchB(config-bgp-af-ipv4)#network 10.2.1.0/24
```

配置 Switch C，bgp 协议会引入一条 192.168.1.0/24 静态路由:

```
SwitchC#configure
SwitchC(config)#ip route-static 192.168.1.0 255.255.255.0 10.2.1.3
SwitchC(config)#bgp 65009
SwitchC(config-bgp)#router-id 172.16.2.2
SwitchC(config-bgp)#peer 10.2.1.1 as-number 65008
SwitchC(config-bgp)#ipv4-family unicast
SwitchC(config-bgp-af-ipv4)#neighbor 10.2.1.1 enable
SwitchC(config-bgp-af-ipv4)#network 10.2.1.0/24
SwitchC(config-bgp-af-ipv4)#import-route static
```

显示各路由器的 IS-IS 路由信息，switchB 中存在一条 BGP 的路由表项，在 switchA 的 IS-IS 和 ip 路由表中会存在 **192.168.1.0/24** 该网段的路由

```
SwitchA(config)#show isis route

ISIS process 1
Level    Dst                 Nexthop          MAC             Cost   Time
Level-1  10.1.1.0/24          10.1.1.1         0000:0000:0000   10
0:12:20
Level-1  10.2.1.0/24          10.1.1.2         f0f1:f2f3:0101   20
0:12:03
Level-1  192.168.1.0/24    10.1.1.2     f0f1:f2f3:0101  10   0:10:30
Level-2  10.1.1.0/24          10.1.1.1         0000:0000:0000   10
0:12:20
Level-2  10.2.1.0/24          10.1.1.2         f0f1:f2f3:0101   20
0:12:03
Level-2  192.168.1.0/24    10.1.1.2     f0f1:f2f3:0101  10   0:10:30


SwitchA(config)#show ip route
 Max Routing Number Limit: 131072
 Max Destination Number Limit: 771
 Routing Tables:
 Destination      Gateway         Pre/Cost     Proto  Interface
Mpls     Status    Vpn-Instance
------------------------------------------------------------------------
------------------------------------
 10.1.1.0/24      10.1.1.1          0/0        local  vlan-10           no
Up       N/A
                  10.1.1.1         50/10       isis   vlan-10           no
Up       N/A
 10.1.1.1/32      10.1.1.1          0/0        local  vlan-10           no
Up       N/A
```

```
 10.2.1.0/24      10.1.1.2         50/20       isis   vlan-10           no
Up       N/A
 127.0.0.1/32     127.0.0.1         0/0        local  loopback-0         no
Up       N/A
 192.168.1.0/24  10.1.1.2     50/10    isis  vlan-10      no    Up    N/A

SwitchB(config)#show isis route

ISIS process 1
Level    Dst                 Nexthop           MAC             Cost    Time
Level-1   10.1.1.0/24          10.1.1.2          0000:0000:0000    10
0:13:51
Level-1   10.2.1.0/24          10.2.1.1          0000:0000:0000    10
0:14:07
Level-2   10.1.1.0/24          10.1.1.2          0000:0000:0000    10
0:13:51
Level-2   10.2.1.0/24          10.2.1.1          0000:0000:0000    10
0:14:07

SwitchB(config)#show ip route
 Max Routing Number Limit: 131072
 Max Destination Number Limit: 771
 Routing Tables:
 Destination      Gateway         Pre/Cost     Proto  Interface
Mpls     Status    Vpn-Instance
------------------------------------------------------------------------------
------------------------------------
 10.1.1.0/24      10.1.1.2          0/0        local  vlan-10           no
Up       N/A
                  10.1.1.2         50/10       isis   vlan-10           no
Up       N/A
 10.1.1.2/32      10.1.1.2          0/0        local  vlan-10           no
Up       N/A
 10.2.1.0/24      10.2.1.1          0/0        local  vlan-20           no
Up       N/A
                  10.2.1.1         50/10       isis   vlan-20           no
Up       N/A
 10.2.1.1/32      10.2.1.1          0/0        local  vlan-20           no
Up       N/A
 127.0.0.1/32     127.0.0.1         0/0        local  loopback-0         no
Up       N/A
 172.16.1.1/32    172.16.1.1        0/0        local  loopback-1         no
Up       N/A
 192.168.1.0/24  10.2.1.2    255/60    bgp  vlan-20      no    Up    N/A
------------------------------------------------------------------------------
------------------------------------
 Total: 9         Static: 0         Down: 0         Destination: 7
```

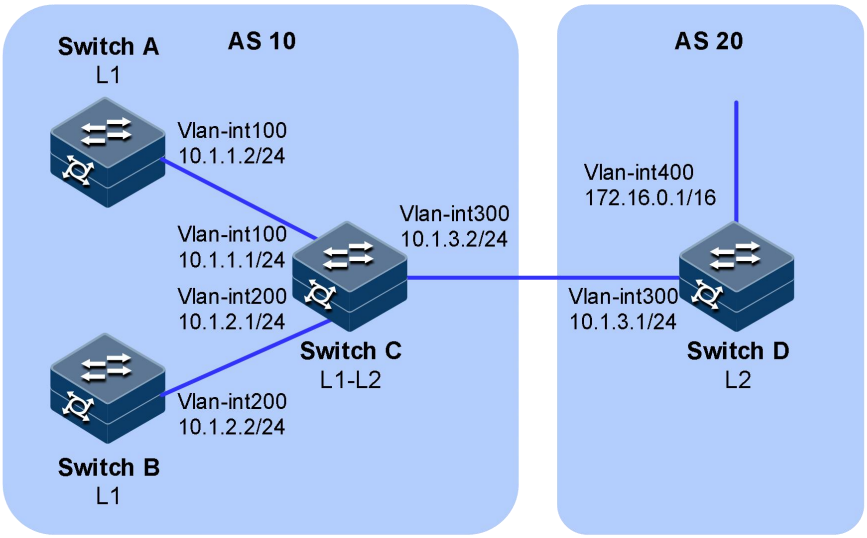
3.10.15 配置 IS-IS 验证配置举例

组网需求

如下图所示，Switch A、Switch B、Switch C 和 Switch D 属于同一路由域，要求它们之间通过 IS-IS 协议达到 IP 网络互连的目的。

其中，Switch A、Switch B 和 Switch C 属于同一个区域，区域号为 10，Switch D 属于另外一个区域，区域号为 20。

在区域 10 内配置区域验证，防止不可信任的路由信息加入到区域 10 的 LSDB 中；在 Switch C 和 Switch D 上配置路由域验证，防止将不可信的路由信息注入当前路由域；分别在 Switch A、SwitchB、Switch C 和 Switch D 上配置邻居关系验证。



配置步骤

步骤 1 配置各接口的 IP 地址。

步骤 2 配置 IS-IS 基本功能。

配置 Switch A:

```
SwitchA#configure
SwitchA(config)#isis 1
SwitchA(config-isis-1)#network-entity 10.0000.0000.0001.00
SwitchA(config-isis-1)#quit
SwitchA(config)#int vlan 100
SwitchA(config-vlanif-100)#isis enable 1
SwitchA(config-vlanif-100)#quit
```

配置 Switch B:

```
SwitchB#configure
SwitchB(config)#isis 1
SwitchB(config-isis-1)#network-entity 10.0000.0000.0002.00
SwitchB(config-isis-1)#quit
SwitchB(config)#int vlan 200
SwitchB(config-vlanif-200)#isis enable 1
SwitchB(config-vlanif-200)#quit
```

配置 Switch C:

```
SwitchC#configure
SwitchC(config)#isis 1
SwitchC(config-isis-1)#network-entity 10.0000.0000.0003.00
SwitchC(config-isis-1)#quit
SwitchC(config)#int vlan 100
SwitchC(config-vlanif-100)#isis enable 1
SwitchC(config-vlanif-100)#quit
SwitchC(config)#int vlan 200
SwitchC(config-vlanif-200)#isis enable 1
SwitchC(config-vlanif-200)#quit
SwitchC(config)#int vlan 300
SwitchC(config-vlanif-300)#isis enable 1
SwitchC(config-vlanif-300)#quit
```

配置 Switch D:

```
SwitchD#configure
SwitchD(config)#isis 1
SwitchD(config-isis-1)#is-type level-2
SwitchD(config-isis-1)#network-entity 20.0000.0000.0004.00
SwitchD(config-isis-1)#quit
SwitchD(config)#int vlan 300
SwitchD(config-vlanif-300)#isis enable 1
SwitchD(config-vlanif-300)#quit
```

步骤 3 在 SwitchA、SwitchB、SwitchC 和 SwitchD 之间建立邻居关系验证。

分别在 Switch A 的 Vlan-interface100、Switch C 的 interface vlan 100 配置邻居关系验证，验证方式为 MD5 明文，验证相同密码能够建立邻居。

```
SwitchA(config)#int vlan 100
SwitchA(config-vlanif-100)#isis authentication-mode md5 plain abcd
SwitchA(config-vlanif-100)#quit
SwitchC(config)#int vlan 100
SwitchC(config-vlanif-100)#isis authentication-mode md5 plain abcd
SwitchC(config-vlanif-100)#quit
```

分别在 Switch B 的 Vlan-interface100、Switch C 的 interface vlan 100 配置邻居关系验证，验证方式为 MD5 明文，验证不同密码不能够建立邻居。

```
SwitchB(config)#int vlan 200
SwitchB(config-vlanif-200)#isis authentication-mode md5 plain abcd
SwitchB(config-vlanif-200)#quit
SwitchC(config)#int vlan 200
SwitchC(config-vlanif-200)#isis authentication-mode md5 plain dcba
SwitchC(config-vlanif-200)#quit
```

分别在 Switch A 和 Switch C 上配置区域验证，验证方式为 MD5 明文验证，验证密码相同时能够正常学到路由，不同则学不到路由。

```
SwitchA(config)#isis 1
SwitchA(config-isis-1)#area-authentication-mode md5 plain abcd
SwitchA(config-isis-1)#quit
SwitchC(config)#isis 1
SwitchC(config-isis-1)#area-authentication-mode md5 plain abcd
SwitchC(config-isis-1)#quit
```

分别在 Switch C 和 Switch D 上配置路由域验证，验证方式为 MD5 明文验，验证密码相同时能够正常学到路由，不同则学不到路由。

```
SwitchC(config)#isis 1
SwitchC(config-isis-1)#domain-authentication-mode md5 plain abcd
SwitchC(config-isis-1)#quit
SwitchD(config)#isis 1
SwitchD(config-isis-1)#domain-authentication-mode md5 plain abcd
SwitchD(config-isis-1)#quit
```

3.11 IS-IS（IPv6）

3.11.1 简介

基本概念

本章只介绍 IS-IS for IPv6 的区别，有关 IS-IS 协议的具体实现原理读者可参见 IS-IS 原理描述。

IS-IS 最初是为 OSI 网络设计的一种基于链路状态算法的动态路由协议。之后为了提供对 IPv4 的路由支持，扩展应用到 IPv4 网络，称为集成 IS-IS。

随着 IPv6 网络的建设，同样需要动态路由协议为 IPv6 报文的转发提供准确有效的路由信息。IS-IS 路由协议结合自身具有良好的扩展性的特点，实现了对 IPv6 网络层协议的支持，可以发现和生成 IPv6 路由。

IETF 的 draft-ietf-isis-ipv6-05 中规定了 IS-IS 为支持 IPv6 所新增的内容。为了支持 IPv6 路由的处理和计算，IS-IS 新增了两个 TLV（Type-Length-Value）和一个新的 NLPID（Network Layer Protocol Identifier）。

新增的两个 TLV 分别是：

- IPv6 Reachability：类型值为 236（0xEC），通过定义路由信息前缀、度量值等信息来说明网络的可达性。
- IPv6 Interface Address：类型值为 232（0xE8），它对应于 IPv4 中的“IP Interface Address”TLV，只不过把原来的 32 比特的 IPv4 地址改为 128 比特的 IPv6 地址。

NLPID 是标识网络层协议报文的一个 8 比特字段，IPv6 的 NLPID 值为 142（0x8E）。如果 IS-IS 路由器支持 IPv6，那么它必须以这个 NLPID 值向外发布路由信息。

3.11.2 配置准备

场景

IS-IS 属于内部网关 IGP。用于自治系统内部。IS-IS 也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。

前提

相邻节点的网络层可达。

3.11.3 缺省配置

设备上 IS-IS（IPv6）的缺省配置如下。

功能	缺省值
IS-IS 全局功能状态	禁用
DIS 优先级	64
设备 level 级别	level-1-2
发送 Hello 报文时间间隔	10s
LSP 刷新时间间隔	900s
LSP 最大有效时间	1200s

3.11.4 配置 IS-IS（IPv6）基本功能

请在设备上进行以下配置。

步骤	配置	说明
1	`JX#config`	进入全局配置模式。
2	`JX(config)#isis` *`Process id`*	创建 IS-IS 进程并进入 IS-IS 视图
3	`JX(config-isis-1)#network-entity` *`net`*	在进入 IS-IS 视图之后，必须完成 IS-IS 进程的 NET 配置，IS-IS 协议才能真正启动

步骤	配置	说明
4	JX(config-isis-1)#**is-type** { **level-1** | **level-1-2** | **level-2** }	配置全局 level 级别。当 Level 级别为 Level-1 时，设备只与属于同一区域的 Level-1 和 Level-1-2 设备形成邻居关系，并且只负责维护 Level-1 的链路状态数据库 LSDB。 当 Level 级别为 Level-2 时，设备可以与同一或者不同区域的 Level-2 设备或者其它区域的 Level-1-2 设备形成邻居关系，并且只维护一个 Level-2 的 LSDB。 同时属于 Level-1 和 Level-2 的路由器称为 Level-1-2 路由器，它可以与同一区域的 Level-1 和 Level-1-2 路由器形成 Level-1 邻居关系，也可以与其他区域的 Level-2 和 Level-1-2 路由器形成 Level-2 的邻居关系。Level-1 路由器必须通过 Level-1-2 路由器才能连接至其他区域。Level-1-2 路由器维护两个 LSDB，Level-1 的 LSDB 用于区域内路由，Level-2 的 LSDB 用于区域间路由。
5	JX(config-isis-1)#**ipv6 enable** [**topology** { **compatible** | **ipv6** | **standard** }]	使能 ISIS 进程的 IPv6 能力，默认使能的是标准的 ipv6 功能。
6	JX(config-vlanif-1)#**isis-ipv6 enable** [*instance-id*]	使能 IS-IS（ipv6）接口
7	JX (config-vlanif-1)#**isis circuit-level** { **level-1** | **level-2** | **level-1-2** | **default** }	配置接口的 level 级别。
8	JX(config-vlanif-1)#**isis dis-priority** { *priority-value* | **default** } [**level-1** | **level-2**]	指定路由器的优先级数值
9	JX(config-vlanif-1)#**isis circuit-type** { **broadcast** | **p2p** }	配置 IS-IS 链路类型，默认链路类型为广播类型

3.11.5 配置 IS-IS（IPv6）网络的安全性

请在设备上进行以下配置。

步骤	配置	说明
1	JX(config-isis-1)#**area-authentication-mode** { **simple** | **md5** } { **cipher** | **plain** } **password** [**snp-packet** { **authentication-avoid** | **send-only** } | **all-send-only**]	配置 IS-IS 区域认证密码，并设置 ISIS 区域按照预定的方式和密码验证收到的 Level-1 路由信息报文（LSP 和 SNP），也可以用来为发送的 Level-1 报文加上认证信息。

步骤	配置	说明
2	JX(config-isis-1)#**domain-authentication-mode** { **simple** | **md5** } { **cipher** | **plain** } **password** [**snp-packet** { **authentication-avoid** | **send-only** } | **all-send-only**]	配置 IS-IS 路由域认证密码，并设置 IS-IS 路由域按照预设的方式和密码验证收到的 Level-2 路由信息报文（LSP 和 SNP）。
3	JX(config-vlanif-1)#**isis authentication-mode** { **simple** | **md5** } { **cipher** | **plain** } **password** { **level-1** | **level-2** | **p2p** }	设置 IS-IS 接口以指定的方式和密码验证 Hello 报文，并在发送的 Hello 报文中添加认证信息。

3.11.6 配置 IS-IS（IPv6）的选路

请在设备上进行以下配置。

步骤	配置	说明
1	JX(config-isis-1)#**ipv6 preference** { *priority* | **default** }	配置 IS-IS（IPv6）协议路由的优先级。
2	JX(config-isis-1)#cost-style { **narrow** | **wide** | **compatible** | **narrow-compatible** | **wide-compatible** | **default** } { **level-1** | **level-2** }	可以接收开销类型为 narrow 和 wide 的路由，但却只发送 narrow 的路由。
3	JX(config-isis-1)#**ipv6 maximum load-balancing** { *load-balancing-number* | **default** }	设置 IPv6 形成负载分担的等价路由的最大条数。
4	JX(config-isis-1)#**import-route level-2 to level-1**	将 Level-2 区域的路由渗透到本地 Level-1 区域。缺省情况下，Level-2 区域的路由信息不渗透到 Level-1 区域。

3.11.7 配置 IS-IS（IPv6）的路由信息的交互

请在设备上进行以下配置。

步骤	配置	说明
1	JX(config-isis-1)#**default-route-originate ipv6** { **none** | **level-1** | **level-2** | **level-1-2** }	设置 IS-IS（IPv6）路由器生成缺省路由。

步骤	配置	说明
2	JX(config-isis-1)#**import-route ipv6** { **connect** | **static** | **rip** | **ospf** | **bgp** | **isis** } { **level-1** | **level-2** | **level-1-2** } [**cost** { **cost-value** | **default** | **inherit** } | **route-policy** *policy-name*]*	引入其它路由信息。
3	JX(config-isis-1)#**filter-policy import route-policy** *route-policy-name*	IS-IS（IPv6）路由加入 IP 路由表时的过滤策略。
4	JX(config-isis-1)#**ipv6 summary** *ip-address/maskLen* { **level-1** | **level-2** }	路由条目过多，会导致在转发数据时降低路由表查找速度，同时会增加管理复杂度，通过配置路由聚合，可以减小路由表的规模。

3.11.8 配置 IS-IS（IPv6）的路由的收敛

请在设备上进行以下配置。

步骤	配置	说明
1	JX(config-vlanif-1)#**isis hello-interval** { *interval-value* | **default** } [**level-1** | **level-2**]	设置 IS-IS 发送 hello 包时间间隔。
2	JX(config-vlanif-1)#**isis hello-multiplier**{ *multiple-value* | **default** } [**level-1** | **level-2**]	配置 IS-IS 的邻居保持时间倍数。
3	JX(config-vlanif-1)#**isis csnp-interval** {*interval-value* | **default** } [**level-1** | **level-2**]	配置指定层级的 csnp 报文发送间隔。
4	JX(config-vlanif-1)#**isis psnp-interval** {*interval-value* | **default** } [**level-1** | **level-2**]	配置指定层级的 psnp 报文发送间隔。
5	JX(config-isis-1)#**lsp-max-lifetime** { *max-life-value* | **default** }	配置路由器的链路状态数据包的刷新时间间隔。
6	JX(config-isis-1)#**lsp-generation max-interval** { *max-interval* | **default** } **init-interval** { *int-interval* | **default** } **incr-interval** { *incr-interval* | **default** } { **level-1** | **level-2** }	设置指定 IS-IS 进程产生 LSP 的延迟时间。

3.11.9 配置 IS-IS（IPv6）的可靠性

请在设备上进行以下配置。

步骤	配置	说明
1	JX(config-vlanif-1)#**isis-ipv6 bfd** { **enable** | **disable** }	使能或去使能 IS-IS（IPv6）在指定接口下配置 BFD（即打开 BFD 开关），建立缺省参数值的 BFD 会话。

3.11.10 配置维护 IS-IS（IPv6）

请在设备上进行以下配置。

步骤	配置	说明
1	JX(config-isis-1)#**reset isis** { *isis-instance* | **all** }	重置所有或者单个 IS-IS 实例。

3.11.11 检查配置

配置完成后，请在设备上执行以下命令检查配置结果。

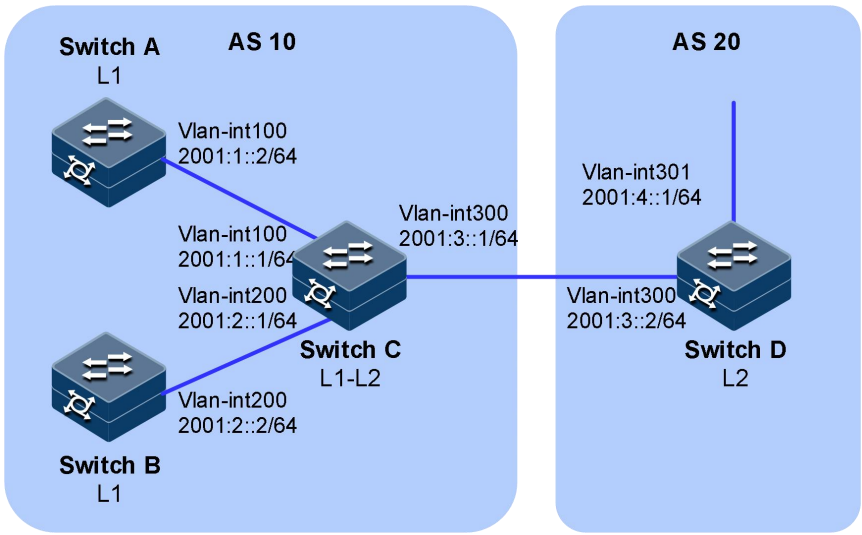
序号	检查项	说明
1	JX#**show isis config**	查看 IS-IS 配置信息。
2	JX#**show isis database**	显示显示链路状态数据库。
3	JX#**show isis interface**	显示 IS-IS 的接口信息或者其接口的详细信息。
4	JX#**show isis-ipv6 route**	显示 IS-IS（IPv6）路由信息
5	JX#**show isis neighbor**	显示 IS-IS 的邻居信息。

3.11.12 配置 IS-IS（IPv6）的基本功能示例

组网需求

如下图所示，Switch A、Switch B、Switch C 和 Switch D 属于同一自治系统，要求它们之间通过 IS-IS 协议达到 IP 网络互连的目的。

Switch A 和 Switch B 为 Level-1 交换机，Switch D 为 Level-2 交换机，Switch C 作为 Level-1-2 交换机将两个区域相连。Switch A、Switch B 和 Switch C 的区域号为 10，Switch D 的区域号为 20。



配置步骤

配置各接口的 IP 地址。

配置 IS-IS（IPv6）。

配置 Switch A:

```
SwitchA#configure
SwitchA(config)#isis 1
SwitchA(config-isis-1)#is-type level-1
SwitchA(config-isis-1)#network-entity 10.0000.0000.0001.00
SwitchA(config-isis-1)#ipv6 enable topology standard
SwitchA(config-isis-1)#quit
SwitchA(config)#int vlan 100
SwitchA(config-vlanif-100)#isis enable 1
SwitchA(config-vlanif-100)#quit
```

配置 Switch B:

```
SwitchB#configure
SwitchB(config)#isis 1
SwitchB(config-isis-1)#is-type level-1
SwitchB(config-isis-1)#network-entity 10.0000.0000.0002.00
SwitchB(config-isis-1)#ipv6 enable topology standard
SwitchB(config-isis-1)#quit
SwitchB(config)#int vlan 200
SwitchB(config-vlanif-200)#isis enable 1
SwitchB(config-vlanif-200)#quit
```

配置 Switch C:

```
SwitchC#configure
SwitchC(config)#isis 1
SwitchC(config-isis-1)#network-entity 10.0000.0000.0003.00
SwitchC(config-isis-1)#ipv6 enable topology standard
SwitchC(config-isis-1)#quit
```

```
SwitchC(config)#int vlan 100
SwitchC(config-vlanif-100)#isis enable 1
SwitchC(config-vlanif-100)#isis enable 1
SwitchC(config-vlanif-100)#isis dis-priority 127
SwitchC(config-vlanif-100)#quit
SwitchC(config)#int vlan 200
SwitchC(config-vlanif-200)#isis enable 1
SwitchC(config-vlanif-100)#isis dis-priority 127
SwitchC(config-vlanif-200)#quit
SwitchC(config)#int vlan 300
SwitchC(config-vlanif-300)#isis enable 1
SwitchC(config-vlanif-100)#isis dis-priority 127
SwitchC(config-vlanif-300)#quit
```

配置 Switch D:

```
SwitchD#configure
SwitchD(config)#isis 1
SwitchD(config-isis-1)#is-type level-2
SwitchD(config-isis-1)#network-entity 20.0000.0000.0004.00
SwitchD(config-isis-1)#ipv6 enable topology standard
SwitchD(config-isis-1)#quit
SwitchD(config)#int vlan 300
SwitchD(config-vlanif-300)#isis enable 1
SwitchD(config-vlanif-300)#quit
```

检查结果

显示各交换机的 IS-IS database 信息，查看 LSP 是否完整

```
SwitchA(config)#show isis database

                  Database Information for IS-IS(1)

                    Level-1 Link State Database
The total number of isis 1 database is: 5
#    ID                  Level   Seq           Checksum  Lifetime  Length
ATT/P/OL
1    0000.0000.0001.00-00   1      5             11192    836       83
0/0/0
2    0000.0000.0002.00-00   1      5             36943    832       83
0/0/0
3    0000.0000.0003.00-00   1      9             11945    818       154
1/0/0
4    0000.0000.0003.01-00   1      1             10925    812       52
0/0/0
5    0000.0000.0003.02-00   1      1             12708    818       52
0/0/0
L1 Database Count :5
L2 Database Count :0

SwitchB(config)##show isis database
```

```
                  Database Information for IS-IS(1)
                    Level-1 Link State Database
The total number of isis 1 database is: 5
#    ID                  Level   Seq           Checksum  Lifetime  Length
ATT/P/OL
1    0000.0000.0001.00-00   1      5             11192    800      83
0/0/0
2    0000.0000.0002.00-00   1      5             36943    834      83
0/0/0
3    0000.0000.0003.00-00   1      9             11945    794      154
1/0/0
4    0000.0000.0003.01-00   1      1             10925    789      52
0/0/0
5    0000.0000.0003.02-00   1      1             12708    794      52
0/0/0
L1 Database Count :5
L2 Database Count :0

SwitchC(config)#show isis database
                  Database Information for IS-IS(1)

                    Level-1 Link State Database
The total number of isis 1 database is: 5
#    ID                  Level   Seq           Checksum  Lifetime  Length
ATT/P/OL
1    0000.0000.0001.00-00   1      5             11192    898      83
0/0/0
2    0000.0000.0002.00-00   1      5             36943    906      83
0/0/0
3    0000.0000.0003.00-00   1      9             11945    893      154
1/0/0
4    0000.0000.0003.01-00   1      1             10925    887      52
0/0/0
5    0000.0000.0003.02-00   1      1             12708    893      52
0/0/0

                    Level-2 Link State Database
The total number of isis 1 database is: 3
#    ID                  Level   Seq           Checksum  Lifetime  Length
ATT/P/OL
6    0000.0000.0003.00-00   2      6             55969    841      143
0/0/0
7    0000.0000.0003.03-00   2      1             18060    821      52
0/0/0
8    0000.0000.0004.00-00   2      6             8674     792      116
0/0/0
L1 Database Count :5
L2 Database Count :3

SwitchD(config)#show isis database

                  Database Information for IS-IS(1)

                    Level-2 Link State Database
```

```
The total number of isis 1 database is: 3
#    ID                  Level   Seq          Checksum  Lifetime  Length
ATT/P/OL
6    0000.0000.0003.00-00    2      6            55969     841      143
0/0/0
7    0000.0000.0003.03-00    2      1            18060     821      52
0/0/0
8    0000.0000.0004.00-00    2      6            8674      792      116
0/0/0
L1 Database Count :0
L2 Database Count :3
```

显示各交换机的 IS-IS 路由信息。Level-1 交换机的路由表中应该有一条缺省路由，且下一跳为 Level-1-2 交换机，Level-2 交换机的路由表中应该有所有 Level-1 和 Level-2 的路由。

```
SwitchA(config)##show isis-ipv6 route

ISIS process 1
Level    Dst                        Nexthop                        MAC
Cost   Time
Level-1  ::/0                       fe80::f2f1:f2ff:fef3:101
f0f1:f2f3:0101   10     0:02:07
Level-1  2001:1::/64                2001:1::2
0000:0000:0000   10     0:03:08
Level-1  2001:2::/64                fe80::f2f1:f2ff:fef3:101
f0f1:f2f3:0101   20     0:02:07
Level-1  2001:3::/64                fe80::f2f1:f2ff:fef3:101
f0f1:f2f3:0101   20     0:02:07
  Total:4

SwitchB(config)#show isis-ipv6 route

ISIS process 1
Level    Dst                        Nexthop                    MAC
Cost   Time
Level-1  ::/0                       fe80::f2f1:f2ff:fef3:101
f0f1:f2f3:0101   10     0:02:46
Level-1  2001:1::/64                 fe80::f2f1:f2ff:fef3:101
f0f1:f2f3:0101   20     0:02:46
Level-1  2001:2::/64                 2001:2::2
0000:0000:0000   10     0:03:18
Level-1  2001:3::/64                 fe80::f2f1:f2ff:fef3:101
f0f1:f2f3:0101   20     0:02:46
  Total:4

SwitchC(config)#show isis-ipv6 route

ISIS process 1
Level    Dst                        Nexthop                    MAC
Cost   Time
Level-1  2001:1::/64                 2001:1::1
0000:0000:0000   10     0:03:55
```

```
Level-1   2001:2::/64                  2001:2::1
0000:0000:0000    10     0:03:55
Level-1   2001:3::/64                  2001:3::1
0000:0000:0000    10     0:05:31
Level-2   2001:1::/64                  2001:1::1
0000:0000:0000    10     0:03:55
Level-2   2001:2::/64                  2001:2::1
0000:0000:0000    10     0:03:55
Level-2   2001:3::/64                  2001:3::1
0000:0000:0000    10     0:05:31
Level-2   2001:4::/64                  fe80::2e0:fcff:feda:53e9
00e0:fcda:53e9    10     0:05:25
  Total:7
```

3.12 RIP

3.12.1 简介

功能

RIP（Routing Information Protocol，路由信息协议）是一种较为简单的内部网关协议（Interior Gateway Protocol，IGP），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。

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 的基本概念

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 使用跳数来衡量到达目的地址的距离，跳数称为度量值。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 不适合应用于大型网络。

为提高性能，防止产生路由环路，RIP 支持水平分割（Split Horizon）和毒性逆转（Poison Reverse）功能。

RIP 的路由数据库

每个运行 RIP 的路由器管理一个路由数据库，该路由数据库包含了到所有可达目的地的路由项，这些路由项包含下列信息：

- 目的地址：主机或网络的地址。
- 下一跳地址：为到达目的地，需要经过的相邻路由器的接口 IP 地址。
- 出接口：本路由器转发报文的出接口。

- 度量值：本路由器到达目的地的开销。
- 路由时间：从路由项最后一次被更新到现在所经过的时间，路由项每次被更新时，路由时间重置为 0。
- 路由标记（Route Tag）：用于标识外部路由，在路由策略中可根据路由标记对路由信息进行灵活的控制。关于路由策略的详细信息，请参见“三层技术-IP 路由配置指导”中的“路由策略”。

RIP 防止路由环路的机制

RIP 协议向邻居通告的是自己的路由表，有可能会发生路由环路，可以通过以下机制来避免：

- 计数到无穷（Counting to infinity）：将度量值等于 16 的路由定义为不可达（infinity）。在路由环路发生时，某条路由的度量值将会增加到 16，该路由被认为不可达。
- 触发更新（Triggered Updates）：RIP 通过触发更新来避免在多个路由器之间形成路由环路的可能，而且可以加速网络的收敛速度。一旦某条路由的度量值发生了变化，就立刻向邻居路由器发布更新报文，而不是等到更新周期的到来。
- 水平分割（Split Horizon）：RIP 从某个接口学到的路由，不会从该接口再发回给邻居路由器。这样不但减少了带宽消耗，还可以防止路由环路。
- 毒性逆转（Poison Reverse）：RIP 从某个接口学到路由后，将该路由的度量值设置为 16（不可达），并从原接口发回邻居路由器。利用这种方式，可以清除对方路由表中的无用信息。

RIP 的版本

RIP 有两个版本：RIP-1 和 RIP-2。

RIP-1 是有类别路由协议（Classful Routing Protocol），它只支持以广播方式发布协议报文。RIP-1 的协议报文无法携带掩码信息，它只能识别 A、B、C 类这样的自然网段的路由，因此 RIP-1 不支持不连续子网（Discontiguous Subnet）。

RIP-2 是一种无类别路由协议（Classless Routing Protocol），与 RIP-1 相比，它有以下优势：

- 支持路由标记，在路由策略中可根据路由标记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，无类域间路由）。
- 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 RIP-2 路由器才能收到更新报文，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。

RIP-2 有两种报文传送方式：广播方式和组播方式，缺省将采用组播方式发送报文，使用的组播地址为 224.0.0.9。当接口运行 RIP-2 广播方式时，也可接收 RIP-1 的报文。

3.12.2 配置准备

场景

完成配置 RIP 的基本功能后，即可以实现通过 RIP 协议构建三层网络

前提

配置接口的 IP 地址，使各相邻节点网络层可达。

3.12.3 缺省配置

设备上 RIP 的缺省配置如下。

功能	缺省值
SNMP Trap	不使能

3.12.4 配置 RIP 基本功能

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**rip** [*process-id*] [**vpn-instance** *NAME*]	创建并进入 RIP 进程配置视图。
3	JX(config-rip-1)#**network** *network-address*	在指定网段上使能 RIP。
4	JX(config-rip-1)#**quit**	退出当前视图，进入全局配置模式。
5	JX(config)#**interface** *interface-type interface-number*	进入接口视图，该接口工作在三层模式的物理接口或者 VLAN。
6	JX(config-vlanif-1)#**rip** *process-id* enable	接口下使能 RIP 协议。
7	JX(config-vlanif-1)#**rip passive-interface** { **enable** | **disable** }	配置接口为被动接口，缺省情况下去去使能，允许所有接口发送路由更新报文，配置为使能接口不发送 RIP 报文。
8	JX(config-vlanif-1)#**rip send-version** { **no-send** | **v1** | **v1-compatible** | **v2** }	配置接口 RIP 发送版本。
9	JX(config-vlanif-1)#**rip receive-version** { **v1** | **v2** | **v1v2** | **no-receive** }	配置接口 RIP 接收版本。

3.12.5 配置 RIP 的路由信息控制

步骤	配置	说明
1	JX#config	进入全局配置模式。
2	JX(config)#**rip** [*process-id*] [**vpn-instance** *NAME*]	创建并进入 RIP 进程配置视图。
3	JX(config-rip-1)#**summary** { **enable** | **disable** }	使能 RIP-2 自动路由聚合功能，默认去使能。
4	JX(config-rip-1)#**default-route originate** {**cost** (<*0-15*>|**default**)|**tag** (<*0-65535*>|**default**)}	配置 RIP 发布缺省路由，默认不发布缺省路由。
5	JX(config-rip-1)#**default-route learning** (**enable** | **disable**)	配置 RIP 学习缺省路由，默认值为使能。
6	JX(config-rip-1)#**import-route** { **rip** | **isis** | **ospf**) <*1-65535*> {**cost** (<*0-15*>|**default** } | **route-policy** *NAME*} JX(config-rip-1)#**import-route** { **static** | **connect** | **ospf** | **bgp** | **isis** | **rip** } { **cost** (<*0-15*>|**default**)|**route-policy** *NAME*}	引入外部路由。
7	JX(config-rip-1)#**default-cost** (<*0-15*>|**default**)	配置引入路由的缺省度量值，默认值为 1。
8	JX(config-rip-1)#**filter-policy import route-policy** *NAME*	配置 RIP 进程下对接收的路由过滤。本命令对从邻居收到的 RIP 路由进行过滤，没有通过过滤的路由将不被加入路由表，也不向邻居发布该路由
9	JX(config-rip-1)#**filter-policy export route-policy** *NAME* JX(config-rip-1)#**filter-policy export** { **rip** | **isis** | **ospf**) <1-65535> **route-policy** *NAME* JX(config-rip-1)#**filter-policy export** { **rip** | **isis** | **ospf**) <1-65535> **route-policy** *NAME* JX(config-rip-1)#**filter-policy export** { **static** | **connect** | **ospf** | **bgp** | **isis** | **rip** } **route-policy** *NAME*	配置 RIP 进程下对发送的路由过滤。本命令对本机所有路由的发布进行过滤，包括使用 **import-route** 引入的路由和从邻居学到的 RIP 路由。
10	JX(config-rip-1)#**preference** (<1-255>|**default**)	配置 RIP 路由优先级，默认值为 100。
11	JX(config-rip-1)#**quit**	退出当前视图，进入全局配置模式。

步骤	配置	说明
12	JX(config)#**interface** *interface-type interface-number*	进入接口视图，该接口工作在三层模式的物理接口或者 VLAN。
13	JX(config-vlanif-1)#**rip metric-in** (<0-15>|**default**)	配置接口接收 RIP 路由时的附加度量值，默认值为 0。
14	JX(config-vlanif-1)#**rip metric-out** (<1-15>|**default**)	配置接口发送 RIP 路由时的附加度量值，默认值为 0。
15	JX(config-vlanif-1)#**rip default-metric** (<0-15>|**default**)	接口发布默认路由开销值，默认值为 1。
16	JX(config-vlanif-1)#**rip summary-address** *A.B.C.D A.B.C.D* [**avoid-feedback**]	接口下配置聚合路由。
17	JX(config-vlanif-1)#**rip filter-policy** (**export** | **import**) **route-policy** *NAME*	配置接口下对发送、接收路由过滤。

3.12.6 配置调整和优化 RIP 网络

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**rip** [*process-id*] [**vpn-instance** *NAME*]	创建并进入 RIP 进程配置视图。
3	JX(config-rip-1)#**rip timer** (<10-5600>|**default**) (<40-14400>|**default**) (<40-14400>|**default**)	缺省情况下，Garbage-collect 定时器的值为 120 秒，Timeout 定时器的值为 180 秒，Update 定时器的值为 30 秒。
4	JX(config-rip-1)#**check-zero** { **enable** | **disable** }	配置 RIP-1 报文的零域检查功能，默认使能。
5	JX(config-rip-1)#**validate-source-address** { **enable** | **disable** }	使能对接收到的 RIP 路由更新报文进行源 IP 地址检查功能，默认使能。
6	JX(config-rip-1)#**neighbor** *neighbor-address*	配置 RIP 邻居。
7	JX(config-rip-1)#**quit**	退出当前视图，进入全局配置模式。
8	JX(config)#**interface** *interface-type interface-number*	进入接口视图，该接口工作在三层模式的物理接口或者 VLAN。
9	JX(config-vlanif-1)#**rip split-horizon** { **enable** | **disable**)	接口下使能水平分割功能，默认使能。
10	JX(config-vlanif-1)#**rip poison-reverse** { **enable** | **disable** }	接口下使能毒性逆转功能，默认去使能。

步骤	配置	说明
11	JX(config-vlanif-1)#**rip authentication** { **md5** | **md5-compatible** } <1-255> (**cipher** | **plain**) *KEY* JX(config-vlanif-1)#**rip authentication simple** { **cipher** | **plain**) *KEY*	配置 RIP 接口认证。
12	JX(config-vlanif-1)#**rip pkt-transmit interval** (<10-500>|**default**) **number** (<1-50>|**default**)	配置接口下 RIP 报文的发送速率，缺省情况下，接口发送 RIP 报文的时间间隔为 200 毫秒，一次最多发送 20 个 RIP 报文。
13	JX(config-vlanif-1)#**rip bfd** { **enable** | **disable** }	接口下使能 RIP BFD 联动功能。

3.12.7 配置检查和维护

步骤	配置	说明
1	JX#**show rip config**	查看 RIP 全部配置信息。
2	JX#**show rip resource**	查看 RIP 资源信息。
3	JX#**show rip** [**process** *process-id*]	查看 RIP 进程信息。
4	JX#**show rip neighbor** [**process** *process-id*|**address** *neighbor-address*]	查看 RIP 邻居信息。
5	JX#**show rip interface** [**process** *process-id* | **vlan** <1-4094>]	查看 RIP 接口信息。
6	JX#**show rip bfd session** [**process** *process-id*]	查看 RIP BFD 会话信息。
7	JX#**show rip** { **database** | **route** } [**process** *process-id* | **dest** *dest-address/dest-mask* | **neighbor** *neighbor-address*]	查看 RIP 路由信息，database 选项查看数据库信息，route 选项查看生效路由信息。

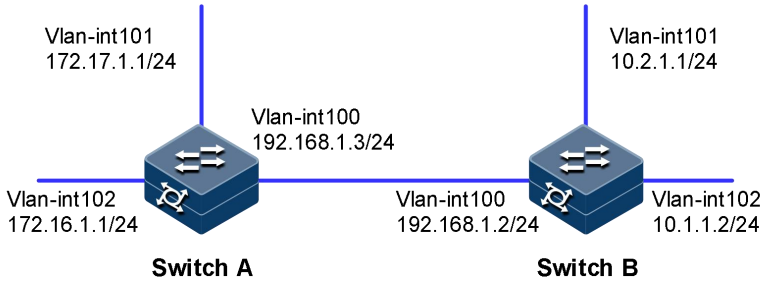
3.12.8 配置 RIP 基本功能实示例

组网需求

如下图所示，在 Switch A 和 Switch B 的所有接口上使能 RIP，并使用 RIP-2 进行网络互连。

在 Switch B 上配置路由出策略，向 Switch A 发布的路由中过滤掉 10.2.1.0/24；Switch B 上配置入策略，使得 Switch B 只接收路由 172.16.1.0/24。

图 3-47 RIP 基本功能组网图



配置步骤

配置各接口的 IP 地址。

配置 RIP 基本功能，使能 RIP 功能在下面采用了两种不同配置方式，请根据实际情况进行选择。

配置 Switch A，在指定网段上使能 RIP。

```
SwitchA#configure
SwitchA(config)#rip 1
SwitchA(config-rip-1)#network 172.16.0.0
SwitchA(config-rip-1)#network 172.17.0.0
SwitchA(config-rip-1)#quit
```

配置 Switch B，在指定接口上使能 RIP。

```
SwitchA#configure
SwitchA(config)#rip 1
SwitchA(config-rip-1)#network 172.16.0.0
SwitchA(config-rip-1)#quit
SwitchA(config)#interface vlan 100
SwitchA(config-vlanif-100)#rip 1 enable
SwitchA(config-vlanif-100)#quit
SwitchA(config)#interface vlan 101
SwitchA(config-vlanif-101)#rip 1 enable
SwitchA(config-vlanif-101)#quit
SwitchA(config)#interface vlan 102
SwitchA(config-vlanif-102)#rip 1 enable
SwitchA(config-vlanif-102)#quit
```

验证配置结果。

Swicth A:

查看接口信息：

```
SwitchA#show rip interface
Interface        Process State Addr/Prefix      Send-Version     Recv-
Version
-------------------------------------------------------------------------
-----------------
vlan-102         1      Up    172.16.1.1/24     RIPv1 Compatible  RIPv1 and
RIPv2
```

```
vlan-101         1      Up    172.17.1.1/24    RIPv1 Compatible  RIPv1 and
RIPv2
vlan-100         1      Up    192.168.1.3/24   RIPv1 Compatible  RIPv1 and
RIPv2
```

查看邻居信息：

```
SwitchA#show rip neighbor
 Process   Address         Interface      Domain   Version Update-Time
RecvBadPkt RecvBadRoute  Key
-------------------------------------------------------------------------
-----------------------------------
 1       192.168.1.2      vlan-100              2      00:06:30    0
0           0
-------------------------------------------------------------------------
-----------------------------------
 The total number of rip neighbor is: 1
Swicth B:
SwitchB#show rip neighbor
 Process   Address         Interface      Domain   Version Update-Time
RecvBadPkt RecvBadRoute  Key
-------------------------------------------------------------------------
-----------------------------------
 1       192.168.1.3      vlan-100              2      00:07:08    0
0           0
-------------------------------------------------------------------------
-----------------------------------
 The total number of rip neighbor is: 1
```

查看数据库信息：

```
SwitchA#show rip database
 Process   Destination    Netmask        Gateway        Metric  Age
State   Proto   Tag
-------------------------------------------------------------------------
--------------------------
 1       10.1.1.0       255.255.255.0   192.168.1.2    1      18
ACTIVE  rip    0
 1       10.2.1.0       255.255.255.0   192.168.1.2    1      18
ACTIVE  rip    0
 1       172.16.1.0     255.255.255.0   172.16.1.1     0      0
ACTIVE  rip    0
 1       172.17.1.0     255.255.255.0   172.17.1.1     0      0
ACTIVE  rip    0
 1       192.168.1.0    255.255.255.0   192.168.1.3    0      0
ACTIVE  rip    0
-------------------------------------------------------------------------
--------------------------
 Total : 5
```

Switch B：

查看接口信息：

```
SwitchA#show rip interface
```

```
Interface        Process State Addr/Prefix      Send-Version     Recv-
Version
-------------------------------------------------------------------------
-----------------
vlan-102         1      Up    172.16.1.1/24     RIPv1 Compatible  RIPv1 and
RIPv2
vlan-101         1      Up    172.17.1.1/24     RIPv1 Compatible  RIPv1 and
RIPv2
vlan-100         1      Up    192.168.1.3/24    RIPv1 Compatible  RIPv1 and
RIPv2
-------------------------------------------------------------------------
-----------------
```

查看邻居信息：

```
SwitchA#show rip neighbor
 Process   Address         Interface      Domain    Version Update-Time
RecvBadPkt RecvBadRoute  Key
-------------------------------------------------------------------------
-------------------------------------
 1       192.168.1.2      vlan-100              2      00:09:41    0
0           0
-------------------------------------------------------------------------
-------------------------------------
 The total number of rip neighbor is: 1
```

查看数据库信息：

```
SwitchA#show rip database
 Process   Destination    Netmask       Gateway       Metric  Age
State   Proto   Tag
-------------------------------------------------------------------------
---------------------------
 1       10.1.1.0       255.255.255.0   192.168.1.2    1      23
ACTIVE  rip    0
 1       10.2.1.0       255.255.255.0   192.168.1.2    1      23
ACTIVE  rip    0
 1       172.16.1.0     255.255.255.0   172.16.1.1     0      0
ACTIVE  rip    0
 1       172.17.1.0     255.255.255.0   172.17.1.1     0      0
ACTIVE  rip    0
 1       192.168.1.0    255.255.255.0   192.168.1.3    0      0
ACTIVE  rip    0
-------------------------------------------------------------------------
---------------------------
 Total : 5
```

配置 RIP 路由过滤。

```
SwitchB(config)#ip prefix-list import index 10 permit 172.16.1.0/24
SwitchB(config)#ip prefix-list export index 10 permit 10.1.1.0/24
SwitchB(config)#route-policy import permit node 1
SwitchB(config-route-policy-import-1)#match destination ip-prefix-list
import
SwitchB(config-route-policy-import-1)#quit
SwitchB(config)#route-policy export permit node 1
```

```
SwitchB(config-route-policy-export-1)#match destination ip-prefix-list
export
SwitchB(config-route-policy-import-1)#quit
SwitchB(config)#rip 1
SwitchB(config-rip-1)#filter-policy import route-policy import
SwitchB(config-rip-1)#filter-policy export route-policy export
```

验证配置结果。

Switch A:

查看数据库信息：

```
SwitchA#show rip database
 Process   Destination    Netmask        Gateway        Metric  Age
State   Proto   Tag
------------------------------------------------------------------------
--------------------------
 1         10.1.1.0       255.255.255.0   192.168.1.2    1      18
ACTIVE  rip    0
 1         172.16.1.0     255.255.255.0   172.16.1.1     0      0
ACTIVE  rip    0
 1         172.17.1.0     255.255.255.0   172.17.1.1     0      0
ACTIVE  rip    0
 1         192.168.1.0    255.255.255.0   192.168.1.3    0      0
ACTIVE  rip    0
------------------------------------------------------------------------
--------------------------
 Total : 4
```

Switch B:

查看数据库信息：

```
SwitchB#show rip database
 Process   Destination    Netmask        Gateway        Metric  Age
State   Proto   Tag
------------------------------------------------------------------------
--------------------------
 1         10.1.1.0       255.255.255.0   10.1.1.2       0      0      ACTIVE
rip    0
 1         10.2.1.0       255.255.255.0   10.2.1.2       0      0      ACTIVE
rip    0
 1         172.16.1.0     255.255.255.0   192.168.1.3    1      20
ACTIVE  rip    0
 1         192.168.1.0    255.255.255.0   192.168.1.2    0      0
ACTIVE  rip    0
------------------------------------------------------------------------
--------------------------
```

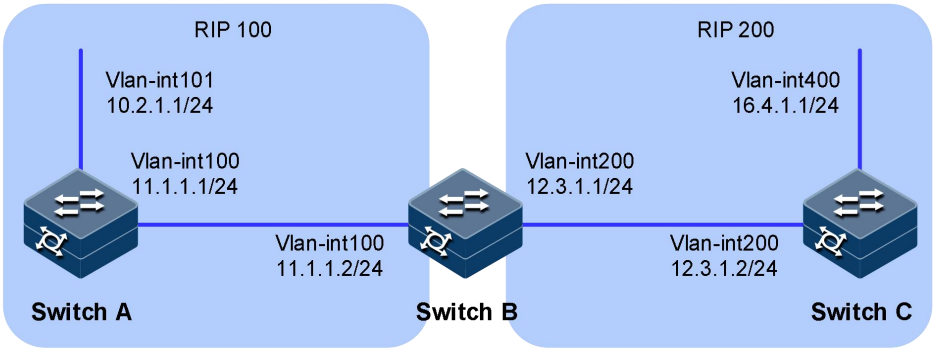
3.12.9 配置 RIP 引入外部路由

组网需求

Switch B 上运行两个 RIP 进程：RIP 100 和 RIP 200。Switch B 通过 RIP 100 和 Switch A 交换路由信息，通过 RIP 200 和 Switch C 交换路由信息。

在 Switch B 上配置 RIP 进程 200 引入外部路由，引入直连路由和 RIP 进程 100 的路由，使得 Switch C 能够学习到达 10.2.1.0/24 和 11.1.1.0/24 的路由，但 Switch A 不能学习到达 12.3.1.0/24 和 16.4.1.0/24 的路由。

图 3-48 RIP 引入外部路由配置组网图



配置步骤

配置各接口的 IP 地址。

设备配置 RIP 功能。

配置 Switch A。

```
SwitchA#configure
SwitchA(config)#rip 100
SwitchA(config-rip-100)#network 10.0.0.0
SwitchA(config-rip-100)#network 11.0.0.0
SwitchA(config-rip-100)#quit
```

配置 Switch B。

```
SwitchB#configure
SwitchB(config)#rip 100
SwitchB(config-rip-100)#network 11.0.0.0
SwitchB(config-rip-100)#quit
SwitchB(config)#rip 200
SwitchB(config-rip-200)#network 12.0.0.0
SwitchB(config-rip-1)#quit
```

配置 Switch C。

```
SwitchC#configure
SwitchC(config)#rip 200
SwitchC(config-rip-200)#network 12.0.0.0
SwitchC(config-rip-200)#network 16.0.0.0
```

```
SwitchC(config-rip-200)#quit
```

验证配置结果。

查看 Switch C RIP 数据库。

```
SwitchC#show rip database
 Process   Destination    Netmask        Gateway        Metric  Age
State   Proto   Tag
------------------------------------------------------------------------------
---------------------------
 200       12.3.1.0       255.255.255.0  12.3.1.2       0       0       ACTIVE
rip     0
 200       16.4.1.0       255.255.255.0  16.4.1.1       0       0       ACTIVE
rip     0
------------------------------------------------------------------------------
---------------------------
```

配置引入路由。

Switch B 进程 200 引入进程 100 路由。

```
SwitchB#configure
SwitchB(config)#rip 200
SwitchB(config-rip-200)#import-route rip 100
SwitchB(config-rip-200)#quit
```

验证配置结果。

查看 Switch C RIP 数据库。

```
SwitchC#show rip database
 Process   Destination    Netmask        Gateway        Metric  Age
State   Proto   Tag
------------------------------------------------------------------------------
---------------------------
 200       10.2.1.0       255.255.255.0  12.3.1.1       2       1       ACTIVE
rip     0
 200       11.1.1.0       255.255.255.0  12.3.1.1       2       1       ACTIVE
rip     0
 200       12.3.1.0       255.255.255.0  12.3.1.2       0       0       ACTIVE
rip     0
 200       16.4.1.0       255.255.255.0  16.4.1.1       0       0       ACTIVE
rip     0
------------------------------------------------------------------------------
---------------------------
 Total : 4
```

3.13 RIPng

3.13.1 简介

功能

RIPng（RIP next generation，下一代 RIP 协议）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。

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

- 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。
- 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 FE80::/10 作为源地址发送 RIPng 路由信息更新报文。

RIPng 工作机制

RIPng 协议是基于距离矢量（Distance-Vector）算法的协议。它通过 UDP 报文交换路由信息，使用的端口号为 521。

RIPng 使用跳数来衡量到达目的地址的距离（也称为度量值或开销）。在 RIPng 中，从一个路由器到其直连网络的跳数为 0，通过与其相连的路由器到达另一个网络的跳数为 1，其余以此类推。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。

RIPng 每 30 秒发送一次路由更新报文。如果在 180 秒内没有收到网络邻居的路由更新报文，RIPng 将从邻居学到的所有路由标识为不可达。如果再过 120 秒内仍没有收到邻居的路由更新报文，RIPng 将从路由表中删除这些路由。

为了提高性能并避免形成路由环路，RIPng 既支持水平分割也支持毒性逆转。此外，RIPng 还可以从其它的路由协议引入路由。

每个运行 RIPng 的路由器都管理一个路由数据库，该路由数据库包含了到所有可达目的地的路由项，这些路由项包含下列信息：

- 目的地址：主机或网络的 IPv6 地址。
- 下一跳地址：为到达目的地，需要经过的相邻路由器的接口 IPv6 地址。
- 出接口：转发 IPv6 报文通过的出接口。
- 度量值：本路由器到达目的地的开销。
- 路由时间：从路由项最后一次被更新到现在所经过的时间，路由项每次被更新时，路由时间重置为 0。
- 路由标记（Route Tag）：用于标识外部路由，以便在路由策略中根据 Tag 对路由进行灵活的控制。

RIPng 报文

RIPng 有两种报文：Request 报文和 Response 报文。

当 RIPng 路由器启动后或者需要更新部分路由表项时，便会发出 Request 报文，向邻居请求需要的路由信息。通常情况下以组播方式发送 Request 报文。

Response 报文包含本地路由表的信息，一般在下列情况下产生：

- 对某个 Request 报文进行响应
- 作为更新报文周期性地发出
- 在路由发生变化时触发更新

收到 Request 报文的 RIPng 路由器会以 Response 报文形式发回给请求路由器。

收到 Response 报文的路由器会更新自己的 RIPng 路由表。为了保证路由的准确性，RIPng 路由器会对收到的 Response 报文进行有效性检查，比如源 IPv6 地址是否是链路本地地址，端口号是否正确等，没有通过检查的报文会被忽略。

3.13.2 配置准备

场景

完成配置 RIPng 的基本功能后，即可以实现通过 RIPng 协议构建 IPv6 三层网络。

前提

配置接口的 IPv6 功能和 IPv6 地址，使各相邻节点网络层可达。

3.13.3 配置 RIPng 基本功能

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**ripng** [*process-id*] [**vpn-instance** *NAME*]	创建并进入 RIPng 进程配置视图。
4	JX(config-rip-1)#**quit**	退出当前视图，进入全局配置模式。
5	JX(config)#**interface** *interface-type interface-number*	进入接口视图，该接口工作在三层模式的物理接口或者 VLAN。
6	JX(config-vlanif-1)#**ripng** *process-id* **enable**	接口下使能 RIPng 协议。

3.13.4 配置 RIPng 的路由特性

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**ripng** [*process-id*] [**vpn-instance** *NAME*]	创建并进入 RIPng 进程配置视图。

步骤	配置	说明
3	JX(config-ripng-1)#**default-route originate** { **enable** | **disable** }	配置 RIPng 发布缺省路由，默认不发布缺省路由。
4	JX(config-ripng-1)#**default-route learning** { **enable** | **disable** }	配置 RIPng 学习缺省路由，默认值为使能。
5	JX(config-ripng-1)#**import-route** { **ripng** | **isis** | **ospf** } <1-65535> {**cost** (<0-15>|**default**)|**route-policy** *NAME*} JX(config-ripng-1)#**import-route** { **static** | **connect** | **bgp** | **ospf** | **isis** | **ripng** } {**cost** (<0-15>|**default**)|**route-policy** *NAME*}	引入外部路由。
6	JX(config-ripng-1)#**default-cost** (<0-15>|**default**)	配置引入路由的缺省度量值，默认值为 1。
7	JX(config-ripng-1)#**filter-policy import route-policy** *NAME*	配置 RIP 进程下对接收的路由过滤。本命令对从邻居收到的 RIP 路由进行过滤，没有通过过滤的路由将不被加入路由表，也不向邻居发布该路由
8	JX(config-ripng-1)#**filter-policy export route-policy** *NAME* JX(config-ripng-1)#**filter-policy export** { **ripng** | **isis** | **ospf** } <1-65535> **route-policy** *NAME* JX(config-ripng-1)#**filter-policy export** { **static** | **connect** | **ospf** | **bgp** | **isis** | **ripng** } **route-policy** *NAME*	配置 RIP 进程下对发送的路由过滤。本命令对本机所有路由的发布进行过滤，包括使用 **import-route** 引入的路由和从邻居学到的 RIPng 路由。
9	JX(config-ripng-1)#**preference** (<1-255>|**default**)	配置 RIPng 路由优先级，默认值为 100。
10	JX(config-ripng-1)#**quit**	退出当前视图，进入全局配置模式。
11	JX(config)#**interface** *interface-type interface-number*	进入接口视图，该接口工作在三层模式的物理接口或者 VLAN。
12	JX(config-vlanif-1)#**ripng metric-in** (<0-15>|**default**)	配置接口接收 RIPng 路由时的附加度量值，默认值为 0。
13	JX(config-vlanif-1)#**rip metric-out** (<1-15>|**default**)	配置接口发送 RIPng 路由时的附加度量值，默认值为 0。
15	JX(config-vlanif-1)#**ripng default-route** { **only** | **originate**) {**cost** (<1-15>|**default**)}	该接口生成一条缺省路由到 RIPng 路由域中。
16	JX(config-vlanif-1)#**ripng summary-address** *X:X::X:X* <0-128> {**avoid-feedback**}	接口下配置聚合路由。
17	JX(config-vlanif-1)#**ripng filter-policy** { **export** | **import**) **route-policy** *NAME*	配置接口下对发送、接收路由过滤。

3.13.5 配置调整和优化 RIPng 网络

步骤	配置	说明
1	JX#config	进入全局配置模式。
2	JX(config)#**ripng** [*process-id*] [**vpn-instance** *NAME*]	创建并进入 RIPng 进程配置视图。
3	JX(config-ripng-1)#**timers update** (<30-500>|**default**) **expire** (<120-2000>|**default**) **garbage** (<120-2000>|**default**)	缺省情况下，Garbage-collect 定时器的值为 120 秒，Timeout 定时器的值为 180 秒，Update 定时器的值为 30 秒。
4	JX(config-ripng-1)#**check-zero** (**enable** | **disable**)	配置 RIPng 报文的零域检查功能，默认使能。
5	JX(config-rip-1)#**quit**	退出当前视图，进入全局配置模式。
6	JX(config)#**interface** *interface-type interface-number*	进入接口视图，该接口工作在三层模式的物理接口或者 VLAN。
7	JX(config-vlanif-1)#**ripng split-horizon** { **enable** | **disable** }	接口下使能水平分割功能，默认使能。
8	JX(config-vlanif-1)#**ripng poison-reverse** { **enable** | **disable** }	接口下使能毒性逆转功能，默认去使能。
9	JX(config-vlanif-1)#**ripng pkt-transmit interval** { <10-500> | **default** } **number** (<1-50>|**default**)	配置接口下 RIPng 报文的发送速率，缺省情况下，接口发送 RIP 报文的时间间隔为 200 毫秒，一次最多发送 10 个 RIP 报文。
10	JX(config-vlanif-1)#**ripng bfd** { **enable** | **disable** }	接口下使能 RIP BFD 联动功能。

3.13.6 配置检查和维护

步骤	配置	说明
1	JX#**show ripng config**	查看 RIPng 全部配置信息。
2	JX#**show ripng resource**	查看 RIPng 资源信息。
3	JX#**show ripng** [**process** *process-id*]	查看 RIPng 进程信息。
4	JX#**show ripng neighbor** [**process** *process-id*|**address** *neighbor-address*]	查看 RIPng 邻居信息。
5	JX#**show ripng interface** [**process** *process-id* | **vlan** <1-4094>]	查看 RIPng 接口信息。

6	JX#**show ripng bfd session** [**process** *process-id*]	查看 RIPngBFD 会话信息。
7	JX#**show ripng** { **database** | **route** } [**process** *process-id* | **dest** *dest-address/dest-mask* | **neighbor** *neighbor-address*]	查看 RIPng 路由信息，database 选项查看数据库信息，route 选项查看生效路由信息。

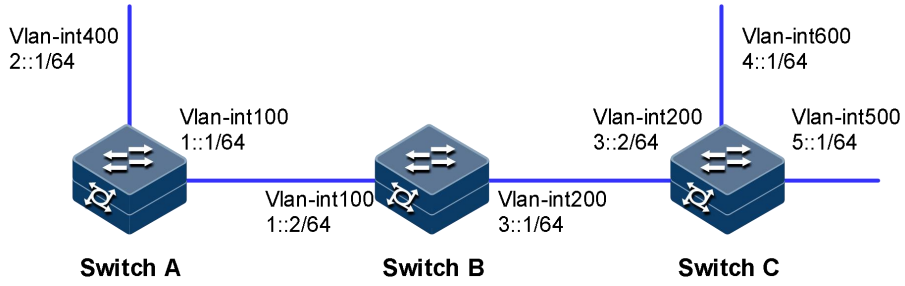
3.13.7 配置 RIPng 基本功能实示例

组网需求

Switch A、Switch B 和 Switch C 相连并通过 RIPng 来学习网络中的 IPv6 路由信息。

在 Switch B 上对接收的 Switch A 的路由（2::/64）进行过滤，使其不加入到 Switch B 的 RIPng 进程的路由表中，发布给 Switch A 的路由只有（4::/64）。

图 3-49 RIPng 基本功能组网图



配置步骤

步骤 1 配置各接口的 IPv6 地址。

步骤 2 配置 RIPng 基本功能。

配置 Switch A。

```
SwitchA#configure
SwitchA(config)#ripng 1
SwitchA(config-ripng-1)#quit
SwitchA(config)#interface vlan 100
SwitchA(config-vlanif-100)#ripng 1 enable
SwitchA(config-vlanif-100)#quit
SwitchA(config)#interface vlan 400
SwitchA(config-vlanif-400)#ripng 1 enable
SwitchA(config-vlanif-400)#quit
```

配置 Switch B。

```
SwitchB#configure
SwitchB(config)#ripng 1
SwitchB(config-ripng-1)#quit
SwitchB(config)#interface vlan 100
SwitchB(config-vlanif-100)#ripng 1 enable
```

```
SwitchB(config-vlanif-100)#quit
SwitchB(config)#interface vlan 200
SwitchB(config-vlanif-200)#ripng 1 enable
SwitchB(config-vlanif-200)#quit
```

配置 Switch C。

```
SwitchC#configure
SwitchC(config)#ripng 1
SwitchC(config-ripng-1)#quit
SwitchC(config)#interface vlan 200
SwitchC(config-vlanif-200)#ripng 1 enable
SwitchC(config-vlanif-200)#quit
SwitchC(config)#interface vlan 500
SwitchC(config-vlanif-500)#ripng 1 enable
SwitchC(config-vlanif-500)#quit
SwitchC(config)#interface vlan 600
SwitchC(config-vlanif-600)#ripng 1 enable
SwitchC(config-vlanif-600)#quit
```

步骤 3 验证配置结果。

Switch A:

查看 RIPng 接口信息：

```
SwitchA#show ripng interface
Interface     Process   State      Passive     Address
------------------------------------------------------------------------------
---------------------------
vlan-100      1         Up         Disable     fe80::f2f1:f2ff:fef3:101
vlan-400      1         Up         Disable     fe80::f2f1:f2ff:fef3:101
------------------------------------------------------------------------------
---------------------------
```

查看 RIPng 邻居信息：

```
SwitchA#show ripng neighbor
Process   Address                    Version   LastUpdate   Interface
InbadPacket   InbadRoute
1       fe80::f2f1:f2ff:fef3:201       Version1  00:06:55    vlan-100
0           0
```

查看 RIPng 路由信息：

```
SwitchA#show ripng database
Process   Prefix/Prefixlen      Nexthop                  Metric  Age
State   Protocol   Tag
------------------------------------------------------------------------------
-----------------------------------------------
1       1::/64              fe80::f2f1:f2ff:fef3:101     0      0
Active  ripng      0
1       2::/64              fe80::f2f1:f2ff:fef3:101     0      0
Active  ripng      0
1       3::/64              fe80::f2f1:f2ff:fef3:201     1      14
Active  ripng      0
```

```
1       4::/64              fe80::f2f1:f2ff:fef3:201     2      14
Active  ripng      0
1       5::/64              fe80::f2f1:f2ff:fef3:201     2      14
Active  ripng      0
------------------------------------------------------------------------
------------------------------------------
 Total : 5
```

Switch B:

查看 RIPng 接口信息：

```
SwitchB#show ripng interface
Interface     Process   State      Passive     Address
------------------------------------------------------------------------
--------------------------
vlan-100      1       Up         Disable     fe80::f2f1:f2ff:fef3:201
vlan-200      1       Up         Disable     fe80::f2f1:f2ff:fef3:201
------------------------------------------------------------------------
--------------------------
```

查看 RIPng 邻居信息：

```
SwitchB#show ripng neighbor
Process   Address                     Version   LastUpdate   Interface
InbadPacket   InbadRoute
1       fe80::f2f1:f2ff:fef3:101      Version1  00:07:35    vlan-100
0           0
1       fe80::f2f1:f2ff:fef3:301      Version1  00:07:21    vlan-200
0           0
```

查看 RIPng 路由信息：

```
SwitchB#show ripng database
Process   Prefix/Prefixlen      Nexthop                  Metric  Age
State   Protocol   Tag
------------------------------------------------------------------------
------------------------------------------
1       1::/64              fe80::f2f1:f2ff:fef3:201     0      0
Active  ripng      0
1       2::/64              fe80::f2f1:f2ff:fef3:101     1      1
Active  ripng      0
1       3::/64              fe80::f2f1:f2ff:fef3:201     0      0
Active  ripng      0
1       4::/64              fe80::f2f1:f2ff:fef3:301     1      9
Active  ripng      0
1       5::/64              fe80::f2f1:f2ff:fef3:301     1      9
Active  ripng      0
------------------------------------------------------------------------
------------------------------------------
 Total : 5
```

Switch C:

查看 RIPng 接口信息：

```
SwitchC#show ripng interface
Interface     Process   State      Passive     Address
```

```
-------------------------------------------------------------------------
--------------------------
vlan-200      1       Up        Disable     fe80::f2f1:f2ff:fef3:301
vlan-500      1       Up        Disable     fe80::f2f1:f2ff:fef3:301
vlan-600      1       Up        Disable     fe80::f2f1:f2ff:fef3:301
-------------------------------------------------------------------------
--------------------------
```

查看 RIPng 邻居信息：

```
SwitchC#show ripng neighbor
Process   Address                     Version   LastUpdate   Interface
InbadPacket   InbadRoute
1       fe80::f2f1:f2ff:fef3:201      Version1  00:08:01    vlan-200
0           0
```

查看 RIPng 路由信息：

```
SwitchC#show ripng database
Process   Prefix/Prefixlen      Nexthop                  Metric  Age
State   Protocol   Tag
-------------------------------------------------------------------------
-----------------------------------------------
1       1::/64              fe80::f2f1:f2ff:fef3:201     1      27
Active  ripng      0
1       2::/64              fe80::f2f1:f2ff:fef3:201     2      27
Active  ripng      0
1       3::/64              fe80::f2f1:f2ff:fef3:301     0      0
Active  ripng      0
1       4::/64              fe80::f2f1:f2ff:fef3:301     0      0
Active  ripng      0
1       5::/64              fe80::f2f1:f2ff:fef3:301     0      0
Active  ripng      0
-------------------------------------------------------------------------
-----------------------------------------------
 Total : 5
```

步骤 4 配置路由过滤。

```
SwitchB(config)#ipv6 prefix-list export index 10 permit 4::/64
SwitchB(config)#ipv6 prefix-list import index 10 permit 2::/64
SwitchB(config)#route-policy export permit node 1
SwitchB(config-route-policy-export-1)#match destination ipv6-prefix-list
export
SwitchB(config-route-policy-export-1)#quit
SwitchB(config)#route-policy import deny node 1
SwitchB(config-route-policy-import-1)#match destination ipv6-prefix-list
import
SwitchB(config-route-policy-import-1)#quit
SwitchB(config)#route-policy import permit node 2
SwitchB(config-route-policy-import-2)#quit
SwitchB(config)#ripng 1
SwitchB(config-ripng-1)#filter-policy import route-policy import
SwitchB(config-ripng-1)#filter-policy export route-policy export
```

步骤 5 验证配置结果。

查看 Switch B 和 Switch A 的 RIPng 路由表。

```
SwitchA#show ripng database
Process   Prefix/Prefixlen      Nexthop                  Metric  Age
State   Protocol    Tag
------------------------------------------------------------------------------
-----------------------------------------------
1       1::/64              fe80::f2f1:f2ff:fef3:101     0      0
Active  ripng      0
1       2::/64              fe80::f2f1:f2ff:fef3:101     0      0
Active  ripng      0
1       4::/64              fe80::f2f1:f2ff:fef3:201     2      12
Active  ripng      0
------------------------------------------------------------------------------
-----------------------------------------------
 Total : 3

SwitchB#show ripng database
Process   Prefix/Prefixlen      Nexthop                  Metric  Age
State   Protocol    Tag
------------------------------------------------------------------------------
-----------------------------------------------
1       1::/64              fe80::f2f1:f2ff:fef3:201     0      0
Active  ripng      0
1       3::/64              fe80::f2f1:f2ff:fef3:201     0      0
Active  ripng      0
1       4::/64              fe80::f2f1:f2ff:fef3:301     1      19
Active  ripng      0
1       5::/64              fe80::f2f1:f2ff:fef3:301     1      19
Active  ripng      0
------------------------------------------------------------------------------
-----------------------------------------------
 Total : 4
```

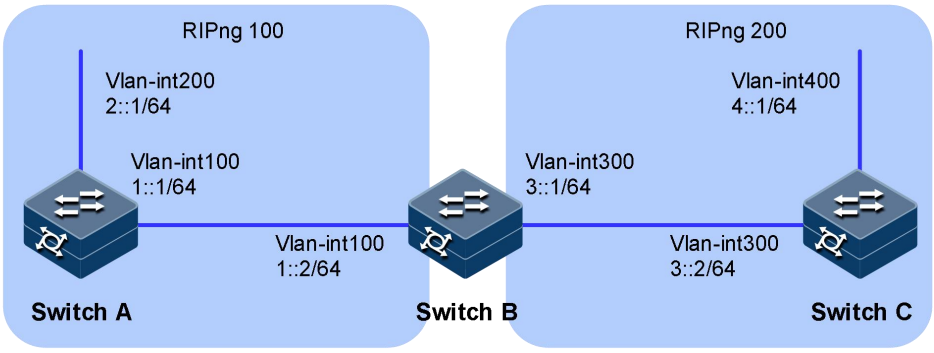
3.13.8 配置 RIPng 引入外部路由

组网需求

Switch B 上运行两个 RIPng 进程：RIPng100 和 RIPng200。Switch B 通过 RIPng100 和 Switch A 交换路由信息，通过 RIPng200 和 Switch C 交换路由信息。

要求在 Switch B 上配置路由引入，将两个不同进程的 RIPng 路由相互引入到对方的 RIPng 进程中。

图 3-50 RIPng 引入外部路由组网图



配置步骤

步骤 1 配置各接口的 IPv6 地址。

步骤 2 配置 RIPng 基本功能。

配置 Switch A。

```
SwitchA#configure
SwitchA(config)#ripng 100
SwitchA(config-ripng-100)#quit
SwitchA(config)#interface vlan 100
SIM-MPU(config-vlanif-100)#ripng 100 enable
SwitchA(config-vlanif-100)#quit
SwitchA(config)#interface vlan 200
SIM-MPU(config-vlanif-200)#ripng 100 enable
SwitchA(config-vlanif-200)#quit
```

配置 Switch B。

```
SwitchA#configure
SwitchA(config)#ripng 100
SwitchA(config-ripng-100)#quit
SwitchA(config)#ripng 200
SwitchA(config-ripng-200)#quit
SwitchA(config)#interface vlan 100
SIM-MPU(config-vlanif-100)#ripng 100 enable
SwitchA(config-vlanif-100)#quit
SwitchA(config)#interface vlan 300
SIM-MPU(config-vlanif-300)#ripng 200 enable
SwitchA(config-vlanif-300)#quit
```

配置 Switch C。

```
SwitchA#configure
SwitchA(config)#ripng 200
SwitchA(config-ripng-200)#quit
SwitchA(config)#interface vlan 300
SIM-MPU(config-vlanif-300)#ripng 200 enable
SwitchA(config-vlanif-300)#quit
```

```
SwitchA(config)#interface vlan 400
SIM-MPU(config-vlanif-400)#ripng 200 enable
SwitchA(config-vlanif-400)#quit
```

步骤 3 验证配置结果。

查看 Switch C RIPng 数据库。

```
SwitchC#show ripng database
Process   Prefix/Prefixlen      Nexthop                  Metric  Age
State   Protocol   Tag
-------------------------------------------------------------------------
-----------------------------------------
200      3::/64              fe80::f2f1:f2ff:fef3:301     0      0
Active  ripng      0
200      4::/64              fe80::f2f1:f2ff:fef3:301     0      0
Active  ripng      0
-------------------------------------------------------------------------
-----------------------------------------
Total : 3
```

步骤 4 配置引入路由。

Switch B 进程 200 引入进程 100 路由。

```
SwitchB#configure
SwitchB(config)#ripng 200
SwitchB(config-ripng-200)#import-route ripng 100
SwitchB(config-ripng-200)#quit
```

步骤 5 验证配置结果。

查看 Switch C RIPng 数据库。

```
SwitchC#show ripng database
Process   Prefix/Prefixlen      Nexthop                  Metric  Age
State   Protocol   Tag
-------------------------------------------------------------------------
-----------------------------------------
200      1::/64              fe80::f2f1:f2ff:fef3:201     2      22
Active  ripng      0
200      2::/64              fe80::f2f1:f2ff:fef3:201     2      22
Active  ripng      0
200      3::/64              fe80::f2f1:f2ff:fef3:301     0      0
Active  ripng      0
200      4::/64              fe80::f2f1:f2ff:fef3:301     0      0
Active  ripng      0
-------------------------------------------------------------------------
-----------------------------------------
 Total : 4
```

3.14 路由策略

3.14.1 简介

功能

在今天的高性能网络中，网络的组织者需要一个领域，可以根据他们自己定义好的路由策略实现对报文的转发和选路。在路由策略的基础上，管理员可以分配网络流量通过指定的路径。

路由策略主要实现了路由过滤和路由属性设置等功能，它通过改变路由属性（包括可达性）来改变网络流量所经过的路径。路由协议在发布、接收和引入路由信息时，根据实际组网需求实施一些策略，以便对路由信息进行过滤和改变路由信息的属性，如：

控制路由的接收和发布，只发布和接收必要、合法的路由信息，以控制路由表的容量，提高网络的安全性。

控制路由的引入，一种路由协议在引入其它路由协议发现的路由信息丰富自己的路由信息时，只引入一部分满足条件的路由信息。

设置特定路由的属性，修改通过路由策略过滤的路由的属性，满足自身需要。

路由策略的实现

路由策略的实现主要包括两个步骤：

定义规则：定义一组匹配规则，可以用路由信息中的不同属性作为匹配规则进行设定。

应用规则：将匹配规则应用于路由的发布、接收和引入等过程的路由策略中。

路由策略的常见方式

访问控制列表，访问控制列表是针对 IP 报文的 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者下一跳地址。

地址前缀列表，地址前缀列表的作用类似于 ACL，但比它更为灵活，且更易于用户理解。使用地址前缀列表过滤路由信息时，其匹配对象为路由信息的目的地址信息域；另外，用户可以指定路由器选项，指明只接收某些路由器发布的路由信息。

AS 路径访问列表（as-path），as-path 仅用于 BGP。BGP 的路由信息中，包含有自治系统路径域。as-path 就是针对自治系统路径域指定匹配条件。

团体属性列表（community-list），community-list 仅用于 BGP。BGP 的路由信息包中，包含一个 community 属性域，用来标识一个团体。community-list 就是针对团体属性域指定匹配条件。

访问控制列表

访问控制列表 ACL 是一系列过滤规则的集合。在每个集合中，所有过滤规则是具有前后顺序的。在定义过滤规则时，根据报文的入接口、源或目的地址、协议类型、源或

目的端口号等属性描述过滤规则，同时指定了拒绝或接收报文。之后，系统根据过滤规则对到达路由器的报文进行分类，并判断报文被拒绝或者接收。

ACL 本身只是一组规则的集合，它只是通过过滤规则对报文进行了分类，因此 ACL 需要与路由策略配合使用，才能实现过滤报文的功能。

在网络设备中（可以是接入设备、核心设备等）部署 ACL 特性，可以保障网络的安全性与稳定性。例如：

防止对网络的攻击，例如防止针对 IP 报文、TCP 报文、ICMP（Internet Control Message Protocol）报文的攻击。

对网络访问行为进行控制，例如控制企业网中内网和外网的通信，用户访问特定网络资源，特定时间段内允许对网络的访问。

限制网络流量和提高网络性能，例如限定网络上行、下行流量的带宽，对用户申请的带宽进行收费，保证高带宽网络资源的充分利用。

地址前缀列表

地址前缀列表是一种包含一组路由信息过滤规则的过滤器，基于路由地址域（ip 地址、地址前缀长度范围、应用规则）提供有序的过滤规则集。地址前缀列表可以应用在各种动态路由协议中，对路由协议发布和接收的路由信息进行过滤。

地址前缀列表和 ACL 相比，配置简单，应用灵活。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。

地址前缀列表包括针对 IPv4 路由的 IPv4 地址前缀列表和针对 IPv6 路由的 IPv6 地址前缀列表，IPv6 地址前缀列表与 IPv4 地址前缀列表的实现相同。地址前缀列表进行匹配的依据有两个：掩码长度和掩码范围。

掩码长度：地址前缀列表匹配的对象是 IP 地址前缀，前缀由 IP 地址和掩码长度共同定义。例如，10.1.1.1/16 这条路由，掩码长度是 16，这个地址的有效前缀为 16 位，即 10.1.0.0。

掩码范围：对于前缀相同，掩码不同的路由，可以指定待匹配的前缀掩码长度范围来实现精确匹配或者在一定掩码长度范围内匹配。

在匹配的过程中，按升序依次检查由 index 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。因此，配置上的先后逻辑顺序错误会导致匹配异常，需要操作人员配置时保证正确性。

AS 路径访问列表

AS 路径访问列表是一组针对 BGP 路由的 AS_Path 属性进行过滤的规则。在 BGP 的路由信息中，包含有 AS_Path 属性，AS_Path 属性按矢量顺序记录了 BGP 路由从本地到目的地址所要经过的所有 AS 编号，因此基于 AS_Path 属性定义一些过滤规则，就可以实现对 BGP 路由信息的过滤。

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。

Route policy 可以由多个节点（node）构成，每个节点是匹配检查的一个单元，在匹配过程中，系统按节点序号升序依次检查各个节点。每个节点可以由一组 match 和 apply 子句组成。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 match 子句是“与”的关系，只有满足节点内所有 match 子句指定的匹配条件，才能通过该节点的匹配测试。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。一个 Route-policy 的不同节点间是“或”的关系，如果通过了其中一节点，就意味着通过该路由映射，不再对其他节点进行匹配测试。

3.14.2 配置准备

场景

配置好路由策略后，在各路由协议发布、接收或者引入路由时应用路由策略，实现路由过滤和属性设置功能。

前提

配置路由协议。

3.14.3 缺省配置

设备上 OSPF 的缺省配置如下。

功能	缺省值
路由策略特性	不使能

3.14.4 配置路由策略基本功能

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**route-policy** *NAME* {**permit** | **deny**} **node** *node*	创建 Route-Policy，并进入 Route-Policy 视图。
3	JX(config-route-policy-a-1)# **match** { **destination** | **next-hop** | **route-source** } **acl-ipv4** *acl-number*	配置 match 子句，ipv4 路由匹配 acl 信息（可选）。

步骤	配置	说明
4	JX(config-route-policy-a-1)# **match { destination | next-hop | route-source } acl-ipv6** *acl-number*	配置 match 子句，ipv6 路由匹配 acl 信息（可选）。
5	JX(config-route-policy-a-1)# **match { destination | next-hop | route-source } ip-prefix-list** *prefixname*	配置 match 子句，ipv4 路由匹配地址前缀列表（可选）。
6	JX(config-route-policy-a-1)# **match { destination | next-hop | route-source } ipv6-prefix-list** *prefixname*	配置 match 子句，ipv6 路由匹配地址前缀列表（可选）。
9	JX(config-route-policy-a-1)# **match as-path-filter** *NAME*	配置 match 子句，bgp 路由匹配 as-path 属性过滤器（可选）。
10	JX(config-route-policy-a-1)# **match community-filter** *filter-number*	配置 match 子句，bgp 路由匹配团体属性过滤器（可选）。
11	JX(config-route-policy-a-1)# **match extcommunity-filter** *filter-number*	配置 match 子句，bgp 路由匹配扩展团体属性过滤器（可选）。
12	JX(config-route-policy-a-1)# **match cost** *cost*	配置 match 子句，匹配路由开销值（可选）。
13	JX(config-route-policy-a-1)# **match process** *process-id*	配置 match 子句，匹配路由实例号（可选）。
14	JX(config-route-policy-a-1)# **match route-type { internal | external-type1 | external-type2 | external-type1or2 | nssa-external-type1 | nssa-external-type2 | nssa-external-type1or2 }**	配置 match 子句，匹配路由类型（可选）。
15	JX(config-route-policy-a-1)# **match tag** *tag*	配置 match 子句，匹配路由标记 TAG 字段（可选）。
16	JX(config-route-policy-a-1)# **apply as-path { additive | replace | delete } {x.y-format** *x.y-format*/**4bytes-format** *4bytes-format*}*	配置 apply 子句，设置 bgp 路由的 As-path 属性（可选）。
17	JX(config-route-policy-a-1)# **apply community { replace | additive } { aann-format** *aann-format* **| 4bytes-format** *4bytes-format* **| internet | no-advertise | no-export | no-export-subconfed }**	配置 apply 子句，设置 bgp 路由的团体属性（可选）。

步骤	配置	说明
18	JX(config-route-policy-a-1)# **apply extcommunity rt** { **replace** | **additive** } { **2bytes-aann-format** *2bytes-aann-format* | **4bytes-aann-format** *4bytes-aann-format* | **ip-format** *ip-format* }	配置 apply 子句，设置 bgp 路由的扩展团体属性（可选）。
19	JX(config-route-policy-a-1)# **apply cost** *cost*	配置 apply 子句，设置路由的开销值（可选）。
20	JX(config-route-policy-a-1)# **apply cost-type** { **type-1** | **type-2** }	配置 apply 子句，设置路由的开销值（可选）。
21	JX(config-route-policy-a-1)# **apply isis** { **level-1** | **level-2** | **level-1-2** }	配置 apply 子句，设置 isis 路由的路由级别（可选）。
22	JX(config-route-policy-a-1)# **apply tag** *tag-value*	配置 apply 子句，设置路由的路由标记 tag 字段（可选）。

3.14.5 配置过滤器

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**ip prefix-list** *list-name* { **deny** | **permit** } *A.B.C.D/M* {**greater-equal** *greater-equal* | **less-equal** *less-equal*}	创建 ipv4 的前缀列表表项，若没有配置 greater-equal 或 less-equal，则表示掩码精准匹配。
3	JX(config)#**ipv6 prefix-list** *list-name* { **deny** | **permit** } *X:X::X:X/M* {**greater-equal** *greater-equal* | **less-equal** *less-equal*}	创建 ipv6 的前缀列表表项，若没有配置 greater-equal 或 less-equal，则表示掩码精准匹配。
4	JX(config)# **as-path-filter** *list-name* { **deny** | **permit** } *EXPRESSION*	创建 AS 路径过滤列表。
5	JX(config)# **ip community-filter** *filter-number* { **permit** | **deny** } {**aann-format** *aann-format*|**4bytes-format** *4bytes-format*|**internet**|**no-advertise**|**no-export**|**no-export-subconfed**}	创建团体属性过滤器。

步骤	配置	说明
6	JX(config)# **ip extcommunity-filter** *filter-number* { **permit** | **deny** } **rt** {**2bytes-aann-format** *2bytes-aann-format* |**4bytes-aann-format** *4bytes-aann-format* |**ip-format** *ip-format*}	创建扩展团体属性过滤器。

3.14.6 检查配置

配置完成后，请在设备上执行以下命令检查配置结果。

步骤	配置	说明
1	JX# **show route-policy** [*list-name*]	查看路由策略的规则信息。
2	JX# **show route-policy information**	查看路由策略的资源信息。
3	JX# **show route-policy config**	查看路由策略的配置信息。
4	JX# **show prefix-list**	查看前缀列表的规则信息。
5	JX# **show prefix-list config**	查看前缀列表的配置信息。
6	JX# **show prefix-list information**	查看前缀列表的资源信息。
7	JX# **show as-path-filter**	查看 AS 路径过滤器的规则信息。
8	JX# **show as-path-filter config**	查看 AS 路径过滤器的配置信息。
9	JX#**show as-path-filter information**	查看 AS 路径过滤器的资源信息。

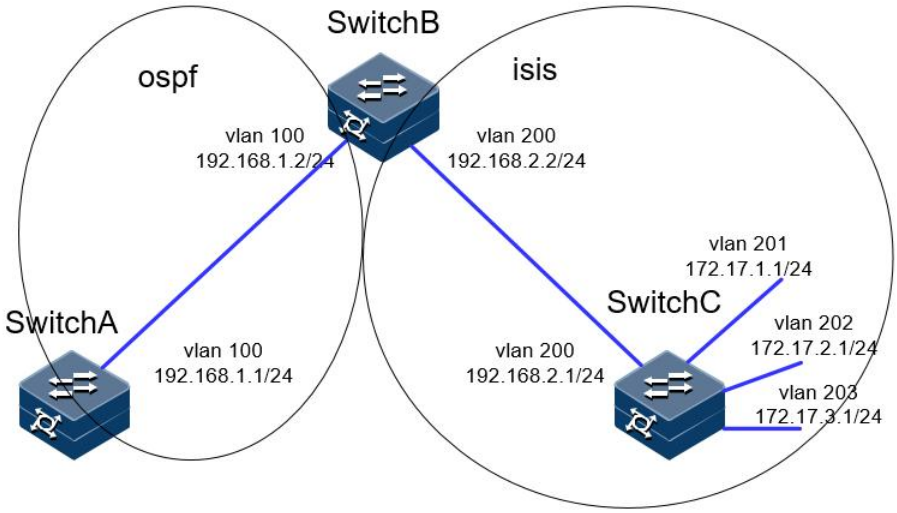
3.14.7 在 IPV4 路由中应用路由策略示例

组网需求

如下图所示，Switch B 与 Switch A 之间通过 OSPF 协议交换路由信息，与 Switch C 之间通过 IS-IS 协议交换路由信息。

要求在 Switch B 上配置路由引入，将 IS-IS 路由引入到 OSPF 中去，并同时使用路由策略设置路由的属性。其中，设置 172.17.1.0/24 的路由的开销为 100，设置 172.17.2.0/24 的路由的 Tag 属性为 20。

图 3-51 ipv4 路由引入中应用路由策略组网图



配置步骤

配置各接口的 IPV6 地址。

配置 ISIS 路由协议。

Switch C：

```
JX-C#configure
JX-C(config)#isis 1
JX-C(config-isis-1)#network-entity 10.0000.0000.0001.00
JX-C(config-isis-1)#exit
JX-C(config)#interface vlan 200
JX-C(config-vlanif-200)#ip address 192.168.2.1/24
JX-C(config-vlanif-200)#isis enable 1
JX-C(config-vlanif-200)#exit
JX-C(config)#interface vlan 201
JX-C(config-vlanif-201)#ip address 172.1.1.1/24
JX-C(config-vlanif-201)#isis enable 1
JX-C(config-vlanif-201)#exit
JX-C(config)#interface vlan 202
JX-C(config-vlanif-202)#ip address 172.1.2.1/24
JX-C(config-vlanif-202)#isis enable 1
JX-C(config-vlanif-202)#exit
JX-C(config)#interface vlan 203
JX-C(config-vlanif-203)#ip address 172.1.3.1/24
JX-C(config-vlanif-203)#isis enable 1
JX-C(config-vlanif-203)#exit
```

Switch B：

```
JX-B#configure
JX-B(config)#isis 1
JX-B(config-isis-1)#network-entity 10.0000.0000.0002.00
```

```
JX-B(config-isis-1)#exit
JX-B(config)#interface vlan 200
JX-C(config-vlanif-200)#ip address 192.168.2.2/24
JX-B(config-vlanif-200)#isis enable 1
JX-B(config-vlanif-200)#exit
```

配置 OSPF 协议以及路由重分配。

Switch A：

```
JX-A#configure
JX-A(config)#ospf 1
JX-A(config-ospf-1)#router-id 1.1.1.1
JX-A(config-ospf-1)#network 192.168.1.0 255.255.255.0 area 0
JX-A(config-ospf-1)#exit
JX-A(config)#
```

Switch B：

```
JX-B#configure
JX-B(config)#ospf 1
JX-B(config-ospf-1)#router-id 2.2.2.2
JX-B(config-ospf-1)#network 192.168.1.0 255.255.255.0 area 0
JX-B(config-ospf-1)#import-route isis 1
JX-B(config-ospf-1)#exit
JX-B(config)#
```

查看配置结果。

使用 show ospf route 查看引入的路由：

```
JX-A(config)#show ospf route
 OSPF Process 1
 RoutType  Prefix           PathType  Cost   Cost2      NextHop
BackupNextHop    AreaId  Time
 ASBR     2.2.2.2/32        INTRA    1      0         192.168.1.2
0.0.0.0         0      0:00:05
 Network   172.17.1.0/24      ASE2     1      1         192.168.1.2
0.0.0.0         N/A    0:00:05
 Network   172.17.2.0/24      ASE2     1      1         192.168.1.2
0.0.0.0         N/A    0:00:18
 Network   172.17.3.0/24      ASE2     1      1         192.168.1.2
0.0.0.0         N/A    0:02:22
 Network   192.168.1.0/24     INTRA    1      0         192.168.1.1
0.0.0.0         N/A    0:24:26
 Network   192.168.2.0/24     ASE2     1      1         192.168.1.2
0.0.0.0         N/A    0:02:22
 Route :
 ABR      ASBR     Network   Intra    Inter    External
 0        1        5         1        0        4

 Path :
 ABR      ASBR     Network   Intra    Inter    External
 0        1        5         1        0        4
```

配置过滤列表。

配置编号为 1002 的 acl，允许 172.17.2.0/24 的路由通过

```
JX-B(config)#acl-ipv4 1002
JX-B(configure-acl-ipv4-1002)#rule 1 ip src-ip 172.17.2.0/24 dst-ip any
JX-B(configure-acl-ipv4-1002)#rule 1 action permit
JX-B(configure-acl-ipv4-1002)#exit
```

配置名为 prefix 的地址前缀列表，允许 172.17.1.0/24 的路由通过

```
JX-B(config)#ip prefix-list prefix permit 172.17.1.0/24
```

配置路由策略。

```
JX-B(config)#route-policy isis2ospf permit node 1
JX-B(config-route-policy-isis2ospf-1)#match destination ip-prefix-list
prefix
JX-B(config-route-policy-isis2ospf-1)#apply cost 100
JX-B(config-route-policy-isis2ospf-1)#exit
JX-B(config)#route-policy isis2ospf permit node 2
JX-B(config-route-policy-isis2ospf-2)#match destination acl-ipv4 1002
JX-B(config-route-policy-isis2ospf-2)#apply tag 20
JX-B(config-route-policy-isis2ospf-2)#exit
JX-B(config)#route-policy isis2ospf permit node 3
JX-B(config-route-policy-isis2ospf-3)#exit
JX-B(config)#
```

在 OSPF 引入路由时应用路由策略。

```
JX-B(config)#ospf 1
JX-B(config-ospf-1)#import-route isis 1 route-policy isis2ospf
```

查看配置结果。

172.17.1.0/24 的 cost 为 200，172.17.2.0/24 的 tag 值为 20（0x14）

```
JX-A (config)#show ospf route
 OSPF Process 1
 RoutType  Prefix            PathType  Cost    Cost2      NextHop
BackupNextHop    AreaId  Time
 ASBR     2.2.2.2/32         INTRA    1       0         192.168.1.2
0.0.0.0          0      0:00:26
 Network   172.17.1.0/24      ASE2     1       100        192.168.1.2
0.0.0.0          N/A    0:02:52
 Network   172.17.2.0/24      ASE2     1       1          192.168.1.2
0.0.0.0          N/A    0:01:56
 Network   172.17.3.0/24      ASE2     1       1          192.168.1.2
0.0.0.0          N/A    0:00:26
 Network   192.168.1.0/24     INTRA     1       0          192.168.1.1
0.0.0.0          N/A    0:39:56
 Network   192.168.2.0/24     ASE2     1       1          192.168.1.2
0.0.0.0          N/A    0:00:26
 Route :
 ABR      ASBR     Network   Intra    Inter    External
 0        1        5         1        0        4
```

```
 Path :
 ABR      ASBR     Network   Intra    Inter    External
 0        1        5         1        0        4

JX-A(config)#show ospf database ls-type as-external-lsa link-id
172.17.2.0 adv-router 2.2.2.2
Database of OSPF Process 1
 Detailed LSA Information

  Buffer: 00 b7 02 05 ac 11 02 00 02 02 02 02 80 00 00 01
  Buffer: 72 83 00 24 ff ff ff 00 80 00 00 01 00 00 00 00
  Buffer: 00 00 00 14
   Lsa Type:(5)AS-External Lsa
   Lsa Age :183
   Options  :(0b 00(DC)(EA)(N/P)(MC)(E)0)0x2
   Link State Id :172.17.2.0(Destination network)
   Advertising Router :2.2.2.2
   Sequence Number :0x80000001
   Checksum:0x7283
   Length :36
   Network Mask:255.255.255.0
   E-bit:Set
   Metric:1
   Forwarding Address:0.0.0.0
   External Route Tag:0x14
```

3.14.8 在 IPV6 路由引入中应用路由策略

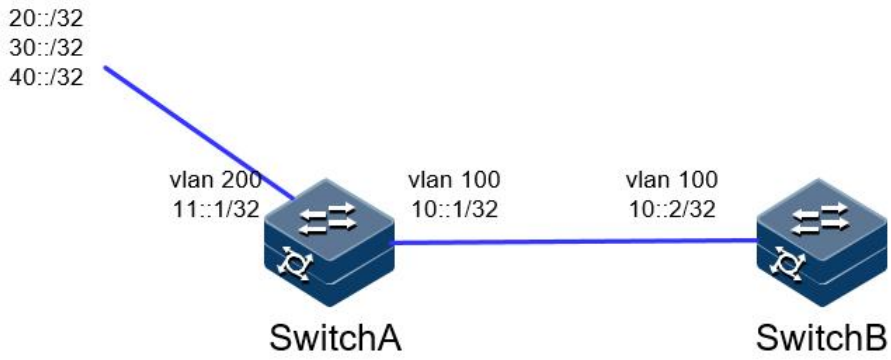
组网需求

在 Switch A 和 Switch B 上使能 RIPng。

在 Switch A 上配置三条静态路由，并设置在引入静态路由时应用路由策略，使三条静态路由部分引入、部分被屏蔽掉——20::/32 和 40::/32 网段的路由是可见的，30::/32 网段的路由则被屏蔽。

通过在 Switch B 上查看 RIPng 路由表，验证路由策略是否生效。

图 3-52 ipv6 路由引入中应用路由策略组网图



配置步骤

配置各接口的 IPV6 地址。

配置 RIPng。

Switch A:

```
JX-A#configure
JX-A(config)#ripng 1
JX-A(config-ripng-1)#exit
JX-A(config)#interface vlan 100
JX-A(config-vlanif-100)#ripng 1 enable
JX-A(config-vlanif-100)#exit
```

Switch B:

```
JX-B#configure
JX-B(config)#ripng 1
JX-B(config-ripng-1)#exit
JX-B(config)#interface vlan 100
JX-B(config-vlanif-100)#ripng 1 enable
JX-B(config-vlanif-100)#exit
JX-B(config)#
```

配置静态路由

```
JX-A(config)#ipv6 route-static 20:: 32 11::2
JX-A(config)#ipv6 route-static 30:: 32 11::2
JX-A(config)#ipv6 route-static 40:: 32 11::2
```

验证路由策略和应用

```
JX-A(config)#ipv6 prefix-list a permit 30::/32
JX-A(config)#route-policy sta2ripng deny node 0
JX-A(config-route-policy-sta2ripng-0)#match destination ipv6-prefix-list
a
JX-A(config-route-policy-sta2ripng-0)#exit
JX-A(config)#route-policy sta2ripng permit node 10
```

```
JX-A(config-route-policy-sta2ripng-10)#exit
JX-A(config)#ripng 1
JX-A(config-ripng-1)#import-route static route-policy sta2ripng
```

查看配置结果

查看 switch B 的路由表。

```
JX-B(config)#show ripng route
Process   Prefix/Prefixlen      Nexthop                  Metric  Age
State   Protocol   Tag
------------------------------------------------------------------------------
-----------------------------------------------
1       10::/32             fe80::f2f1:f2ff:fef3:101     0      0
Active  ripng      0
1       20::/32             fe80::f2f1:f2ff:fef3:201     2      25
Active  ripng      0
1       40::/32             fe80::f2f1:f2ff:fef3:201     2      25
Active  ripng      0
------------------------------------------------------------------------------
-----------------------------------------------
 Total : 3
```

3.15 ECMP

3.15.1 简介

ECMP 应用于多条不同链路到达同一目的地址的网络环境中。如果使用传统的路由技术，发往该目的地址的数据包只能利用其中的一条链路，其它链路处于备份状态或无效状态，并且在动态路由环境下相互的切换需要一定时间。而 ECMP 可以在该网络环境下同时使用多条链路，不仅增加了传输带宽，并且可以无时延无丢包地备份失效链路的数据传输。

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，那么就满足负载分担的条件。ECMP 提供普通的算法配置方式和增强的模版配置方式，普通型 ECMP 负载分担方式提供了基于源 IP 地址、目的 IP 地址、传输层源端口号和目的端口号进行负载分担的方式，而增强型 ECMP 负载分担方式则提供了更多负载分担方式的模板。

3.15.2 配置准备

场景

无

前提

无

3.15.3 缺省配置

功能	缺省值
ECMP 算法	不使能
ECMP 模版	使能默认模版

3.15.4 配置 ECMP 分流算法

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)# **ecmp load-balance { src-ip | default }**	配置 ECMP 分流算法： src-ip：基于源 IP 分担 default：默认基于源 IP 和目的 IP 分担

3.15.5 配置 ECMP 模版

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**ecmp-profile default**	配置 ECMP 分流模版

步骤	配置	说明
3	JX(config-ecmp-profile-defaut)# **l3 field { src-ip | dst-ip | src-mac | dst-mac | src-port | vlan | l4-srcport | l4-dstport | protocol | all | default }**	配置 ECMP 分担模板中 L3 报文的负载分担方式。 src-ip：指定根据源 IP 地址进行负载分担 dst-ip：指定根据目的 IP 地址进行负载分担 src-mac：指定根据源 MAC 地址进行负载分担 dst-mac：指定根据目的 MAC 地址进行负载分担 src-port：指定根据源端口进行负载分担 vlan：指定根据 VLAN 进行负载分担 l4-srcport：指定根据传输层源端口进行负载分担 l4-dstport：指定根据传输层目的端口进行负载分担 protocol：指定根据协议进行负载分担 all：指定根据以上所有项目进行负载分担 default：默认根据源 IP 地址和目的 IP 地址进行负载分担 缺省情况下，IPv4 报文负载分担方式为 src-ip、dst-ip、l4-srcport、l4-dstport 和 protocol
4	JX(config-route-policy-a-1)# **ecmp-profile default**	应用 ecmp 分流模版。

3.15.6 检查配置

步骤	配置	说明
1	JX#**show ecmp-profile**	显示 ECMP 模版配置信息
2	JX# **show ecmp load-balance**	显示 ECMP 模版绑定信息。

4 DHCP

本章介绍 DHCP 的基本原理和配置过程，并提供相关的配置案例。

- DHCP Client
- DHCP Server
- DHCP Relay

4.1 DHCP Client

4.1.1 简介

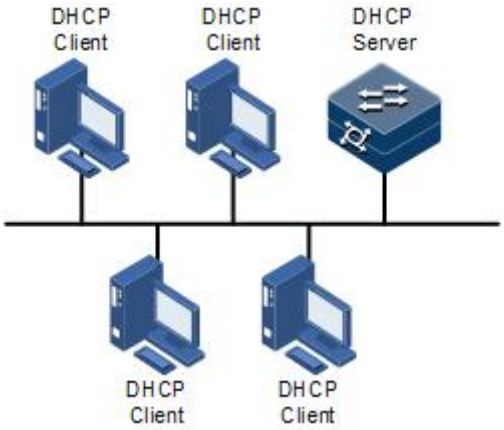
DHCP（Dynamic Host Configuration Protocol，动态主机配置协议）在 TCP/IP 网络上给用户动态分配 IP 地址等配置信息的协议。它是基于 BOOTP（Bootstrap Protocol）协议，并在 BOOTP 协议的基础上添加了自动分配可用网络地址、网络地址重复使用以及其他扩展配置选项等功能。

随着网络规模的不断扩大和网络复杂度的提高，计算机的数量经常超过可供分配的 IP 地址数量。同时随着便携机及无线网络的广泛使用，计算机的位置也经常变化，相应的 IP 地址也必须经常更新，从而导致网络配置越来越复杂。DHCP 就是为解决这些问题而发展起来的。

DHCP 采用客户端/服务器通信模式，由客户端向服务器提出配置申请（包括 IP 地址、子网掩码、缺省网关等参数），服务器返回为客户端分配的 IP 地址等相应的配置信息，以实现 IP 地址等信息的动态配置。

在 DHCP 的典型应用中，一般包含一台 DHCP 服务器和多台客户端（如 PC 和便携机），如图 4-1 所示。

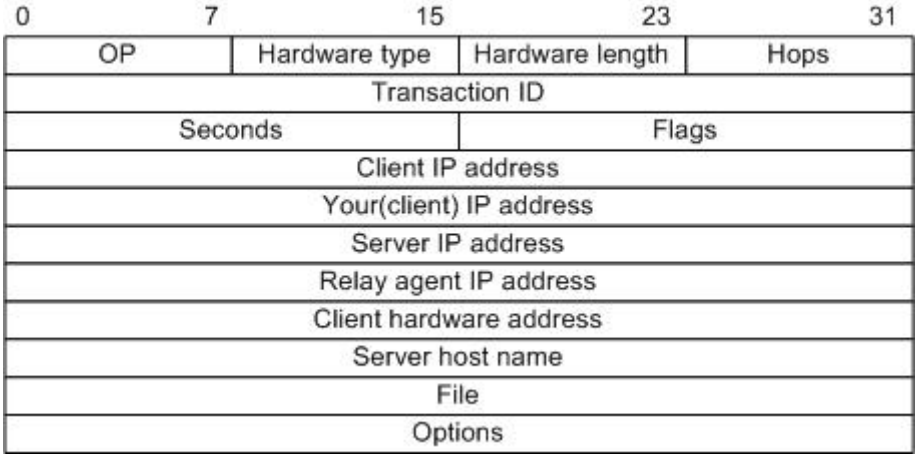
图 4-1 DHCP 典型应用组网示意图



DHCP 技术保证了 IP 地址的合理分配问题，从而避免了 IP 地址的浪费，提高了整网的 IP 地址使用率。

DHCP 报文格式如图 4-2 所示。DHCP 报文被封装在 UDP 数据报中。

图 4-2 DHCP 报文结构示意图



DHCP 报文中的各个字段含义如表 4-1 所示。

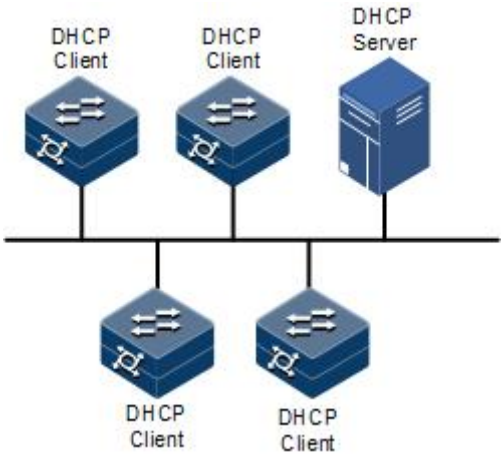
表 4-1 DHCP 报文字段含义列表

字段名	字段长度（Byte）	描述
OP	1	报文类型。 • 取值为 1 时，表示该报文为客户端请求报文； • 取值为 2 时，表示该报文为服务器端回应报文。

字段名	字段长度（Byte）	描述
Hardware type	1	DHCP 客户端的硬件地址类型。
Hardware length	1	DHCP 客户端的硬件地址长度。
Hops	1	DHCP 报文经过的 DHCP 中继的数目。 DHCP 请求报文每经过一个 DHCP 中继，该字段就会加 1。
Transaction ID	4	客户端发起一次请求时选择的随机数，用来标识一次地址请求过程。
Seconds	2	DHCP 客户端开始 DHCP 请求后所经过的时间。目前没有使用，固定为 0。
Flags	2	第 1 个比特为广播回应标识位，用来标识 DHCP 服务器回应报文是采用单播还是广播方式发送。 • 0 表示采用单播方式； • 1 表示采用广播方式。 其它比特保留不用。
Client IP address	4	DHCP 客户端的 IP 地址，只有当客户端在绑定，更新或重新绑定状态时进行填充，且可以用于回应 ARP 请求。
Your(client) IP address	4	DHCP 服务器分配给客户端的 IP 地址。
Server IP address	4	DHCP 服务器的 IP 地址。
Relay agent IP address	4	DHCP 客户端发出请求报文后经过的第一个 DHCP 中继的 IP 地址。
Client hardware address	16	DHCP 客户端的硬件地址。
Server host name	64	DHCP 服务器的名称。
File	128	DHCP 服务器为 DHCP 客户端指定的启动配置文件名称及路径信息。
Options	可变	可选变长选项字段，包含报文的类型、有效租期、DNS（Domain Name System，域名系统）服务器的 IP 地址、WINS（Windows Internet Name Server ，Windows 网际命名服务）服务器的 IP 地址等配置信息。

设备支持作为 DHCP 客户端，从 DHCP 服务器获取 IP 地址，以便后续对该设备进行管理，如图 4-3 所示。

图 4-3 DHCP 客户端组网示意图



4.1.2 配置准备

场景

设备作为 DHCP 客户端时，从指定的 DHCP 服务器获取 IP 地址，可以用于后续对该设备的管理。

当采用动态地址分配方式时，DHCP 客户端所分配到的 IP 地址有一定的租借期限。租借期满后，DHCP 服务器会收回该 IP 地址。如果 DHCP 客户端希望继续使用该地址，需要续租 IP 地址。如果租借期未满，DHCP 客户端不需要继续使用该地址，可以释放 IP 地址。

若 DHCP 客户端需要通过多台 DHCP 中继向 DHCP 服务器获取 IP 地址，建议 DHCP 中继的台数不超过 4 台。

前提

在配置 DHCP 客户端之前，需完成以下任务：

- 创建 VLAN 并将接口加入 VLAN；
- 设备未启动 DHCP Snooping 功能。DHCP 客户端的缺省配置

设备上 DHCP 客户端的缺省配置如下。

功能	缺省值
hostname	设备主机名
class-id	缺省值可由启动参数指定

功能	缺省值
client-id	01 MAC

4.1.3 配置 DHCP 客户端

DHCP 客户端申请 IP 地址，需先创建 VLAN，同时配置好 DHCP 服务器，否则接口通过 DHCP 获取 IP 地址会失败。

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**configure**	进入全局配置模式。
2	JX (config)#**interface vlan** *vlan-id*	进入三层物理接口配置模式或者 VLAN 接口配置模式，以下以 VLAN 接口配置模式为例进行说明。
4	JX (config-vlanif-*)#**dhcp client** { **class-id ascii** *class-id* | **client-id ascii** *client-id* | **hostname ascii** *host-name* }	配置 DHCP 客户端信息，可以配置的信息有类型标识符、客户端标识符和主机名称。 **注意** DHCP 客户端获取到 IP 地址后，不允许修改客户端信息。
4	JX (config-vlanif-*)#**ip address dhcp enable**	配置通过 DHCP 方式申请 IP 地址。
6	JX(config-vlan*)#**ip address dhcp renew**	续租 IP 地址。 若此前设备的 VLAN 接口通过 DHCP 方式获取了 IP 地址，在 IP 地址租约到期时将会自动续租。
7	JX(config-vlan*)#**ip address dhcp release**	释放 IP 地址。

4.1.4 检查配置

配置完成后，请在设备上执行以下命令检查配置结果。

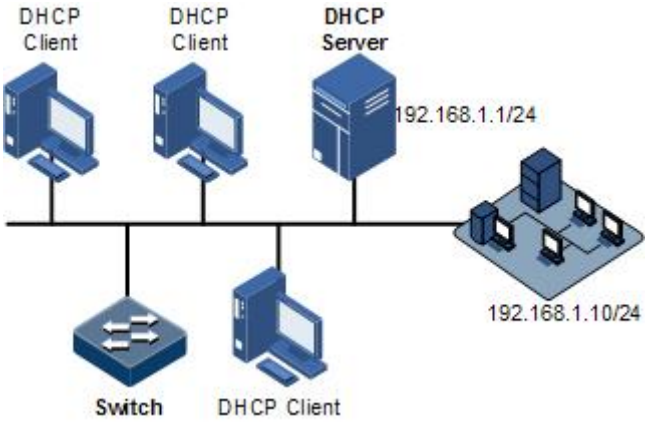
序号	检查项	说明
1	JX#**show dhcp client** [*interface-type interface-number* | **vlan** *vlan-id*]	查看 DHCP 客户端配置信息。

4.1.5 配置 DHCP 客户端示例

组网需求

如图 4-4 所示，Switch 作为 DHCP 客户端，主机名为 JX，接入 DHCP 服务器和网管平台。需要由 DHCP 服务器分配 IP 地址给 Switch，从而使网管平台能够管理 Switch。

图 4-4 配置 DHCP 客户端组网示意图



配置步骤

步骤 1 配置 DHCP 客户端信息。

```
JX#configure
JX(config)#interface vlan 1
JX(config-vlanif-1)#dhcp client hostname ascii JX
```

步骤 2 配置通过 DHCP 方式申请 IP 地址。

```
JX(config-vlanif-1)#ip address dhcp enable
```

检查结果

通过 **show dhcp client** 命令查看 DHCP 客户端配置是否正确。

```
JX#show dhcp client
Interface vlan-1 dhcp client information :
-------------------------------------------------------------
  Current state : Bound
  Allocated IP : 192.168.1.2
  Subnet Mask : 255.255.255.0
  Server IP : 192.168.1.1
  Allocated lease : 86400 seconds
  Lease T1 time : 43200 seconds
```

```
  Lease T2 time : 75600 seconds
  Lease Obtained : 2023/07/18  Tue  18:49:37
  Lease  timeout : 2023/07/19  Wed  18:49:37
  Transaction ID : 0x5558ec
  Client ID :
  Class ID :
  Hostname : JX
  DNS :
  Getway :
  Domain :
  Lease : 0 days 23 hours 59 minutes 49 seconds.
------------------------------------------------------------
```

4.2 DHCP Server

4.2.1 简介

DHCP（Dynamic Host Configuration Protocol，动态主机配置协议）是在 TCP/IP 网络上给用户动态分配 IP 地址等配置信息的协议。它是基于 BOOTP（Bootstrap Protocol）协议，并在 BOOTP 协议的基础上添加了自动分配可用网络地址、网络地址重复使用以及其他扩展配置选项等功能。

随着网络规模的不断扩大和网络复杂度的提高，计算机的数量经常超过可供分配的 IP 地址数量。同时随着便携机及无线网络的广泛使用，计算机的位置也经常变化，相应的 IP 地址也必须经常更新，从而导致网络配置越来越复杂。DHCP 就是为解决这些问题而发展起来的。

DHCP 采用客户端/服务器通信模式，由客户端向服务器提出配置申请（包括 IP 地址、子网掩码、缺省网关等参数）服务器端返回为客户端分配的 IP 地址等相应的配置信息，以实现 IP 地址等信息的动态配置。

在 DHCP 的客户端/服务器通信模式中指定专门的主机分配网络地址，传送网络配置参数给需要的网络主机，被指定的主机称为 DHCP Server。

DHCP 应用

一般情况下，在以下场合会利用 DHCP Server 来完成 IP 地址分配：

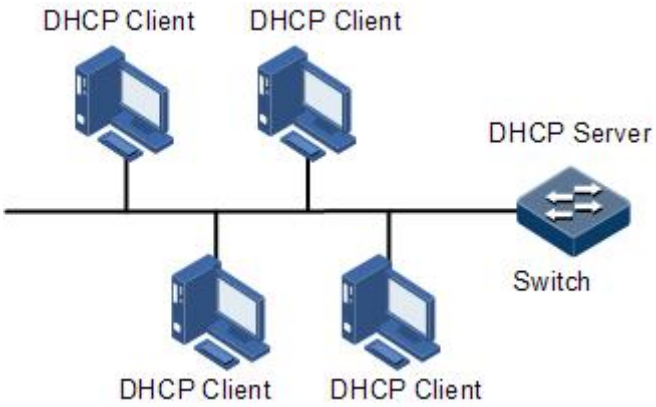
- 网络规模较大，手工配置需要很大的工作量，并难以对整个网络进行集中管理。
- 网络中主机数目大于该网络支持的 IP 地址数量，无法给每个主机分配一个固定的 IP 地址，且对同时接入网络的用户数目也有限制。
- 网络中只有少数主机需要固定的 IP 地址，大多数主机没有固定 IP 地址需求。

DHCP Client 从 DHCP Server 获得 IP 地址后，并不能永久使用其获得的 IP 地址，而是有一个固定的使用期限，称为租约时间。

DHCP 技术解决了 IP 地址的合理分配问题，从而避免了 IP 地址的浪费，提高了整网的 IP 地址使用率。

设备支持作为 DHCP Server，向客户端提供动态 IP 地址，如图 4-5 所示。

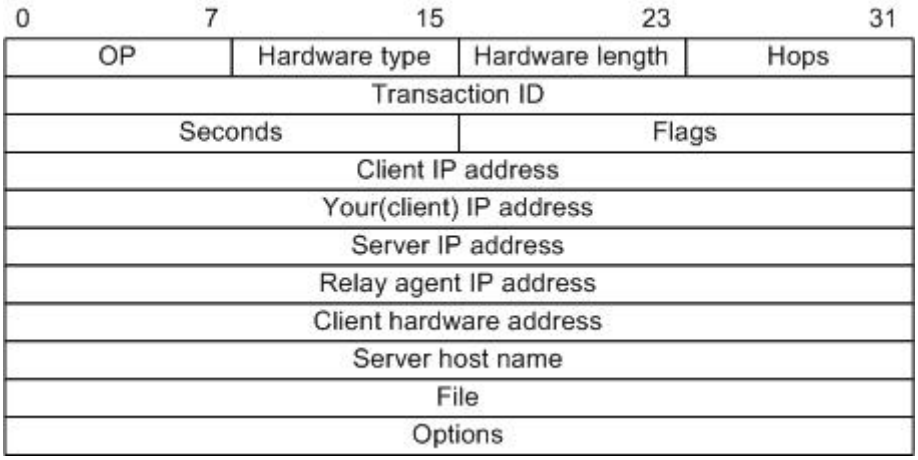
图 4-5 DHCP Server 和 DHCP Client 应用组网示意图



DHCP 报文

DHCP 报文格式如图 4-6 所示。DHCP 报文被封装在 UDP 数据报中。

图 4-6 DHCP 报文结构示意图



DHCP 报文中的各个字段含义如表 4-2 所示。

表 4-2 DHCP 报文字段含义列表

字段名	字段长度（Byte）	描述
OP	1	报文类型。 • 取值为 1 时，表示该报文为客户端请求报文； • 取值为 2 时，表示该报文为服务器端回应报文。

字段名	字段长度（Byte）	描述
Hardware type	1	DHCP 客户端的硬件地址类型。
Hardware length	1	DHCP 客户端的硬件地址长度。
Hops	1	DHCP 报文经过的 DHCP 中继的数目。 DHCP 请求报文每经过一个 DHCP 中继，该字段就会加 1。
Transaction ID	4	客户端发起一次请求时选择的随机数，用来标识一次地址请求过程。
Seconds	2	DHCP 客户端开始 DHCP 请求后所经过的时间。目前没有使用，固定为 0。
Flags	2	第 1 个比特为广播回应标识位，用来标识 DHCP 服务器回应报文是采用单播还是广播方式发送。 • 0 表示采用单播方式； • 1 表示采用广播方式。 其它比特保留不用。
Client IP address	4	DHCP 客户端的 IP 地址，只有当客户端在绑定，更新或重新绑定状态时进行填充，且可以用于回应 ARP 请求。
Your(client) IP address	4	DHCP 服务器分配给客户端的 IP 地址。
Server IP address	4	DHCP 服务器的 IP 地址。
Relay agent IP address	4	DHCP 客户端发出请求报文后经过的第一个 DHCP 中继的 IP 地址。
Client hardware address	16	DHCP 客户端的硬件地址。
Server host name	64	DHCP 服务器的名称。
File	128	DHCP 服务器为 DHCP 客户端指定的启动配置文件名称及路径信息。
Options	可变	可选变长选项字段，包含报文的类型、有效租期、DNS（Domain Name System，域名系统）服务器的 IP 地址、WINS（Windows Internet Name Server ，Windows 网际命名服务）服务器的 IP 地址等配置信息。

4.2.2 配置准备

场景

设备作为 DHCPv4 服务器时，DHCPv4 客户端可以向设备获取 IP 地址。

前提

接口上必须配置 IP 地址，且该接口未启动 DHCP 客户端和 DHCP Relay。

4.2.3 创建并配置 IPv4 地址池

请在需要创建并配置 IPv4 地址池的设备上进行以下配置。

步骤	配置	说明
1	JX#**configure**	进入全局配置模式。
2	JX(config)#**dhcp start**	使能全局 DHCP 功能。
3	JX(config)#**dhcp server pool** *pool-name*	创建 IPv4 地址池并进入地址池配置模式。
4	JX(config-dhcp-pool-*)#**ip range** *start-ip-address end-ip-address* **mask** *mask*	配置 IPv4 地址池的地址范围。
5	JX(config-dhcp-pool-*)#**dhcp server exclude-ip** *start-ip-address* [*end-ip-address*]	配置 IPv4 地址池排除的地址范围。
6	JX(config-dhcp-pool-*)#**lease-time** { *hour* | **day** *day* **hour** *hour* **minute** *minute* | **unlimited** | **default** }	配置 IPv4 地址池的地址租期。
7	JX(config-dhcp-pool-*)#**dns** *ip-address* [**backup**]	配置 IPv4 地址池的 DNS 服务器。
8	JX(config-dhcp-pool-*)#**gateway** *ip-address*	配置 IPv4 地址池的缺省网关。
9	JX(config-dhcp-pool-*)#**option** *option-code* [**sub-option** *option-code*] { **ascii** *ascii-string* | **hex** *hex-string* | **ip-address** ip-address }	配置 Option 携带信息。
10	JX(config)#**dhcp server static-bind** *ip-address mac-address*	配置 IPv4 地址池中的 IP 地址与用户 MAC 地址绑定。

4.2.4 配置接口的 DHCP Server 功能

请在需要配置接口的 DHCP 服务器功能的设备上进行以下配置。

步骤	配置	说明
1	JX#**configure**	进入全局配置模式。
2	JX(config)#**interface vlan** *vlan-id*	进入 VLAN 接口配置模式或者三层物理接口配置模式，以下以 VLAN 接口配置模式为例进行说明。
3	JX(config-vlanif-*)#**dhcp enable server**	使能接口的 DHCPv4 服务器功能。

4.2.5 回收 IP 地址或地址池

请在需要回收 IP 地址池的设备上进行以下配置。

步骤	配置	说明
1	JX#**configure**	进入全局配置模式。
2	JX(config)#**no dhcp server pool** *pool-name*	回收 IP 地址池。
3	JX(config)#**no dhcp server user-bind**	回收 DHCPv4 服务器的地址绑定表。
4	JX(config)#**no dhcp server dynamic-bind** *ip-address*	回收与用户 MAC 地址绑定的 IP 地址。

4.2.6 配置 DHCPv4 Server Ping 功能

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**configure**	进入全局配置模式。
2	JX(config)#**dhcp server address-check-time** { *period* | **default** }	配置 DHCP Server 分配地址前检查地址是否被使用所等待的超时时间。0 表示不检查地址是否被使用，默认 500ms

4.2.7 检查配置

配置完成后，请在设备上执行以下命令检查配置结果。

序号	检查项	说明
1	JX#**show dhcp server config**	查看 DHCP 服务器的配置信息。
2	JX#**show dhcp information**	查看 DHCP 服务器的详细信息。
3	JX#**show dhcp server user-bind**	查看已分配的 IP 地址及其客户端信息。

序号	检查项	说明
4	JX#show dhcp server statistics	查看 DHCP 服务器的报文统计信息。
5	JX#show dhcp server pool	查看 DHCP 服务器的地址池配置信息。

4.2.8 维护

用户可以通过以下命令，维护设备 DHCP Server 特性的运行情况和配置情况。

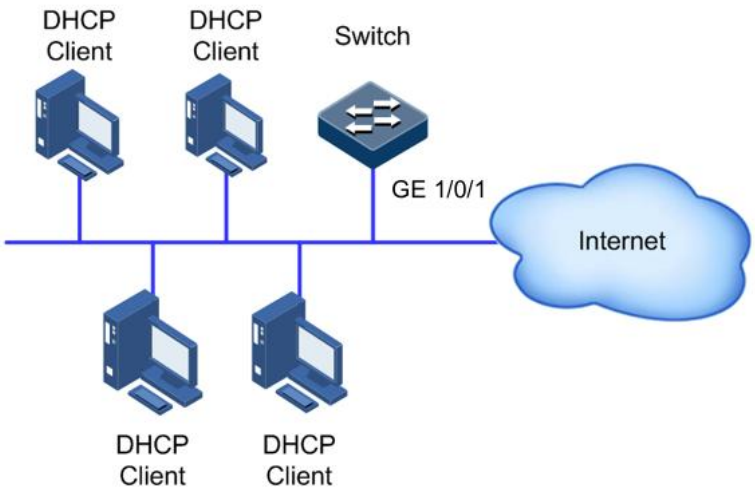
命令	描述
JX(config)# reset dhcp server statistics	清空 DHCP Server 统计信息。

4.2.9 配置 DHCPv4 服务器示例

组网需求

如图 4-7 所示，Switch 设备作为 DHCP 服务器分配 IP 地址给 DHCP 客户端，租期为 8 小时，IP 地址池名称为 pool，IP 地址范围是 172.31.1.2～172.31.1.100，DNS 服务器的 IP 地址为 172.31.100.1。

图 4-7 配置 DHCP 服务器组网示意图



配置步骤

步骤 1 创建并配置 IP 地址池。

```
JX#configure
```

```
JX(config)#dhcp start
JX(config-dhcp-pool-1)#dhcp server pool 1
JX(config-dhcp-pool-1)# ip range 172.31.1.2 172.31.1.100 mask
255.255.255.0
JX(config-dhcp-pool-1)#lease-time day 0 hour 8 minute 0
JX(config-dhcp-pool-1)#dns 172.31.100.1
JX(config-dhcp-pool-1)#exit
```

步骤 2 配置接口的 DHCP 服务器功能。

```
JX(config)#interface vlan 1
JX(config-vlanif-1)#ip address 172.31.1.1/24
JX(config-vlanif-1)#dhcp enable server
```

检查结果

通过 **show dhcp server config** 命令查看 DHCP 服务器配置是否正确。

```
JX# show dhcp server config
!
dhcp start
dhcp server pool 1
ip range 172.31.1.2 172.31.1.100 mask 255.255.255.0
dns 172.31.100.1
lease-time day 0 hour 8 minute 0
!
interface vlan 1
dhcp enable server
```

通过 **show dhcp server pool** 命令查看 DHCP 服务器的地址池配置是否正确。

```
JX# show dhcp server pool

DHCP Pool 1 information :
-------------------------------------------------------------
 IP range               : 172.31.1.2 to 172.31.1.100
 Mask                  : 255.255.255.0
 Gateway                : 0.0.0.0
 DNS                   : main : 172.31.100.1 , backup : 0.0.0.0
 Lease                 : 0 days 8 hours 0 minutes
 Total Number           : 99
 Used Number            : 0
-------------------------------------------------------------
```

4.3 DHCP Relay

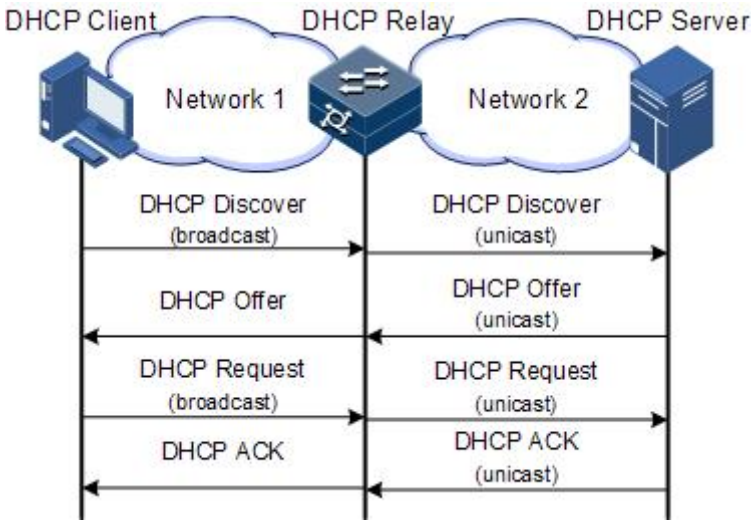
4.3.1 简介

最初的 DHCP 协议要求客户端和 DHCP 服务器只能在一个网段内，不可以跨网段工作。因此，为进行动态主机配置，需要在所有网段上都设置一个 DHCP 服务器，这显然是不经济的。

DHCP Relay（DHCP 中继）的引入解决了这个问题，它可使处于不同网段间的 DHCP 客户端和 DHCP 服务器之间承担中继服务，将 DHCP 协议报文跨网段中继到目的 DHCP 服务器，于是处在不同网段的 DHCP 客户端可以共同使用同一个 DHCP 服务器。

DHCP Relay 工作原理如图 4-8 所示。

图 4-8 DHCP Relay 工作原理示意图



DHCP 客户端发送请求报文给 DHCP 服务器，DHCP 中继在收到该报文并适当处理后，发送给指定网段上的 DHCP 服务器。服务器根据请求报文中提供的必要信息，通过 DHCP 中继返回给客户端，完成对客户端的动态配置。

4.3.2 配置准备

场景

当 DHCP 客户端和 DHCP 服务器不在同一网段时，可以使用 DHCP Relay 功能解决这一问题。它可使处于不同网段间的 DHCP 客户端和 DHCP 服务器之间承担中继服务，将 DHCP 协议报文跨网段中继到目的 DHCP 服务器，于是处在不同网段的 DHCP 客户端可以共同使用同一个 DHCP 服务器。

前提

无

4.3.3 DHCP Relay 的缺省配置

设备上 DHCP Relay 的缺省配置如下。

功能	缺省值
全局 DHCP Relay 功能状态	禁止

功能	缺省值
接口 DHCP Relay 功能状态	禁止
全局 DHCPv6 Relay 功能状态	禁止
接口 DHCPv6 Relay 功能状态	禁止

4.3.4 配置接口的 DHCP Relay 功能

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**configure**	进入全局配置模式。
2	JX(config)#**dhcp start**	使能全局 DHCP 功能。
3	JX(config)#**interface vlan** *vlan-id*	进入 VLAN 接口配置模式或者三层物理接口配置模式，以下以 VLAN 接口配置模式为例进行说明。
3	JX(config-vlanif-*)#**dhcp enable relay**	使能接口下 DHCP Relay 功能。
4	JX(config-vlanif-*)#**dhcp relay server-ip** *ip-address*	配置 DHCP 服务器的 IP 地址。

4.3.5 配置接口的 DHCPv6 Relay 功能

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**configure**	进入全局配置模式。
2	JX(config)#**dhcpv6 start**	使能全局 DHCPv6 功能。
2	JX(config)#**interface vlan** *vlan-id*	进入 VLAN 接口配置模式或者三层物理接口配置模式，以下以 VLAN 接口配置模式为例进行说明。
3	JX(config-vlanif-*)#**dhcpv6 enable relay**	使能接口下 DHCPv6 Relay 功能。
4	JX(config-vlanif-*)#**dhcpv6 relay destination** *ipv6-address*	配置 DHCPv6 服务器的 IPv6 地址。

4.3.6 配置 DHCP Relay 支持 Option 82 功能

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**configure**	进入全局配置模式。
2	JX(config)#**interface vlan** *vlan-id*	进入 VLAN 接口配置模式或者三层物理接口配置模式，以下以 VLAN 接口配置模式为例进行说明。
3	JX(config-vlanif-*)#**dhcp relay option82 enable**	配置 DHCP Relay 支持 Option 82。
4	JX(config-vlanif-*)#**dhcp relay option82** { **drop** | **keep** | **replace** }	配置 DHCP Relay 对含 Option 82 的 DHCP 请求报文处理策略。
5	JX(config-vlanif-*)#**dhcp relay option82 circuit-id format** { **default** | **user-defined** format-string }	配置 DHCP Relay Option82 的 circuitID 字段内容
6	JX(config-vlanif-*)#**dhcp relay option82 remote-id format** { **default** | **user-defined** format-string }	配置 DHCP Relay Option82 的 remoteID 字段内容

4.3.7 配置 DHCPv6 Relay 支持 Option 18/37 功能

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**configure**	进入全局配置模式。
2	JX(config)#**dhcpv6 relay interface-id format** { **default** | **user-defined** format-string }	配置 DHCPv6 报文中的 interface-id 选项（option 18）的格式。
3	JX(config)#**dhcpv6 relay remote-id format** { **default** | **user-defined** format-string }	配置 DHCPv6 报文中的 remote-id 选项（option 37）的格式。
4	JX(config)#**interface vlan** *vlan-id*	进入 VLAN 接口配置模式或者三层物理接口配置模式，以下以 VLAN 接口配置模式为例进行说明。
5	JX(config-vlanif-*)#**dhcpv6 relay interface-id enable**	配置 DHCPv6 Relay 支持 Option 18。
3	JX(config-vlanif-*)#**dhcpv6 relay remote-id enable**	配置 DHCPv6 Relay 支持 Option 37。

4.3.8 检查配置

配置完成后，请在设备上执行以下命令检查配置结果。

序号	检查项	说明
1	JX#show dhcp relay config	查看 DHCP Relay 功能配置信息。
2	JX#show dhcp relay user-bind	查看 DHCP Relay 的绑定信息。
3	JX#show dhcp relay statistics	查看 DHCP Relay 的报文统计信息。
4	JX#show dhcpv6 config	查看 DHCPv6 Relay 的配置信息。
5	JX#show dhcpv6 statistics	查看 DHCPv6 Relay 的报文统计信息。

4.3.9 维护

用户可以通过以下命令，维护设备 DHCP RELAY 特性的运行情况和配置情况。

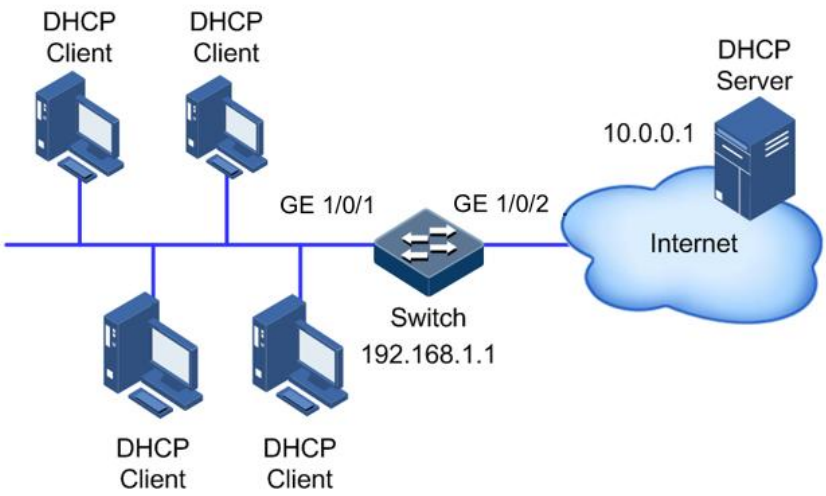
命令	描述
JX(config)#reset dhcp relay statistics	清空 DHCP Relay 统计信息。
JX(config)#reset dhcpv6 statistics	清空 DHCPv6 统计信息。

4.3.10 配置 DHCPv4 中继示例

组网需求

如图 4-9 所示，Switch 设备作为 DHCP 中继，主机名为 JX，通过业务接口接入 DHCP 服务器。需要由 DHCP 服务器分配 IP 地址给客户端设备，从而使网管平台能够发现并管理该设备。

图 4-9 配置 DHCP 中继组网示意图



配置步骤

步骤 1 配置接口的 DHCP Relay 功能。

```
JX#config
JX(config)#dhcp start
JX(config)#interface vlan 1
JX(config-vlanif-1)#ip address 192.168.1.1/24
JX(config-vlanif-1)#dhcp enable relay
JX(config-vlanif-1)#dhcp relay server-ip 10.0.0.1
JX(config-vlanif-1)#exit
```

检查结果

通过 **show dhcp relay config** 命令查看 DHCP 中继配置是否正确。

```
JX#show dhcp relay config
!
dhcp start
!
interface vlan 1
dhcp enable relay
dhcp relay server-ip 10.0.0.1
```

5 QoS

本章介绍 QoS 的基本原理和配置过程，并提供相关的配置案例。

- 简介
- 优先级映射
- 队列调度
- 拥塞避免
- 流量限速
- 带宽限速

5.1 简介

随着网络应用种类的日益丰富，用户对不同的网络应用提出了不同的服务质量需求，这就需要网络能够根据用户的需求为不同的网络应用分配和调度资源。QoS（Quality of Service，服务质量）技术的产生，使网络在发生过载或拥塞时，能够确保重要业务的实时性和完整性，同时保证整个网络的高效运行。

QoS 由一组流量管理技术组成：

- 服务模型
- 优先级信任
- 流分类
- 流策略
- 优先级映射
- 拥塞管理
- 拥塞避免

5.1.1 服务模型

QoS 技术服务模型：

- Best-effort Service（尽力而为服务模型）

- Differentiated Service（区分服务模型，简称 DiffServ）

Best-effort

Best-effort 服务模型是基于储存转发机制的 Internet（IPv4 标准）最基本、最简单的服务模型。在 Best-effort 服务模型中，应用程序在任何时刻，发送任意数量的报文，而且事先不需要经过批准，也不需要通知网络。对 Best-effort 服务来说，网络尽最大可能来发送报文，但对时延、可靠性等不做任何保证。

Best-effort 模型是现在 Internet 缺省的服务模型，适用于大多数网络应用，如 FTP，E-mail 等，它通过先入先出（FIFO）队列来实现。

DiffServ

DiffServ 模型是一个多服务模型，它可以满足不同的 QoS 需求。

对于 DiffServ 模型来说，它不需要为每个流维护状态。它根据每个报文的 QoS 分类来提供差异化的服务。可以使用不同的方法进行报文的 QoS 分类，如 IP 报文的优先级（IP Precedence）、报文的源地址或目的地址等。

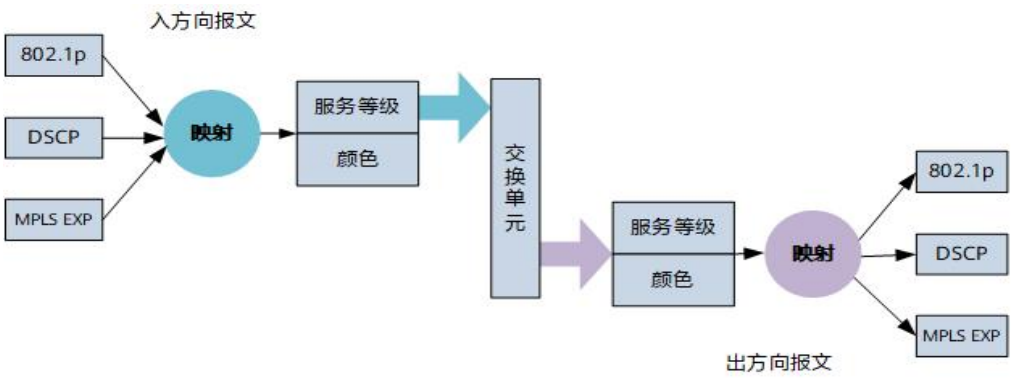
DiffServ 一般用来为一些重要的应用提供端到端的 QoS 服务。主要通过以下技术来实现：

- CAR（Committed Access Rate，承诺访问速率）：根据预先设定的报文匹配规则。如 IP 报文的优先级、报文的源地址或目的地址等，进行报文的分类。如果是符合令牌桶流量规则的报文，就继续发送。如果是超出流量规定的报文，则被丢弃或者重新标记 IP Precedence、DSCP、EXP 等。CAR 不仅可以进行流量控制，还可以对报文进行标记和重标记。
- 队列技术：SP、WRR、DRR、SP+WRR、SP+DRR 等队列技术对拥塞的报文进行缓存和调度，实现拥塞管理。

5.2 优先级映射

5.2.1 简介

优先级映射



服务等级是指报文在设备内部的服务质量，它决定了报文在设备内部所属的队列类型。服务等级有 8 种取值，即 8 种 PHB（Per-Hop Behavior），优先级从高到低依次为 CS7、CS6、EF、AF4、AF3、AF2、AF1、BE。PHB 行为的详细描述，参见 PHB 行为。

颜色是指报文在设备内部的丢弃优先级，用于决定同一个队列内部当队列发生拥塞时报文的丢弃顺序。颜色有 3 种取值，IEEE 定义的优先级从低到高依次为 Green、Yellow、Red。丢弃优先级的高低实际取决于对应参数的配置。

在每一个 DS 节点上对报文的处理称为 PHB。PHB 描述了 DS 节点对报文采用的外部可见的转发行为。PHB 可以用优先级来定义，也可以用一些可见的服务特征如报文延迟、抖动或丢包率来定义。PHB 只定义了一些外部可见的转发行为，没有指定特定的实现方式。

RFC 定义了四种标准的 PHB：CS（Class Selector），EF（Expedited Forwarding），AF（Assured Forwarding）和 BE（Best-Effort）。其中，BE 是缺省的 PHB。

在 RFC 2474 中，CS 又被划分为两个等级，即 CS6 和 CS7；在 RFC 2597 中，AF 又被划分为四个等级，即为 AF1～AF4。至此，PHB 共有 8 个细分级别，每个 PHB 在设备内部都有对应的服务等级，不同的服务等级将决定不同流的拥塞管理策略。同时每个 PHB 又再被划分为三个颜色（Color，也可以叫丢弃优先级），分别用 Green、Yellow 和 Red 表示，不同的颜色将决定不同流的拥塞避免策略。

优先级信任

优先级信任是指设备采用报文自身携带的优先级作为分类依据，对报文进行后续的 QoS 管理操作。

信任的优先级种类有：

- 基于 DSCP（Differentiated Services Code Point，差分服务代码点）优先级
- 基于 8021p inner 优先级
- 基于 8021p outer 优先级

5.2.2 配置准备

场景

对于来自上游设备的报文，用户可以选择信任报文携带的优先级，对于不信任其优先级的报文可以交由流分类和流策略处理。配置优先级信任模式后，设备可以依据不同优先级对报文进行相应的操作，提供相应的服务。

为报文指定本地优先级是进行队列调度的前提条件，对于来自上游设备的报文，用户可以将报文携带的外部优先级映射到不同的本地优先级，也可以基于接口直接配置报文的本地优先级，之后设备将依据本地优先级对报文进行队列调度。通常来说，对于 IP 报文，需要配置 IP 优先级或 DSCP 优先级与本地优先级的映射关系；对于 VLAN 报文，需要配置 8021p outer 优先级与本地优先级的映射关系。

前提

无

5.2.3 基本 QoS 的缺省配置

设备上基本 QoS 的缺省配置如下。

功能	缺省值
全局 QoS 功能状态	使能
入接口信任的优先级类型	信任 8021p outer 优先级
出接口优先级重标记	去使能

表 5-1 缺省情况下 8021p 入方向和本地优先级及颜色的映射关系

802.1p 优先级	PHB 行为	Color
0	BE	green
1	AF1	green
2	AF2	green
3	AF3	green
4	AF4	green
5	EF	green
6	CS6	green
7	CS7	green

表 5-2 缺省情况下 8021p 出方向和本地优先级及颜色的映射关系

PHB 行为	Color	802.1p 优先级
BE	green	0
BE	yellow	0
BE	red	0
AF1	green	1
AF1	yellow	1
AF1	red	1
AF2	green	2
AF2	yellow	2
AF2	red	2
AF3	green	3

AF3	yellow	3
AF3	red	3
AF4	green	4
AF4	yellow	4
AF4	red	4
EF	green	5
EF	yellow	5
EF	red	5
CS6	green	6
CS6	yellow	6
CS6	red	6
CS7	green	7
CS7	yellow	7
CS7	red	7

表 5-3 缺省情况下 DSCP 入方向和本地优先级及颜色的映射关系

DSCP	PHB 行为	Color	DSCP	PHB 行为	Color
0	BE	green	32	AF4	green
1	BE	green	33	BE	green
2	BE	green	34	AF4	green
3	BE	green	35	BE	green
4	BE	green	36	AF4	yellow
5	BE	green	37	BE	green
6	BE	green	38	AF4	red
7	BE	green	39	BE	green
8	AF1	green	40	EF	green
9	BE	green	41	BE	green
10	AF1	green	42	BE	green
11	BE	green	43	BE	green
12	AF1	yellow	44	BE	green
13	BE	green	45	BE	green
14	AF1	red	46	EF	green

DSCP	PHB 行为	Color	DSCP	PHB 行为	Color
15	BE	green	47	BE	green
16	AF2	green	48	CS6	green
17	BE	green	49	BE	green
18	AF2	green	50	BE	green
19	BE	green	51	BE	green
20	AF2	yellow	52	BE	green
21	BE	green	53	BE	green
22	AF2	red	54	BE	green
23	BE	green	55	BE	green
24	AF3	green	56	CS7	green
25	BE	green	57	BE	green
26	AF3	green	58	BE	green
27	BE	green	59	BE	green
28	AF3	yellow	60	BE	green
29	BE	green	61	BE	green
30	AF3	red	62	BE	green
31	BE	green	63	BE	green

表 5-4 缺省情况下 DSCP 出方向和本地优先级及颜色的映射关系

PHB 行为	Color	DSCP
BE	green	0
BE	yellow	0
BE	red	0
AF1	green	10
AF1	yellow	12
AF1	red	14
AF2	green	18
AF2	yellow	20
AF2	red	22
AF3	green	26

PHB 行为	Color	DSCP
AF3	yellow	28
AF3	red	30
AF4	green	34
AF4	yellow	36
AF4	red	38
EF	green	46
EF	yellow	46
EF	red	46
CS6	green	48
CS6	yellow	48
CS6	red	48
CS7	green	56
CS7	yellow	56
CS7	red	56

5.2.4 配置入接口信任的优先级类型

请在需要的设备上进行以下配置。

步骤	配置	说明
1	JX#**configure**	进入全局配置模式。
2	JX(config)#**interface** *interface-type interface-number*	进入二层物理接口配置模式或聚合组接口配置模式。以下步骤以物理接口配置模式为例。
3	JX(config-ge-1/0/*)#**trust** [**dot1p** { **inner** | **outer** } | **dscp**]	在接口上设置信任的报文优先级 dot1p inner : vlan 报文的内层优先级 dot1p outer: vlan 报文的外层优先级 dscp：ip 报文优先级

5.2.5 配置 DIFFSERV 模版

请在需要的设备上进行以下配置。

步骤	配置	说明
1	JX#**configure**	进入全局配置模式。
2	JX(config)#**diffserv domain default**	创建 diffserv 域模版或者进入 default 模版 default　默认模版 ds-domain-name　需要创建的模版
3	JX(config-dsdomain-ds1)#**8021p-inbound 8021p-value phb { BE | AF1 | AF2 | AF3 | AF4 | EF | CS6 | CS7 } [green | yellow | red]**	8021p-value：　8021p 优先级 BE|AF1|AF2|AF3|AF4|EF|CS6|CS7：8 个队列 green|yellow|red：3 种颜色
4	JX(config-dsdomain-ds1)#**8021p-outbound { BE | AF1 | AF2 | AF3 | AF4 | EF | CS6 | CS7 } [green | yellow | red] map 8021p-value**	8021p-value：　8021p 优先级 BE|AF1|AF2|AF3|AF4|EF|CS6|CS7：8 个队列 green|yellow|red：3 种颜色
5	JX(config-dsdomain-ds1)#**ip-dscp-inbound dscp-value phb { BE | AF1 | AF2 | AF3 | AF4 | EF | CS6 | CS7 } [green | yellow | red]**	dscp-value ：　dscp 优先级 BE|AF1|AF2|AF3|AF4|EF|CS6|CS7：8 个队列 green|yellow|red：3 种颜色
6	JX(config-dsdomain-ds1)#**ip-dscp-outbound { BE | AF1 | AF2 | AF3 | AF4 | EF | CS6 | CS7 } [green | yellow | red] map dscp-value**	dscp-value ：　dscp 优先级 BE|AF1|AF2|AF3|AF4|EF|CS6|CS7：8 个队列 green|yellow|red：3 种颜色

5.2.6 配置出接口 8021P/DSCP 重标记

请在需要的设备上进行以下配置。

步骤	配置	说明
1	JX#**configure**	进入全局配置模式。
2	JX(config)#**interface** *interface-type interface-number*	进入二层物理接口配置模式或聚合组配置模式。以下步骤以物理接口配置模式为例。

步骤	配置	说明
3	JX(config-ge-1/0/*)#**qos phb marking { 8021p | dscp } { enable | disable }**	使能/去使能再出接口上的内部优先级到外部优先级的重标记 **8021p：重标记 8021p 优先级** **Dscp：重标记 dscp 优先级**

5.2.7 检查配置

配置完成后，请在设备上执行以下命令检查配置结果。

序号	检查项	说明
1	JX#**show differv domain [ds-domain-name]**	显示配置的 diffserv 域模版信息
2	JX#**show differv interface** *interface-type interface-number*	显示接口下的 diffserv 配置

5.3 队列调度

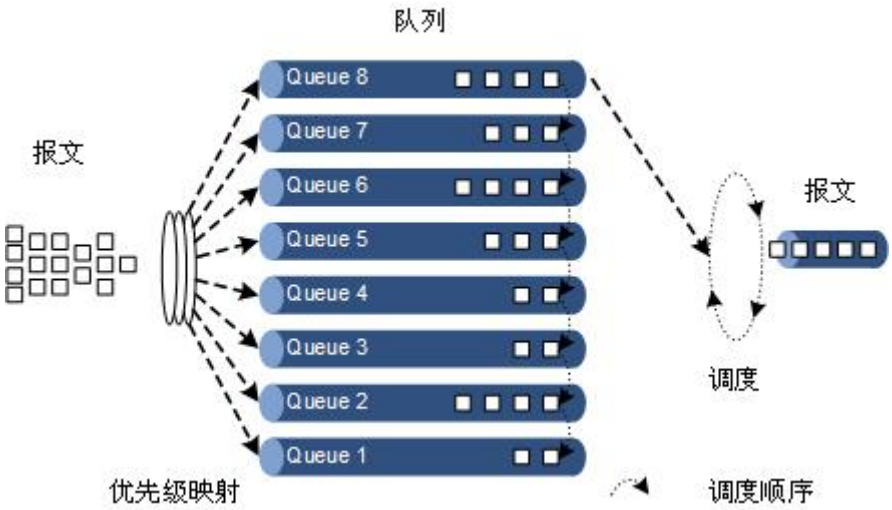
5.3.1 简介

当时延敏感业务要求得到比非时延敏感业务更高质量的 QoS 服务，或网络间歇性的出现拥塞时，需要进行队列调度。

队列调度是指使用不同的调度算法来发送队列中的报文流。设备支持的队列调度算法有 SP（Strict-Priority，严格优先级调度）、WRR（Weight Round Robin，加权循环调度）、WFQ（Weighted Fair Queueing，加权公平队列）、SP+WRR、SP+WFQ。每种调度算法都是为了解决特定网络流量的问题，并对带宽资源的分配、延迟、抖动等有不同的影响。

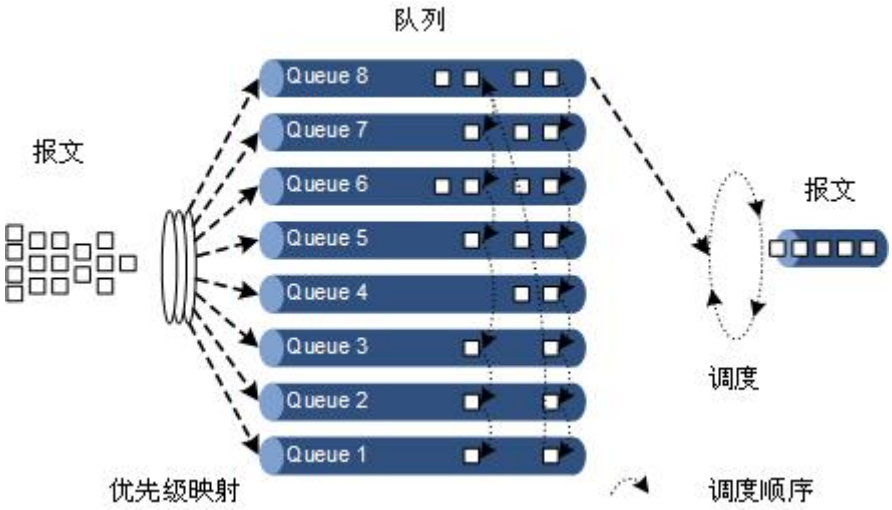
- SP：设备严格按照队列优先级的高低顺序进行调度。只有高优先级队列中的报文全部调度完毕后，低优先级队列才有调度的机会。如下图所示。

图 5-2 SP 调度示意图



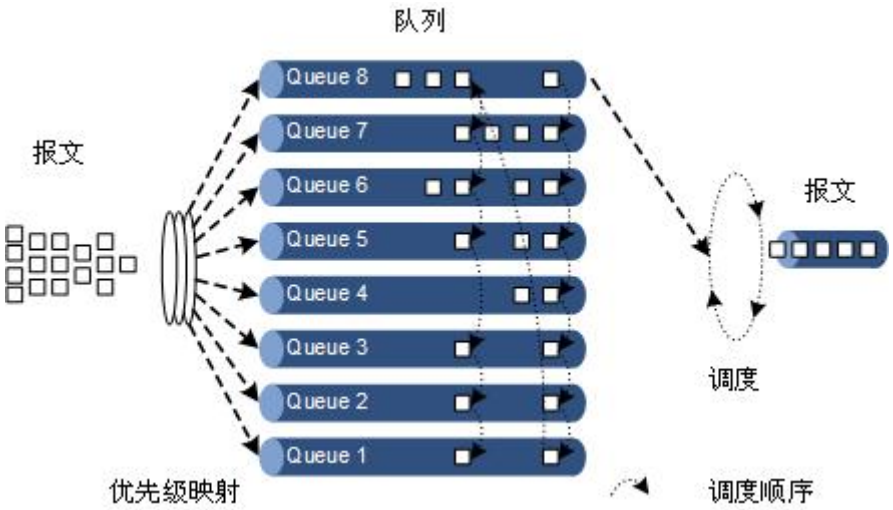
- WRR：在按照队列的优先级次序以循环方式调度每个队列报文的基础上，根据每个队列的权重，以报文个数来调度各队列中的报文。如下图所示。

图 5-3 WRR 调度示意图



- WFQ：在按照队列的优先级次序以循环方式调度每个队列报文的基础上，根据每个队列的权重，以 bit 为单位来调度各队列中的报文。如下图所示。

图 5-4 WFQ 调度示意图



- SP+WRR：SP 调度和 WRR 调度相结合的调度方式，将设备接口上的队列分为两组，用户可以指定其中的某几组队列进行 SP 调度，其他队列进行 WRR 调度。
- SP+WFQ：SP 调度和 WFQ 调度相结合的调度方式，将设备接口上的队列分为两组，用户可以指定其中的某几组队列进行 SP 调度，其他队列进行 WFQ 调度。

5.3.2 配置准备

场景

当网络发生拥塞时，若用户希望能均衡各类报文的延迟和延迟抖动，关键业务（如视频业务、语音业务）的报文能够得到优先处理；同时对于非关键业务（如 E-Mail）的报文，保证相同优先级业务得到公平处理，不同优先级业务按照各自权值处理。可以通过配置队列调度来实现。具体选择何种调度算法，需要依据当时的业务情况与客户需求。

前提

全局 QoS 功能使能。

5.3.3 拥塞管理的缺省配置

设备上拥塞管理的缺省配置如下。

功能	缺省值
队列调度模式	SP
队列权重	• WRR 调度 8 个队列的权重均为 1 • WFR 调度 8 个队列的权重均为 1

5.3.4 配置 SP 队列调度

请在需要的设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**interface** *interface-type interface-number*	进入二层物理接口配置模式或聚合组接口配置模式。以下步骤以二层物理接口配置模式为例。
3	JX(config-ge-1/0/*)#**queue scheduling sp**	配置接口队列调度方式为 SP。

5.3.5 配置 WRR 或 SP+WRR 队列调度

请在需要的设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**interface** *interface-type interface-number*	进入二层物理接口配置模式或聚合组接口配置模式。以下步骤以二层物理接口配置模式为例。
3	JX(config-ge-1/0/*)#**queue scheduling wrr**	配置接口队列调度方式为 WRR。
4	JX(config-ge-1/0/*)#**queue** *queue-id* **weight** *weight-value*	配置各队列的权重。

5.3.6 配置 DRR 或 SP+DRR 队列调度

请在需要配置 DRR 或 SP+DRR 队列调度的设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**interface** *interface-type interface-number*	进入二层物理接口配置模式或聚合组接口配置模式。以下步骤以二层物理接口配置模式为例。
3	JX(config-ge-1/0/*)#**queue scheduling wfq**	配置接口队列调度方式为 WFQ。
4	JX(config-ge-1/0/*)#**queue** *queue-id* **weight** *weight-value*	配置各队列的权重。

5.3.7 配置队列带宽保证

请在需要的设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**interface** *interface-type interface-number*	进入二层物理接口配置模式或聚合组接口配置模式。以下步骤以二层物理接口配置模式为例。
3	JX(config-ge-1/0/1)# **queue** *queue-id* **min-bandwidth** { **kbps** | **mbps** | **gbps** } *bandwidth*	配置基于接口队列的带宽保证。

5.3.8 检查配置

配置完成后，请在设备上执行以下命令检查配置结果。

序号	检查项	说明
1	JX#**show queue interface**	查看接口的队列信息。
2	JX#**show queue statistics interface** *interface-type interface-number*	查看接口下队列的统计信息。

5.3.9 维护

用户可以通过以下命令，维护队列调度。

命令	描述
JX(config)#**reset queue statistics interface** *interface-type interface-number*	清空接口下队列对包数统计。

5.3.10 配置队列调度示例

组网需求

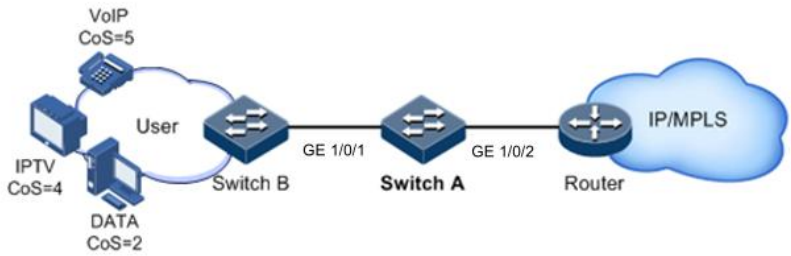
如下图所示，来自 User 的业务类型有语音、视频和数据。

语音业务的 CoS 优先级为 5，视频业务的 CoS 优先级为 4，数据业务的 CoS 优先级为 2。

在 Switch A 处容易发生拥塞，为了减轻网络拥塞造成的影响，根据不同的业务类型，需要制定如下规则：

- 对于语音业务，需要对其进行 SP 调度，优先保证这部分流量通过；
- 对于视频业务，需要对其进行 WRR 调度，权重值为 50；
- 对于数据业务，需要对其进行 WRR 调度，权重值为 20。

图 5-5 配置队列调度组网示意图



配置步骤

步骤 1 配置接口报文根据 8021p 映射内部优先级。

```
SwitchA(config)#interface ge 1/0/2
SwitchA(config-ge-1/0/2)#trust 8021p outer
```

步骤 2 配置在 GE 1/0/2 出方向进行 SP+WRR 队列调度。

```
SwitchA(config)#interface ge 1/0/2
SwitchA(config-ge-1/0/2)#queue scheduling sp+wrr queue 5
SwitchA(config-ge-1/0/2)#queue 4 weight 50
SwitchA(config-ge-1/0/2)#queue 2 weight 20
SwitchA(config-ge-1/0/2)#quit
```

检查结果

查看接口优先级信任模式。

```
JX#show queue interface ge 1/0/2
  Current scheduling algorithm is 'sp+wrr',queue list 5
  'Max-BW' means 'Max Bandwidth'
  'Min-BW' means 'Min Bandwidth'
  Interface     Queue  Max-BW       Min-BW       Weight
-------------------------------------------------------------------
  ge-1/0/2      0      0M           0M           1
  ge-1/0/2      1      0M           0M           1
  ge-1/0/2      2      0M           0M           20
  ge-1/0/2      3      0M           0M           1
  ge-1/0/2      4      0M           0M           50
  ge-1/0/2      5      0M           0M           --
  ge-1/0/2      6      0M           0M           1
  ge-1/0/2      7      0M           0M           1
-------------------------------------------------------------------
```

5.4 拥塞避免

5.4.1 简介

拥塞避免（Congestion Avoidance）是指通过监视网络资源（如队列或内存缓冲区）的使用情况，在拥塞发生或有加剧的趋势时主动丢弃报文，通过调整网络的流量来解除网络过载的一种流量控制机制。

传统的丢包策略采用尾部丢弃（Tail-Drop）的方法，同等的对待所有的报文，不对服务等级进行区分。在拥塞发生期间，队列尾部的数据包将被丢弃，直到拥塞解决。

这种丢弃策略会引发 TCP 全局同步现象，使网络流量忽大忽小，影响链路利用率。

RED

RED（Random Early Detection，随机早期检测）技术通过随机地丢弃报文，让多个 TCP 连接不同时降低发送速度，从而避免了 TCP 的全局同步现象。

在 RED 技术的算法中，为每个队列的长度都设定了阈值上下限，并规定：

- 当队列的长度小于阈值下限时，不丢弃报文。
- 当队列的长度大于阈值上限时，丢弃所有收到的报文。
- 当队列的长度在阈值上限和阈值下限之间时，随机丢弃到来的报文。队列越长，报文被丢弃的概率越高。

WRED

加权随机早期检测 WRED（Weighted Random Early Detection）技术也是通过随机丢弃报文来避免 TCP 的全局同步现象，但该技术生成的随机丢弃参数是基于队列的优先级，它可以依据报文的不同颜色来区别丢弃策略，考虑了高优先级报文的利益并使其被丢弃的概率相对较小。

5.4.2 配置准备

场景

为避免网络拥塞的发生，解决 TCP 全局同步的问题，可以通过配置拥塞避免，调整网络流量，解除网络过载。

前提

5.4.3 拥塞避免的缺省配置

设备上拥塞避免的缺省配置如下。

功能	缺省值
接口 WRED 功能状态	不使能

5.4.4 配置 WRED

请在需要的设备上进行以下配置。

步骤	配置	说明
1	JX#**configure**	进入全局配置模式。
2	JX(config)#**drop-profile** *drop-profile-name*	创建 WRED 模板，并进入 WRED 配置模式。
3	JX(config-drop-profile)#**color** { **green** | **non-tcp** | **red** | **yellow** } **low-limit** low-limit-percentage **high-limit** high-limit-percentage **discard-percentage** *discard-percentage*	{ green | non-tcp | red | yellow }:设置针对绿色/非 TCP/红色/黄色报文的 WRED 参数 low-limit-percentag :WRED 丢弃的低门限百分比，即 WRED 丢弃的低门限值占队列长度的百分比。 high-limit-percentage :WRED 丢弃的高门限值百分比，即 WRED 丢弃的高门限值占队列长度的百分比。 discard-percentage :WRED 的最大丢弃概率 缺省情况下，WRED 丢弃模板的高低门限百分比以及最大丢弃概率的取值均为 100
4	JX(config)#**interface** *interface-type interface-number*	进入二层物理层接口配置模式或聚合组接口配置模式。以下步骤以二层物理接口配置模式为例。
5	JX(config-ge-1/0/*)#**qos wred** *drop-profile-name*	配置接口上应用 WRED 模板。
6	JX(config-ge-1/0/*)#**qos queue** *queue-index* **wred** *drop-profile-name*	在接口队列下应用丢弃模版 queue-index 接口队列号 0-7 对应 BE AF1 AF2 AF3 AF4 EF CS6 CS7 drop-profile-name 应用的丢弃模板名称

5.4.5 检查配置

配置完成后，请在设备上执行以下命令检查配置结果。

序号	检查项	说明
1	JX#**show drop-profile** { **all** | **name** *drop-profile-name* }	查看 WRED 模板信息。

5.5 流量限速

5.5.1 简介

设备基于流策略对报文流量进行限速外，还支持基于接口、基于 VLAN、基于接口+VLAN 对报文流量进行限速。与基于流策略的流量限速类似，设备对超出部分流量采取丢弃措施。

5.5.2 配置准备

场景

为避免网络发生拥塞或缓解网络拥塞状况，用户可以配置基于接口的流量限速。通过限制某一接口的突发流量，使之以比较均匀的速率发送。

前提

无

5.5.3 配置流量限速模式

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**rate-limit mode** { **l1** | **l2** }	配置流量限速的工作模式。

5.5.4 配置基于物理接口的流量限速

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**interface** *interface-type interface-number*	进入物理接口配置模式。
3	JX(config-ge-1/0/*)#**rate-limit** { **ingress** | **egress** } **cir** *cir-value*	配置入方向或出方向基于接口的流量限速。
3	JX(config-ge-1/0/*)#**rate-limit** { **ingress** | **egress** } **cir** *cir-value* **cbs** *cbs-value*	配置入方向或出方向基于接口的流量限速。

说明

- 缺省情况下，未配置接口的带宽限制。
- 入接口限速资源超过设定的限速值后的处理方式 drop 丢弃。
- 接口配置限速值和突发值时，在配置的限速值小于 256kbit/s 时，突发值设置不能太大，否则会出现不连续的情况。
- 限速值很小时，建议限速值和突发值在数字上成 4 倍的关系，如果出现了报文不连续的情况，此时请减小突发值，或者增大限速值。
- 出接口的限速丢包会统计到入接口的丢包统计上。
- 聚合组接口模式下不支持出方向的流量限速。

5.5.5 配置基于 VLAN 的流量限速

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**interface vlan** *vlan-id*	进入 VLAN 接口配置模式。
3	JX(config-vlan*)#**rate-limit ingress cir** *cir-value*	配置入方向基于 VLAN 接口的流量限速。
	JX(config-vlan*)#**rate-limit ingress cir** *cir-value* **cbs** *cbs-value*	配置入方向基于 VLAN 接口的流量限速。

5.5.6 配置基于聚合组的流量限速

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**interface port-channel** *port-channel-id*	进入聚合组接口配置模式。
3	JX(config-port-channel*)#**rate-limit ingress cir** *cir-value*	配置入方向基于聚合组接口的流量限速。
	JX(config-port-channel*)#**rate-limit ingress cir** *cir-value* **cbs** *cbs-value*	配置入方向基于聚合组接口的流量限速。

5.5.7 检查配置

配置完成后，请在设备上执行以下命令检查配置结果。

序号	检查项	说明
1	JX#**show rate-limit interface** [{ *interface-type interface-number* | **vlan** *vlan-id* } [**ingress** | **egress**]]	查看基于接口的流量限速。
2	JX#**show rate-limit mode**	查看接口的带宽限速模式。

5.5.8 配置基于接口的流量限速示例

组网需求

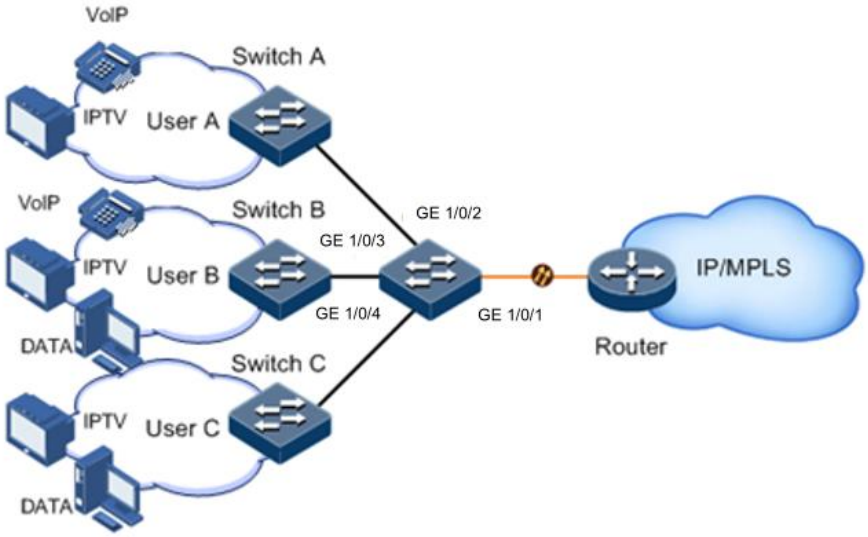
如图 5-5 所示，User A、User B、User C 分别通过 Switch A、Switch B、Switch C 和交换机相连。

来自 User A 的业务类型有语音和视频，来自 User B 的业务类型有语音、视频和数据，来自 User C 的业务类型有视频和数据。

根据各用户的业务需求，需要制定如下规则：

- 对于 User A，须为其提供保证带宽 25M，突发流量允许 100kB，丢弃多余的流量；
- 对于 User B，须为其提供保证带宽 35M，突发流量允许 100kB，丢弃多余的流量；
- 对于 User C，须为其提供保证带宽 30M，突发流量允许 100kB，丢弃多余的流量。

图 5-6 配置基于接口的流量限速组网示意图



配置步骤

步骤 1 配置基于接口的流量限速。

```
JX#config
JX(config)#interface ge 1/0/1
JX(config-ge-1/0/1)#rate-limit ingress cir 25000 cbs 100
JX(config-ge-1/0/1)#exit
JX(config)#interface ge 1/0/2
JX(config-ge-1/0/2)#rate-limit ingress cir 35000 cbs 100
JX(config-ge-1/0/2)#exit
JX(config)#interface ge 1/0/3
JX(config-ge-1/0/3)#rate-limit ingress cir 30000 cbs 100
JX(config-ge-1/0/3)#exit
```

检查结果

通过 **show rate-limit interface** 命令查看基于接口的流量限速配置是否正确。

```
JX(config)#show rate-limit interface
Interface       Direction Cir(kbps)    Cbs(kb)   CirOper(kbps)  CbsOper(kb)
-------------------------------------------------------------------------
ge-1/0/1  ingress   25000       100       25024          101
ge-1/0/2  ingress   35000       100       30016          101
ge-1/0/3  ingress   30000       100       30016           101
```

5.6 带宽限速

5.6.1 简介

带宽限速是 QoS 的一个子功能，它比基本 QoS 更加灵活，在交换机设备上应用广泛。

带宽限速的主要功能如下：

- 入端口
  - 带宽保证：实现基于端口或基于流的带宽服务，同时支持分层带宽保证，细化不同业务流的带宽。
  - 识别：当流从带宽保证端口进入时，决定是否对报文进行颜色识别。
- 出端口
  - 带宽保证：实现基于端口或基于流的带宽服务，不支持分层带宽保证。
  - 标记：当流从带宽保证端口转出时，决定是否对报文进行颜色标记。

带宽保证

带宽保证功能能够保证接入网络的业务流量保持在规定的范围内，对于超出的流量进行“惩罚”，如丢弃或选择调度。带宽保证既能满足用户对业务带宽的要求，也能保护网络资源和运营商的利益。

用户通过配置带宽保证模板并在端口上应用模板，可以实现对端口的流量进行颜色（绿色、黄色和红色）标记。设备会根据流量的颜色作出不同处理，绿色流量保证转发，黄色流量选择调度，红色流量直接丢弃。

分层带宽保证

分层带宽保证是一种更灵活的带宽保证，用户不仅可以为每个流量单独配置带宽保证，还可以通过分层带宽对多个流量的总和进行带宽保证。

颜色识别与标记

使能颜色识别功能，设备会处于 Color-aware 状态，可以识别从上游设备流入端口的流量是否带有颜色。如果禁用颜色识别功能，设备处于 Color-blind 状态，将忽略进入端口的流量是否带有颜色，设备会重新判断该流量的颜色。

颜色标记是设备根据用户在带宽保证模板中设定的 CIR、CBS、EIR、EBS 参数来判断业务流量属于哪一种颜色，并根据 802.1ad 标准定义的报文格式，修改流量报文中相关标志位，使之带有颜色。

5.6.2 配置准备

场景

带宽限速功能主要用来保证用户对业务带宽的要求，并保护网络资源和运营商的利益。

前提

无

5.6.3 带宽限速功能的缺省配置

设备上带宽限速功能的缺省配置如下。

功能	缺省值
颜色识别	禁用

5.6.4 配置带宽保证功能

创建带宽保证模板

请在需要的设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**bandwidth-profile** *bwp-profile-id* **cir** *cir* **cbs** *cbs* [**color-aware**]	创建带宽保证模板。
3	JX(config)#**bandwidth-profile** *bwp-profile-id* **description** *word*	配置带宽保证模板的描述信息。
4	JX(config)#**interface** *interface-type interface-number*	进入二层物理接口配置模式。
5	JX(config-ge-1/0/*)#**bandwidth ingress** *bwp-profile-id*	在入方向二层物理接口上应用带宽保证模板。
	JX(config-ge-1/0/*)#**bandwidth egress** *bwp-profile-id*	在出方向二层物理接口上应用带宽保证模板。
6	JX(config-ge-1/0/*)#**bandwidth color-aware** { **enable** | **disable** }	使能带宽保证接口入方向报文的颜色识别功能，使用 **disable** 格式禁止该功能。
7	JX(config-ge-1/0/*)#**exit** JX(config)#**interface vlan** *vlan-id*	进入 VLAN 接口配置模式。
8	JX(config-vlan*)#**bandwidth ingress** *bwp-profile-id*	在入方向 VLAN 接口上应用带宽保证模板。
	JX(config-vlan*)#**bandwidth egress** *bwp-profile-id*	在出方向 VLAN 接口上应用带宽保证模板。
9	JX(config-vlan*)#**exit** JX(config)#**interface port-channel** *port-channel-id*	进入聚合组接口配置模式。

步骤	配置	说明
10	JX(config-port-channel*)#**bandwidth ingress** *bwp-profile-id*	在入方向聚合组接口上应用带宽保证模板。

配置端口+VLAN 带宽保证

请在需要的设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**bandwidth-profile** *bwp-profile-id* **cir** *cir* **cbs** *cbs* [**color-aware**]	创建带宽保证模板。
3	JX(config)#**interface** *interface-type interface-number*	进入二层物理接口配置模式。
4	JX(config-ge-1/0/*)#**bandwidth ingress vlan** *vlan-id bwp-profile-id*	在入方向端口+VLAN 上应用带宽保证模板。
	JX(config-ge-1/0/*)#**bandwidth egress vlan** *vlan-id bwp-profile-id*	在出方向端口+VLAN 上应用带宽保证模板。
5	JX(config-ge-1/0/*)#**exit** JX(config)#**interface** *port-channel-id*	进入聚合组接口配置模式。
6	JX(config-port-channel*)#**bandwidth ingress vlan** *vlan-id bwp-profile-id*	在入方向端口+VLAN 上应用带宽保证模板。

配置端口+ CoS 带宽保证

请在需要的设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**bandwidth-profile** *bwp-profile-id* **cir** *cir* **cbs** *cbs* [**color-aware**]	创建带宽保证模板。
3	JX(config)#**interface** *interface-type interface-number*	进入二层物理接口配置模式。
4	JX(config-ge-1/0/*))#**bandwidth ingress coslist** *cos-value-list bwp-profile-id*	在入方向端口+CoS 上应用带宽保证模板。

步骤	配置	说明
	JX(config-ge-1/0/*))#**bandwidth egress coslist** *cos-value-list bwp-profile-id*	在出方向端口+CoS 上应用带宽保证模板。
5	JX(config-ge-1/0/*)#**exit** JX(config)#**interface port-channel** *port-channel-id*	进入聚合组接口配置模式。
6	JX(config-port-channel*)#**bandwidth ingress coslist** *cos-value-list bwp-profile-id*	在入方向端口+CoS 上应用带宽保证模板。
7	JX(config- port-channel*)#**exit** JX(config)#**interface vlan** *vlan-id*	进入 VLAN 接口配置模式。
8	JX(config-vlan*))#**bandwidth ingress coslist** *cos-value-list bwp-profile-id*	在入方向端口+CoS 上应用带宽保证模板。
	JX(config-vlan*))#**bandwidth egress coslist** *cos-value-list bwp-profile-id*	在出方向端口+CoS 上应用带宽保证模板。

说明

当删除带宽保证模板时，如果该模板被其它分层模板引用或该模板已被应用，则不能删除。

配置端口+VLAN+CoS 带宽保证

请在需要的设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**bandwidth-profile** *bwp-profile-id* **cir** *cir* **cbs** *cbs* [**color-aware**]	创建带宽保证模板。
3	JX(config)#**interface** *interface-type interface-number*	进入二层物理接口配置模式。
4	JX(config-ge-1/0/*)#**bandwidth ingress vlan** *vlan-id* **coslist** *cos-value-list bwp-profile-id*	在入方向端口+VLAN+CoS 上应用带宽保证模板。
	JX(config-ge-1/0/*)#**bandwidth egress vlan** *vlan-id* **coslist** *cos-value-list bwp-profile-id*	在出方向端口+VLAN+CoS 上应用带宽保证模板。
5	JX(config-ge-1/0/*)#**exit** JX(config)#**interface port-channel** *port-channel-id*	进入聚合组接口配置模式。
6	JX(config-port-channel*)#**bandwidth ingress vlan** *vlan-id* **coslist** *cos-value-list bwp-profile-id*	在入方向端口+VLAN+CoS 上应用带宽保证模板。

5.6.5 检查配置

配置完成后，请在设备上执行以下命令检查配置结果。

序号	检查项	说明
1	JX#**show bandwidth-profile** [*bwp-profile-id*]	查看带宽保证模板信息。
2	JX#**show bandwidth interface** *interface-type interface-number*	查看端口的带宽保证配置信息。

6 组播

本章介绍了组播特性的原理和配置过程，并提供相关的配置案例。

- 组播概述
- IGMP Snooping
- MLD Snooping
- IGMP
- PIM
- MLD
- PIMv6

6.1 组播概述

随着 Internet 网络的不断发展，一方面网络中交互的各种数据、语音和视频信息越来越多；另一方面新兴的电子商务、网上会议、网上拍卖、视频点播、远程教学等服务逐渐兴起，这些服务对网络带宽、信息安全性和有偿性提出了更高的要求。传统的单播和广播方式无法很好的满足这些要求，组播及时满足了这些需求。

组播（Multicast）是一种点到多点的数据传输方式。组播技术能够有效地解决单点发送，多点接收的问题，在网络数据传输时，能够节约网络资源，提高信息安全性。

三种通信方式的比较

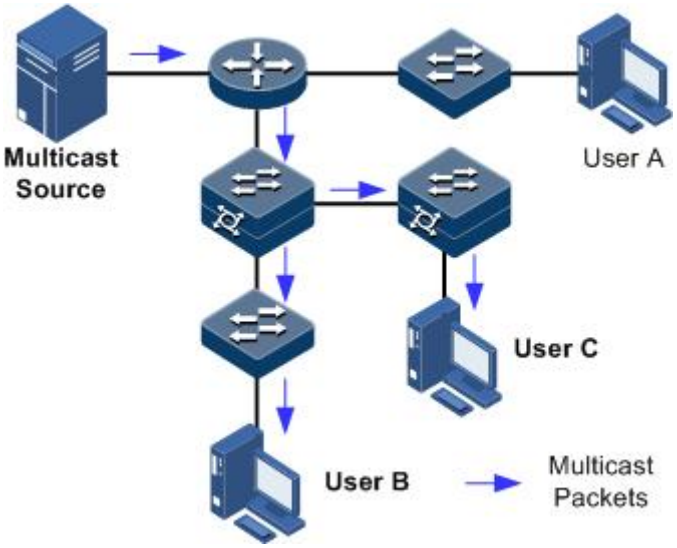
组播是与单播、广播并列的通信方式：

- 单播方式：系统为每个需要信息的用户单独建立一条数据传送通路，并为该用户发送一份独立的拷贝信息。利用单播方式，网络中传输的信息量和需要该信息的用户量成正比，因此当需要该信息的用户量庞大时，网络中将出现多份相同信息。此时，带宽将成为重要瓶颈，单播方式不利于信息规模化发送。
- 广播方式：系统把信息传送给网络中的所有用户，不管他们是否需要，任何用户都会接收到广播来的信息。利用广播方式，信息源将把信息传递给网段中所有用户，用户信息安全性和有偿服务得不到保障。另外，当同一网络中需要该信息的用户量很小时，网络资源利用率将非常低，带宽浪费严重。

- 组播方式：当网络中的某些用户需求特定信息时，组播信息发送者仅发送一份信息，被传递的信息在尽可能远的分叉路口才开始复制和分发。

组播方式传输如下图所示。假设用户 B、C 需要信息，采用组播方式传输，将用户 B、C 组成一个接收者集合，信息源只需发送一份信息，由网络中各交换机根据 IGMP 报文建立自己的组播转发表，信息只传输给实际需要的接收者 B、C。

图 6-1 组播方式传输信息示意图



综上所述，单播方式适合用户稀少的网络，广播方式适合用户稠密的网络，当网络中需要某信息的用户量不确定时，单播和广播方式效率很低。组播方式传输，用户数量成倍增长时，主干带宽不需要随之增加，而且只将信息传递给需要的用户，这些优点使它成为当前网络技术中的研究热点之一。

组播优势与应用

与单播和广播通信方式相比，组播的优势主要在于：

- 提高效率：降低网络流量，减轻了服务器和 CPU 负荷。
- 优化性能：减少冗余量，保障了信息的安全性。
- 分布式应用：解决了点到多点数据传输问题。

组播技术主要应用在以下几个方面：

- 多媒体、流媒体的应用，如：网络电视、网络电台、实时视/音频会议。
- 培训、联合作业场合的通信，如：远程教育、远程医疗。
- 数据仓库、金融应用（股票）。
- 其它任何“点到多点”的应用。

组播中的基本概念

介绍组播中的基本概念：

- 组播组

组播组是指使用一个 IP 组播地址标识的接收者集合。任何用户主机（或其他接收设备）加入一个组播组，就成为了该组成员，可以识别并接收以该 IP 组播地址为目的地址的组播数据。

- 组播组成员

所有加入某组播组的主机便成为该组播组的成员。组播组中的成员是动态的，主机可以在任何时刻加入或离开组播组。组成员可能广泛分布在网络中的任何地方。

- 组播源

以组播组地址为目的地址，发送 IP 报文的信源称为组播源。一个组播源可以同时向多个组播组发送数据，多个组播源可以同时向一个组播组发送数据。

- 组播路由器

在组播传输中提供三层组播功能的路由器。组播路由器能够实现组播路由，指导组播报文转发，也能够在与用户连接的末梢网段上提供组播组成员管理功能。

- 路由器接口

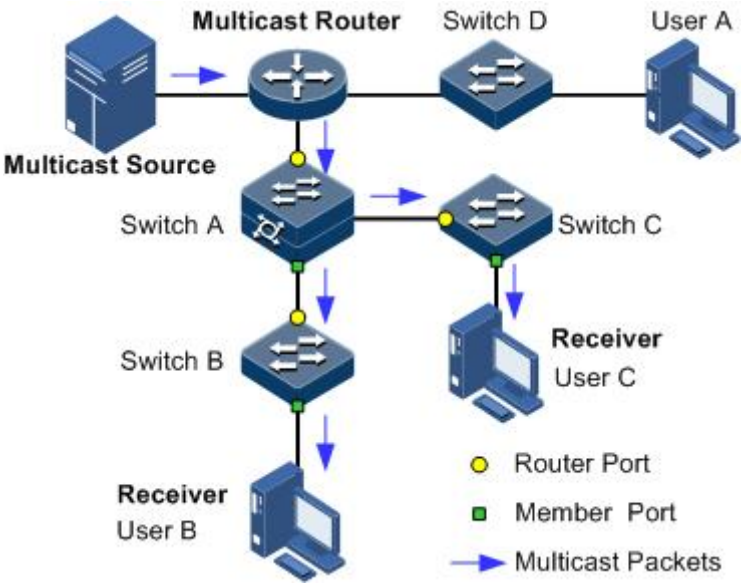
是指组播路由器和主机之间的设备上，朝向组播路由器的接口，设备从该接口接收组播报文。

- 成员接口

也称接收接口，是指组播路由器和主机之间的设备上，朝向主机的接口，设备从该接口发出组播报文。

组播基本概念中路由器接口，成员接口位置标示如下图所示。

图 6-2 组播基本概念在网络中相应位置标示示意图



组播地址

为了让组播源和组播组成员进行通信，需要提供网络层组播地址和链路层组播地址，即 IP 组播地址和组播 MAC 地址。需要注意的是组播地址都只能作为目的地址，而不能作为源地址来使用。

- IP 组播地址

IANA（Internet Assigned Numbers Authority，互联网编号分配委员会）将 D 类地址空间分配给 IPv4 组播使用，IPv4 组播地址范围 224.0.0.0～239.255.255.255。

- 组播 MAC 地址

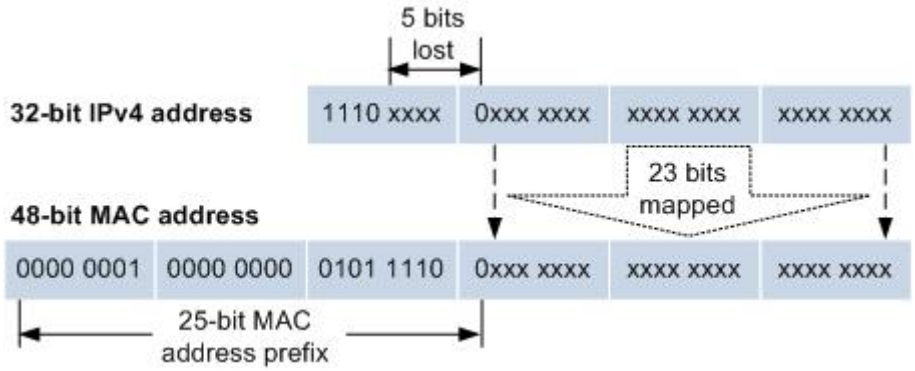
以太网传输单播 IP 报文时，目的 MAC 地址使用的是接收者的 MAC 地址。但是在传输组播数据包时，其目的地不再是一个具体的接收者，而是一个成员不确定的组，所以要使用组播 MAC 地址。

组播 MAC 地址用于在链路层上标识属于同一组播组的接收者。

IANA 规定，组播 MAC 地址的高 24 位为 0x01005E，第 25 位为 0，低 23 位为 IPv4 组播地址的低 23 位。

IP 组播地址和 MAC 地址以一种映射关系相关联，映射关系如下图所示。

图 6-3 IPv4 组播地址和组播 MAC 地址的映射关系



由于 IP 组播地址的前 4 位是 1110，代表组播标识，而后 28 位中只有 23 位被映射到 MAC 地址，这样 IP 地址中就有 5 位信息丢失，可能导致 32 个 IP 组播地址映射到同一 MAC 地址上。因此在二层处理过程中，设备可能要接收一些本 IPv4 组播组以外的组播数据，而这些多余的组播数据就需要设备的上层进行过滤了。

组播协议基础

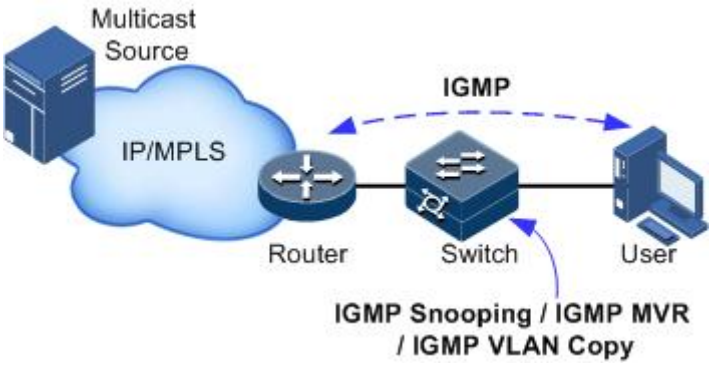
实现一套完整的组播服务，需要在网络各个位置部署多种组播协议相互配合，共同运作。

通常，把工作在网络层的 IP 组播称为“三层组播”，相应的组播协议称为“三层组播协议”，包括 IGMP（Internet Group Management Protocol，因特网组管理协议）等；把工作在数据链路层的 IP 组播称为“二层组播”，相应的组播协议称为“二层组播协

议”，包括 IGMP Snooping（Internet Group Management Protocol Snooping，因特网组管理协议监听）等。

IGMP 和二层组播特性运行位置如下图所示。

图 6-4 IGMP 和二层组播特性运行位置示意图



IGMP 是 TCP/IP 协议族中负责 IPv4 组播成员管理的协议。IGMP 运行在组播路由器和主机之间，该协议定义了主机与组播路由器之间建立、维护组播组成员关系的机制。IGMP 不包括组播路由器之间的组成员关系信息的传播与维护，这部分工作由组播路由协议完成。

IGMP 通过在主机和组播路由器之间交互 IGMP 报文实现组成员管理功能。IGMP 报文封装在 IP 报文中，IGMP 报文类型有 Query 查询报文、Report 报告报文和 Leave 离开报文。IGMP 基本功能：

- 主机发送 Report 报文加入组播组，发送 Leave 报文离开组播组，自主决定接收哪些组播组的报文。
- 组播路由器周期发送 Query 报文，并接收主机反馈的 Report 报文和 Leave 报文，了解连接的网段上有哪些组播组成员。如果有组播组成员，应将组播数据转发到这个网段；如果没有组播组成员则不转发。

到目前为止，IGMP 有三个版本：IGMPv1 版本、IGMPv2 版本和 IGMPv3 版本，新版本完全兼容旧版本。目前应用最广泛的是 IGMPv2，其中 Leave 报文 IGMPv1 版本不支持。

二层组播运行在主机和组播路由器之间的二层设备上。

二层组播通过监听和分析主机和组播路由器之间交互的 IGMP 报文来管理和控制组播组，实现组播数据在二层的转发，抑制组播数据在二层网络中的扩散。

二层组播基础

二层组播基础功能包含：

- 指定组播路由器接口；
- 使能立即离开功能；
- 设置组播转发表项和路由器接口的老化时间；

与二层组播基础功能有关的概念说明。

- 组播路由器接口

在运行组播功能的二层交换机上，可以动态学习（需要在组播路由器上开启组播路由协议，通过组播 Query 报文学习）到路由器接口，也可以手工设置，以使下游的组播报告、离开等报文可以转发到该路由器接口。

动态学习到的路由器接口有老化时间，手工配置的路由器接口不会老化。

- 老化时间

设置的老化时间同时作用于组播转发表项的老化时间和路由器接口的老化时间。

在运行组播功能的二层交换机上，每个动态学习到的路由器接口会启动一个定时器。到达老化时间时未收到 Query 报文，该路由器接口将被删除；收到 Query 报文时，更新路由器接口的超时时间。

每条组播转发表项会启动一个定时器，也就是组播成员的老化时间，到达老化时间时未收到 Report 报文，该组播成员将会被删除；收到 Report 报文时，更新组播转发表项的超时时间。

- 立即离开功能

在运行组播功能的二层交换机上，当用户发送 Leave 报文时，并不立即删除对应的组播转发表项，而是等待表项老化时才删除。当下游用户数量较多，并且加入离开比较频繁时，可以打开此功能，这样对应的组播转发表项被快速删除。

6.2 IGMP Snooping

6.2.1 简介

IGMP Snooping 是运行在二层设备上的组播约束机制，用于管理和控制组播组，实现二层组播。

IGMP Snooping 允许交换机监听主机和组播路由器之间的 IGMP 会话。当交换机监听到主机发往某个组的 IGMP Report，交换机将主机所在的接口加入到这个组的转发列表中，同样，当转发表项达到老化时间时，就将主机所在的接口从转发表中删除。

IGMP Snooping 利用二层组播转发表进行组播数据转发，当交换机收到组播数据时，会直接根据组播转发表项的相应的接收接口进行转发，并不向所有接口洪泛，因此有效地节省了交换机的带宽。

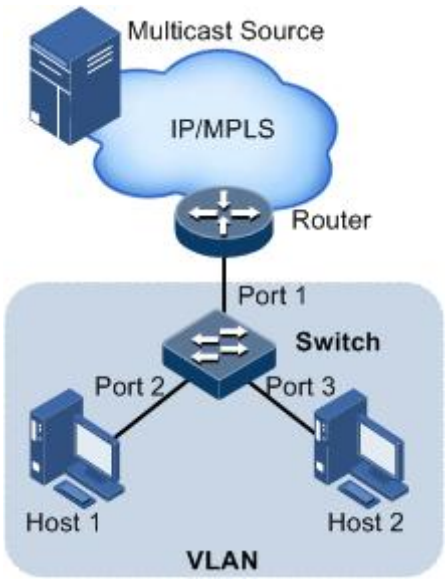
IGMP Snooping 建立二层组播转发表，可以通过 IGMP Snooping 动态学习，也可以进行手工配置。

6.2.2 配置准备

场景

如下图所示，多个主机接收组播源的数据，多个主机属于同一个 VLAN。可以在组播路由器和主机相连的交换机上运行 IGMP Snooping，通过监听组播路由器和主机之间的 IGMP 报文，建立和维护组播转发表，实现二层组播。

图 6-5 IGMP Snooping 应用场景



前提

在配置 IGMP Snooping 之前，需完成以下任务：

- 创建 VLAN，并将相应接口加入 VLAN。

6.2.3 IGMP Snooping 的缺省配置

设备上 IGMP Snooping 的缺省配置如下。

功能	缺省值
全局 IGMP Snooping 状态	禁用
VLAN 的 IGMP Snooping 状态	禁用
接口的 IGMP Snooping 状态	禁用
IGMP Snooping 协议版本	v2
通用查询间隔	60s
健壮系数	2

功能	缺省值
最大响应时间	10s
路由器端口老化时间	180s
接口的最大组播组限制	无限制

6.2.4 配置 IGMP Snooping 基本功能

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**igmp-snooping start**	全局使能 IGMP Snooping。
3	JX(config)#**igmp-snooping router-aging-time** *aging-time*	（可选）配置路由器端口老化时间
4	JX(config)#**vlan** *vlan-id*	进入 VLAN 配置视图
5	JX(config-vlan-10)#**igmp-snooping enable**	VLAN 使能 IGMP Snooping
6	JX(config-vlan-10)#**igmp-snooping version { v1 | v2 | V3 }**	（可选）配置协议版本
7	JX(config-vlan-10)#**igmp-snooping forwarding-mode { ip | mac }**	（可选）配置组播表项转发模式 基于组播 ip 或者基于组播 mac
8	JX(config-vlan-10)#**igmp-snooping require-router-alert enable**	（可选）配置 Router-Alert 选项检查
9	JX(config)#**interface** *interface-type interface-number*	进入物理接口配置模式。
10	JX(config-ge-1/0/1)#**igmp-snooping enable**	接口使能 IGMP Snooping

6.2.5 配置 IGMP Snooping 查询器功能

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**igmp-snooping query-interval** *interval*	（可选）配置通用查询间隔。

3	JX(config)#**igmp-snooping robust-count** *count*	（可选）配置健壮系数
4	JX(config)#**vlan** *vlan-id*	进入 VLAN 配置视图
5	JX(config-vlan-10)#**igmp-snooping querier enable**	配置查询器使能 使能了查询器才会发送通用查询和特定组查询
6	JX(config-vlan-10)#**igmp-snooping send-query source-address** *ip-address*	（可选）配置设备发送查询报文的源 IP 地址
7	JX(config-vlan-10)#**igmp-snooping max-response-time** *response-time*	（可选）配置查询报文最大响应时间
8	JX(config-vlan-10)#**igmp-snooping lastmember-query-interval** *interval*	（可选）配置特定组查询报文发送间隔
9	JX(config-vlan-10)#**igmp-snooping lastmember-query-number** *number*	（可选）配置特定组查询报文发送个数

6.2.6 配置 IGMP Snooping 报文抑制功能

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
4	JX(config)#**vlan** *vlan-id*	进入 VLAN 配置视图
8	JX(config-vlan-10)#**igmp-snooping report-suppres enable**	配置 Report 和 Leave 报文抑制

6.2.7 配置 IGMP Snooping 组播复制功能

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**vlan** *vlan-id*	进入 VLAN 配置视图
3	JX(config-vlan-10)#**igmp-snooping forwarding-mode ip**	配置转发模式为 IP 模式
4	JX(config-vlan-10)#**igmp-snooping multicast-duplicate enable**	配置组播复制使能
5	JX(config-vlan-10)#**igmp-snooping multicast-user-vlan** *VLANLIST*	配置组播复制关联的用户 VLAN

6.2.8 配置 IGMP Snooping 静态组播成员功能

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**interface** *interface-type interface-number*	进入物理接口配置模式。
3	JX(config-ge-1/0/1)#**igmp-snooping static-group group-address** g*roup-address* [**source-address** *source-address*] **vlan** *vlan-id* [**user-vlan** *vlan-list*]	配置接口为静态组播成员

说明

如果需要配置 source-address 的静态组播成员，需要协议版本配置为 v3。

如果需要配置 user-vlan，需要使能组播复制

6.2.9 配置 IGMP Snooping Proxy 功能

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**vlan** *vlan-id*	进入 VLAN 配置视图
3	JX(config-vlan-10)#**igmp-snooping workmode igmp-proxy**	配置工作模式为 Proxy 模式
4	JX(config-vlan-10)#**igmp-snooping querier enable**	配置查询器使能 Proxy 模式必须打开查询器，否则组播表项可能会老化
5	JX(config-vlan-10)#**igmp-snooping proxy-ip** *ip-address*	（可选）配置 Proxy 模式的代理 IP 地址

6.2.10 配置 IGMP Snooping 接口组播组个数限制

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**interface** *interface-type interface-number*	进入物理接口配置模式。

3	JX(config-ge-1/0/1)#**igmp-snooping group-limit** *number* [**action** { **delay** | **replace** }]	配置 Report 和 Leave 报文抑制

6.2.11 配置 IGMP Snooping 组播策略

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**acl-ipv4** *list-id*	创建并配置 ACL 策略模板 策略只保护对目的 IP 地址的 permit 和 deny
3	JX(config)#**interface** *interface-type interface-number*	进入物理接口配置模式。
4	JX(config-ge-1/0/1)#**igmp-snooping group-policy acl-ipv4** *list-id*	接口组播策略绑定 ACL 模板
5	JX(config)#**vlan** *vlan-id*	进入 VLAN 配置视图
6	JX(config-vlan-10)#**igmp-snooping group-policy acl-ipv4** *list-id*	VLAN 组播策略绑定 ACL 模板

说明

VLAN 和接口同时配置组播策略时，如果 VLAN 的策略为 deny 则不再检查接口的策略。

6.2.12 配置 IGMP Snooping SSM Mapping 功能

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**acl-ipv4** *list-id*	创建并配置 ACL 策略模板 策略只保护对目的 IP 地址的 permit 和 deny
3	JX(config)#**vlan** *vlan-id*	进入 VLAN 配置视图
4	JX(config-vlan-10)#**igmp-snooping forwarding-mode ip**	配置转发模式为 IP 模式
5	JX(config-vlan-10)#**igmp-snooping version v3**	配置协议版本为 v3
6	JX(config-vlan-10)#**igmp-snooping ssm-mapping enable**	使能 SSM Mapping 功能

7	JX(config-vlan-10)#**igmp-snooping ssm-mapping acl-ipv4** *list-id* **source-addres** *ip-address*	配置组地址与源地址的映射 ACL 动作为 permit 的组地址进行映射

6.2.13 检查配置

配置完成后，请在设备上进行以下命令检查配置结果。

序号	检查项	说明
1	JX#**show igmp-snooping config**	查看 IGMP Snooping 配置信息。
2	JX#**show igmp-snooping interface**	查看 IGMP Snooping 接口信息
3	JX#**show igmp-snooping vlan**	查看 IGMP Snooping 组播 VLAN 信息
4	JX#**show igmp-snooping forwarding-table**	查看 IGMP Snooping 组播转发表项

6.2.14 维护

用户可以通过以下命令维护 IGMP Snooping。

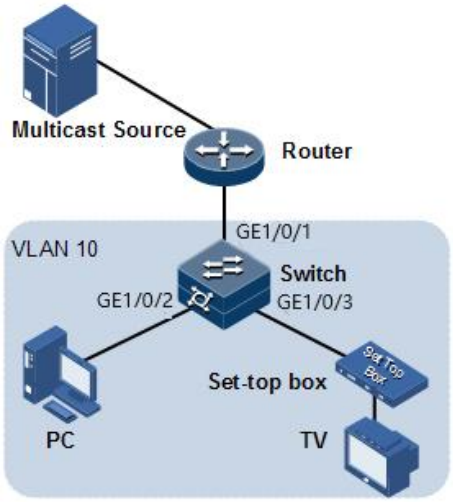
命令	描述
JX(config)#**reset igmp-snooping forwarding-table**	清除动态学习的组播转发表项。
JX(config)#**reset igmp-snooping statistics**	清除 IGMP Snooping 协议报文统计。

6.2.15 配置 IGMP Snooping 基本功能应用示例

组网需求

如下图所示，交换机接口 Gigaethernet 1/0/1 连接组播路由器，接口 Gigaethernet 1/0/2，Gigaethernet 1/0/3 连接用户，用户在同一个 VLAN 10 中，配置 IGMP Snooping 功能

图 6-6 IGMP Snooping 基本功能组网应用示意图



配置步骤

步骤 1 创建 VLAN，并将接口加入 VLAN。

```
JX#config
JX(config)#vlan 10
JX(config-vlan-10)#quit
JX(config)#interface ge 1/0/1
JX(config-ge-1/0/1)#port link-type trunk
JX(config-ge-1/0/1)#port trunk allow-pass vlan 10
JX(config-ge-1/0/1)#quit
JX(config)#interface ge 1/0/2
JX(config-ge-1/0/2)#port link-type access
JX(config-ge-1/0/2)#port default vlan 10
JX(config-ge-1/0/2)#quit
JX(config)#interface ge 1/0/3
JX(config-ge-1/0/3)#port link-type access
JX(config-ge-1/0/3)#port default vlan 10
JX(config-ge-1/0/3)#quit
```

步骤 2 配置使能 IGMP Snooping。

```
JX(config)#igmp-snooping start
JX(config)#vlan 10
JX(config-vlan-10)#igmp-snooping enable
JX(config-vlan-10)#quit
JX(config)#interface ge 1/0/1
JX(config-ge-1/0/1)#igmp-snooping enable
JX(config-ge-1/0/1)#quit
JX(config)#interface ge 1/0/2
JX(config-ge-1/0/2)#igmp-snooping enable
JX(config-ge-1/0/2)#quit
JX(config)#interface ge 1/0/3
JX(config-ge-1/0/3)#igmp-snooping enable
JX(config-ge-1/0/3)#quit
```

检查结果

查看 IGMP Snooping 配置信息是否正确。

```
JX#show igmp-snooping config
!
igmp-snooping start
!
vlan 10
 igmp-snooping enable
!
interface ge 1/0/1
 igmp-snooping enable
!
interface ge 1/0/2
 igmp-snooping enable
!
interface ge 1/0/3
 igmp-snooping enable
```

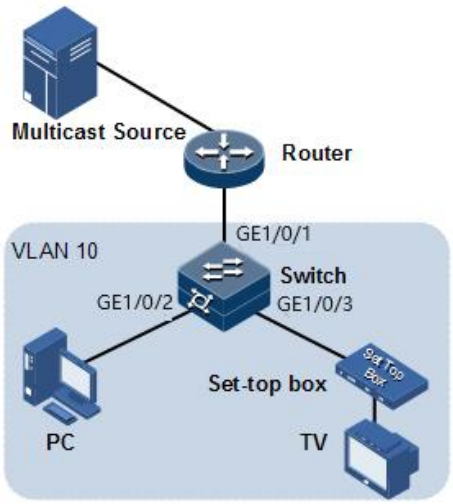
6.2.16 配置 IGMP Snooping 静态成员应用示例

组网需求

如下图所示，交换机接口 Gigaethernet 1/0/1 连接组播路由器，接口 Gigaethernet 1/0/2，Gigaethernet 1/0/3 连接用户，用户在同一个 VLAN 10 中，配置 IGMP Snooping 功能

其中 Gigaethernet 1/0/2 下用户希望长期稳定接收 225.1.1.1 ~ 225.1.1.3 的组播数据。

图 6-7 IGMP Snooping 静态成员功能组网应用示意图



配置步骤

步骤 1 创建 VLAN，并将接口加入 VLAN。

```
JX#config
JX(config)#vlan 10
JX(config-vlan-10)#quit
JX(config)#interface ge 1/0/1
JX(config-ge-1/0/1)#port link-type trunk
JX(config-ge-1/0/1)#port trunk allow-pass vlan 10
JX(config-ge-1/0/1)#quit
JX(config)#interface ge 1/0/2
JX(config-ge-1/0/2)#port link-type access
JX(config-ge-1/0/2)#port default vlan 10
JX(config-ge-1/0/2)#quit
JX(config)#interface ge 1/0/3
JX(config-ge-1/0/3)#port link-type access
JX(config-ge-1/0/3)#port default vlan 10
JX(config-ge-1/0/3)#quit
```

步骤 2 配置使能 IGMP Snooping。

```
JX(config)#igmp-snooping start
JX(config)#vlan 10
JX(config-vlan-10)#igmp-snooping enable
JX(config-vlan-10)#quit
JX(config)#interface ge 1/0/1
JX(config-ge-1/0/1)#igmp-snooping enable
JX(config-ge-1/0/1)#quit
JX(config)#interface ge 1/0/2
JX(config-ge-1/0/2)#igmp-snooping enable
JX(config-ge-1/0/2)#quit
JX(config)#interface ge 1/0/3
JX(config-ge-1/0/3)#igmp-snooping enable
JX(config-ge-1/0/3)#quit
```

步骤 3 配置 IGMP Snooping 静态组播成员

```
    JX(config)#interface ge 1/0/2
    JX(config-ge-1/0/1)#igmp-snooping static-group group-address
225.1.1.1 vlan 10
    JX(config-ge-1/0/1)#igmp-snooping static-group group-address
225.1.1.2 vlan 10
    JX(config-ge-1/0/1)#igmp-snooping static-group group-address
225.1.1.3 vlan 10
```

检查结果

查看 IGMP Snooping 配置信息是否正确。

```
JX#show igmp-snooping config
!
igmp-snooping start
!
vlan 10
 igmp-snooping enable
!
interface ge 1/0/1
 igmp-snooping enable
!
```

```
interface ge 1/0/2
 igmp-snooping enable
 igmp-snooping static-group group-address 225.1.1.1 vlan 10
 igmp-snooping static-group group-address 225.1.1.2 vlan 10
 igmp-snooping static-group group-address 225.1.1.3 vlan 10
!
interface ge 1/0/3
 igmp-snooping enable
```

查看 IGMP Snooping 静态转发表是否正确

JX#show igmp-snooping forwarding-table

S:Static, D:dynamic, E:Exclude, I:Include

Vlan	(Source,Group)	Port	OutVlan	Flag	Expires
10	(*,225.1.1.1)	ge-1/0/2	10	S/E	--
10	(*,225.1.1.2)	ge-1/0/2	10	S/E	--
10	(*,225.1.1.3)	ge-1/0/2	10	S/E	--

Total Group Number: 3

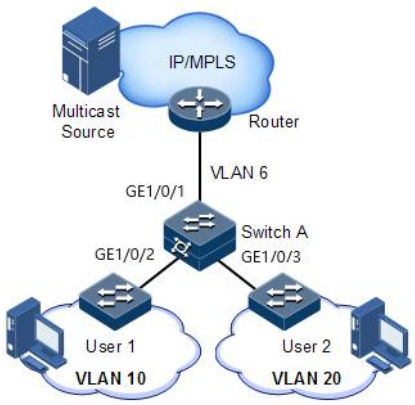
6.2.17 配置 IGMP Snooping 组播复制应用示例

组网需求

如下图所示，交换机接口 Gigaethernet 1/0/1 连接组播路由器，接口 Gigaethernet 1/0/2，Gigaethernet 1/0/3 连接用户，组播 VLAN 和用户 VLAN 不同，配置 IGMP Snooping 组播复制功能，

其中 GE1/0/1 加入组播 VLAN 6， 用户 1 加入用户 VLAN 10，用户 2 加入 VLAN 20。

图 6-8 IGMP Snooping 组播复制功能组网应用示意图



配置步骤

步骤 1 创建 VLAN，并将接口加入 VLAN。

```
JX#config
```

```
JX(config)#vlan 6,10,20
JX(config)#interface ge 1/0/1
JX(config-ge-1/0/1)#port link-type trunk
JX(config-ge-1/0/1)#port trunk allow-pass vlan 6
JX(config-ge-1/0/1)#quit
JX(config)#interface ge 1/0/2
JX(config-ge-1/0/2)#port link-type access
JX(config-ge-1/0/2)#port default vlan 10
JX(config-ge-1/0/2)#quit
JX(config)#interface ge 1/0/3
JX(config-ge-1/0/3)#port link-type access
JX(config-ge-1/0/3)#port default vlan 20
JX(config-ge-1/0/3)#quit
```

步骤 2 配置使能 IGMP Snooping 功能

```
JX(config)#igmp-snooping start
JX(config)#vlan 6
JX(config-vlan-6)#igmp-snooping enable
JX(config-vlan-6)#quit
JX(config)#interface ge 1/0/1
JX(config-ge-1/0/1)#igmp-snooping enable
JX(config-ge-1/0/1)#quit
JX(config)#interface ge 1/0/2
JX(config-ge-1/0/2)#igmp-snooping enable
JX(config-ge-1/0/2)#quit
JX(config)#interface ge 1/0/3
JX(config-ge-1/0/3)#igmp-snooping enable
JX(config-ge-1/0/3)#quit
```

步骤 3 配置使能 IGMP Snooping 开启组播复制功能

```
JX(config)#vlan 6
JX(config-vlan-6)#igmp-snooping forwarding-mode ip
JX(config-vlan-6)#igmp-snooping multicast-duplicate enable
JX(config-vlan-6)#igmp-snooping multicast-user-vlan 10,20
JX(config-vlan-6)#quit
```

检查结果

查看 IGMP Snooping 配置信息是否正确。

```
JX#show igmp-snooping config
!
igmp-snooping start
!
vlan 6
 igmp-snooping enable
 igmp-snooping forwarding-mode ip
 igmp-snooping multicast-duplicate enable
 igmp-snooping multicast-user-vlan 10,20
!
interface ge 1/0/1
 igmp-snooping enable
!
interface ge 1/0/2
```

```
 igmp-snooping enable
!
interface ge 1/0/3
 igmp-snooping enable
```

6.2.18 配置 IGMP Snooping Proxy 应用示例

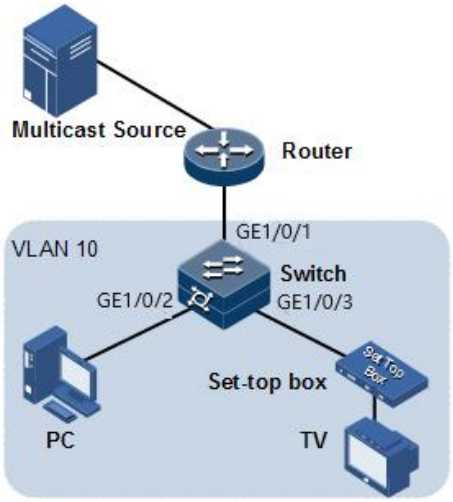
组网需求

如下图所示，交换机接口 Gigaethernet 1/0/1 连接组播路由器，接口 Gigaethernet 1/0/2，Gigaethernet 1/0/3 连接用户，用户在同一个 VLAN 10 中，配置 IGMP Snooping 功能

在交换机上开启 IGMP Proxy 功能，减少主机和组播路由器之间的通信，同时又不影响组播功能实现。

图中 PC 和机顶盒加入同一个组播组时，交换机收到两份 IGMP Report 报文，只向组播路由器发送一份 IGMP Report 报文。组播路由器发送的 IGMP Query 报文不再向下游转发，而是由交换机定时发送 IGMP Query 报文。

图 6-9 IGMP Snooping Proxy 功能组网应用示意图



配置步骤

步骤 1 创建 VLAN，并将接口加入 VLAN。

```
JX#config
JX(config)#vlan 10
JX(config-vlan-10)#quit
JX(config)#interface ge 1/0/1
JX(config-ge-1/0/1)#port link-type trunk
JX(config-ge-1/0/1)#port trunk allow-pass vlan 10
JX(config-ge-1/0/1)#quit
JX(config)#interface ge 1/0/2
JX(config-ge-1/0/2)#port link-type access
JX(config-ge-1/0/2)#port default vlan 10
```

```
JX(config-ge-1/0/2)#quit
JX(config)#interface ge 1/0/3
JX(config-ge-1/0/3)#port link-type access
JX(config-ge-1/0/3)#port default vlan 10
JX(config-ge-1/0/3)#quit
```

步骤 2 配置使能 IGMP Snooping 功能

```
JX(config)#igmp-snooping start
JX(config)#vlan 10
JX(config-vlan-10)#igmp-snooping enable
JX(config-vlan-10)#quit
JX(config)#interface ge 1/0/1
JX(config-ge-1/0/1)#igmp-snooping enable
JX(config-ge-1/0/1)#quit
JX(config)#interface ge 1/0/2
JX(config-ge-1/0/2)#igmp-snooping enable
JX(config-ge-1/0/2)#quit
JX(config)#interface ge 1/0/3
JX(config-ge-1/0/3)#igmp-snooping enable
JX(config-ge-1/0/3)#quit
```

步骤 3 配置使能 IGMP Snooping 开启 IGMP Proxy 功能

```
JX(config)#vlan 10
JX(config-vlan-10)#igmp-snooping workmode igmp-proxy
JX(config-vlan-10)#igmp-snooping querier enable
JX(config-vlan-10)#quit
```

检查结果

查看 IGMP Snooping 配置信息是否正确。

```
JX#show igmp-snooping config
!
igmp-snooping start
!
vlan 10
 igmp-snooping enable
 igmp-snooping workmode igmp-proxy
 igmp-snooping querier enable
!
interface ge 1/0/1
 igmp-snooping enable
!
interface ge 1/0/2
 igmp-snooping enable
!
interface ge 1/0/3
 igmp-snooping enable
```

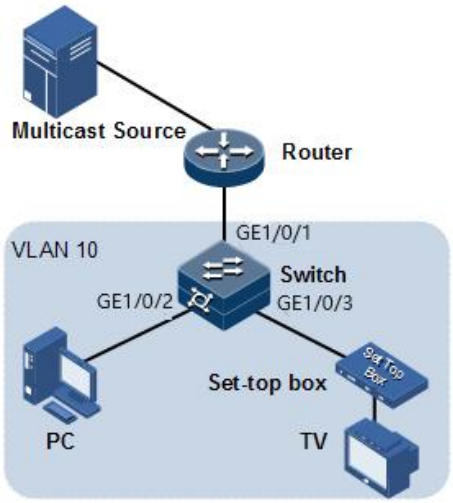
6.2.19 配置 IGMP Snooping 策略应用示例

组网需求

如下图所示，交换机接口 Gigaethernet 1/0/1 连接组播路由器，接口 Gigaethernet 1/0/2，Gigaethernet 1/0/3 连接用户，用户在同一个 VLAN 10 中，配置 IGMP Snooping 功能

在交换机上开启组播策略功能，接口 Gigaethernet 1/0/2 的用户允许加入 225.1.1.1～225.1.1.3，接口 Gigaethernet 1/0/3 的用户允许加入 225.1.1.4～225.1.1.6。

图 6-10 IGMP Snooping 组播策略功能组网应用示意图



配置步骤

步骤 1 创建 VLAN，并将接口加入 VLAN。

```
JX#config
JX(config)#vlan 10
JX(config-vlan-10)#quit
JX(config)#interface ge 1/0/1
JX(config-ge-1/0/1)#port link-type trunk
JX(config-ge-1/0/1)#port trunk allow-pass vlan 10
JX(config-ge-1/0/1)#quit
JX(config)#interface ge 1/0/2
JX(config-ge-1/0/2)#port link-type access
JX(config-ge-1/0/2)#port default vlan 10
JX(config-ge-1/0/2)#quit
JX(config)#interface ge 1/0/3
JX(config-ge-1/0/3)#port link-type access
JX(config-ge-1/0/3)#port default vlan 10
JX(config-ge-1/0/3)#quit
```

步骤 2 配置使能 IGMP Snooping 功能

```
JX(config)#igmp-snooping start
JX(config)#vlan 10
JX(config-vlan-10)#igmp-snooping enable
```

```
JX(config-vlan-10)#quit
JX(config)#interface ge 1/0/1
JX(config-ge-1/0/1)#igmp-snooping enable
JX(config-ge-1/0/1)#quit
JX(config)#interface ge 1/0/2
JX(config-ge-1/0/2)#igmp-snooping enable
JX(config-ge-1/0/2)#quit
JX(config)#interface ge 1/0/3
JX(config-ge-1/0/3)#igmp-snooping enable
JX(config-ge-1/0/3)#quit
```

步骤 3 配置 GE1/0/2 用户的 ACL 模板

```
JX(config)#acl-ipv4 1001
JX(configure-acl-ipv4-1001)#rule 1 ip src-ip any dst-ip 225.1.1.1/32
JX(configure-acl-ipv4-1001)#rule 1 action permit
JX(configure-acl-ipv4-1001)#rule 2 ip src-ip any dst-ip 225.1.1.2/32
JX(configure-acl-ipv4-1001)#rule 2 action permit
JX(configure-acl-ipv4-1001)#rule 3 ip src-ip any dst-ip 225.1.1.3/32
JX(configure-acl-ipv4-1001)#rule 3 action permit
JX(configure-acl-ipv4-1001)#quit
```

步骤 4 配置 GE1/0/3 用户的 ACL 模板

```
JX(config)#acl-ipv4 1002
JX(configure-acl-ipv4-1002)#rule 1 ip src-ip any dst-ip 225.1.1.4/32
JX(configure-acl-ipv4-1002)#rule 1 action permit
JX(configure-acl-ipv4-1002)#rule 2 ip src-ip any dst-ip 225.1.1.5/32
JX(configure-acl-ipv4-1002)#rule 2 action permit
JX(configure-acl-ipv4-1002)#rule 3 ip src-ip any dst-ip 225.1.1.6/32
JX(configure-acl-ipv4-1002)#rule 3 action permit
JX(configure-acl-ipv4-1002)#quit
```

步骤 5 配置接口绑定组播策略 ACL 模板

```
JX(config)#interface ge 1/0/2
JX(config-ge-1/0/2)#igmp-snooping group-policy acl-ipv4 1001
JX(config-ge-1/0/2)#quit
JX(config)#interface ge 1/0/3
JX(config-ge-1/0/3)#igmp-snooping group-policy acl-ipv4 1002
JX(config-ge-1/0/3)#quit
```

检查结果

查看 IGMP Snooping 配置信息是否正确。

```
JX#show igmp-snooping config
!
igmp-snooping start
!
vlan 10
 igmp-snooping enable
!
interface ge 1/0/2
 igmp-snooping enable
 igmp-snooping group-policy acl-ipv4 1001 version 1-3
!
```

```
interface ge 1/0/3
 igmp-snooping enable
 igmp-snooping group-policy acl-ipv4 1002 version 1-3
```

查看 ACL 模板配置是否正确。

```
JX#show acl config
!
acl-ipv4 1001
 rule 1 ip src-ip any dst-ip 225.1.1.1/32
 rule 1 action permit
 rule 2 ip src-ip any dst-ip 225.1.1.2/32
 rule 2 action permit
 rule 3 ip src-ip any dst-ip 225.1.1.3/32
 rule 3 action permit
acl-ipv4 1002
 rule 1 ip src-ip any dst-ip 225.1.1.4/32
 rule 1 action permit
 rule 2 ip src-ip any dst-ip 225.1.1.5/32
 rule 2 action permit
 rule 3 ip src-ip any dst-ip 225.1.1.6/32
 rule 3 action permit
```

6.3 MLD Snooping

6.3.1 简介

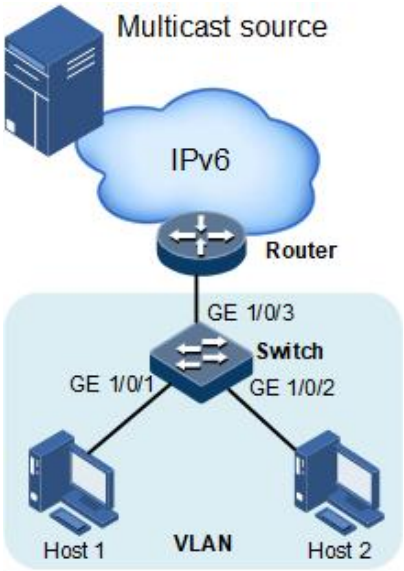
MLD Snooping 是一种 IPv6 二层组播协议，该协议侦听三层组播设备和用户主机之间发送的组播协议报文，维护组播报文出端口的信息，进而管理和控制组播数据报文的转发。

6.3.2 配置准备

场景

多个主机接收组播源的数据，多个主机属于同一个 VLAN。可以在组播路由器和主机相连的交换机上运行 MLD Snooping，通过监听组播路由器和主机之间的 MLD 报文，建立和维护组播转发表，实现二层组播。

图 6-11 MLD Snooping 应用场景



前提

在配置 MLD Snooping 之前，需完成以下任务：

- 创建 VLAN，并将相应接口加入 VLAN。

6.3.3 MLD Snooping 的缺省配置

功能	缺省值
全局 MLD Snooping 状态	禁用
VLAN 的 MLD Snooping 状态	禁用
接口的 MLD Snooping 状态	禁用
MLD Snooping 协议版本	v1
通用查询间隔	60s
健壮系数	2
最大响应时间	10s
路由器端口老化时间	180s
接口的最大组播组限制	无限制

6.3.4 配置 MLD Snooping 基本功能

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**mld-snooping start**	全局使能 MLD Snooping。
3	JX(config)#**mld-snooping router-aging-time** *aging-time*	（可选）配置路由器端口老化时间
4	JX(config)#**vlan** *vlan-id*	进入 VLAN 配置视图
5	JX(config-vlan-10)#**mld-snooping enable**	VLAN 使能 MLD Snooping
6	JX(config-vlan-10)#**mld-snooping version** { **v1** | **v2** }	（可选）配置协议版本
7	JX(config-vlan-10)#**mld-snooping forwarding-mode** { **ip** | **mac** }	（可选）配置组播表项转发模式 基于组播 ip 或者基于组播 mac
8	JX(config-vlan-10)#**mld-snooping require-router-alert enable**	（可选）配置 Router-Alert 选项检查
9	JX(config)#**interface** *interface-type interface-number*	进入物理接口配置模式。
10	JX(config-ge-1/0/1)#**mld-snooping enable**	接口使能 MLD Snooping

6.3.5 配置 MLD Snooping 查询器功能

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**mld-snooping query-interval** *interval*	（可选）配置通用查询间隔。
3	JX(config)#**mld-snooping robust-count** *count*	（可选）配置健壮系数
4	JX(config)#**vlan** *vlan-id*	进入 VLAN 配置视图
5	JX(config-vlan-10)#**mld-snooping querier enable**	配置查询器使能 使能了查询器才会发送通用查询和特定组查询
6	JX(config-vlan-10)#**mld-snooping send-query source-address** *ipv6-address*	（可选）配置设备发送查询报文的源 IP 地址
7	JX(config-vlan-10)#**mld-snooping max-response-time** *response-time*	（可选）配置查询报文最大响应时间
8	JX(config-vlan-10)#**mld-snooping lastmember-query-interval** *interval*	（可选）配置特定组查询报文发送间隔

9	JX(config-vlan-10)#**mld-snooping lastmember-query-number** *number*	（可选）配置特定组查询报文发送个数

6.3.6 配置 MLD Snooping 报文抑制功能

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
4	JX(config)#**vlan** *vlan-id*	进入 VLAN 配置视图
8	JX(config-vlan-10)#**mld-snooping report-suppres enable**	配置 Report 和 Leave 报文抑制

6.3.7 配置 MLD Snooping 组播复制功能

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**vlan** *vlan-id*	进入 VLAN 配置视图
3	JX(config-vlan-10)#**mld-snooping forwarding-mode ip**	配置转发模式为 IP 模式
4	JX(config-vlan-10)#**mld-snooping multicast-duplicate enable**	配置组播复制使能
5	JX(config-vlan-10)#**mld-snooping multicast-user-vlan** *VLANLIST*	配置组播复制关联的用户 VLAN

6.3.8 配置 MLD Snooping 静态组播成员功能

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**interface** *interface-type interface-number*	进入物理接口配置模式。
3	JX(config-ge-1/0/1)#**mld-snooping static-group group-address** *group-address* **vlan** *vlan-id* [**user-vlan** *vlan-list*]	配置接口为静态组播成员

说明

1. 如果需要配置 user-vlan，需要使能组播复制

6.3.9 配置 MLD Snooping Proxy 功能

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**vlan** *vlan-id*	进入 VLAN 配置视图
3	JX(config-vlan-10)#**mld-snooping workmode mld-proxy**	配置工作模式为 Proxy 模式
4	JX(config-vlan-10)#**mld-snooping querier enable**	配置查询器使能 Proxy 模式必须打开查询器，否则组播表项可能会老化
5	JX(config-vlan-10)#**mld-snooping proxy-ip** *ipv6-address*	（可选）配置 Proxy 模式的代理 IP 地址

6.3.10 配置 MLD Snooping 接口组播组个数限制

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**interface** *interface-type interface-number*	进入物理接口配置模式。
3	JX(config-ge-1/0/1)#**mld-snooping group-limit** *number* [**action** { **delay** | **replace** }]	配置 Report 和 Leave 报文抑制

6.3.11 配置 MLD Snooping 组播策略

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。

2	JX(config)#**acl-ipv6** *list-id*	创建并配置 ACL 策略模板 策略只保护对目的 IPv6 地址的 permit 和 deny
3	JX(config)#**interface** *interface-type interface-number*	进入物理接口配置模式。
4	JX(config-ge-1/0/1)#**mld-snooping group-policy acl-ipv6** *list-id*	接口组播策略绑定 ACL 模板
5	JX(config)#**vlan** *vlan-id*	进入 VLAN 配置视图
6	JX(config-vlan-10)#**mld-snooping group-policy acl-ipv6** *list-id*	VLAN 组播策略绑定 ACL 模板

说明

VLAN 和接口同时配置组播策略时，如果 VLAN 的策略为 deny 则不再检查接口的策略。

6.3.12 配置 MLD Snooping SSM Mapping 功能

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**acl-ipv6** *list-id*	创建并配置 ACL 策略模板 策略只保护对目的 IP 地址的 permit 和 deny
3	JX(config)#**vlan** *vlan-id*	进入 VLAN 配置视图
4	JX(config-vlan-10)#**mld-snooping forwarding-mode ip**	配置转发模式为 IP 模式
5	JX(config-vlan-10)#**mld-snooping version v2**	配置协议版本为 v2
6	JX(config-vlan-10)#**mld-snooping ssm-mapping enable**	使能 SSM Mapping 功能
7	JX(config-vlan-10)#**mld-snooping ssm-mapping acl-ipv6** *list-id* **source-addres** *ipv6-address*	配置组地址与源地址的映射 ACL 动作为 permit 的组地址进行映射

6.3.13 检查配置

配置完成后，请在设备上进行以下命令检查配置结果。

序号	检查项	说明
1	JX#**show mld-snooping config**	查看 MLD Snooping 配置信息。

序号	检查项	说明
2	JX#show mld-snooping interface	查看 MLD Snooping 接口信息
3	JX#show mld-snooping vlan	查看 MLD Snooping 组播 VLAN 信息
4	JX#show mld-snooping forwarding-table	查看 MLD Snooping 组播转发表项

6.3.14 维护

用户可以通过以下命令维护 MLD Snooping。

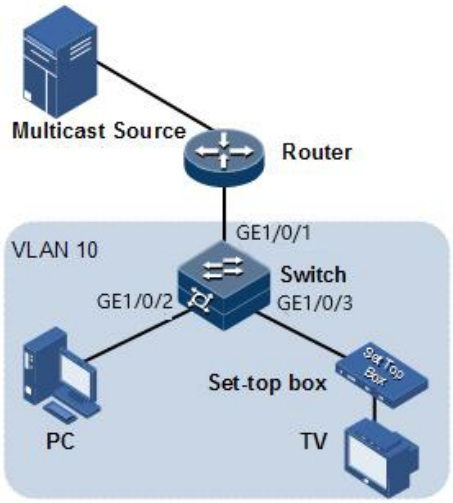
命令	描述
JX(config)#reset mld-snooping forwarding-table	清除动态学习的组播转发表项。

6.3.15 配置 MLD Snooping 基本功能应用示例

组网需求

如下图所示，交换机接口 Gigaethernet 1/0/1 连接组播路由器，接口 Gigaethernet 1/0/2，Gigaethernet 1/0/3 连接用户，用户在同一个 VLAN 10 中，配置 MLD Snooping 功能

图 6-12 MLD Snooping 基本功能组网应用示意图



配置步骤

步骤 1 创建 VLAN，并将接口加入 VLAN。

```
JX#config
JX(config)#vlan 10
JX(config-vlan-10)#quit
```

```
JX(config)#interface ge 1/0/1
JX(config-ge-1/0/1)#port link-type trunk
JX(config-ge-1/0/1)#port trunk allow-pass vlan 10
JX(config-ge-1/0/1)#quit
JX(config)#interface ge 1/0/2
JX(config-ge-1/0/2)#port link-type access
JX(config-ge-1/0/2)#port default vlan 10
JX(config-ge-1/0/2)#quit
JX(config)#interface ge 1/0/3
JX(config-ge-1/0/3)#port link-type access
JX(config-ge-1/0/3)#port default vlan 10
JX(config-ge-1/0/3)#quit
```

步骤 2 配置使能 MLD Snooping。

```
JX(config)#mld-snooping start
JX(config)#vlan 10
JX(config-vlan-10)#mld-snooping enable
JX(config-vlan-10)#quit
JX(config)#interface ge 1/0/1
JX(config-ge-1/0/1)#mld-snooping enable
JX(config-ge-1/0/1)#quit
JX(config)#interface ge 1/0/2
JX(config-ge-1/0/2)#mld-snooping enable
JX(config-ge-1/0/2)#quit
JX(config)#interface ge 1/0/3
JX(config-ge-1/0/3)#mld-snooping enable
JX(config-ge-1/0/3)#quit
```

检查结果

查看 MLD Snooping 配置信息是否正确。

```
JX#show mld-snooping config
!
mld-snooping start
!
vlan 10
 mld-snooping enable
!
interface ge 1/0/1
 mld-snooping enable
!
interface ge 1/0/2
 mld-snooping enable
!
interface ge 1/0/3
 mld-snooping enable
```

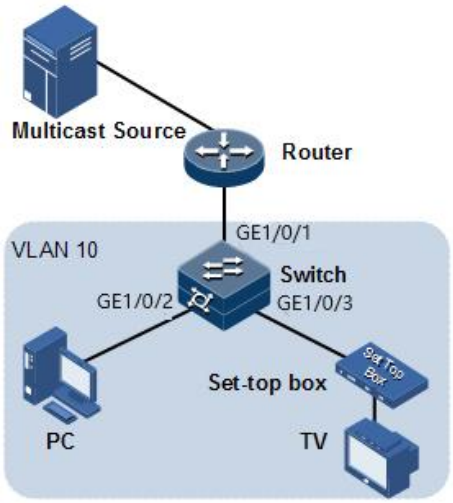
6.3.16 配置 MLD Snooping 静态成员应用示例

组网需求

如下图所示，交换机接口 Gigaethernet 1/0/1 连接组播路由器，接口 Gigaethernet 1/0/2，Gigaethernet 1/0/3 连接用户，用户在同一个 VLAN 10 中，配置 MLD Snooping 功能

其中 Gigaethernet 1/0/2 下用户希望长期稳定接收 ff1e::1 ~ ff1e::3 的组播数据。

图 6-13 MLD Snooping 静态成员功能组网应用示意图



配置步骤

步骤 1 创建 VLAN，并将接口加入 VLAN。

```
JX#config
JX(config)#vlan 10
JX(config-vlan-10)#quit
JX(config)#interface ge 1/0/1
JX(config-ge-1/0/1)#port link-type trunk
JX(config-ge-1/0/1)#port trunk allow-pass vlan 10
JX(config-ge-1/0/1)#quit
JX(config)#interface ge 1/0/2
JX(config-ge-1/0/2)#port link-type access
JX(config-ge-1/0/2)#port default vlan 10
JX(config-ge-1/0/2)#quit
JX(config)#interface ge 1/0/3
JX(config-ge-1/0/3)#port link-type access
JX(config-ge-1/0/3)#port default vlan 10
JX(config-ge-1/0/3)#quit
```

步骤 2 配置使能 MLD Snooping。

```
JX(config)#mld-snooping start
JX(config)#vlan 10
JX(config-vlan-10)#mld-snooping enable
JX(config-vlan-10)#quit
```

```
JX(config)#interface ge 1/0/1
JX(config-ge-1/0/1)#mld-snooping enable
JX(config-ge-1/0/1)#quit
JX(config)#interface ge 1/0/2
JX(config-ge-1/0/2)#mld-snooping enable
JX(config-ge-1/0/2)#quit
JX(config)#interface ge 1/0/3
JX(config-ge-1/0/3)#mld-snooping enable
JX(config-ge-1/0/3)#quit
```

步骤 3 配置 MLD Snooping 静态组播成员

```
JX(config)#interface ge 1/0/2
JX(config-ge-1/0/1)#mld-snooping static-group group-address ff1e::1 vlan
10
JX(config-ge-1/0/1)#mld-snooping static-group group-address ff1e::2 vlan
10
JX(config-ge-1/0/1)#mld-snooping static-group group-address ff1e::3 vlan
10
```

检查结果

查看 MLD Snooping 配置信息是否正确。

```
JX#show mld-snooping config
!
mld-snooping start
!
vlan 10
 mld-snooping enable
!
interface ge 1/0/1
 mld-snooping enable
 mld-snooping static-group group-address ff1e::1 vlan 10
 mld-snooping static-group group-address ff1e::2 vlan 10
 mld-snooping static-group group-address ff1e::3 vlan 10
!
interface ge 1/0/2
 mld-snooping enable
!
interface ge 1/0/3
 mld-snooping enable
```

查看 MLD Snooping 静态转发表是否正确

```
JX#show mld-snooping forwarding-table
 S:Static,  D:dynamic,   E:Exclude,   I:Include
 Vlan     (Source,Group)         Port         OutVlan       Flag   Expires
-------------------------------------------------------------------------
-------------------------------------
 10       (*,ff1e::1)            ge-1/0/1      10           S/E    --
 10       (*,ff1e::2)            ge-1/0/1      10           S/E    --
 10       (*,ff1e::3)            ge-1/0/1      10           S/E    --
-------------------------------------------------------------------------
-------------------------------------
```

```
Total Group Number: 3
```

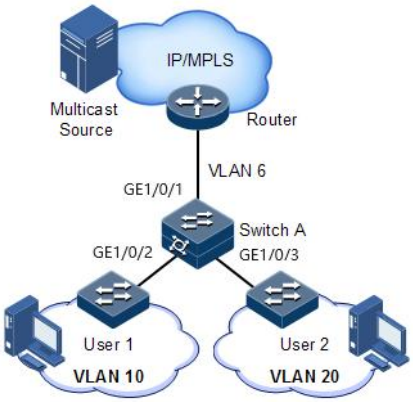
6.3.17 配置 MLD Snooping 组播复制应用示例

组网需求

如下图所示，交换机接口 Gigaethernet 1/0/1 连接组播路由器，接口 Gigaethernet 1/0/2，Gigaethernet 1/0/3 连接用户，组播 VLAN 和用户 VLAN 不同，配置 MLD Snooping 组播复制功能，

其中 GE1/0/1 加入组播 VLAN 6， 用户 1 加入用户 VLAN 10，用户 2 加入 VLAN 20。

图 6-14 MLD Snooping 组播复制功能组网应用示意图



配置步骤

步骤 1 创建 VLAN，并将接口加入 VLAN。

```
JX#config
JX(config)#vlan 6,10,20
JX(config)#interface ge 1/0/1
JX(config-ge-1/0/1)#port link-type trunk
JX(config-ge-1/0/1)#port trunk allow-pass vlan 6
JX(config-ge-1/0/1)#quit
JX(config)#interface ge 1/0/2
JX(config-ge-1/0/2)#port link-type access
JX(config-ge-1/0/2)#port default vlan 10
JX(config-ge-1/0/2)#quit
JX(config)#interface ge 1/0/3
JX(config-ge-1/0/3)#port link-type access
JX(config-ge-1/0/3)#port default vlan 20
JX(config-ge-1/0/3)#quit
```

步骤 2 配置使能 MLD Snooping 功能

```
JX(config)#mld-snooping start
JX(config)#vlan 6
JX(config-vlan-6)#mld-snooping enable
```

```
JX(config-vlan-6)#quit
JX(config)#interface ge 1/0/1
JX(config-ge-1/0/1)#mld-snooping enable
JX(config-ge-1/0/1)#quit
JX(config)#interface ge 1/0/2
JX(config-ge-1/0/2)#mld-snooping enable
JX(config-ge-1/0/2)#quit
JX(config)#interface ge 1/0/3
JX(config-ge-1/0/3)#mld-snooping enable
JX(config-ge-1/0/3)#quit
```

步骤 3 配置使能 MLD Snooping 开启组播复制功能

```
JX(config)#vlan 6
JX(config-vlan-6)#mld-snooping forwarding-mode ip
JX(config-vlan-6)#mld-snooping multicast-duplicate enable
JX(config-vlan-6)#mld-snooping multicast-user-vlan 10,20
JX(config-vlan-6)#quit
```

检查结果

查看 MLD Snooping 配置信息是否正确。

```
JX#show mld-snooping config
!
mld-snooping start
!
vlan 10
 mld-snooping enable
 mld-snooping forwarding-mode ip
 mld-snooping multicast-duplicate enable
 mld-snooping multicast-user-vlan 10,20
!
interface ge 1/0/1
 mld-snooping enable
!
interface ge 1/0/2
 mld-snooping enable
!
interface ge 1/0/3
 mld-snooping enable
```

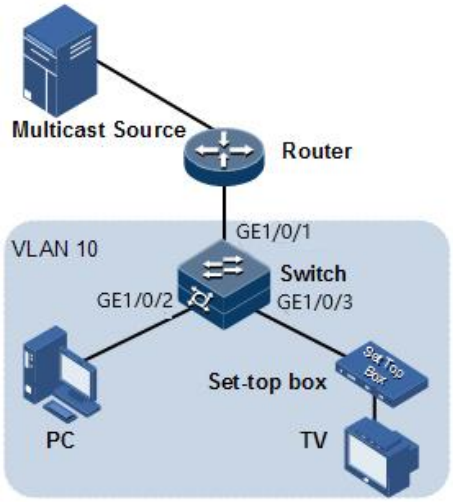
6.3.18 配置 MLD Snooping Proxy 应用示例

组网需求

如下图所示，交换机接口 Gigaethernet 1/0/1 连接组播路由器，接口 Gigaethernet 1/0/2，Gigaethernet 1/0/3 连接用户，用户在同一个 VLAN 10 中，配置 MLD Snooping 功能

在交换机上开启 MLD Proxy 功能，减少主机和组播路由器之间的通信，同时又不影响组播功能实现。

图 6-15 MLD Snooping Proxy 功能组网应用示意图



配置步骤

步骤 1 创建 VLAN，并将接口加入 VLAN。

```
JX#config
JX(config)#vlan 10
JX(config-vlan-10)#quit
JX(config)#interface ge 1/0/1
JX(config-ge-1/0/1)#port link-type trunk
JX(config-ge-1/0/1)#port trunk allow-pass vlan 10
JX(config-ge-1/0/1)#quit
JX(config)#interface ge 1/0/2
JX(config-ge-1/0/2)#port link-type access
JX(config-ge-1/0/2)#port default vlan 10
JX(config-ge-1/0/2)#quit
JX(config)#interface ge 1/0/3
JX(config-ge-1/0/3)#port link-type access
JX(config-ge-1/0/3)#port default vlan 10
JX(config-ge-1/0/3)#quit
```

步骤 2 配置使能 MLD Snooping 功能

```
JX(config)#mld-snooping start
JX(config)#vlan 10
JX(config-vlan-10)#mld-snooping enable
JX(config-vlan-10)#quit
JX(config)#interface ge 1/0/1
JX(config-ge-1/0/1)#mld-snooping enable
JX(config-ge-1/0/1)#quit
JX(config)#interface ge 1/0/2
JX(config-ge-1/0/2)#mld-snooping enable
JX(config-ge-1/0/2)#quit
JX(config)#interface ge 1/0/3
JX(config-ge-1/0/3)#mld-snooping enable
JX(config-ge-1/0/3)#quit
```

步骤 3 配置开启 MLD Proxy 功能

```
JX(config)#vlan 10
JX(config-vlan-10)#mld-snooping workmode mld-proxy
JX(config-vlan-10)#mld-snooping querier enable
JX(config-vlan-10)#quit
```

检查结果

查看 MLD Snooping 配置信息是否正确。

```
JX#show mld-snooping config
!
mld-snooping start
!
vlan 10
 mld-snooping enable
 mld-snooping workmode mld-proxy
 mld-snooping querier enable
!
interface ge 1/0/1
 mld-snooping enable
!
interface ge 1/0/2
 mld-snooping enable
!
interface ge 1/0/3
 mld-snooping enable
```

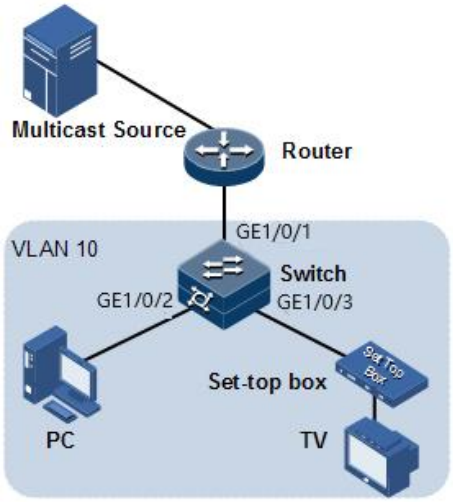
6.3.19 配置 MLD Snooping 策略应用示例

组网需求

如下图所示，交换机接口 Gigaethernet 1/0/1 连接组播路由器，接口 Gigaethernet 1/0/2，Gigaethernet 1/0/3 连接用户，用户在同一个 VLAN 10 中，配置 MLD Snooping 功能

在交换机上开启组播策略功能，接口 Gigaethernet 1/0/2 的用户允许加入 ff1e::1 ~ ff1e::3，接口 Gigaethernet 1/0/3 的用户允许加入 ff1e::4 ~ ff1e::6。

图 6-16 MLD Snooping 组播策略功能组网应用示意图



配置步骤

步骤 1 创建 VLAN，并将接口加入 VLAN。

```
JX#config
JX(config)#vlan 10
JX(config-vlan-10)#quit
JX(config)#interface ge 1/0/1
JX(config-ge-1/0/1)#port link-type trunk
JX(config-ge-1/0/1)#port trunk allow-pass vlan 10
JX(config-ge-1/0/1)#quit
JX(config)#interface ge 1/0/2
JX(config-ge-1/0/2)#port link-type access
JX(config-ge-1/0/2)#port default vlan 10
JX(config-ge-1/0/2)#quit
JX(config)#interface ge 1/0/3
JX(config-ge-1/0/3)#port link-type access
JX(config-ge-1/0/3)#port default vlan 10
JX(config-ge-1/0/3)#quit
```

步骤 2 配置使能 MLD Snooping 功能

```
JX(config)#mld-snooping start
JX(config)#vlan 10
JX(config-vlan-10)#mld-snooping enable
JX(config-vlan-10)#quit
JX(config)#interface ge 1/0/1
JX(config-ge-1/0/1)#mld-snooping enable
JX(config-ge-1/0/1)#quit
JX(config)#interface ge 1/0/2
JX(config-ge-1/0/2)#mld-snooping enable
JX(config-ge-1/0/2)#quit
JX(config)#interface ge 1/0/3
JX(config-ge-1/0/3)#mld-snooping enable
JX(config-ge-1/0/3)#quit
```

步骤 3 配置 GE1/0/2 用户的 ACL 模板

```
JX(config)#acl-ipv6 3001
JX(configure-acl-ipv6-3001)#rule 1 ip src-ip any dst-ip ff1e::1/128
JX(configure-acl-ipv6-3001)#rule 1 action permit
JX(configure-acl-ipv6-3001)#rule 2 ip src-ip any dst-ip ff1e::2/128
JX(configure-acl-ipv6-3001)#rule 2 action permit
JX(configure-acl-ipv6-3001)#rule 3 ip src-ip any dst-ip ff1e::3/128
JX(configure-acl-ipv6-3001)#rule 3 action permit
JX(configure-acl-ipv6-3001)#quit
```

步骤 4 配置 GE1/0/3 用户的 ACL 模板

```
JX(config)#acl-ipv6 3002
JX(configure-acl-ipv6-3002)#rule 1 ip src-ip any dst-ip ff1e::4/128
JX(configure-acl-ipv6-3002)#rule 1 action permit
JX(configure-acl-ipv6-3002)#rule 2 ip src-ip any dst-ip ff1e::5/128
JX(configure-acl-ipv6-3002)#rule 2 action permit
JX(configure-acl-ipv6-3002)#rule 3 ip src-ip any dst-ip ff1e::6/128
JX(configure-acl-ipv6-3002)#rule 3 action permit
JX(configure-acl-ipv6-3002)#quit
```

步骤 5 配置接口绑定组播策略 ACL 模板

```
JX(config)#interface ge 1/0/2
JX(config-ge-1/0/2)#mld-snooping group-policy acl-ipv6 3001
JX(config-ge-1/0/2)#quit
JX(config)#interface ge 1/0/3
JX(config-ge-1/0/3)#mld-snooping group-policy acl-ipv6 3002
JX(config-ge-1/0/3)#quit
```

检查结果

查看 MLD Snooping 配置信息是否正确。

```
JX#show mld-snooping config
!
mld-snooping start
!
vlan 10
 mld-snooping enable
!
interface ge 1/0/2
 mld-snooping enable
 mld-snooping group-policy acl-ipv6 3001 version 1-2
!
interface ge 1/0/3
 mld-snooping enable
 mld-snooping group-policy acl-ipv6 3002 version 1-2
```

查看 ACL 模板配置是否正确。

```
JX#show acl config
!
acl-ipv6 3001
 rule 1 ip src-ip any dst-ip ff1e::1/128
 rule 1 action permit
 rule 2 ip src-ip any dst-ip ff1e::2/128
 rule 2 action permit
```

```
 rule 3 ip src-ip any dst-ip ff1e::3/128
 rule 3 action permit
acl-ipv6 3002
 rule 1 ip src-ip any dst-ip ff1e::4/128
 rule 1 action permit
 rule 2 ip src-ip any dst-ip ff1e::5/128
 rule 2 action permit
 rule 3 ip src-ip any dst-ip ff1e::6/128
 rule 3 action permit
```

6.4 IGMP

6.4.1 简介

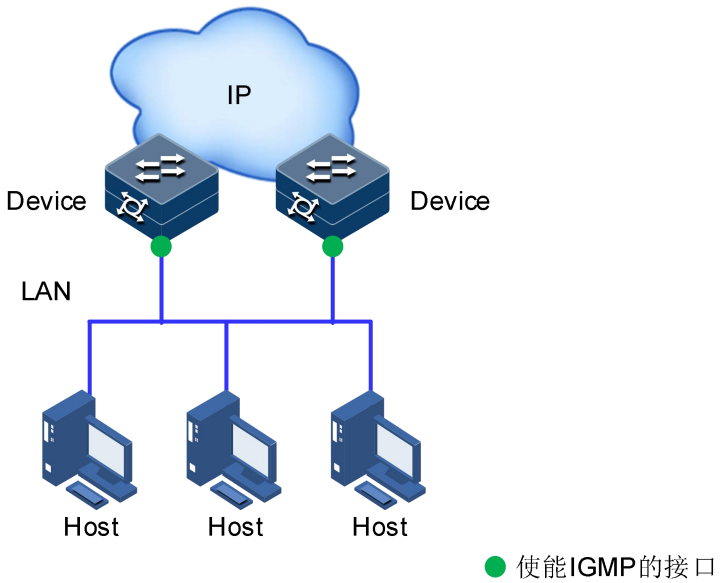
IP 组播通信的特点是报文从一个源发出，被转发到一组特定的接收者。但在组播通信模型中，发送者不关注接收者的位置信息。要使组播报文最终能够到达接收者 Host，需要某种机制使连接接收者网段的组播设备 Device 能够了解到该网段存在哪些组播接收者，同时保证接收者可以加入相应的组播组中。IGMP 就是用来在接收者主机和与其所在网段直接相邻的组播设备之间建立、维护组播组成员关系的协议

6.4.2 配置准备

场景

如下图所示，多个主机接收组播源的数据。

图 6-17 IGMP 在组播网络中的部署位置



前提

在配置 IGMP 之前，需完成以下任务：

- 在使能 IGMP 的接口配置 PIM 使能

6.4.3 IGMP 的缺省配置

设备上 IGMP 的缺省配置如下。

功能	缺省值
全局 IGMP 状态	禁用
接口的 IGMP 状态	禁用
IGMP 协议版本	v2
通用查询间隔	125s
健壮系数	2
最大响应时间	10s
其他查询器存活时间	255s

6.4.4 配置 IGMP 使能

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**pim** { **vpn-instance** *vpn-name* }	创建 pim 实例 注：igmp 功能必须依赖 pim 功能
3	JX(config)#**igmp** { **vpn-instance** *vpn-name* }	创建 igmp 实例。
4	JX(config)#**interface vlan** *vlan-id*	进入 VLANIF 配置视图
5	JX(config-vlanif-10)#**pim enable** { **sm** | **dm** }	使能 PIM 功能
6	JX(config-vlanif-10)#**igmp enable**	使能 IGMP 功能
7	JX(config-vlanif-10)#**igmp version** { **v1** | **v2** | **v3** }	（可选）配置协议版本

6.4.5 配置 IGMP 查询器参数

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**interface vlan** *vlan-id*	进入 VLANIF 配置视图
3	JX(config-vlanif-10)#**igmp timer query** *interval*	配置查询间隔，单位：秒
4	JX(config-vlanif-10)#**igmp robust-count** *robust-value*	配置查询器的健壮系数
5	JX(config-vlanif-10)#**igmp timer other-querier-present** *interval*	配置其他 IGMP 查询器存活时间

6.4.6 配置 IGMP 快速离开

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**interface vlan** *vlan-id*	进入 VLANIF 配置视图
3	JX(config-vlanif-10)#igmp fast-leave enable	配置快速离开使能： 注：快速离开只对 IGMPv2 生效

6.4.7 检查配置

配置完成后，请在设备上进行以下命令检查配置结果。

序号	检查项	说明
1	JX#**show igmp config**	查看 IGMP 配置信息。

6.4.8 维护

用户可以通过以下命令维护 IGMP。

命令	描述
JX#**show igmp interface**	查看 IGMP 接口信息。
JX#**show igmp routing-table**	查看 IGMP 组播表信息
JX#**show igmp** { **public-net** | **vpn-instance** *vpn-name* }	查看 IGMP 实例信息

6.4.9 配置 IGMP 基本功能应用示例

组网需求

如下图所示，交换机接口 Gigaethernet 1/0/1 连接组播路由器，接口 Gigaethernet 1/0/2，Gigaethernet 1/0/3 连接用户，组播 VLAN 和用户 VLAN 不同，配置 IGMP 基本功能，

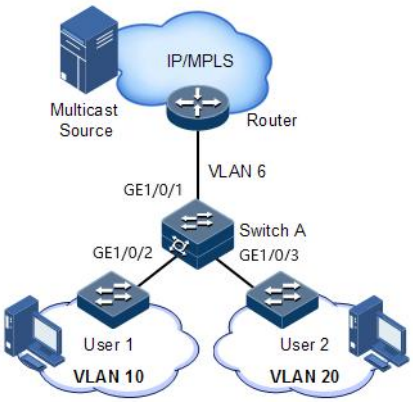
其中 GE1/0/1 加入 VLAN 6， 用户 1 加入 VLAN 10，用户 2 加入 VLAN 20。

组播路由器接口 ip 地址 203.6.1.1/24

用户 1 接口 ip 地址 10.1.1.1/24

用户 2 接口 ip 地址 20.1.1.1/24

图 6-18 IGMP 基本功能功能组网应用示意图



配置步骤

步骤 1 创建 VLAN，并将接口加入 VLAN。

```
JX#config
JX(config)#vlan 6,10,20
JX(config)#interface ge 1/0/1
JX(config-ge-1/0/1)#port link-type trunk
JX(config-ge-1/0/1)#port trunk allow-pass vlan 6
JX(config-ge-1/0/1)#quit
JX(config)#interface ge 1/0/2
JX(config-ge-1/0/2)#port link-type access
JX(config-ge-1/0/2)#port default vlan 10
JX(config-ge-1/0/2)#quit
JX(config)#interface ge 1/0/3
JX(config-ge-1/0/3)#port link-type access
JX(config-ge-1/0/3)#port default vlan 20
JX(config-ge-1/0/3)#quit
```

步骤 2 配置接口 IP 地址

```
JX(config)#interface vlan 6
JX(config-vlanif-6)#ip address 203.6.1.1/24
JX(config-vlanif-6)#quit
JX(config)#interface vlan 10
JX(config-vlanif-10)#ip address 10.1.1.1/24
JX(config-vlanif-10)#quit
JX(config)#interface vlan 20
JX(config-vlanif-20)#ip address 20.1.1.1/24
JX(config-vlanif-20)#quit
```

步骤 3 配置 PIM-SM 和 IGMP 基本功能

```
JX(config)#pim
JX(config-pim)#quit
JX(config)#igmp
JX(config-igmp)#quit
JX(config)#interface vlan 6
JX(config-vlanif-6)#pim enable sm
JX(config-vlanif-6)#igmp enable
JX(config-vlanif-6)#quit
JX(config)#interface vlan 10
JX(config-vlanif-10)#pim enable sm
JX(config-vlanif-10)#igmp enable
JX(config-vlanif-10)#quit
JX(config)#interface vlan 20
JX(config-vlanif-20)#pim enable sm
JX(config-vlanif-20)#igmp enable
JX(config-vlanif-20)#quit
```

检查结果

查看 IGMP 配置信息是否正确。

```
JX#show igmp config
!
igmp
!
interface vlan 6
 igmp enable
!
interface vlan 10
 igmp enable
!
interface vlan 20
 igmp enable
```

查看 PIM 配置信息是否正确。

```
JX#show pim config
!
pim
!
interface vlan 6
 pim enable sm
```

```
!
interface vlan 10
 pim enable sm
!
interface vlan 20
 pim enable sm
```

6.5 PIM

6.5.1 简介

PIM（Protocol Independent Multicast，协议无关组播）是域内组播路由协议，利用单播路由信息，对组播消息进行 RPF 检查，创建组播路由表项。为 IP 组播提供路由信息的协议可以是任何单播路由协议，比如静态路由、RIP、OSPF、IS-IS、BGP 等。组播路由和单播路由协议无关，只是通过单播路由表产生相应组播路由表项。

在网络中，组播数据从组播源到接收者，中间需要通过组播网络来实现组播报文的复制和转发。为了构建组播网络，需要在网络中配置组播协议，PIM 是目前应用最为广泛的域内组播协议。通过 PIM 协议构建的组播分发树，可以用来指导网络中的组播数据转发。

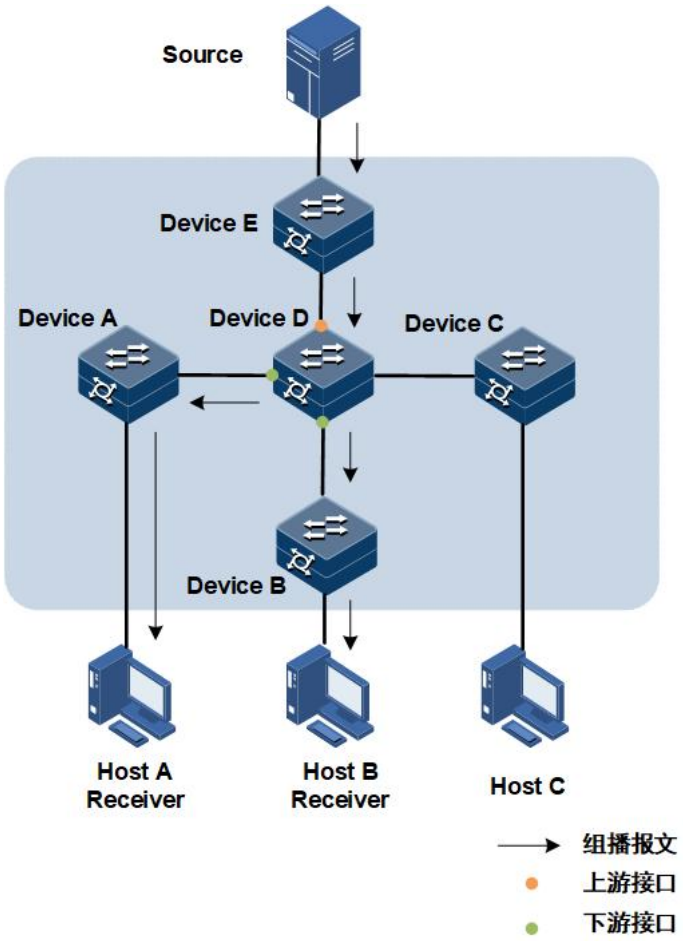
PIM 协议可实现按需创建组播路由、动态响应网络拓扑变化、按照路由表项执行转发等功能。

6.5.2 配置准备

场景

如下图所示。

图 6-19 PIM 在组播网络中的部署位置



前提

无

6.5.3 PIM 的缺省配置

设备上 PIM 的缺省配置如下。

功能	缺省值
全局 PIM 状态	禁用
接口的 PIM 状态	禁用
DR 优先级	1
Hello 报文间隔	30 s
SPT 切换条件	RP 或组成员端 DR 接收到第一个组播数据报文时就进行 SPT 切换

功能	缺省值
BFD	禁用
PIM Slient	禁用

6.5.4 配置 PIM-DM 功能

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**pim** { **vpn-instance** *vpn-name* }	创建 pim 实例
3	JX(config)#**interface vlan** *vlan-id*	进入 VLANIF 配置视图
4	JX(config-vlanif-10)#**pim enable dm**	使能 PIM-DM 功能

6.5.5 配置 PIM-SM 功能

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**pim** { **vpn-instance** *vpn-name* }	创建 pim 实例
3	JX(config-pim)#**rp-address** *ip-address* **group all-multicast**	（可选）配置静态 rp
4	JX(config)#**interface vlan** *vlan-id*	进入 VLANIF 配置视图
5	JX(config-vlanif-10)#**pim enable sm**	使能 PIM-SM 功能
6	JX(config-vlanif-10)#**pim c-rp group all-multicast**	（可选）配置候选 rp
7	JX(config-vlanif-10)#**pim c-bsr group all-multicast**	（可选）配置候选 bsr

6.5.6 配置 PIM-SM 得 BSR 管理域功能

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**pim** { **vpn-instance** *vpn-name* }	创建 pim 实例
3	JX(config-pim)#**c-bsr admin-scope enable**	使能 BSR 管理域功能
4	JX(config)#**interface vlan** *vlan-id*	进入 VLANIF 配置视图
5	JX(config-vlanif-10)#**pim enable sm**	使能 PIM-SM 功能
6	JX(config-vlanif-10)#**pim c-rp group** *group-address/group-masklen*	（可选）配置管理域候选 rp 服务的组地址范围
7	JX(config-vlanif-10)#**pim c-bsr group** *group-address/group-masklen*	（可选）配置管理域候选 bsr 服务的组地址范围
8	JX(config-vlanif-10)#**pim bsr-boundary group** *group-address/group-masklen*	（可选）配置 BSR 管理域边界

6.5.7 检查配置

配置完成后，请在设备上进行以下命令检查配置结果。

序号	检查项	说明
1	JX#**show pim config**	查看 PIM 配置信息。

6.5.8 维护

用户可以通过以下命令维护 PIM。

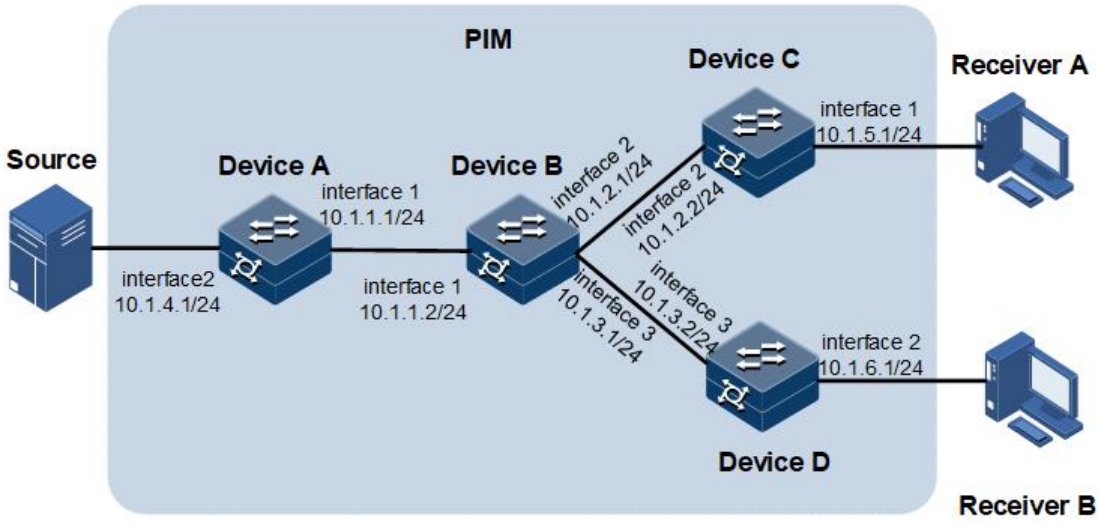
命令	描述
JX#**show pim interface**	查看 PIM 接口信息。
JX#**show pim routing-table**	查看 PIM 组播表信息
JX#**show pim** { **public-net** | **vpn-instance** *vpn-name* }	查看 PIM 实例信息

6.5.9 配置 PIM-DM 功能应用示例

组网需求

要求通过在设备上部署组播功能，使网络中的用户主机能够通过组播方式接收视频点播信息。

图 6-20 PIM-DM 功能组网应用示意图



配置步骤

步骤 1 创建 VLAN，并将接口加入 VLAN。

```
Device A:
JX#config
JX(config)#vlan 10,20
JX(config)#interface ge 1/0/1
JX(config-ge-1/0/1)#port hybrid vlan 10 untagged
JX(config-ge-1/0/1)#port hybrid pvid 10
JX(config-ge-1/0/1)#quit
JX(config)#interface ge 1/0/2
JX(config-ge-1/0/2)#port hybrid vlan 20 untagged
JX(config-ge-1/0/2)#port hybrid pvid 20
JX(config-ge-1/0/2)#quit

Device B
JX#config
JX(config)#vlan 10,20,30
JX(config)#interface ge 1/0/1
JX(config-ge-1/0/1)#port hybrid vlan 10 untagged
JX(config-ge-1/0/1)#port hybrid pvid 10
JX(config-ge-1/0/1)#quit
JX(config)#interface ge 1/0/2
JX(config-ge-1/0/2)#port hybrid vlan 20 untagged
JX(config-ge-1/0/2)#port hybrid pvid 20
JX(config-ge-1/0/2)#quit
JX(config)#interface ge 1/0/3
JX(config-ge-1/0/3)#port hybrid vlan 30 untagged
JX(config-ge-1/0/3)#port hybrid pvid 30
JX(config-ge-1/0/3)#quit

Device C
JX#config
JX(config)#vlan 10,20
JX(config)#interface ge 1/0/1
```

```
JX(config-ge-1/0/1)#port hybrid vlan 10 untagged
JX(config-ge-1/0/1)#port hybrid pvid 10
JX(config-ge-1/0/1)#quit
JX(config)#interface ge 1/0/2
JX(config-ge-1/0/2)#port hybrid vlan 20 untagged
JX(config-ge-1/0/2)#port hybrid pvid 20
JX(config-ge-1/0/2)#quit

Device D
JX#config
JX(config)#vlan 20,30
JX(config)#interface ge 1/0/2
JX(config-ge-1/0/2)#port hybrid vlan 20 untagged
JX(config-ge-1/0/2)#port hybrid pvid 20
JX(config-ge-1/0/2)#quit
JX(config)#interface ge 1/0/3
JX(config-ge-1/0/3)#port hybrid vlan 30 untagged
JX(config-ge-1/0/3)#port hybrid pvid 30
JX(config-ge-1/0/3)#quit
```

步骤 2 配置接口 IP 地址

```
Device A
JX(config)#interface vlan 10
JX(config-vlanif-10)#ip address 10.1.1.1/24
JX(config-vlanif-10)#quit
JX(config)#interface vlan 20
JX(config-vlanif-20)#ip address 10.1.4.1/24
JX(config-vlanif-20)#quit

Device B
JX(config)#interface vlan 10
JX(config-vlanif-10)#ip address 10.1.1.2/24
JX(config-vlanif-10)#quit
JX(config)#interface vlan 20
JX(config-vlanif-20)#ip address 10.1.2.1/24
JX(config-vlanif-20)#quit
JX(config)#interface vlan 30
JX(config-vlanif-30)#ip address 10.1.3.1/24
JX(config-vlanif-30)#quit

Device C
JX(config)#interface vlan 10
JX(config-vlanif-10)#ip address 10.1.5.1/24
JX(config-vlanif-10)#quit
JX(config)#interface vlan 20
JX(config-vlanif-20)#ip address 10.1.2.2/24
JX(config-vlanif-20)#quit

Device D
JX(config)#interface vlan 20
JX(config-vlanif-20)#ip address 10.1.6.1/24
JX(config-vlanif-20)#quit
JX(config)#interface vlan 30
```

```
JX(config-vlanif-30)#ip address 10.1.3.2/24
JX(config-vlanif-30)#quit
```

步骤 3 配置 OSPF 单播路由

```
Device A
JX(config)#ospf 1
JX(config-ospf-1)#router-id 1.1.1.1
JX(config-ospf-1)#network 10.0.0.0 255.0.0.0 area 0

Device B
JX(config)#ospf 1
JX(config-ospf-1)#router-id 2.2.2.2
JX(config-ospf-1)#network 10.0.0.0 255.0.0.0 area 0

Device A
JX(config)#ospf 1
JX(config-ospf-1)#router-id 3.3.3.3
JX(config-ospf-1)#network 10.0.0.0 255.0.0.0 area 0

Device A
JX(config)#ospf 1
JX(config-ospf-1)#router-id 4.4.4.4
JX(config-ospf-1)#network 10.0.0.0 255.0.0.0 area 0
```

步骤 4 配置 PIM-DM 和 IGMP 功能

```
Device A
JX(config)#pim
JX(config-pim)#quit
JX(config)#interface vlan 10
JX(config-vlanif-10)#pim enable dm
JX(config-vlanif-10)#quit
JX(config)#interface vlan 20
JX(config-vlanif-20)#pim enable dm
JX(config-vlanif-20)#quit

Device B
JX(config)#pim
JX(config-pim)#quit
JX(config)#interface vlan 10
JX(config-vlanif-10)#pim enable dm
JX(config-vlanif-10)#quit
JX(config)#interface vlan 20
JX(config-vlanif-20)#pim enable dm
JX(config-vlanif-20)#quit
JX(config)#interface vlan 30
JX(config-vlanif-30)#pim enable dm
JX(config-vlanif-30)#quit

Device C
JX(config)#pim
JX(config-pim)#quit
JX(config)#igmp
JX(config-igmp)#quit
```

```
JX(config-pim)#quit
JX(config)#interface vlan 10
JX(config-vlanif-10)#pim enable dm
JX(config-vlanif-10)#igmp enable
JX(config-vlanif-10)#quit
JX(config)#interface vlan 20
JX(config-vlanif-20)#pim enable dm
JX(config-vlanif-20)#quit
```

```
Device D
JX(config)#pim
JX(config-pim)#quit
JX(config)#igmp
JX(config-igmp)#quit
JX(config)#interface vlan 20
JX(config-vlanif-20)#pim enable dm
JX(config-vlanif-20)#igmp enable
JX(config-vlanif-20)#quit
JX(config)#interface vlan 30
JX(config-vlanif-30)#pim enable dm
JX(config-vlanif-30)#quit
```

检查结果

Device A:

查看 PIM 配置信息是否正确。

```
JX#show pim config
!
pim
!
interface vlan 10
 pim enable dm
!
interface vlan 20
 pim enable dm
```

Device B:

查看 PIM 配置信息是否正确。

```
JX#show pim config
!
pim
!
interface vlan 10
 pim enable dm
!
interface vlan 20
 pim enable dm
!
interface vlan 30
 pim enable dm
```

Device C:

查看 PIM 配置信息是否正确。

```
JX#show pim config
!
pim
!
interface vlan 10
 pim enable dm
!
interface vlan 20
 pim enable dm
```

Device D:

查看 PIM 配置信息是否正确。

```
JX#show pim config
!
pim
!
interface vlan 20
 pim enable dm
!
interface vlan 30
 pim enable dm
```

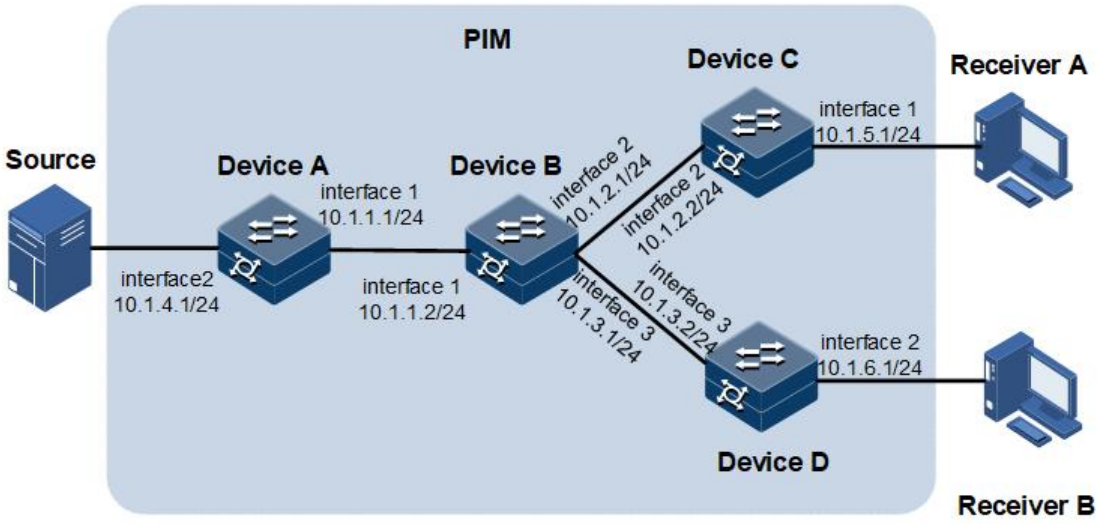
6.5.10 配置 PIM-SM 功能应用示例

组网需求

要求通过在设备上部署组播功能，使网络中的用户主机能够通过组播方式接收视频点播信息。

其中在 DeivceA 和 Deivce B 上配置候选 RP 和候选 BSR

图 6-21 PIM-SM 功能功能组网应用示意图



配置步骤

步骤 1 创建 VLAN，并将接口加入 VLAN。

```
Device A:
JX#config
JX(config)#vlan 10,20
JX(config)#interface ge 1/0/1
JX(config-ge-1/0/1)#port hybrid vlan 10 untagged
JX(config-ge-1/0/1)#port hybrid pvid 10
JX(config-ge-1/0/1)#quit
JX(config)#interface ge 1/0/2
JX(config-ge-1/0/2)#port hybrid vlan 20 untagged
JX(config-ge-1/0/2)#port hybrid pvid 20
JX(config-ge-1/0/2)#quit

Device B
JX#config
JX(config)#vlan 10,20,30
JX(config)#interface ge 1/0/1
JX(config-ge-1/0/1)#port hybrid vlan 10 untagged
JX(config-ge-1/0/1)#port hybrid pvid 10
JX(config-ge-1/0/1)#quit
JX(config)#interface ge 1/0/2
JX(config-ge-1/0/2)#port hybrid vlan 20 untagged
JX(config-ge-1/0/2)#port hybrid pvid 20
JX(config-ge-1/0/2)#quit
JX(config)#interface ge 1/0/3
JX(config-ge-1/0/3)#port hybrid vlan 30 untagged
JX(config-ge-1/0/3)#port hybrid pvid 30
JX(config-ge-1/0/3)#quit

Device C
JX#config
JX(config)#vlan 10,20
JX(config)#interface ge 1/0/1
```

```
JX(config-ge-1/0/1)#port hybrid vlan 10 untagged
JX(config-ge-1/0/1)#port hybrid pvid 10
JX(config-ge-1/0/1)#quit
JX(config)#interface ge 1/0/2
JX(config-ge-1/0/2)#port hybrid vlan 20 untagged
JX(config-ge-1/0/2)#port hybrid pvid 20
JX(config-ge-1/0/2)#quit

Device D
JX#config
JX(config)#vlan 20,30
JX(config)#interface ge 1/0/2
JX(config-ge-1/0/2)#port hybrid vlan 20 untagged
JX(config-ge-1/0/2)#port hybrid pvid 20
JX(config-ge-1/0/2)#quit
JX(config)#interface ge 1/0/3
JX(config-ge-1/0/3)#port hybrid vlan 30 untagged
JX(config-ge-1/0/3)#port hybrid pvid 30
JX(config-ge-1/0/3)#quit
```

步骤 2 配置接口 IP 地址

```
Device A
JX(config)#interface vlan 10
JX(config-vlanif-10)#ip address 10.1.1.1/24
JX(config-vlanif-10)#quit
JX(config)#interface vlan 20
JX(config-vlanif-20)#ip address 10.1.4.1/24
JX(config-vlanif-20)#quit

Device B
JX(config)#interface vlan 10
JX(config-vlanif-10)#ip address 10.1.1.2/24
JX(config-vlanif-10)#quit
JX(config)#interface vlan 20
JX(config-vlanif-20)#ip address 10.1.2.1/24
JX(config-vlanif-20)#quit
JX(config)#interface vlan 30
JX(config-vlanif-30)#ip address 10.1.3.1/24
JX(config-vlanif-30)#quit

Device C
JX(config)#interface vlan 10
JX(config-vlanif-10)#ip address 10.1.5.1/24
JX(config-vlanif-10)#quit
JX(config)#interface vlan 20
JX(config-vlanif-20)#ip address 10.1.2.2/24
JX(config-vlanif-20)#quit

Device D
JX(config)#interface vlan 20
JX(config-vlanif-20)#ip address 10.1.6.1/24
JX(config-vlanif-20)#quit
JX(config)#interface vlan 30
```

```
JX(config-vlanif-30)#ip address 10.1.3.2/24
JX(config-vlanif-30)#quit
```

步骤 3 配置 OSPF 单播路由

```
Device A
JX(config)#ospf 1
JX(config-ospf-1)#router-id 1.1.1.1
JX(config-ospf-1)#network 10.0.0.0 255.0.0.0 area 0

Device B
JX(config)#ospf 1
JX(config-ospf-1)#router-id 2.2.2.2
JX(config-ospf-1)#network 10.0.0.0 255.0.0.0 area 0

Device A
JX(config)#ospf 1
JX(config-ospf-1)#router-id 3.3.3.3
JX(config-ospf-1)#network 10.0.0.0 255.0.0.0 area 0

Device A
JX(config)#ospf 1
JX(config-ospf-1)#router-id 4.4.4.4
JX(config-ospf-1)#network 10.0.0.0 255.0.0.0 area 0
```

步骤 4 配置 PIM-SM 和 IGMP 功能

```
Device A
JX(config)#pim
JX(config-pim)#quit
JX(config)#interface vlan 10
JX(config-vlanif-10)#pim enable sm
JX(config-vlanif-10)#pim c-rp group all-multicast
JX(config-vlanif-10)#pim c-bsr group all-multicast
JX(config-vlanif-10)#quit
JX(config)#interface vlan 20
JX(config-vlanif-20)#pim enable sm
JX(config-vlanif-20)#quit

Device B
JX(config)#pim
JX(config-pim)#quit
JX(config)#interface vlan 10
JX(config-vlanif-10)#pim enable sm
JX(config-vlanif-10)#pim c-rp group all-multicast
JX(config-vlanif-10)#pim c-bsr group all-multicast
JX(config-vlanif-10)#quit
JX(config)#interface vlan 20
JX(config-vlanif-20)#pim enable sm
JX(config-vlanif-20)#quit
JX(config)#interface vlan 30
JX(config-vlanif-30)#pim enable sm
JX(config-vlanif-30)#quit

Device C
JX(config)#pim
```

```
JX(config-pim)#quit
JX(config)#igmp
JX(config-igmp)#quit
JX(config-pim)#quit
JX(config)#interface vlan 10
JX(config-vlanif-10)#pim enable sm
JX(config-vlanif-10)#igmp enable
JX(config-vlanif-10)#quit
JX(config)#interface vlan 20
JX(config-vlanif-20)#pim enable sm
JX(config-vlanif-20)#quit
```

```
Device D
JX(config)#pim
JX(config-pim)#quit
JX(config)#igmp
JX(config-igmp)#quit
JX(config)#interface vlan 20
JX(config-vlanif-20)#pim enable sm
JX(config-vlanif-20)#igmp enable
JX(config-vlanif-20)#quit
JX(config)#interface vlan 30
JX(config-vlanif-30)#pim enable sm
JX(config-vlanif-30)#quit
```

检查结果

Device A:

查看 PIM 配置信息是否正确。

```
JX#show pim config
!
pim
!
interface vlan 10
 pim enable sm
 pim c-bsr group all-multicast
 pim c-rp group all-multicast
!
interface vlan 20
 pim enable sm
```

Device B:

查看 PIM 配置信息是否正确。

```
JX#show pim config
!
pim
!
interface vlan 10
 pim enable sm
 pim c-bsr group all-multicast
 pim c-rp group all-multicast
```

```
!
interface vlan 20
 pim enable sm
!
interface vlan 30
 pim enable sm
```

Device C:

查看 PIM 配置信息是否正确。

```
JX#show pim config
!
pim
!
interface vlan 10
 pim enable sm
!
interface vlan 20
 pim enable sm
```

Device D:

查看 PIM 配置信息是否正确。

```
JX#show pim config
!
pim
!
interface vlan 20
 pim enable sm
!
interface vlan 30
 pim enable sm
```

6.6 MLD

6.6.1 简介

IP 组播通信的特点是报文从一个源发出，被转发到一组特定的接收者。但在组播通信模型中，发送者不关注接收者的位置信息。要使组播报文最终能够到达接收者 Host，需要某种机制使连接接收者网段的组播设备 Device 能够了解到该网段存在哪些组播接收者，同时保证接收者可以加入相应的组播组中。MLD 就是用来在接收者主机和与其所在网段直接相邻的组播设备之间建立、维护组播组成员关系的协议

6.6.2 配置准备

场景

参考 IGMP 测试场景

前提

在配置 MLD 之前，需完成以下任务：

- 在使能 MLD 的接口配置 PIM-Ipv6 使能

6.6.3 MLD 的缺省配置

设备上 MLD 的缺省配置如下。

功能	缺省值
全局 MLD 状态	禁用
接口的 MLD 状态	禁用
MLD 协议版本	v2
通用查询间隔	125s
健壮系数	2
最大响应时间	10s
其他查询器存活时间	255s

6.6.4 配置 MLD 使能

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**pim-ipv6** { **vpn-instance** *vpn-name* }	创建 pim-ipv6 实例 注：mld 功能必须依赖 pim-ipv6 功能
3	JX(config)#**mld** { **vpn-instance** *vpn-name* }	创建 mld 实例。
4	JX(config)#**interface vlan** *vlan-id*	进入 VLANIF 配置视图
5	JX(config-vlanif-10)#**ipv6 enable**	使能 IPV6 功能
6	JX(config-vlanif-10)#**pim-ipv6 enable** { **sm** | **dm** }	使能 PIM 功能
7	JX(config-vlanif-10)#**mld enable**	使能 MLD 功能
8	JX(config-vlanif-10)#**mld version** { **v1** | **v2** }	（可选）配置协议版本

6.6.5 配置 MLD 查询器参数

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**interface vlan** *vlan-id*	进入 VLANIF 配置视图
3	JX(config-vlanif-10)#**mld timer query** *interval*	配置查询间隔，单位：秒
4	JX(config-vlanif-10)#**mld robust-count** *robust-value*	配置查询器的健壮系数
5	JX(config-vlanif-10)#**mld timer other-querier-present** *interval*	配置其他 MLD 查询器存活时间

6.6.6 配置 MLD 快速离开

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**interface vlan** *vlan-id*	进入 VLANIF 配置视图
3	JX(config-vlanif-10)#**mld fast-leave enable**	配置快速离开使能： 注：快速离开只对 MLDv1 生效

6.6.7 检查配置

配置完成后，请在设备上进行以下命令检查配置结果。

序号	检查项	说明
1	JX#**show mld config**	查看 MLD 配置信息。

6.6.8 维护

用户可以通过以下命令维护 MLD。

命令	描述
JX#**show mld interface**	查看 MLD 接口信息。
JX#**show mld routing-table**	查看 MLD 组播表信息
JX#**show mld** { **public-net** | **vpn-instance** *vpn-name* }	查看 MLD 实例信息

6.6.9 配置 MLD 基本功能应用示例

参考 IGMP 基本功能应用示例

6.7 PIMv6

6.7.1 简介

参考 PIM 简介。

6.7.2 配置准备

场景

参考 PIM 配置准备

前提

无

6.7.3 PIMv6 的缺省配置

设备上 PIMv6 的缺省配置如下。

功能	缺省值
全局 PIMv6 状态	禁用
接口的 PIMv6 状态	禁用
DR 优先级	1
Hello 报文间隔	30 s
SPT 切换条件	RP 或组成员端 DR 接收到第一个组播数据报文时就进行 SPT 切换
BFD	禁用

6.7.4 配置 PIMv6-DM 功能

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**pim-ipv6** { **vpn-instance** *vpn-name* }	创建 pim-ipv6 实例
3	JX(config)#**interface vlan** *vlan-id*	进入 VLANIF 配置视图
4	JX(config-vlanif-10)#**ipv6 enable**	使能 IPV6 功能
5	JX(config-vlanif-10)#**pim-ipv6 enable dm**	使能 PIMv6-DM 功能

6.7.5 配置 PIM-SM 功能

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**pim-ipv6** { **vpn-instance** *vpn-name* }	创建 pim-ipv6 实例
3	JX(config-pim6)#**rp-address** *ipv6-address* **group all-multicast**	（可选）配置静态 rp
4	JX(config)#**interface vlan** *vlan-id*	进入 VLANIF 配置视图
5	JX(config-vlanif-10)#**ipv6 enable**	使能 IPV6 功能
6	JX(config-vlanif-10)#**pim-ipv6 enable sm**	使能 PIMv6-SM 功能
7	JX(config-vlanif-10)#**pim-ipv6 c-rp group all-multicast**	（可选）配置候选 rp
8	JX(config-vlanif-10)#**pim-ipv6 c-bsr group all-multicast**	（可选）配置候选 bsr

6.7.6 配置 PIM-SM 得 BSR 管理域功能

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**pim-ipv6** { **vpn-instance** *vpn-name* }	创建 pim-ipv6 实例
3	JX(config-pim6)#**c-bsr admin-scope enable**	使能 BSR 管理域功能
4	JX(config)#**interface vlan** *vlan-id*	进入 VLANIF 配置视图

5	JX(config-vlanif-10)#**ipv6 enable**	使能 IPV6 功能
6	JX(config-vlanif-10)#**pim-ipv6 enable sm**	使能 PIMv6-SM 功能
7	JX(config-vlanif-10)#**pim-ipv6 c-rp group** *group-address/group-masklen*	（可选）配置管理域候选 rp 服务的组地址范围
8	JX(config-vlanif-10)#**pim-ipv6 c-bsr group** *group-address/group-masklen*	（可选）配置管理域候选 bsr 服务的组地址范围
9	JX(config-vlanif-10)#**pim-ipv6 bsr-boundary group** *group-address/group-masklen*	（可选）配置 BSR 管理域边界

6.7.7 检查配置

配置完成后，请在设备上进行以下命令检查配置结果。

序号	检查项	说明
1	JX#**show pim-ipv6 config**	查看 PIMv6 配置信息。

6.7.8 维护

用户可以通过以下命令维护 PIMv6。

命令	描述
JX#**show pim-ipv6 interface**	查看 PIMv6 接口信息。
JX#**show pim-ipv6 routing-table**	查看 PIMv6 组播表信息
JX#**show pim-ipv6** { **public-net** | **vpn-instance** *vpn-name* }	查看 PIMv6 实例信息

6.7.9 配置 PIMv6-DM 功能应用示例

参考 pim-dm 功能应用示例

6.7.10 配置 PIM-SM 功能应用示例

参考 pim-sm 功能应用示例

7 OAM

本章介绍 OAM 特性的基本原理和配置过程，并提供相关的配置案例。

- 简介
- EFM
- 故障转移
- VRRP
- CFM

7.1 简介

以太网最初为局域网设计，由于规模较小，所以 OAM（Operation，Administration and Maintenance，运行、管理和维护）能力较弱，且只有网元级的管理系统。随着以太网技术的不断发展，以太网在电信级网络中的应用也越来越广泛，但电信级网络在链路长度和网络规模上都较局域网大很多，有效管理维护机制的缺乏，已成为以太网技术在电信级网络中应用的严重障碍。

为了在以太网层能确定以太网虚链接的连通性，有效地检测、确认并定位以太网层网络内部的故障，并且可以衡量网络的利用率以及度量网络的性能，从而能根据与用户签订的 SLA（Service Level Agreement，服务等级协议）提供业务，在以太网上实现 OAM 机制已经成为必然的发展趋势。

工作模式

使能 EFM OAM 功能的接口称为 OAM 实体。EFM OAM 支持以下两种工作连接模式：

- 主动模式：连接过程由处于主动模式的 OAM 实体发起；
- 被动模式：处于被动模式的 OAM 实体只能等待对端 OAM 实体的连接请求。如果链路两端的 OAM 实体都处于被动模式，则无法建立 OAM 连接。

OAM 发现

OAM 发现阶段是 OAM 实体发现对端设备的 OAM 实体，并与之建立稳定对话的过程。

本阶段由主动模式的 OAM 实体发起，两端通过交互配置信息 OAM PDU 通报各自的 OAM 配置信息及本地节点支持的 OAM 能力信息，并决定是否同意建立 OAM 连接。如果两端都同意建立 OAM 连接，则 OAM 协议将在链路层开始正常工作。

以太网 OAM 连接建立后，两端的 OAM 实体通过发送 Information OAM PDU 保持连接。若在超时时间内未收到对端 OAM 实体的 Information OAM PDU，则认为连接超时，需要重新建立 OAM 连接。

对端故障通知

当设备发生故障或不可用时将会导致网络中断，因此，OAM PDU 定义了一个标志位（Flag 域）允许 OAM 实体不断发送 Information OAM PDU 给对端，告知此故障信息。

- 链路故障（Link Fault）：对端链路信号丢失，每秒发送 1 次 OAM PDU。
- 致命故障（Dying Gasp）：设备发生导致系统不能恢复的不可预知的故障，不间断发送 OAM PDU。例如电源中断等。
- 严重事件（Critical Event）：设备发生不能确定的紧急事件，不间断发送 OAM PDU。例如温度异常等。

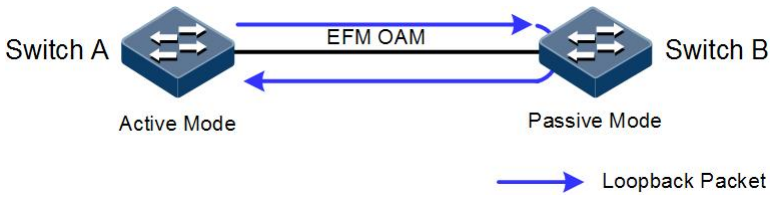
远端环回

远端环回功能可用于定位故障发生的区域，同时借助仪器仪表还可以对链路质量进行测试。定期地进行环回检测可以及时发现网络故障，并通过分段环回检测来定位故障发生的具体区域，有助于用户排除故障。

OAM 环回只有在以太网 OAM 连接建立完成后才能实现。在连接建立的情况下，主动模式的 OAM 实体发起 OAM 环回命令，对端实体对该命令进行响应。当对端处于环回模式时，除 OAM PDU 报文以外的所有报文都将按原路返回。

如图 7-1 所示，本地处于主动模式的 Switch A 设备将通过返回报文的情况，确定链路状况。

图 7-1 OAM 环回示意图



7.2 EFM

7.2.1 简介

遵循 IEEE 802.3ah 协议的 EFM（Ethernet in the First Mile，第一公里以太网）是一种链路级以太网 OAM 技术，针对两台直连设备之间的链路，提供链路连通性检测功能、

链路故障监控功能、远端故障通知功能等。EFM 主要用于用户接入的网络边缘的以太网链路。

7.2.2 配置准备

场景

在直连设备之间部署 EFM 特性可以有效提高对以太网链路的管理和维护能力，保障网络的稳定运行。

前提

需要连接接口并配置接口的物理参数，使接口的物理层状态为 Up。

7.2.3 缺省配置

设备上 EFM 的缺省配置如下。

功能	缺省值
EFM 工作模式	主模式
报文发送间隔	10×100ms
链路超时时间	5s
EFM 远端环回状态	不响应
误帧事件监控窗口	1s
误帧事件监控阈值	1 个错误帧
误帧周期事件监控窗口	1000ms
误帧周期事件监控阈值	1 个错误帧
链路误帧秒统计事件的监控窗口	100s
链路误帧秒统计事件的监控阈值	1s
误符号周期事件监控窗口	1s
误符号周期事件监控阈值	1s
故障指示功能状态	使能

7.2.4 配置 EFM 基本功能

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**configure**	进入全局配置模式。

步骤	配置	说明
2	JX(config)#**interface** *interface-type interface-number* Example: JX(config)#**interface** ge **1/0/1**	进入物理接口配置模式。 interface-type：接口类型 interface-number：接口号
3	JX(config-ge-1/0/1)#**efm max-rate value** Example: JX(config-ge-1/0/1)#**efm max-rate 5**	（可选）最大发送速率限制 EFM 占用的带宽，保证在一定时间间隔内最多只能发送一定数量的 EFMPDU，设置范围为 1 到 10，默认为 10. value：单位时间内发送个数
4	JX(config-ge-1/0/1)#**efm min-rate value** Example: JX(config-ge-1/0/1)#**efm min-rate 5**	（可选）如果本地 EFM 实体在发现超时时间内没有收到对端的 EFMPDU，认为发现连接失败，重启发现过程。设置范围为 2 到 30，默认为 5 秒。 value：超时时间
5	JX(config-ge-1/0/1)#**efm mode** { **active** | **passive** } JX(config-ge-1/0/1)#**exit** Example: JX(config-ge-1/0/1)#**oam active**	配置 EFM 的工作模式。 配置时至少有一端为主动模式，否则链路检测无法进行。 active：主动模式，接口主动发送 OAM PDU（Protocol Data Unit，协议数据单元），从而发起对端发现或远端环回过程 passive：被动模式，接口被动等待对端发送的 OAM PDU
6	JX(config-ge-1/0/1)#**exit**	进入全局配置模式。
7	JX(config)#**efm** { **enable** | **disable** } Example: JX(config-ge-1/0/1)#**efm enable**	使能接口的 EFM OAM 功能。 enable：使能链路的 EFM OAM 功能 disable：禁用链路的 EFM OAM 功能

7.2.5 配置 EFM 端口环回功能

配置 OAM 远端环回功能

OAM 提供链路层远端环回机制，可用于链路错误定位和性能以及质量测试。当处于链路环回状态时，设备将该链路收到的除了 OAM 报文外的所有报文环回到对端设备。本端设备通过 OAM 远端环回命令发起或者关闭远端环回，对端设备则通过环回配置命令控制是否响应环回命令。

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**configure**	进入全局配置模式。
2	JX(config)#**interface** *interface-type interface-number*	进入物理接口配置模式。
3	JX(config-ge-1/0/1)#**efm remote-loopback timeout value** Example: JX(config-ge-1/0/1)#**efm remote-loopback timeout 5**	（可选）本地等待响应的时间即为此响应超时时间，如果在此时间内没有收到对方以处于环回状态的响应，设置将失败。设置范围为 1 到 10 秒，默认为 10s value：超时时间
4	JX(config-ge-1/0/1)#**efm remote-loopback** { **supported** | **unsupported** } Example: JX(config-ge-1/0/*)#**efm min-rate 5**	配置环回功能支持与否，默认为不支持 supported ： 支持 unsupported：不支持
5	JX(config-ge-1/0/1)#**efm remote-loopback start holdtime** { <0-1000> | **default** } Example: JX(config-ge-1/0/1)#**efm remote-loopback start holdtime 100**	配置环回功能开始，为避免由于用户忘记停止 EFM 远端环回而造成链路长时间无法正常转发业务数据， EFM 远端环回具有超时自动取消功能。holdtime 表示远端环回持续时间。缺省情况下，远端环回的持续时间是 20 分钟，到达此时间，远端环回自动取消。
6	JX(config-ge-1/0/1)#**efm mode remote-loopback stop** Example: JX(config-ge-1/0/1)#**efm mode remote-loopback stop**	配置环回功能结束

7.2.6 配置 EFM 链路性能监测功能

配置设备 EFM 链路性能监测

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**configure**	进入全局配置模式。
2	JX(config)#**interface** *interface-type interface-number*	进入物理接口配置模式。

步骤	配置	说明
3	JX(config-ge-1/0/*)#**efm link-monitor { supported | unsupported }** Example: JX(config-ge-1/0/*)#**efm link-monitor supported**	使能端口的 efm 链路性能监测 使能接口的 EFM 协议后，默认为支持链路性能检测，默认每 100ms 采集一次数据，CPU 性能限制的设备每秒遍历所有使能 dot3ah 协议的端口采集一次数据 supported ：支持 unsupported：不支持
4	JX(config-ge-1/0/*)#**efm link-monitor error-frame threshold** *hold-value* **window** *window-value*	frame ：错误帧的检测默认为使能状态,窗口大小默认为 10s（即 100ms 的 100 倍），门限大小默认为 1 个错误帧，即默认当每 10s 检测到 1 个错误帧就会检测到一个错误帧的链路错误。 hold-value：错误帧事件阈值，取值范围是 1～65535，整数形式，单位是帧数 window-value：错误帧事件的监控窗口，取值范围是 10-600，整数形式，默认值为 100，单位 100ms 的倍数
5	JX(config-ge-1/0/*)#**efm link-monitor error-frame-period threshold** *hold-value* **window** *window-value*	配置错误帧周期的窗口与门限 hold-value：错误帧周期事件阈值，取值范围是 1～65535，整数形式，单位是帧数 window-value：错误帧周期事件的监控窗口，取值范围是 1～65535，整数形式，默认值为 1，单位 10,000 帧的倍数的倍数
6	JX(config-ge-1/0/*)#**efm link-monitor error-frame-second threshold** *hold-value* **window** *window-value*	配置错误帧秒的窗口与门限 hold-value：错误帧周期事件阈值，取值范围是 1～65535，整数形式，单位是帧数 window-value：错误帧秒事件的监控窗口，取值范围是 1～65535，整数形式，默认值为 1000，单位 100 毫秒的倍数

步骤	配置	说明
7	JX(config-ge-1/0/*)#**efm link-monitor trigger { error-down | trap | all }**	本地发生四种链路性能错误或者两种种紧急链路故障（Link Fault，Critical Event）时，或者收到远端的这几种相同的故障消息时，可以在本地通过网管 Trap 向服务器告警或者直接关闭此接口（紧急事件不支持），或者同时执行上述操作，也可以任何操作都不执行，默认为不执行任何操作

7.2.7 配置 EFM 链路故障检测功能

序号	检查项	说明
1	JX#**configure**	进入全局配置模式。
2	JX(config)#**interface** *interface-type interface-number* Example: JX(config)#**interface** ge **1/0/1**	进入物理接口配置模式。 interface-type：接口类型 interface-number：接口号
3	JX(config-ge-1/0/1)#**efm critical-event { supported | unsupported }** Example: JX(config-ge-1/0/1)#**efm critical-event supported**	使能端口的 efm 紧急事件监测 supported ：支持 unsupported：不支持

7.2.8 检查配置

配置完成后，请在设备上执行以下命令检查配置结果。

序号	检查项	说明
1	JX#**show efm session { interface** *interface-type interface-number* **}**	显示两端 EFM 会话信息
2	JX#**show efm interface** *interface-type interface-number*	显示接口 EFM 统计信息
3	JX#**show efm fault-logs { interface** *interface-type interface-number* **}**	显示两端 EFM 错误日志信息

7.2.9 维护

用户可以通过以下命令，维护设备 EFM OAM 特性的运行情况和配置情况。

命令	描述
JX(config-ge-1/0/*)#**efm fault-logs clear all**	清除 EFM OAM 链路事件日志信息。

7.3 故障转移

7.3.1 简介

故障转移功能提供了一种接口联动方案，可以扩展链路备份的范围，即通过监控上行链路并对下行链路进行同步设置，将上、下行接口加入到一个故障转移组中，使上层设备的故障迅速传达给下层，从而触发主备切换。故障转移功能可避免因上行链路故障无法被下层设备感知而出现的流量丢失。

一旦上行接口全部故障，则下行接口就会被置为 Down 状态，并且只要有一个上行接口恢复后，下行接口就将恢复 Up 状态，从而及时的将上行链路的故障情况通知到下层设备。下行接口故障时不影响上行接口。

7.3.2 配置准备

场景

中间设备上行链路故障时，如无法及时通知下层设备，会导致流量不能切换到备份路径，从而产生流量中断。

故障转移特性将中间设备的上行接口和下行接口加入同一故障转移组，并实时监控上行接口，当上行接口全部故障时，使上层设备的故障迅速传达给下层，保证主链路向备份链路的快速切换，以尽可能减少流量丢失。

前提

无

7.3.3 故障转移功能的缺省配置

设备上故障转移功能的缺省配置如下。

功能	缺省值
故障转移组	无
接口故障处理动作	无
故障转移组 Trap 告警功能	禁止

7.3.4 配置故障转移

说明

故障转移支持物理接口及聚合组接口配置。

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**monitor-link group** *group-number*	创建转移组并使能故障转移功能。
3	JX(config-monitorlink-*)#**snmp-trap** { **enable** | **disable** }	配置故障转移上报 Trap 功能。
5	JX(config-monitorlink-*)#**add interface** *interface-type interface-number* **role uplink**	配置 uplink
6	JX(config-monitorlink-*)#**add interface** *interface-type interface-number* **role downlink**	配置 downlink

说明

一个故障转移组中可以有多个上行接口，只要有一个上行接口为 Up 状态就不会发生故障转移；只有当全部上行接口都为 Down 的状态时才发生故障转移。

在全局节点下，使用 **remove interface** *interface-type interface-number* 命令从故障转移组中删除一个接口。

在物理层接口节点下，使用 **no monitor-link group** *group-number* 命令从故障转移组中删除一个接口。

7.3.5 检查配置

配置完成后，请在设备上执行以下命令检查配置结果。

序号	检查项	说明
1	JX#**show monitor-link group** [*group-number*]	查看故障转移组配置和状态信息。

7.3.6 配置故障转移示例

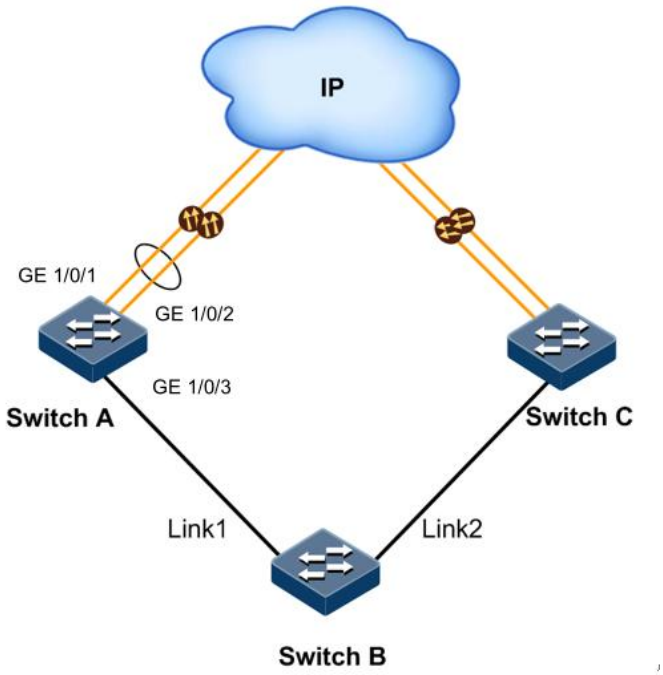
组网需求

如下图所示，为提高网络可靠性，Switch B 通过 Link1 和 Link2 两条链路分别连接到 Switch A 和 Switch C，Link1 为主链路，Link2 为备份链路，只有在 Link1 故障时，才启用 Link2 转发数据。

Switch A 和 Switch C 上行通过链路聚合的方式与上级网络相连，在 Switch A 或 Switch C 上行链路全部故障时，需要及时让 Switch B 感知，从而及时将流量切换到备份链路。

所以需要在 Switch A 和 Switch C 上部署故障转移特性。

图 7-2 故障转移应用组网示意图



配置步骤

配置 Switch A 和 Switch C 的故障转移功能，配置步骤相同，以配置 Switch A 为例。

步骤 1 创建链路聚合组 1，并将上行口 GE 1/0/1 和 GE 1/0/2 加入到链路聚合组中。

```
JX#config
JX(config)#int eth-trunk 1
JX(config-eth-trunk-1)#add interface ge 1/0/1
JX(config-eth-trunk-1)#add interface ge 1/0/2
```

步骤 2 创建故障转移组 1，将链路聚合组接口加入到故障转移组中。

```
JX(config)#monitor-link group 1
JX(config-monitorlink-1)#add interface eth-trunk 1 role uplink
```

步骤 3 将下行接口 GE 1/0/3 加入到故障转移组中。

```
JX(config-monitorlink-1)#add interface ge 1/0/3 role downlink
```

检查结果

以 Switch A 为例，通过 **show monitor-link group** 查看故障转移组配置是否正确。

```
SwitchA#show monitor-link group 1
    Mlink group 1 :
    ------------------------------------------------------------------------
-----------
     Snmp trap          : enable
     HoldOff time       : 3
     Uplink-select      : first-up
     Member            Role         State    Status   Linkstate
     ge-1/0/3          downlink      forward  active    up/up
     eth-trunk-1       uplink        forward  active    up/up
    ------------------------------------------------------------------------
-----------
```

Switch A 上行链路均故障后，再次通过 **show monitor-link group** 查看故障转移组配置可以看到已经发生故障转移。

```
SwitchA#show link-state-tracking group 1
```

Mlink group 1 :

Snmp trap : enable

HoldOff time : 3

Uplink-select : first-up

Member	Role	State	Status	Linkstate
ge-1/0/3	downlink	block	active	up/down
eth-trunk-1	uplink	block	active	up/down

7.4 VRRP

7.4.1 简介

内部网络中的所有主机都设置一条相同的缺省路由，指向出口网关，实现主机与外部网络的通信。如果网关发生故障，以该网关为缺省路由的主机将无法与外部进行通信。

VRRP（Virtual Router Redundancy Protocol，虚拟路由冗余协议）是为消除在静态缺省路由环境下，缺省路由设备单点故障引起的网络失效而设计的主备模式协议，有效避免单一链路发生故障引起的网络中断，且无需修改相应的路由协议等配置。

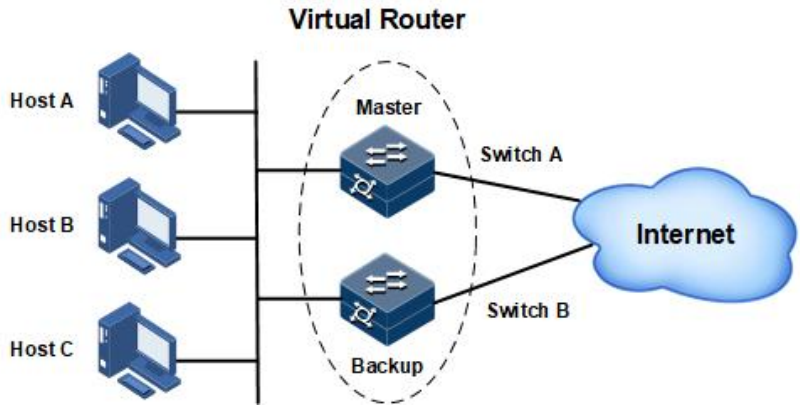
VRRP 备份组

VRRP 协议将局域网内的两台或多台路由设备虚拟成一个设备（包括一个 Master 设备和若干个 Backup 设备），组成一个 VRRP 备份组。VRRP 备份组功能上相当于一台虚拟路由器，对外提供虚拟路由设备的 IP 地址。

- Master 设备：拥有对外 IP 地址的设备如果工作正常，则为 Master 设备，或者通过算法选举产生。Master 设备实现针对虚拟路由设备 IP 地址的各种网络功能。VRRP 协议运行时 Master 设备定时向 Backup 设备发送 VRRP 通告报文，表示其工作正常。
- Backup 设备：不拥有对外 IP 地址或者优先级低的设备，则为 Backup 设备。只接收通告报文，不进行发送。当 Master 设备失效时，从 Backup 设备中重新选举 Master 设备，并接管原先 Master 设备的网络功能。

配置 VRRP 协议时需要为每台路由设备配置备份组号和优先级，使用备份组号将设备进行分组，具有相同备份组号的设备为同一个组。同一组中的设备通过优先级来选举 Master 设备，优先级大的为 Master 设备，如下图所示。

图 7-3 VRRP 原理示意图



在上图中，Switch A 和 Switch B 组成一个虚拟路由设备，此设备拥有自己的 IP 地址。局域网内的主机将虚拟路由设备设置为缺省网关。Switch A 和 Switch B 中优先级最高的设备为 Master 设备，承担网关的功能，另一台设备为 Backup 设备。

VRRP 工作模式

在 VRRP 备份组中，设备具有以下两种工作模式：

- 非抢占模式：只要 Master 设备正常，Backup 设备即使被配置了更高的优先级也不会成为 Master 设备。
- 抢占模式：VRRP 备份组中，设备一旦发现自己的优先级比当前 Master 设备的优先级高，就会对外发送 VRRP 通告报文，导致备份组内设备重新选举 Master，并最终取代原有的 Master 设备。相应地，原来的 Master 设备将会变成 Backup 设备。

VRRP 工作过程

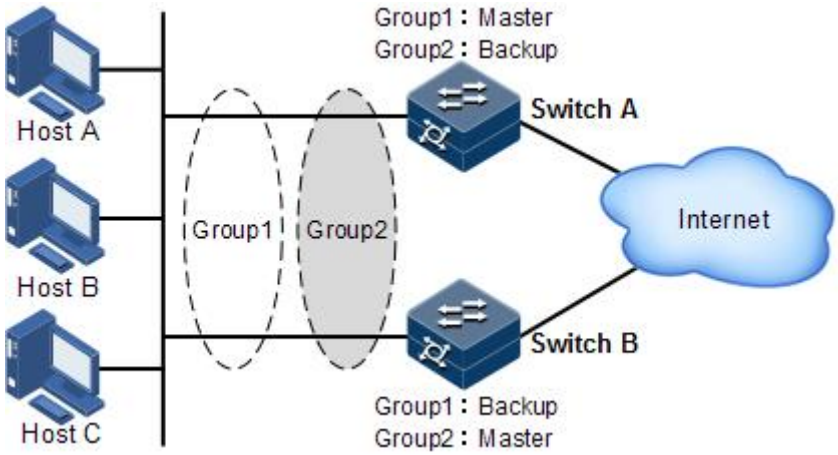
VRRP 的工作过程如下：

1. 设备使能 VRRP 功能后，通过优先级来确定自身在 VRRP 备份组中的角色。优先级高的为 Master，优先级低的为 Backup。Master 定期发送 VRRP 通告报文，通知备份组内的其他设备自身工作正常；Backup 则启动定时器等待通告报文的到来。
2. 抢占模式下，Backup 收到 VRRP 通告报文后，将自身优先级与通告报文中的优先级进行比较。如果小于通告报文中的优先级，则保持 Backup 状态；否则将成为 Master 设备。
3. 非抢占模式下，只要 Master 设备没有出现故障，备份组中的设备状态始终保持不变。
4. 如果 Backup 设备的定时器超时，则认为 Master 设备已经无法正常工作，此时 Backup 设备会认为自己是 Master 设备，并对外发送 VRRP 通告报文，进行新一轮 Master 设备的选举。新选举出来的 Master 设备将接管原先 Master 设备的网络功能，承担报文转发功能。

负载分担

VRRP 负载分担是指建立两个或者多个 VRRP 备份组，多台设备同时承担业务。允许一台设备为多个备份组作备份，在不同备份组中具有不同的优先级。通过多个虚拟设备可以实现负载分担。各个备份组的 Master 可以不同，如下图所示。

图 7-4 VRRP 负载分担原理示意图



其中：

- Switch A 为 VRRP 备份组 1 的 Master 设备，同时为备份组 2 的 Backup 设备。
- Switch B 为 VRRP 备份组 2 的 Master 设备，同时为备份组 1 的 Backup 设备。

网络内部分主机使用备份组 1 作为网关，例如 Host A 和 Host B，部分主机使用备份组 2 作为网关，例如 Host C。这样，既达到相互备份的目的，又可以分担网络流量。

7.4.2 配置准备

场景

在网络环境中，通常会为同局域网内的所有主机配置一条指向出口网关的缺省路由，实现局域网内主机与外部网络之间的通信。如果该网关发生故障，则局域网内主机与外部网络的通信就会中断。

VRRP 技术将多台路由器组合到一起形成备份组，用户通过为备份组配置虚拟 IP 地址，使用时只需要将局域网主机的缺省网关配置为备份组的虚拟 IP 地址，即可实现局域网内主机与外部网络之间的通信。

VRRP 功能的部署可以提高网络的可靠性，有效避免因为单一链路中断而造成的网络中断的问题，也无需因为链路中断而更改路由配置。

前提

无

7.4.3 VRRP 的缺省配置

设备上 VRRP 的缺省配置如下。

功能	缺省值
VRRP 功能状态	使能
VRRP 的 Trap 功能状态	去使能
接口下 VRRP 组	无
VRRP 备份组描述	无
VRRP 备份组功能状态	去使能
设备优先级	100
IP 拥有者优先级	255
VRRP 工作模式	抢占模式
VRRP 组的抢占延时	0s
VRRP 组的报文发送间隔	1s

7.4.4 配置 VRRP 备份组

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**configure**	进入全局配置模式。

步骤	配置	说明
2	JX(config)#**interface vlan** *vlan-id*	进入 VLAN 接口配置模式。
3	JX(config-vlan*)#**vrrp** *group-id* **associate-address** *ipv4-address*	配置 VRRP 备份组的虚 IPv4 地址
4	JX(config-vlan*)#**vrrp** group-id **advertise-interval** { **interval** | **default** }	（可选）配置协议包发送间隔 group-id：VRRP 实例号。 (interval|default)：协议包发送间隔时间，单位为秒，默认值为 1
5	JX(config-vlan*)#**vrrp group-id check-ttl** { **enable** | **disable** }	可选）配置检测协议包 ttl 使能 group-id：VRRP 实例号 enable:使能 disable：去使能 默认使能
6	JX(config-vlan*)#**vrrp** group-id **priority** { value | **default** }	（可选）VRRP 实例优先级 group-id：VRRP 实例号 (value|default)：优先级大小，默认为 100.
7	JX(config-vlan*)#**vrrp group-id track bfd-session** *bfd-id* [{ **increased** | **reduced** } { **value** | **default** }]	（可选）绑定 bfd group-id：VRRP 实例号 bfd-id:bfd 实例号 （可选）increased：提高优先级 （可选）increased：降低优先级 （可选）value|default：配置当 bfd 状态 down 时，当前 vrrp 实例需减少的优先级大小，默认为 10
8	JX(config-vlan*)#**vrrp** *group-id* **track interface vlan** *vlan-id*	（可选）配置绑定关联 vlan 接口 group-id：VRRP 实例号 vlan-id：绑定的 vlan 索引
9	JX(config-vlan*)#**vrrp** group-id **track admin-vrrp interface vlan** *vlan-id vrrp-id*	（可选）配置绑定关联管理 vrrp 组 group-id：VRRP 实例号 vlan-id：绑定的 vlan 索引 vrrp-id：管理 vrrp 实例号
10	JX(config-vlan*)#**vrrp** group-id **send-packet-mode** { **v2** | **v3** | **v2v3** }	（可选）配置发送的 vrrp 报文版本 group-id：VRRP 实例号 v2：发送 v2 报文 v3：发送 v3 报文 v2v3:同时发送 v2 和 v3 报文

步骤	配置	说明
11	JX(config-vlan*)#**vrrp group-id authentication-mode { simple | md5 } { cipher | plain } key**	（可选）vrrp 认证，只对 v2 实例生效 group-id：VRRP 实例号 simple：简单字符认证 md5：md5 认证 cipher：密文显示 plain: 明文显示 key：认证字符
12	JX(config-vlan*)#**vrrp** *group-id* **holding-multiplier [holding-multiplier-value]**	（可选）vrrp backup 超时倍数 group-id：VRRP 实例号 holding-multiplier-value：超时倍数（3-10），默认为 3
13	JX(config-vlan*)#**vrrp** *group-id* **role admin**	（可选）配置 vrrp 角色管理 group-id：VRRP 实例号
14	JX(config-vlan*)# **vrrp** *member-group-id* **track admin-vrrp interface vlan** *vlan-id* **vrid** *group-id*	（可选）配置 vrrp 角色管理 group-id：管理 VRRP 实例号 vlan-id：管理的 vlan 接口索引 member-group-id：关联 vrrp 实例号

7.4.5 配置 VRRP6 备份组

步骤	配置	说明
1	JX#**configure**	进入全局配置模式。
2	JX(config)#**interface vlan** *vlan-id*	进入 VLAN 接口配置模式。
3	JX(config-vlan*)#**vrrp6** *group-id* **associate-address** *ipv6-address*	配置 VRRP 备份组的虚 IPv6 地址
4	JX(config-vlan*)#**vrrp-ipv6** *group-id* **advertise-interval { interval | default }**	（可选）配置协议包发送间隔 group-id：VRRP6 实例号。 (interval|default)：协议包发送间隔时间，单位为秒，默认值为 1
5	JX(config-vlan*)#**vrrp-ipv6 group-id check-ttl { enable | disable }**	（可选）配置检测协议包 ttl 使能 group-id：VRRP6 实例号 enable:使能 disable：去使能 默认使能

步骤	配置	说明
6	JX(config-vlan*)#**vrrp-ipv6** *group-id* **priority** { **value** | **default** }	（可选）VRRP6 实例优先级 group-id：VRRP 实例号 (value|default)：优先级大小，默认为 100.
7	JX(config-vlan*)#**vrrp-ipv6 group-id track bfd-session** *bfd-id* [{ **increased** | **reduced** } { **value** | **default** }]	（可选）绑定 bfd group-id：VRRP6 实例号 bfd-id:bfd 实例号 （可选）increased：提高优先级 （可选）increased：降低优先级 （可选）value|default：配置当 bfd 状态 down 时，当前 vrrp 实例需减少的优先级大小，默认为 10
8	JX(config-vlan*)#**vrrp-ipv6** *group-id* **track interface vlan** *vlan-id*	（可选）配置绑定关联 vlan 接口 group-id：VRRP6 实例号 vlan-id：绑定的 vlan 索引
9	JX(config-vlan*)#**vrrp-ipv6** *group-id* **track admin-vrrp interface vlan** *vlan-id vrrp-id*	（可选）配置绑定关联管理 vrrp 组 group-id：VRRP6 实例号 vlan-id：绑定的 vlan 索引 vrrp-id：管理 vrrp 实例号
10	JX(config-vlan*)#**vrrp-ipv6** *group-id* **holding-multiplier** [**holding-multiplier-value**]	（可选） backup 超时倍数 group-id：VRRP6 实例号 holding-multiplier-value：超时倍数（3-10），默认为 3
11	JX(config-vlan*)#**vrrp6vrrp-ipv6** *group-id* **role admin**	（可选）配置 vrrp 6 角色管理 group-id：VRRP 实例号
12	JX(config-vlan*)#**vrrp-ipv6 admin-vrrp** *admin-group-id* **associate-vrrp interface vlan** *vlan-id member-group-id*	（可选）配置 vrrp6 角色管理 group-id：管理 VRRP 实例号 vlan-id：管理的 vlan 接口索引 member-group-id：关联 vrrp 实例号

7.4.6 配置 VRRP 的 Trap 功能

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**configure**	进入全局配置模式。

步骤	配置	说明
2	JX(config)#**vrrp snmp-trap** { **enable** | **disable** }	使能/去使能 VRRP 的 Trap 功能。

7.4.7 配置 VRRP 监视接口

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**configure**	进入全局配置模式。
2	JX(config)#**interface vlan** *vlan-id*	进入 VLAN 接口配置模式。
3	JX(config-vlan*)#**vrrp** *group-id* **track** { *interface-type interface-number* | **vlan** *vlan-id* } [{ **increased** | **reduced** } { **value** | **default** }]	配置 VRRP 监视接口功能。

说明

reduced *priority*：当被监视接口从 UP 状态变为 DOWN 状态时，优先级减少的数值，整数形式，取值范围是 1～255，不配置该参数，设备在备份组中的优先级在原有基础上降低 10，减少后的优先级范围是 1～254

当被监视接口从 DOWN 状态变为 UP 状态时，优先级恢复原来的数值，建议在 Master 设备进行配置。

7.4.8 配置 BFD for VRRP

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**configure**	进入全局配置模式。
2	JX(config)#**interface vlan** *vlan-id*	进入 VLAN 接口配置模式。
3	JX(config-vlan*)#**vrrp group-id track bfd-session** *bfd-id* [{ **increased** | **reduced** } { **value** | **default** }]	配置 VRRP 备份组对 BFD 会话进行监测，以达到快速切换的目的。

说明

increased *priority*：配置当被监视的 BFD 会话状态变为 DOWN 时，优先级增加的数值，整数形式，取值范围是 1～255，增加后的优先级范围是 1～254。如果从 DOWN 状态变为 UP 状态，则优先级恢复原来的数值。建议在 Backup 设备上进行配置。

reduced *priority*：配置当被监视的 BFD 会话状态变为 DOWN 时，优先级降低的数值，整数形式，取值范围是 1～255，减少后的优先级范围是 1～254。如果从 DOWN 状态变为 UP 状态，则优先级恢复原来的数值。建议在 Master 设备上进行配置。

7.4.9 检查配置

配置完成后，请在设备上执行以下命令检查配置结果。

序号	检查项	说明
1	JX#**show vrrp session { role { admin | member | normal } }**	显示 VRRP 基本状态信息。
2	JX#**show vrrp admin-vrrp**	显示管理 VRRP 信息
3	JX#**show vrrp statistics**	显示 VRRP 统计信息
4	JX#**show vrrp interface vlan** *vlan-id*	显示 VLAN 接口上的 vrrp 信息。
5	JX#**show vrrp associate interface vlan** *vlan-id group-id*	显示接口上指定 group id 的 VRRP 信息
6	JX#**show vrrp binding { admin-vrrp interface vlan** *vlan-id* group-id }	显示 VRRP 管理组绑定信息

7.5 CFM

7.5.1 简介

CFM 是一种网络级以太网 OAM 技术，针对网络实现端到端的连通性故障检测、故障通告、故障判定和故障定位功能。用于对 EVC（Ethernet Virtual Connection，以太网虚连接）进行主动的故障诊断，并通过使用故障管理功能有效降低网络维护成本，提高以太网的可维护性。

交换机设备遵循 IEEE 802.1ag 的 CFM（Connectivity Fault Management，连通错误管理）协议和 ITU-T 的 Y.1731 协议。

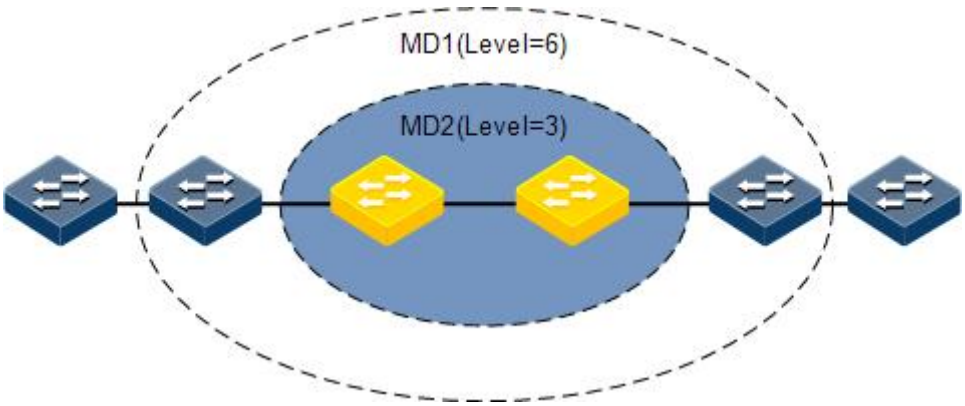
CFM 由如下组件组成：

MD

维护域 MD（Maintenance Domain，又称 MEG，Maintenance Entity Group，维护实体组）是一个运行 CFM 功能的网络，它确定了进行 OAM 管理的网络范围。维护域具有级别属性，共分为 8 级（0～7），数字越大表示维护域级别越高，对应维护域的范围越大。低级别 MD 的协议报文进入高级别的 MD 后被丢弃，（如果高级别 MD 中不存在 MEP，而只有 MIP，则报文能够通过。）高级别 MD 的协议报文可以穿越低级别的 MD。在同一 VLAN 范围内，不同的维护域之间可以相邻、嵌套，但不能交叉。

如下图所示，MD2 包含在 MD1 中，MD1 的协议报文需要穿越 MD2。因此，将 MD1 的级别配置为 6，MD2 的级别配置为 3，这样 MD1 内的协议报文就可以穿越 MD2 实现整个 MD1 的连通性故障管理，而 MD2 的协议报文不会扩散到 MD1 中。MD2 为服务器层，MD1 为客户层。

图 7-5 不同级别 MD 网络示意图



MA

MA（Maintenance Association，维护联盟）是 MD 的一部分，一个 MD 可划分为一个或多个 MA。MA 以“MD 名称＋MA 名称”来标识。

MA 可以服务于指定的 VLAN，也可以不服务于任何 VLAN，分别称为带 VLAN 属性和不带 VLAN 属性的 MA。

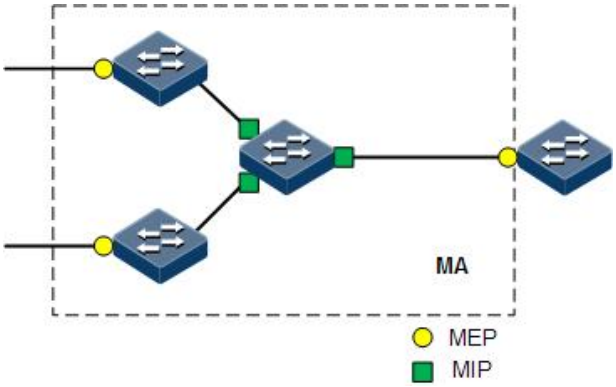
MEP

如下图所示，MEP（Maintenance association End Point，维护端点）确定了 MA 的边界，以“MEP ID”来标识。MEP 具有方向性，分为内向 MEP 和外向 MEP 两种：

内向 MEP 通过除其所在的接口以外的所有接口向外发送 CFM 协议报文，即在其所属 MA 所服务的 VLAN 中进行广播。

外向 MEP 则直接通过其所在的接口向外发送 CFM 协议报文。

图 7-6 不同级别 MD 网络示意图



MIP

如上图所示，MIP（Maintenance association Intermediate Point，维护中间点）位于 MA 的内部，不能主动发出 CFM 协议报文，但可以处理和响应 CFM 协议报文。MIP 由设备自动创建，可以配合 MEP 完成类似于 ping 和 tracert 的功能。

MP

MEP 和 MIP 统称为维护节点 MP（Maintenance Point）。

CFM 能够提供以下 OAM 功能：

故障检测功能

故障检测功能是指使用 CC（Continuity Check，连续性检测）协议来检测一个以太网虚连接的连通性，确定 MP 之间的连接状态。该功能通过 MEP 周期性地发送 CCM（Continuity Check Message，连续性检测报文）实现，同一服务实例内其他 MEP 接收该报文，由此确定 RMEP 的状态。如果设备故障或者链路中间配置错误，会导致 MEP 无法正常接收和处理 RMEP 发送的 CCM。如果 MEP 在 3.5 个 CCM 间隔周期内未收到远端的 CCM 报文，则认为链路存在故障，会根据告警优先级配置发送故障告警。

故障确认功能

故障确认功能利用 LB（LoopBack，环回功能），通过源 MEP 发送 LBM 和目的 MP 回应 LBR 以确定两个 MP 之间的连通性。源 MEP 发送 LBM 给要进行故障确认的 MP，当该 MP 收到 LBM 报文后，发送一个 LBR 给源 MEP，如果源 MEP 接收到 LBR，则确认路径是连通的，如果源 MEP 没有接收到 LBR，则确认存在连通性故障。

故障定位功能

故障定位功能利用 LT（LinkTrace，链路跟踪），通过源 MEP 发送 LTM 给目的 MP，LTM 传输路径上的每个 MP 设备都会回应 LTR 给源 MEP，通过记录有效的 LTR 和 LTM 定位故障点。

告警抑制功能

告警抑制功能用来减少 MEP 故障告警的数量。如果 MEP 在 3.5 个 CCM 报文发送周期内未收到远端 MEP 发来的 CCM 报文，便立刻开始周期性地发送 AIS（Alarm Indication Signal，告警指示信号）报文，该报文的发送方向与 CCM 报文相反。其它

MEP 在收到 AIS 报文后，会抑制本端的故障告警，并继续发送 AIS 报文。此后，如果 MEP 收到了 CCM 报文，便停止发送 AIS 报文并恢复故障告警。AIS 报文是组播报文。

单向丢包测试功能

单向丢包测试（Loss Measurement，LM）功能用来检测 MEP 之间的单向丢包情况，其实现方式是：由源 MEP 发送 LMM（Loss Measurement Message，丢包测量报文）报文给目标 MEP，目标 MEP 收到该报文后，会发送 LMR（Loss Measurement Reply，丢包测量应答）报文给源 MEP，源 MEP 则根据两个连续的 LMR 报文来计算源 MEP 和目标 MEP 间的丢包数，即源 MEP 从收到第二个 LMR 报文开始，根据本 LMR 报文和前一个 LMR 报文的统计计数来计算源 MEP 和目标 MEP 间的丢包数。LMM 报文和 LMR 报文都是单播报文。

帧时延测试功能

帧时延测试（Delay Measurement，DM）功能用来检测 MEP 之间报文传输的时延情况，分为单向时延测试、双向时延测试两种，目前仅支持双向时延测试。双向时延测试功能的实现方式是：源 MEP 发送 DMM（Delay Measurement Message，时延测量报文）报文给目标 MEP，该报文中携带有其发送时间。目标 MEP 收到该报文后记录其接收时间，然后再发送 DMR（Delay Measurement Reply，时延测量应答）报文给源 MEP，该报文中携带有 DMM 报文的发送和接收时间，以及 DMR 报文的发送时间。源 MEP 收到 DMR 报文后记录其接收时间，并据此计算出链路传输的时延和抖动。

总之，CFM 实现了在端到端服务层面的 OAM 技术，降低了服务提供商的运行维护成本，在一定程度上可以提高服务提供商的竞争优势。

7.5.2 配置准备

场景

为拓展以太网技术在电信级网络中的应用，以太网需要达到与电信级传送网相同的服务水平。CFM 通过为电信级以太网提供全面的 OAM 工具解决了此问题。

前提

在配置 CFM 之前，需完成以下任务：

- 连接接口并配置接口的物理参数，使接口的物理层状态为 Up。
- 创建 VLAN。
- 将接口加入 VLAN。

7.5.3 缺省配置

设备上 CFM 的缺省配置如下。

功能	缺省值
CFM 全局功能状态	禁用

功能	缺省值
CCM 报文发送时间间隔	1s
MEP 发送 CCM 报文	禁用
CFM OAM 报文优先级	0

7.5.4 配置 CFM 基本功能

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**configure**	进入全局配置模式。
2	JX(config)#**cfm start**	全局使能 CFM 功能。
3	JX(config)#**cfm md** *md-name* **level** *level* **format** { **dns** *dns-format-string* | **mac** *mac-format-string* | **string** string | **none** } JX(config)#**cfm md** *md-name* [**level** *level*]	创建 MD，并进入 MD 视图。指定了 MD 的名称，指定发送 CCM 报文中使用的 MD 名称的格式和内容，维护域名称和发送报文中使用的 MD 名称内容必须全局唯一，否则将导致维护域配置失败。
4	JX(config-cfm-md-*)#**mip create-type** { **default** | **explicit** | **none** }	配置当前 MA 的 MIP 生成规则。 default：如果端口不存在更高级别的 MEP，并且不存在更低级别的 MIP，则可以在该端口上创建 MIP。在本类型下，接口上没有配置 MEP 也可创建 MIP。 explicit：如果端口存在更低级的 MEP 但不存在更高级别的 MEP，而且不存在更低级别的 MIP，则可以创建 MIP。在本类型下，接口上只有已配置了更低级别的 MEP 才可能创建 MIP。 none：如果端口的 MIP 的生成规则为 none 类型，即不自动创建 MIP。
5	JX(config-cfm-md-*)#**ma** *ma-name* **format** { **string** | **icc** } *format-string*	在 MD 内创建 MA，并进入 MA 视图。指定了 MA 的名称以及指定发送 CCM 报文中填充的 MA 名称的格式和内容。同一 MD 内，MA 的名称不能重复。
6	JX(config-cfm-md-**ma*-*)#**map vlan** *vlan-id*	配置当前 MA 关联的 VLAN。

步骤	配置	说明
7	JX(config-cfm-md-*-*ma-*)#**mep mep-id** *mep-id* **interface** *interface-type interface-number* { **inward** | **outward** }	在 MA 内创建 MEP。 在同一个 MA 内，对创建 MEP 的数量和类型要求如下： inward 型普通 MEP 和 outward 型普通 MEP 不能同时存在。 只能创建 1 个 outward 接口型 MEP，可以创建多个 inward 接口型 MEP，但是同一个接口上只能创建 1 个 inward 接口型 MEP。

7.5.5 配置故障检测功能

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**configure**	进入全局配置模式。
2	JX(config)#**cfm md** *md-name*	进入 MD 视图。
3	JX(config-cfm-md-*)#**ma** *ma-name*	进入 MA 视图。
4	JX(config-cfm-md-*-*ma-*)#**ccm interval** { **3.3ms** | **10ms** | **100ms** | **1s** | **10s** | **1min** | **10min** | **default** }	配置当前 MA 内 MEP 发送 CCM 消息的时间间隔。 部署 1s 以下的发送 CCM 消息的时间间隔，需要交换芯片支持 CFM。
5	JX(config-cfm-md-*-*ma-*)#**ccm send** [**mep-id** *mep-id*] **enable**	使能 MEP 发送 CCM 报文。
6	JX(config-cfm-md-*-*ma-*)#**rmep mep-id** *mep-id* **mac** *mac-address*	配置静态远端 MEP。配合 CCM 报文检测功能使用。
7	JX(config-cfm-md-*-*ma-*)#**ccm send** [**mep-id** *mep-id*] **priority** { priority | **default** }	配置 MA 内 MEP 发送 CCM 报文的优先级。

7.5.6 配置故障确认功能

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**configure**	进入全局配置模式。
2	JX(config)#**cfm md** *md-name*	进入 MD 视图。
3	JX(config-cfm-md-*)#**ma** *ma-name*	进入 MA 视图。
4	JX(config-cfm-md-**ma*-*)#**ping mep-id** *mep-id* **rmep-id** *mep-id* [**count** *count* | **tlv-type** { **null** | **null-crc** | **prbs** | **prbs-crc** } | **tlv-len** *len* | **priority** *priority*]* JX(config-cfm-md-**ma*-*)#**ping mep-id** *mep-id* **mac** *mac-address* [**count** *count* | **tlv-type** { **null** | **null-crc** | **prbs** | **prbs-crc** } | **tlv-len** *len* | **priority** *priority*]* JX(config-cfm-md-**ma*-*)#**ping mep-id** *mep-id* **mac multicast** [**count** *count* | **tlv-type** { **null** | **null-crc** | **prbs** | **prbs-crc** } | **tlv-len** *len* | **priority** *priority*]*	执行二层 ping 功能，用于故障确认。

7.5.7 配置故障定位功能

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**configure**	进入全局配置模式。
2	JX(config)#**cfm md** *md-name*	进入 MD 视图。
3	JX(config-cfm-md-*)#**ma** *ma-name*	进入 MA 视图。
4	JX(config-cfm-md-**ma*-*)#**trace mep-id** *mep-id* **mac** *mac-address* [**ttl** *ttl* | **fdb** *<0-1>*]* JX(config-cfm-md-**ma*-*)# **trace mep-id** *mep-id* **rmep-id** *mep-id* [**ttl** *ttl* | **fdb** *<0-1>*]*	执行二层 Traceroute 功能，用于故障定位。

7.5.8 配置告警抑制功能

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**configure**	进入全局配置模式。
2	JX(config)#**cfm md** *md-name*	进入 MD 视图。
3	JX(config-cfm-md-*)#**ma** *ma-name*	进入 MA 视图。
4	JX(config-cfm-md-**ma-*)#**ais [mep** *mep-id* **] interval { 1s | 1min | default }**	配置当前 MA 内 MEP 向高级别 MA 内 MEP 发送 AIS 报文的时间间隔。
5	JX(config-cfm-md-**ma-*)#**ais [mep** *mep-id* **] md-level {** *level* **| defalut }**	配置当前 MA 内 MEP 发送 AIS 报文的级别。
6	JX(config-cfm-md-**ma-*)#**ais [mep** *mep-id* **] priority {** *priority* **| defalut }**	配置当前 MA 内 MEP 发送 AIS 报文的优先级。
7	JX(config-cfm-md-**ma-*)#**ais [mep** *mep-id* **] enable**	使能 MEP 发送 AIS 报文。

7.5.9 配置单向丢包测试功能

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**configure**	进入全局配置模式。
2	JX(config)#**cfm md** *md-name*	进入 MD 视图。
3	JX(config-cfm-md-*)#**ma** *ma-name*	进入 MA 视图。
4	JX(config-cfm-md-**ma-*)#**loss-measure mep-id** *mep-id* **rmep-id** *mep-id* **[interval { 100ms | 1s } | priority** *priority* **| count** *<1-100>* **]*** JX(config-cfm-md-**ma-*)#**loss-measure mep-id** *mep-id* **mac** *mac-address* **[interval { 100ms | 1s } | priority** *priority* **| count** *<1-100>* **]***	执行单向丢包测试功能，用于检测 MEP 之间的单向丢包情况。

7.5.10 配置双向时延测试功能

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**configure**	进入全局配置模式。
2	JX(config)#**cfm md** *md-name*	进入 MD 视图。

步骤	配置	说明
3	JX(config-cfm-md-*)#**ma** *ma-name*	进入 MA 视图。
4	JX(config-cfm-md-**ma*-*)#**delay-measure mep-id** *mep-id* **rmep-id** *mep-id* [**interval** { **100ms** | **1s** } | **priority** *priority* | **frame-len** *len* | **count** *count*]* JX(config-cfm-md-**ma*-*)#**delay-measure mep-id** *mep-id* **mac** *mac-address* [**interval** { **100ms** | **1s** } | **priority** *priority* | **frame-len** *len* | **count** *count*]*	执行双向时延测试功能，用于检测 MEP 之间报文传输的时延情况。

7.5.11 检查配置

配置完成后，请在设备上执行以下命令检查配置结果。

序号	检查项	说明
1	JX#**show cfm config**	查看 CFM 配置信息。
2	JX#**show cfm md**	查看 CFM MD 信息。
3	JX#**show cfm ma**	查看 CFM MA 信息。
4	JX#**show cfm mep**	查看 CFM MEP 信息。
5	JX#**show cfm rmep**	查看 CFM 远端 MEP 信息。
6	JX#**show cfm mip**	查看 CFM MIP 信息。

7.5.12 配置 CFM 示例

组网需求

图 7-7 由五台设备组成的网络被划分为 MD_A 和 MD_B 两个 MD，其级别分别为 5 和 3，各设备的所有端口都属于 VLAN 100，且各 MD 中的 MA 均服务于该 VLAN，并假定 Device A～Device E 的 MAC 地址依次为 00:03:56:00:00:01、00:03:56:00:00:02、00:03:56:00:00:03、00:03:56:00:00:04 和 00:03:56:00:00:05。

MD_A 的边界端口为 Device A 的 GE1/0/1、Device D 的 GE1/0/3 和 Device E 的 GE1/0/4，这些端口上都是内向 MEP；MD_B 的边界端口为 Device B 的 GE1/0/3 和 Device D 的 GE1/0/1，这些端口都是外向 MEP。

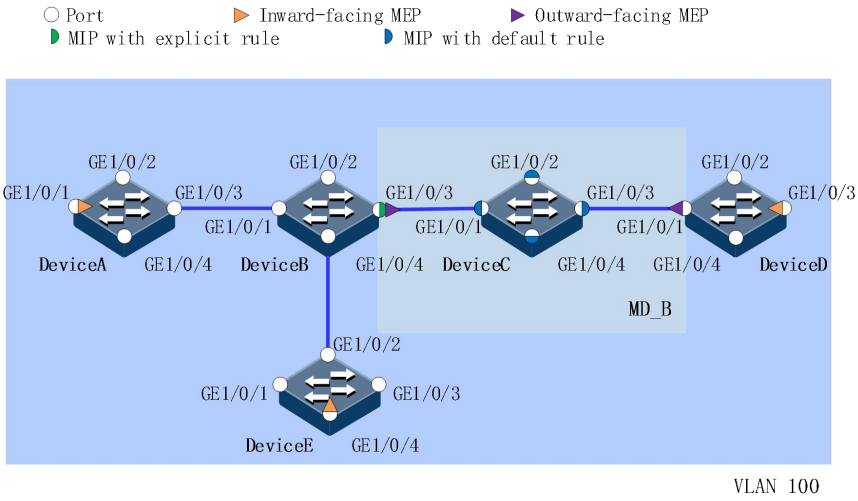
要求将 MD_A 的 MIP 规划在 Device B 上，并只在端口上有低级别 MEP 时配置。根据此规划，由于 Device B 的 GE1/0/3 上配置有 MD_B 的 MEP，因此在 Device B 上采用 Explicit 规则来创建 MD_A 的 MIP，其他 Device 上采用 None 规则。

要求将 MD_B 的 MIP 规划在 Device C 上，并在其所有端口上配置。根据此规划，在 Device C 上配置 MD_B 的 MIP，且其创建规则为 Default 规则，其他 Device 上采用 None 规则。

要求通过使用连续性检测功能来检测 MD_A 和 MD_B 中各 MEP 之间的连通状态，当检测到链路故障时，使用环回功能进行故障定位，并通过告警抑制功能和以太网告警抑制功能来减少故障告警的数量。

要求在获取到整个组网的状态后，分别使用链路跟踪功能、单向丢包测试功能、双向时延测试功能进行各种链路故障检测。

图 7-7 CFM 典型配置组网图



配置步骤

步骤 1 配置 VLAN 和端口。

请按照上图在各设备上分别创建 VLAN 100，并配置端口 GE1/0/1～GE1/0/4 都属于 VLAN 100。

步骤 2 开启 CFM 基本功能。

```
DeviceA#configure
DeviceA(config)#cfm start
DeviceA(config)#cfm md MD_A level 5 format none
DeviceA(config-cfm-md-MD_A)#mip create-type none
DeviceA(config-cfm-md-MD_A)#ma 1 format icc 1
DeviceA(config-cfm-md-MD_A-ma-1)#map vlan 100
DeviceA(config-cfm-md-MD_A-ma-1)#mep mep-id 1001 interface ge 1/0/1
inward
DeviceA(config-cfm-md-MD_A-ma-1)#end
DeviceB#configure
DeviceB(config)#cfm start
DeviceB(config)#cfm md MD_A level 5 format none
DeviceB(config-cfm-md-MD_A)#mip create-type explicit
DeviceB(config-cfm-md-MD_A)#ma 1 format icc 1
DeviceB(config-cfm-md-MD_A-ma-1)#map vlan 100
DeviceB(config-cfm-md-MD_A-ma-1)#end
DeviceB#configure
```

```
DeviceB(config)#cfm md MD_B level 3 format none
DeviceB(config-cfm-md-MD_B)#ma 2 format icc 2
DeviceB(config-cfm-md-MD_B-ma-2)#map vlan 100
DeviceB(config-cfm-md-MD_B-ma-2)#mep mep-id 2001 interface ge 1/0/3
outward
DeviceB(config-cfm-md-MD_B-ma-2)#end

DeviceC#configure
DeviceC(config)#cfm start
DeviceC(config)#cfm md MD_B level 3 format none
DeviceC(config-cfm-md-MD_B)#ma 2 format icc 2
DeviceC(config-cfm-md-MD_B-ma-2)#map vlan 100
DeviceC(config-cfm-md-MD_B-ma-2)#end

DeviceD#configure
DeviceD(config)#cfm start
DeviceD(config)#cfm md MD_A level 5 format none
DeviceD(config-cfm-md-MD_A)#mip create-type none
DeviceD(config-cfm-md-MD_A)#ma 1 format icc 1
DeviceD(config-cfm-md-MD_A-ma-1)#map vlan 100
DeviceD(config-cfm-md-MD_A-ma-1)#mep mep-id 4002 interface ge 1/0/3
inward
DeviceD(config-cfm-md-MD_A-ma-1)#end
DeviceD#configure
DeviceD(config)#cfm md MD_B level 3 format none
DeviceD(config-cfm-md-MD_B)#mip create-type none
DeviceD(config-cfm-md-MD_B)#ma 2 format icc 2
DeviceD(config-cfm-md-MD_B-ma-2)#map vlan 100
DeviceD(config-cfm-md-MD_B-ma-2)#mep mep-id 4001 interface ge 1/0/1
outward
DeviceD(config-cfm-md-MD_B-ma-2)#end

DeviceE#configure
DeviceE(config)#cfm start
DeviceE(config)#cfm md MD_A level 5 format none
DeviceE(config-cfm-md-MD_A)#mip create-type none
DeviceE(config-cfm-md-MD_A)#ma 1 format icc 1
DeviceE(config-cfm-md-MD_A-ma-1)#map vlan 100
DeviceE(config-cfm-md-MD_A-ma-1)#mep mep-id 5001 interface ge 1/0/4
inward
DeviceE(config-cfm-md-MD_A-ma-1)#end
```

步骤 3 配置连续性检测功能。

```
DeviceA#configure
DeviceA(config)#cfm md MD_A
DeviceA(config-cfm-md-MD_A)#ma 1
DeviceA(config-cfm-md-MD_A-ma-1)#ccm send mep-id 1001 enable
DeviceA(config-cfm-md-MD_A-ma-1)#end

DeviceB#configure
DeviceB(config)#cfm md MD_B
DeviceB(config-cfm-md-MD_B)#ma 2
DeviceB(config-cfm-md-MD_B-ma-2)#ccm send mep-id 2001 enable
DeviceB(config-cfm-md-MD_B-ma-2)#end
```

```
DeviceD#configure
DeviceD(config)#cfm md MD_A
DeviceD(config-cfm-md-MD_A)#ma 1
DeviceD(config-cfm-md-MD_A-ma-1)#ccm send mep-id 4002 enable
DeviceD(config-cfm-md-MD_A-ma-1)#end
DeviceD#configure
DeviceD(config)#cfm md MD_B
DeviceD(config-cfm-md-MD_B)#ma 2
DeviceD(config-cfm-md-MD_B-ma-2)#ccm send mep-id 4001 enable
DeviceD(config-cfm-md-MD_B-ma-2)#end

DeviceE#configure
DeviceE(config)#cfm md MD_A
DeviceE(config-cfm-md-MD_A)#ma 1
DeviceE(config-cfm-md-MD_A-ma-1)#ccm send mep-id 5001 enable
DeviceE(config-cfm-md-MD_A-ma-1)#end
```

步骤 4 验证环回功能。

在 Device A 上启用环回功能，检查 MA 1 内 MEP 1001 到 5001 的链路状况。

```
DeviceA#configure
DeviceA(config)#cfm md MD_A
DeviceA(config-cfm-md-MD_A)#ma 1
DeviceA(config-cfm-md-MD_A-ma-1)#ping mep-id 1001 rmep-id 5001 count 5
tlv-type null tlv-len 50 priority 0

Pinging 00-03-56-00-00-05 with tlv len 50 of data:
Reply from 00-03-56-00-00-05: bytes=59 time=2ms
Reply from 00-03-56-00-00-05: bytes=59 time=3ms
Reply from 00-03-56-00-00-05: bytes=59 time=2ms
Reply from 00-03-56-00-00-05: bytes=59 time=3ms
Reply from 00-03-56-00-00-05: bytes=59 time=2ms

Packets: Sent = 5, Received = 5, Lost = 0 <0.00% loss>
Minimum = 2ms, Maximum = 3ms, Average = 2ms
```

步骤 5 验证链路跟踪功能。

在 Device A 的 MA 1 内查找 MEP 1001 到 5001 的路径。

```
DeviceA#configure
DeviceA(config)#cfm md MD_A
DeviceA(config-cfm-md-MD_A)#ma 1
DeviceA(config-cfm-md-MD_A-ma-1)#trace mep-id 1001 rmep-id 5001 ttl 16
fdb 0

Tracing the route to 00-03-56-00-00-05 over a maximum of 16 hops:
Hop-Num   TTL     MAC              Last-MAC          Ismep   Relay Action
1         15     0003:5600:0005     0003:5600:0001     IsMep   Hit
```

步骤 6 验证单向丢包测试功能。

在 Device A 上测试 MA 1 内 MEP 1001 到 4002 的单向丢包情况。

```
DeviceA#configure
```

```
DeviceA(config)#cfm md MD_A
DeviceA(config-cfm-md-MD_A)#ma 1
DeviceA(config-cfm-md-MD_A-ma-1)#loss-measure mep-id 1001 rmep-id 4002
interval 1s priority 0 count 5
Info: Single-ended loss measurement will take some time.

Single-ended loss measurement statistics for remote mep 4002 in md MD_A
ma 1:
Packets: Sent = 5, Received = 5, Lost = 0
Far-end frame loss rate : Minimum = 0%, Maximum = 0%, Average = 0%
Near-end frame loss rate: Minimum = 0%, Maximum = 0%, Average = 0%
```

步骤 7 验证双向时延测试功能。

在 Device A 上测试 MA 1 内 MEP 1001 到 4002 的双向时延。

```
DeviceA#configure
DeviceA(config)#cfm md MD_A
DeviceA(config-cfm-md-MD_A)#ma 1
DeviceA(config-cfm-md-MD_A-ma-1)#delay-measure mep-id 1001 rmep-id 4002
interval 1s priority 0 frame-len 64 count 5
Info: Two-way delay measurement will take some time.

Two-way delay measurement statistics for remote mep 4002 in md MD_A ma 1:
Packets: Sent = 5, Received = 5, Lost = 0
Delay Time    : Minimum = 1396455us, Maximum = 2219010us, Average =
1785922us
Delay variation: Minimum = 146221us, Maximum = 707088us, Average =
436789us
```

检查结果

通过 **show cfm config** 命令查看 CFM 配置是否正确。

```
DeviceA#show cfm config
!
cfm start
cfm md MD_A level 5 format none
  mip create-type none
  ma 1 format icc 1
   map vlan 100
   mep mep-id 1001 interface ge 1/0/1 inward
   ccm send mep-id 1001 enable
```

8 安全性

本章介绍安全特性的基本原理和配置过程，并提供相关的配置案例。

- ACL
- AAA
- 802.1x
- PPPoE+
- 安全 MAC
- 风暴抑制与风暴控制
- ARP 防攻击
- ND Snooping
- DHCP Snooping
- IP Source Guard
- CPU 防攻击
- URPF
- Timerange

8.1 ACL

8.1.1 简介

ACL（Access Control List，访问控制列表）是一系列有序规则的集合，通过应用这些规则控制设备接收或拒绝某些数据报文。

在网络中为了控制非法报文对网络的影响，需要在设备上配置一系列的规则，以决定什么样的数据包能够通过，这些规则就是通过 ACL 定义的。

访问控制列表是由 permit | 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 MAC 地址、目的 MAC 地址、源 IP 地址、目的 IP 地址、端口号等来描述。设备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。

8.1.2 配置准备

场景

网络设备为了过滤数据包，需要配置 ACL，以识别需要过滤的对象。在识别出特定的对象之后，才能根据预先设定的策略允许或禁止相应的数据包通过，，丢弃动作支持报文上送 CPU。当 ACL 否定目的 MAC 地址后，相应报文的源 MAC 地址不进行学习且不显示。

- 基本 IPv4 ACL：根据数据包 IP 头所携带的源 IP、目的 IP 地址制定分类规则。
- 扩展 IPv4 ACL：根据数据包 IP 头所携带的源 IP、目的 IP、承载的协议类型、使用的 TCP 或 UDP 端口号（默认为 0）等数据包的属性信息制定分类规则，支持限制 Telnet/SSH 登录。
- MAC ACL：根据数据包二层帧头携带的源 MAC 地址、目的 MAC 地址、二层协议类型等二层信息制定分类规则，ACL 拒绝的目的 MAC 地址报文，源 MAC 地址也不再学习、不显示。
- User ACL：可以以报文的报文头、IP 头等为基准，指定从第几个字节开始与掩码进行“与”操作，将从报文提取出来的字符串与用户定义的字符串进行比较，从而找到相匹配的报文，支持匹配以太网帧前 64 字节任意字段的信息。
- IPv6 ACL：根据数据包 IP 头所携带源 IPv6 地址信息、目的 IPv6 地址信息、IPv6 承载的协议类型、使用的 TCP 或 UDP 端口号（默认为 0）等数据包的属性信息制定分类规则，支持 IPv6 ACL 限制 Telnet/SSH 登录。
- 高级 ACL：根据数据包二层帧头携带的源 MAC 地址、目的 MAC 地址、IP 头所携带的源 IP、目的 IP 等数据包的属性信息制定分类规则。

根据实际场景的差异，应用 ACL 的方式有四种：基于整个设备、基于接口、基于从入接口到出接口的流和基于 VLAN。

前提

无

8.1.3 配置 ACL

请在设备上进行以下配置。步骤 3～步骤 7 请根据需要选择配置。

步骤	配置	说明
1	`JX#config`	进入全局配置模式。

步骤	配置	说明
2	JX(config)#**acl-l2** *acl-number* [**name** *acl-name*] JX(config)#**acl-ipv4** *acl-number* [**name** *acl-name*] JX(config)#**acl-ipv6** *acl-number* [**name** *acl-name*] JX(config)#**acl-hybrid** *acl-number* [**name** *acl-name*] JX(config)#**acl-mpls** *acl-number* [**name** *acl-name*] JX(config)#**acl-userdefined** *acl-number* [**name** *acl-name*]	创建 ACL，进入 ACL 配置模式。 • 取值在 1～1000 之间时，进入基本 MAC ACL 配置模式 • 取值在 1001～2000 之间时，进入基本 IPV4 ACL 配置模式 • 取值在 2001～3000 之间时，进入扩展 HYBRID ACL 配置模式 • 取值在 3001～4000 之间时，进入 IPV6 ACL 配置模式 • 取值在 4001～5000 之间时，进入 MPLS ACL 配置模式 • 取值在 5001～6000 之间时，进入用户自定义 ACL 配置模式
3	JX(configure-acl-ipv4-*)#**rule** *rule-id* **ip src-ip** *ip-address* **dst-ip** *ip-address* JX(configure-acl-ipv4-*)#**rule** *rule-id* **icmp src-ip** *ip-address* **dst-ip** *ip-address* **icmp-type** *icmp-type* **icmp-code** *icmp-code* **[fragment]** JX(configure-acl-ipv4-*)#**rule** *rule-id* **tcp src-ip** *ip-address* **src-port** *port* **dst-ip** *ip-address* **dst-port** *port* { **syn** | **synack** | **ack** | **fin** } **[fragment]** JX(configure-acl-ipv4-*)#**rule** *rule-id* **igmp src-ip** *ip-address* **dst-ip** *ip-address* **[fragment]** JX(configure-acl-ipv4-*)#**rule** *rule-id* **udp src-ip** *ip-address* **src-port** *port* **dst-ip** *ip-address* **dst-port** *port* **[fragment]**	配置 IP ACL 的匹配规则。
4	JX(configure-acl-l2-*)#**rule** *rule-id* **ethernet src-mac** *mac-address* **dst-mac** *mac-address* JX(configure-acl-l2-*)#**rule** *rule-id* **src-mac ethernet** *mac-address* **dst-mac** *mac-address* **eth-type** { **ip** | **arp** | *ethtype* } JX(configure-acl-l2-*)#**rule** *rule-id* **src-mac ethernet** *mac-addr* **dst-mac** *mac-address* { **inner-vlan** | **outer-vlan** } *vlan-id* **8021p** *cos*	配置 MAC ACL 的规则。

步骤	配置	说明
5	JX(configure-acl-ipv6-*)#**rule** *rule-id* **ip src-ip** *ipv6-address* **dst-ip** *ipv6-address* JX(configure-acl-ipv6-*)#**rule** *rule-id* **tcp src-ip** *ipv6-address* **src-port** *port* **dst-ip** *ipv6-address* **dst-port** *port* { **syn** | **synack** | **ack** | **fin** } JX(configure-acl-ipv6-*)# **rule** *rule-id* **udp src-ip** *ipv6-address* **src-port** *port* **dst-ip** *ipv6-address* **dst-port** *port* JX(configure-acl-ipv6-*)#**rule** *rule-id* **icmp src-ip** *ipv6-address* **dst-ip** *ipv6-address* **icmp-type** *icmp-type* **icmp-code** *icmp-code* **[fragment]**	配置 IPv6 ACL 的规则。
6	JX(configure-acl-ipv6-*)#**rule** *rule-id* **field1** *data/mask/offset* {**field2** *data/mask/offset* | **field3** *data/mask/offset* | **field4** *data/mask/offset*}*	配置用户自定义 ACL 的规则。 data 表示需要匹配的信息，mask 表示对 data 做掩码操作，offset 表示该匹配信息距离报文头部的字节数，offset 必须能整除 4。
7	JX(configure-acl-ipv4-*)#**rule** *rule-id* **action** { **permit** | **deny** | **mirror** | **redirect** | **counter** }	配置基本 IP ACL 的动作。

8.1.4 应用 ACL

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**interface** *interface-type interface-number*	进入物理接口配置模式或者 VLAN 接口配置模式，以下以物理接口模式为例进行说明。
3	JX(config-ge-1/0/*)#**acl-l2 in** *acl-num*	在接口上应用 ACL。
	JX(config-ge-1/0/*)#**acl-l2 in** *name*	在接口上应用 ACL。
4	JX(config-ge-1/0/*)#**exit**	返回全局配置模式。

8.1.5 配置统计

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**counter** *counter-id* { **packet** | **byte** | **all** } **sort** { **green** | **red** | **greenred** | **greenyellow** | **redyellow** | **total** }	创建统计模版
3	JX(configure-acl-l2-*)#**rule** *rule-id* **action counter** *counter-id*	在 acl 中应用统计模版

8.1.6 配置限速

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**meter** *meter-id* **pps** *pps-value* **color** { **aware** | **blind** }	创建基于包个数的限速模版
3	JX(config)#**meter** *meter-id* **cir** { **kbps** | **mbps** | **gbps** } *cir-value* **cbs** { **bytes** | **kbytes** | **mbytes** } *cbs-value* **ebs** { **bytes** | **kbytes** | **mbytes** } *ebs-value* **color** { **aware** | **blind** }	创建单速三色的限速模版
4	JX(config)#**meter** *meter-id* **cir** { **kbps** | **mbps** | **gbps** } *cir-value* **cbs** { **bytes** | **kbytes** | **mbytes** } *cbs-value* **pir** { **kbps** | **mbps** | **gbps** } *pir-value* **pbs** { **bytes** | **kbytes** | **mbytes** } *pbs-value* **color** { **aware** | **blind** }	创建双速三色的限速模版
3	JX(configure-acl-l2-*)#**rule** *rule-id* **meter** *meter-id* **outaction** { **red-drop** | **yellow-drop** }	在 acl 中应用限速模版

8.1.7 配置时间段

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**acl-l2** *acl-id*	进入 acl list 视图

步骤	配置	说明
3	JX(configure-acl-l2-*)#**rule** *rule-id* **time-range** *timerange-list-id*	在 acl 中应用时间段

8.1.8 检查配置

配置完成后，请在设备上执行以下命令检查配置结果。

序号	检查项	说明
1	JX#**show acl config**	查看 ACL 配置信息。
2	JX#**show acl interface**	查看接口 acl 配置信息。
3	JX#**show acl statistics**	查看 acl 统计信息。
4	JX#**show time-range list**	查看时间段配置相关信息。

8.1.9 维护

用户可以通过以下命令，维护设备 ACL 特性的运行情况和配置情况。

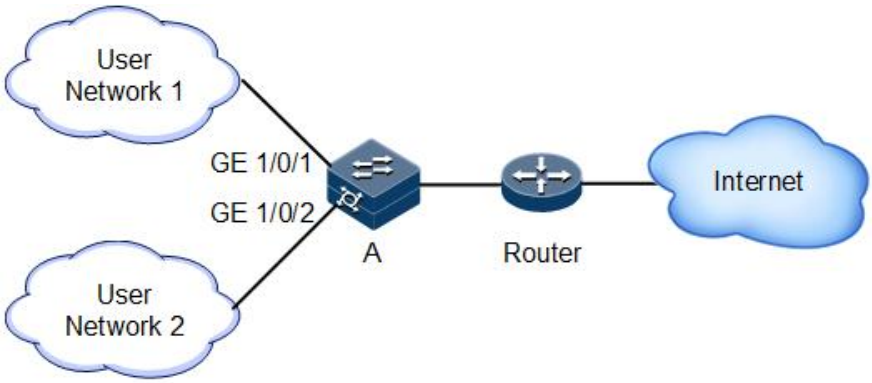
命令	描述
JX(config)#**reset acl statistics** [**acl** *acl-id* |**rule** *rule-id* | **direction** { **in** | **out** }]	清除全局 ACL 统计信息。
JX(config)#**reset acl statistics interface** *interface-type interface-number* [**acl** *acl-id* |**rule** *rule-id* | **direction** { **in** | **out** }]	清除接口 ACL 统计信息。

8.1.10 配置 ACL 示例

组网需求

如下图所示，想要控制在每天 00:00 到 08:00 之间，拒绝通过 SwitchA 访问互联网的请求，在每天 08:00 到 12:00 之间，限制访问速度为 10000pps

图 8-1 ACL 示例



配置步骤

步骤 1 配置 timerange。

```
JX#configure
JX(config)#timerange list 1
JX(config-timerange-1)#time-range 1 everyday 00:00:00 to 08:00:00
JX(config-timerange-1)#configure
JX(config)#timerange list 2
JX(config-timerange-1)#time-range 1 everyday 08:00:00 to 12:00:00
```

步骤 2 配置 meter。

```
JX#configure
JX(config)#meter 1 pps 10000 color blind
```

步骤 3 配置 acl 匹配规则和动作。

```
JX#configure
JX(config)#acl-l2 1 name test
JX(configure-acl-l2-1)#rule 1 src-mac any dst-mac any
JX(configure-acl-l2-1)#rule 1 time-range 1
JX(configure-acl-l2-1)#rule 1 action deny
JX(configure-acl-l2-1)#rule 2 src-mac any dst-mac any
JX(configure-acl-l2-1)#rule 2 time-range 2
JX(configure-acl-l2-1)#rule 2 meter 1 outaction red-drop yellow-drop
```

步骤 4 在接口上应用 acl。

```
JX#configure
JX(config)#interface ge 1/0/1 to ge 1/0/2
JX(config-ge-1/0/1->ge-1/0/2)#acl-l2 in 1
```

检查结果

```
JX#show acl config
!
acl-l2 1 name test
 rule 1 src-mac any dst-mac any
 rule 1 time-range 1
 rule 1 action deny
 rule 2 src-mac any dst-mac any
```

```
 rule 2 time-range 2
 rule 2 meter 1 outaction red-drop yellow-drop
!
interface ge 1/0/1
 acl-l2 in name test
!
interface ge 1/0/2
 acl-l2 in name test
```

8.2 AAA

8.2.1 简介

AAA

AAA（Authentication、Authorization、Accounting，认证、授权、计费）是网络安全的一种管理机制，提供了认证、授权、计费三种安全功能。

- 认证：确认访问网络的远程用户的身份，判断访问者是否为合法的网络用户。
- 授权：对不同用户赋予不同的权限，限制用户可以使用的服务。例如，管理员授权办公用户才能对服务器中的文件进行访问和打印操作，而其它临时访客不具备此权限。
- 计费：记录用户使用网络服务过程中的所有操作，包括使用的服务类型、起始时间、数据流量等，用于收集和记录用户对网络资源的使用情况，并可以实现针对时间、流量的计费需求，也对网络起到监视作用。

AAA 采用客户端/服务器结构，客户端运行于 NAS（Network Access Server，网络接入服务器）上，负责验证用户身份与管理用户接入，服务器上则集中管理用户信息。

AAA 可以通过多种协议来实现，这些协议规定了 NAS 与服务器之间如何传递用户信息。目前设备支持 RADIUS（Remote Authentication Dial-In User Service，远程认证拨号用户服务）协议和 TACACS+（Terminal Access Controller Access Control System，终端访问控制器访问控制系统）。

RADIUS

RADIUS（Remote Authentication Dial In User Service，远程用户拨号认证系统）是一种用于对远程访问用户进行集中鉴别的标准化通信协议。RADIUS 使用 UDP 作为传输协议（端口 1812、1813），具有良好的实时性；同时也支持重传机制和备用服务器机制，从而具有较好的可靠性。

- 认证功能

RADIUS 使用客户端/服务器模式，网络访问设备作为 RADIUS 服务器的客户端。RADIUS 服务器负责接收用户的连接请求、对用户进行鉴别，然后将所有客户端所需的配置信息传回，以便为用户提供服务。通过这种方式可以控制用户对设备和网络的访问，提高网络的安全性。

客户端与 RADIUS 服务器之间的通信是通过共享密钥的使用来鉴别的，这个共享密钥不会通过网络传送。此外，任何用户口令在客户机和 RADIUS 服务器间发送时都需要进行加密过程，以避免有人通过嗅探非安全网络得到用户密码。

- RADIUS 计费功能

RADIUS 计费功能主要针对通过 RADIUS 认证的用户进行。在用户登录时给 RADIUS 计费服务器发送一个开始计费的报文，在登录期间根据计费策略给 RADIUS 计费服务器发送计费更新报文，退出登录时，给 RADIUS 计费服务器发送停止计费报文，报文里面包含用户的登录时间。通过这些报文，RADIUS 计费服务器可以记录每个用户的访问时间和操作。

TACACS+

TACACS+（Terminal Access Controller Access Control System，终端访问控制器访问控制系统）是一种与 RADIUS 类似的网络接入认证协议。其区别如下：

- TACACS+使用 TCP 端口 49，相对于 RADIUS 使用的 UDP 端口，具有更高的传输可靠性。
- TACACS+加密数据包除标准的 TACACS+头部外的整体，而包头中有一个区域会指示数据包是否加密。相对于 RADIUS 的只加密用户密码，安全性更高。
- TACACS+的认证功能与授权、计费功能相分离，部署更灵活。

综上所述，TACACS+较 RADIUS 更加安全、可靠，但是 RADIUS 作为一种开放性的协议，在网络中的应用更加广泛。

8.2.2 配置准备

场景

为了控制用户对设备和网络的访问，可以在网络中部署 RADIUS/TACACS+服务器对用户进行认证和计费。本设备可以作为 RADIUS/TACACS+服务器的代理设备，根据 RADIUS/TACACS+服务器反馈的结果对用户访问进行授权。TACACS+较 RADIUS 更加安全、可靠。

前提

无

8.2.3 缺省配置

设备上 AAA 的缺省配置如下。

功能	缺省值
RADIUS 认证服务器未响应之后的失效时间	60 秒
RADIUS 报文重传次数	3
RADIUS 报文重传时间间隔	2 秒

功能	缺省值
RADIUS 认证服务器未响应之后的失效时间	60 秒
TACACS 认证服务器 TCP 连接超时和收包超时时间	2 秒

8.2.4 配置 RADIUS 服务器

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)# **aaa**	进入 AAA 配置模式。
3	JX(config-aaa)#**radius-server deadtime** { *second* | **default** }	全局配置服务器未响应之后的失效时间，单位为秒，缺省值为 60 秒。
4	JX(config-aaa)#**radius-server max-retransmit** { *count* | **default** }	全局配置发送请求失败后的重传次数，缺省值为 3 次。
5	JX(config-aaa)#**radius-server retransmit-interval** { *interval* | **default** }	全局配置重传时间间隔，缺省值为 2 秒。
6	JX(config-aaa)#**radius-server host** *server-name* **ip-address** *ip-address* **key** *key* [**acct-port** *port-id* | **auth-port** *port-id* | **deadtime** { *second* | **default** } | **max-retransmit** { *count* | **default** } | **retransmit-interval** { *interval* | **default** } | **source-ip** *ip-address* [**source-vpn-instance** *vpn-name*] | **vpn-instance** *vpn-name*]*	创建 IPv4 RADIUS 服务器。
7	JX(config-aaa)#**radius-server host** *server-name* **ip6-address** *ipv6-address* **key** *key* [**acct-port** *port-id* | **auth-port** *port-id* | **deadtime** { *second* | **default** } | **max-retransmit** { *count* | **default** } | **retransmit-interval** { *interval* | **default** } | **source-ip6** *ipv6-address* [**source-vpn-instance** *vpn-name*] | **vpn-instance** *vpn-name*]*	创建 IPv6 RADIUS 服务器。

8.2.5 配置 TACACS+服务器

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**aaa**	进入 AAA 配置模式。
3	JX(config-aaa)#**tacacs-server deadtime** { *second* | **default** }	全局配置服务器未响应之后的失效时间，单位为秒，缺省值为 300 秒。
4	JX(config-aaa)#**tacacs-server timeout** { *second* | **default** }	全局配置与服务器的 TCP 连接超时和收包超时时间，缺省值为 2 秒。
5	JX(config-aaa)#**tacacs-server host** *server-name* **ip-address** *ip-address* **key** *key* [**deadtime** { *second* | **default** } | **port** *port-id* | **single-connection** { **enable** | **disable** } | **timeout** { *second* | **default** } | **source-ip** *ip-address* [**source-vpn-instance** *vpn-name*] | **vpn-instance** *vpn-name*]*	创建 IPv4 TACACS 服务器。
6	JX(config-aaa)#**tacacs-server host** *server-name* **ip6-address** *ipv6-address* **key** *key* [**deadtime** { *second* | **default** } | **port** *port-id* | **single-connection** { **enable** | **disable** } | **timeout** { *second* | **default** } | **source-ip6** *ipv6-address* [**source-vpn-instance** *vpn-name*] | **vpn-instance** *vpn-name*]*	创建 IPv6 TACACS 服务器。

8.2.6 配置 AAA 服务器组

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**aaa**	进入 AAA 配置模式。
3	JX (config-aaa)#**server-group** *group-name* { **radius-server** | **tacacs-server** } *server-name*	创建 AAA 服务器组或向 AAA 服务器组添加服务器

8.2.7 配置 AAA 方法

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。

步骤	配置	说明
2	JX(config)#**aaa**	进入 AAA 配置模式。
3	JX(config-aaa)#**aaa authentication** { **dot1x** | **login** | **mac-authen** | **enable** } **method** *method-name* **first** { *group-name* | **local** } **second** { *group-name* | **local** | **none** } **third** { *group-name* | **local** | **none** }	配置 AAA 认证方法。 注意事项：服务器组最多配置两个。
4	JX(config-aaa)#**aaa authorization** { **login** | **cmd** } **method** *method-name* **first** { *group-name* | **local** } **second** { *group-name* | **local** | **none** } **third** { *group-name* | **local** | **none** }	配置 AAA 授权方法。 注意事项：服务器组最多配置两个。
5	JX (config-aaa)#**aaa accounting** { **dot1x** | **login** | **mac-authen** } **method** *method-name* **first** { *group-name* | **local** } **second** { *group-name* | **local** | **none** } **third** { *group-name* | **local** | **none** }	配置 AAA 计费方法。 注意事项：服务器组最多配置两个。

8.2.8 检查配置

配置完成后，请在设备上执行以下命令检查配置结果。

序号	检查项	说明
1	JX#**show aaa config**	查看 AAA 配置信息。
2	JX#**show aaa information**	查看 AAA 全局信息。
3	JX#**show aaa server**	查看 AAA 服务器信息。
4	JX#**show aaa server-group**	查看 AAA 服务器组信息。
5	JX#**show aaa method**	查看 AAA 方法信息。

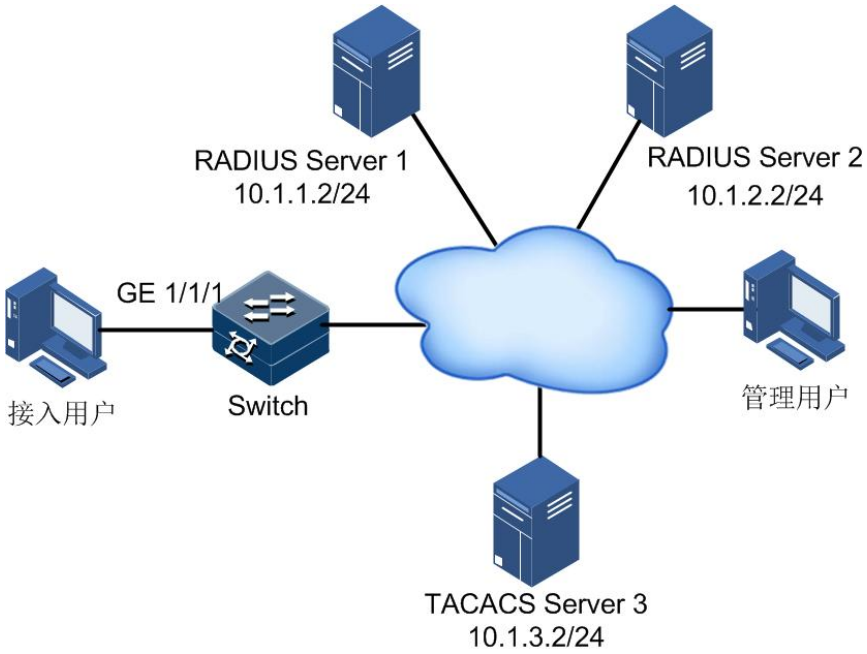
8.2.9 配置 AAA 示例

组网需求

如下图所示，实现接入用户和管理用户接入不同的服务器，要求配置如下：

- 在 Switch 上配置 IP 地址、Vlan 和路由，用于用户连接及认证。
- 创建本地用户，配置管理用户所用的 AAA 模板和方案，采用 TACACS Server 3 进行认证和授权，RADIUS Server 2 进行计费。
- 开启 Dot1x 功能，配置接入用户所使用的 AAA 模板和方案，采用 RADIUS Server 1 进行认证、计费和授权。

图 8-2 分域认证应用组网示意图



配置步骤

步骤 1 配置 IP 地址。

```
JX#config
JX(config)#interface vlan 1
JX(config-vlanif-1)#ip address 10.1.0.254/24
JX(config-vlanif-1)#exit
JX(config)ip route-static 0.0.0.0 0.0.0.0 10.1.0.1
```

步骤 2 配置 AAA 服务器。

```
JX(config)#aaa
JX(config-aaa)#radius-server host Server1 ip-address 10.1.1.2
JX(config-aaa)#radius-server host Server2 ip-address 10.1.2.2
JX(config-aaa)#tacacs-server host Server3 ip-address 10.1.3.2
```

步骤 3 配置 AAA 服务器组。

```
JX(config-aaa)#server-group Group1 radius-server Server1
JX(config-aaa)#server-group Group2 radius-server Server2
JX(config-aaa)#server-group Group3 tacacs-server Server3
```

步骤 4 配置 AAA 方法。

```
JX(config-aaa)#aaa authentication login method Method1 first Group3
JX(config-aaa)#aaa authorization login method Method2 first Group3
JX(config-aaa)#aaa accounting login method Method3 first Group2
JX(config-aaa)#aaa authentication dot1x method Method4 first Group1
JX(config-aaa)#aaa accounting dot1x method Method5 first Group1
switch(config-aaa)#exit
```

步骤 5 配置管理用户认证、授权、计费方法。

```
JX(config)#line vty * *
JX(config-line)#login authentication aaa method Method1
JX(config-line)#login authorization aaa method Method2
JX(config-line)#login accounting aaa method Method3
JX(config-line)#exit
```

步骤 6 配置接入用户认证、授权、计费方法。

```
JX(config)#dot1x start
JX(config)#interface ge 1/0/1
JX(config-ge-1/0/1)#dot1x enable
JX(config-ge-1/0/1)#dot1x aaa-authentication Method4
JX(config-ge-1/0/1)#dot1x aaa-accounting Method5
JX(config-ge-1/0/1)#exit
```

8.2.10 检查结果

步骤 1 通过 **show aaa config** 查看 AAA 配置是否正确。

```
JX#show aaa config
Version  : AAA_V7.00.00.00
!
aaa
 radius-server host Server1 ip-address 10.1.1.2
 radius-server host Server2 ip-address 10.1.2.2
 tacacs-server host Server3 ip-address 10.1.3.2
 server-group Group1 radius-server Server1
 server-group Group2 radius-server Server2
 server-group Group3 tacacs-server Server3
 aaa authentication login method Method1 first Group3
 aaa authorization login method Method2 first Group3
 aaa accounting login method Method3 first Group2
 aaa authentication dot1x method Method4 first Group1
 aaa accounting dot1x method Method5 first Group1
```

8.3 802.1x

8.3.1 简介

802.1x 是基于 IEEE 802.1x 协议即基于接口的网络接入控制技术。802.1x 功能的主要目的是解决局域网用户的接入认证和安全问题。

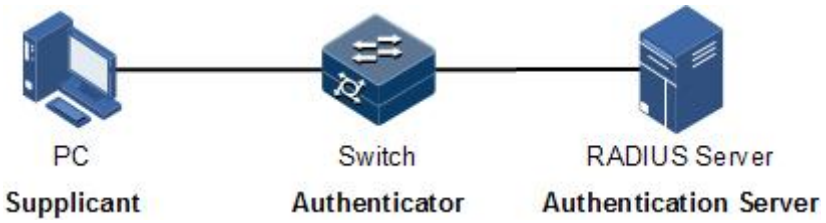
在网络设备的物理接入层对接入设备进行认证和控制，仅定义了设备接口和用户设备之间的点到点连接方式。连接在接口上的用户设备如果能够通过认证，就可以访问局域网中的资源；如果不能通过认证，则无法通过交换机访问网络中的资源。

802.1x 体系结构

802.1x 认证采用客户端/服务器模式，如下图所示，包括以下 3 个部分：

- 申请者（Supplicant）：需要安装 802.1x 客户端软件（例如 Windows XP 自带的 802.1x 客户端）的用户侧设备，如计算机等。
- 认证者（Authenticator）：提供 802.1x 认证功能的接入控制设备，如交换机等。
- 认证服务器（Authentication Server）：用于对用户进行认证、授权和计费，通常使用 RADIUS 服务器作为 802.1x 认证服务器。

图 8-3 802.1x 认证体系结构



接口接入控制模式

认证者利用认证服务器对需要接入局域网的客户端进行认证，并根据认证结果对接入接口授权或者非授权状态进行控制。用户可以通过配置接口的接入控制模式来控制接口的接入状态。802.1x 认证支持三种接口接入控制模式：

- 协议授权模式（auto）：由协议状态机决定认证授权结果，在认证成功之前，仅允许收发 EAPoL 报文，不允许用户访问网络资源和交换机提供的服务。如果认证通过，则接口切换到授权状态，允许用户访问网络资源和交换机提供的服务。
- 强制接口授权模式（authorized-force）：接口始终处于授权状态，允许用户不经认证授权即可访问网络资源和交换机提供的服务。
- 强制接口非授权模式（unauthorized-force）：接口始终处于非授权状态，不允许用户访问网络资源和交换机提供的服务，即不允许用户进行认证。

802.1x 认证过程

802.1x 系统支持 EAP 中继和 EAP 终结两种方式完成与 RADIUS 服务器之间的认证过程。

- EAP 中继方式

申请者与认证服务器之间通过 EAP（Extensible Authentication Protocol，可扩展认证协议）报文交换信息。申请者与认证者之间则以 IEEE802.1x 协议所定义的 EAPoL（EAP over LAN，基于局域网的 EAP）报文交换信息。EAP 报文中封装了认证数据，该认证数据将被封装在 RADIUS 协议的报文中，以穿越复杂的网络到达认证服务器，这一过程称为 EAP 中继。

认证者或申请者均能发起 802.1x 认证过程。以申请者发起认证过程为例，EAP 中继认证过程如下：

1. 用户输入用户名和密码，申请者向认证者发送一个 EAPoL-Start 报文，开始一次 802.1x 认证；
2. 认证者向申请者发送 EAP-Request/Identity 报文，询问请求者的用户名；

3. 申请者响应一个 EAP-Response/Identity 给认证者，其中包括用户名信息；
4. 认证者将 EAP-Response/Identity 报文封装到 RADIUS 协议报文中，发送给认证服务器；
5. 认证服务器将接收到的用户名信息与数据库中的用户名表进行比对，找到该用户的口令信息，利用随机生成的加密字对口令信息进行加密处理。同时，认证服务器将此加密字发送给认证者，认证者再将此加密字发送给申请者；
6. 申请者利用接收到的加密字对口令进行加密，并通过认证者发送给认证服务器；
7. 认证服务器对比收到的加密口令与自身生成的加密口令否一致。如果认证成功，认证者将接口改为授权状态，允许用户通过接口访问网络，并发送 EAP-Success 报文给申请者；如果认证失败，则接口为非授权状态，并发送 EAP-Failure 报文给通知申请者。

- EAP 终结方式

将 EAP 报文在设备端终结并映射到 RADIUS 报文中，利用标准 RADIUS 协议完成认证、授权和计费过程。设备端支持与 RADIUS 服务器之间采用 PAP 或者 CHAP 认证方法。

在 EAP 终结方式中，用来对用户密码信息进行加密处理的随机加密字由设备端生成，之后设备端会把用户名、随机加密字和客户端加密后的密码信息共同发送给 RADIUS 服务器，进行相关的认证处理。

802.1x 定时器

802.1x 认证过程中，认证设备上涉及到 5 个定时器：

- Reauth-period：重认证定时器。在该定时器超时后，会重新发起 802.1x 认证。
- Quiet-period：静默定时器。用户认证失败以后，认证设备需要静默一段时间，静默定时器超时后再重新发起认证。在静默期间，交换机不处理认证报文。
- Tx-period：请求报文发送超时定时器。当交换机向用户请求端发送 Request/Identity 请求报文后，会启动该定时器，在该定时器超时后，用户端软件未成功发送认证应答报文，则设备重发认证请求报文，此报文共重发 3 次。
- Supp-timeout：申请者认证超时定时器。当交换机向用户请求端发送了用于请求用户端 MD5 加密密文的 Request/Challenge 请求报文后，交换机启动该定时器。若在该定时器设置的时长内用户请求端未成功响应，交换机将重发该报文，此报文共重发两次。
- Server-timeout：认证服务器超时定时器。该定时器定义认证者和认证服务器会话超时的总时长，此定时器超时后认证者结束同认证服务器会话，重新开始一次新的认证过程。

802.1x guest-vlan

在网络中，用户在未经过 802.1x 认证时只能访问有限资源，当配置 guest-vlan 功能后，设备对 untag 报文添加 guest-vlan，并允许该报文通过端口。

- 基于端口认证：认证时认证通过会删除端口的 guest-vlan。
- 基于用户认证：认证通过会保留端口的 guest-vlan。
- guset-vlan 不能是 supervlan，也不能是 voice-vlan。

8.3.2 配置准备

场景

为了实现对局域网用户的接入认证，并解决接入用户的安全问题，需要在设备上配置802.1x 认证。

对于认证通过的用户，允许其访问网络中的资源；如果认证未通过，则该用户无法访问网络资源。通过对用户接入接口的认证控制，达到对用户管理的目的。

前提

配置 802.1x 认证之前，如果使用 RADIUS 认证服务器，需要完成以下任务：

- 配置 RADIUS 服务器 IP 地址和 RADIUS 公有密钥。
- 交换机能够与 RADIUS 服务器 Ping 通。

8.3.3 802.1x 功能的缺省配置

设备上 802.1x 功能的缺省配置如下。

功能	缺省值
全局 802.1x 功能状态	禁止
接口 802.1x 功能状态	禁止
全局认证方式	**chap**
接口接入控制模式	**auto**
接口认证方式	**macbased**
RADIUS 服务器超时定时器时间	10s
802.1x 重认证功能状态	允许
802.1x 重认证定时器时间	5600s
802.1x 静默定时器时间	60s
请求报文重传定时器时间	30s
最大用户数	128

8.3.4 配置 802.1x 基本功能

注意

- 一个接口同一时刻只能处理一个用户认证请求。

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**dot1x** { **start** | **stop** }	使能或去使能全局 802.1x 功能。
3	JX(config)#**dot1x aaa authentication method** *method-name*	绑定全局 802.1x 认证时的 AAA 方法。
4	JX(config)#**dot1x max-user** *user-number*	配置全局 802.1x 认证最大用户数。
5	JX(config)#**interface** *interface-type interface-number*	进入二层物理接口配置模式。
6	JX(config-ge-1/0/*)#**dot1x enable**	使能接口 802.1x 功能。
7	JX(config-ge-1/0/*)#**dot1x port-control** { **auto** | **force-auth** | **force-unauth** }	配置接口接入控制模式。
8	JX(config-ge-1/0/*)#**dot1x port-method** { **mac** | **port** }	配置接口认证方式。
9	JX(config-ge-1/0/*)#**dot1x max-user** *user-number*	配置 802.1x 端口允许认证的最大用户数。
10	JX(config-ge-1/0/*)#**nac guest-vlan** *vlan-id*	配置指定端口的 Guest VLAN，对 802.1x 协议和 mac 认证都生效。
11	JX(config-ge-1/0/*)#**dot1x critical-vlan** *vlan-id*	配置指定端口的 802.1x Critical VLAN。
12	JX(config-ge-1/0/*)#**dot1x restrict-vlan** *vlan-id*	配置指定端口的 802.1x Restrict VLAN。
13	JX(config-ge-1/0/*)#**dot1x reauthenticate all user**	手动触发指定端口下 802.1x 用户进行重认证。
14	JX(config-ge-1/0/*)#**dot1x delete all user**	强制将端口下 802.1x 用户下线。
15	JX(config-ge-1/0/*)#**dot1x quiet** { **disable** | **enable** }	使能或去使能端口 802.1x 用户静默功能。
16	JX(config-ge-1/0/*)#**dot1x quiet-times** *times*	配置端口 802.1x 用户触发静默功能的认证失败次数，默认 3 次。

说明

如果全局或接口模式下未使能 802.1x 功能，则接口下不能使能 802.1x 功能。

8.3.5 配置 802.1x 重认证

注意

重认证功能是针对已授权的用户发起的，所以在使能重认证功能之前，应该保证使能全局和接口 802.1x 功能。处于授权状态的接口在重认证过程中仍保持授权状态，如果重认证失败，才进入非授权状态。

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**interface** *interface-type interface-number*	进入二层物理接口配置模式。
3	JX(config-ge-1/0/*)#**dot1x reauthenticate { enable | disable }**	使能 802.1x 重认证功能。

8.3.6 配置 802.1x 定时器

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**interface** *interface-type interface-number*	进入二层物理接口配置模式。
3	JX(config-ge-1/0/*)#**dot1x reauthenticate period** *time*	配置重认证定时器时间。
4	JX(config-ge-1/0/*)#**dot1x timer quiet-period** *time*	配置静默定时器时间。
5	JX(config-ge-1/0/*)#**dot1x supp period** *time*	配置 Request/MD5 Challenge 请求报文超时定时器。
6	JX(config-ge-1/0/*)#**dot1x server-timeout period** *time*	配置认证服务器超时定时器时间。
7	JX(config-ge-1/0/*)#**dot1x tx period** *time*	配置 Request/Identity 请求报文超时定时器。

8.3.7 检查配置

配置完成后，请在设备上执行以下命令检查配置结果。

序号	检查项	说明
1	JX#show dot1x config	查看所有 802.1x 相关配置信息。
2	JX#show dot1x information	查看 802.1x 协议统计信息。
3	JX#show dot1x user	查看 802.1x 协议认证的用户信息。
4	JX#show dot1x *interface-type interface-number*	查看接口下 802.1x 相关统计及配置信息。

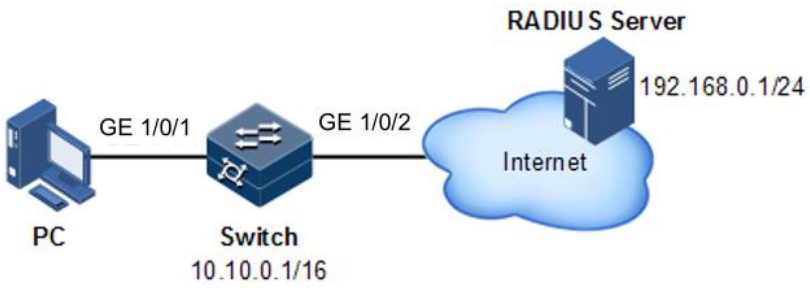
8.3.8 配置 802.1x 示例

组网需求

为了使用户访问外部网络，如下图所示，在交换机上配置 802.1x 认证，具体要求如下：

- 交换机的 IP 地址是 10.10.0.1，掩码是 255.255.0.0，缺省网关地址为 10.10.0.2。
- 通过 RADIUS 服务器进行认证和授权，RADIUS 服务器的 IP 地址是 192.168.0.1，密码是 JX。
- 在认证通过后，可以在 600s 后自动发起重认证过程。

图 8-4 802.1x 应用组网示意图



配置步骤

步骤 1 配置交换机 IP 地址及 RADIUS 服务器地址。

```
JX#config
JX(config)#interface vlan 1
JX(config-vlan1)#ip address 10.10.0.1/16
JX(config-vlan1)#exit
JX(config)#ip route 0.0.0.0 0.0.0.0 10.10.0.2
JX(config)#exit
JX(config)#aaa
JX(config-aaa)#radius-server host server1 ip-address 192.168.0.1 key
12345
JX(config-aaa)#server-group grp1 radius-server server1
```

```
JX(config-aaa)#aaa authentication dot1x method d1 first grp1
```

步骤 2 使能全局及接口 802.1x 认证功能。

```
JX#config
JX(config)#dot1x start
JX(config)#dot1x aaa authentication method d1
JX(config)#interface ge 1/0/1
JX(config-ge-1/0/1)#dot1x enable
JX(config-ge-1/0/1)#dot1x reauthenticate period 600
```

检查结果

通过 **show dot1x** 命令查看设备上 802.1x 功能的配置结果。

```
JX#show dot1x interface ge 1/0/1
Interface                  : ge 1/0/1
 Authentication Guest Vlan   : n/a
 Max User Num              : 1
 Default Max User Num       : 1
 Current User Num          : 1
 Authen Success User Num    : 0
 Authen Fail User Num       : 1
 Authen Timeout User Num    : 1
 Authenting User Num        : 0
 Authentication Method      : n/a
 Accounting Method          : n/a
 Quiet                     : Disable
 Reauthentication           : Enable
 Reauthentication Period    : 5600
 Mac Bypass                : Disable
 Offline Detect            : Disable
 Restrict Vlan             : n/a
 Critical Vlan             : n/a
 TX Period                 : 30
 Supp Timeout Period        : 5
 Server time Period         : 120
 Port Control              : Auto
 Port Method               : Mac Based
 Port Auth State           : Unauthenticated
 Auth Method               : Chap
 Not Eapol Trigger          : Disable
 Trigger Authen Type        : None
 Trigger Auth Pkt Type      : ARP NDP DHCP DHCP6
 Rx Eapol Start Pkt Num     : 61
 Rx Eapol Logoff Pkt Num    : 0
 Rx Eap Idenitity Pkt Num   : 12
 Rx Eap MD5 Pkt Num         : 5
 Tx Eap Success Pkt Num     : 0
 Tx Eap Fail Pkt Num        : 11
 Tx Eap Idenitity Pkt Num   : 6
 Tx Eap MD5 Pkt Num         : 5
```

8.4 PPPoE+

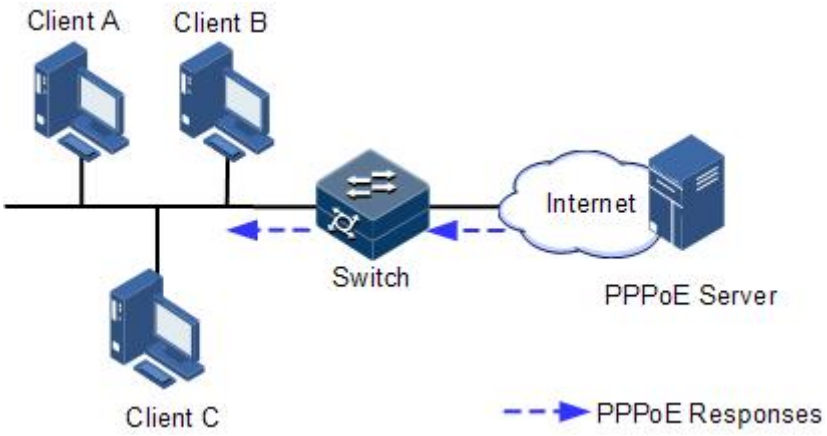
8.4.1 简介

PPPoE+（PPPoE Intermediate Agent，PPPoE 中继代理）协议用于对 PPPoE（Point to Point Protocol Over Ethernet，基于以太网承载点到点协议）认证报文进行处理，即对 PPPoE 报文附加更多接入设备信息，使服务器能获得足够的信息辨别用户。PPPoE+协议可以有效防止在 PPPoE 认证过程中的账号共享或者账号盗用问题，保证了网络的安全性。

使用 PPPoE 拨号方式连接网络，用户通过设备的不同接口，只要通过同一个认证服务器认证成功，就可以使用这个账号访问网络。但是服务器仅根据包含用户名和密码的认证信息，很难对用户进行区分。增加 PPPoE+特性以后，认证时除了需要用户名和密码信息外，在认证报文中还将携带设备接口等信息。如果认证服务器识别的接口号等信息与配置不一致，则认证不通过，这样就可以防止非法用户盗用其他合法用户的账号进行上网。

PPPoE 协议采用客户端/服务器模式，如下图所示，Switch 起到中继代理的作用，用户通过 PPPoE 认证连接网络。如果服务器需要定位用户，在认证报文中则需要更多的客户信息。

图 8-5 用户通过 PPPoE 认证连接网络示意图



用户通过 PPPoE 访问网络需要经过两个阶段：第一个阶段是发现阶段，即认证阶段，第二个阶段是会话阶段。PPPoE+功能需要处理的就是发现阶段的报文。

- 客户端通过 PPPoE 访问网络，首先会发送一个广播报文 PADI（PPPoE Active Discovery Initiation，PPPoE 活动发现发起报文），该报文的作用是查找认证服务器；
- 收到 PADI 报文的认证服务器会发送一个单播 PADO（PPPoE Active Discovery Offer，PPPoE 活动发现提供报文）报文响应；
- 如果有多个认证服务器发送了 PADO 报文，客户端会从中选择一个，发送单播 PADR（PPPoE Active Discovery Request，PPPoE 活动发现请求报文）报文请求认证；

- 认证服务器收到 PADR 报文后，如果判定用户合法，则发送一个单播报文 PADS（PPPoE Active Discovery Session-confirmation，PPPoE 活动发现会话确认报文）作为 PADR 的响应。至此，发现阶段完毕。

PPPoE+的主要功能是在 PADI 和 PADR 报文中添加用户标识信息，服务器可以判定标识信息是否和用户账号匹配，决定是否分配资源。

8.4.2 配置准备

场景

为了防止在 PPPoE 认证过程中有非法用户接入，需要配置 PPPoE+功能，在 PPPoE 协议报文中加入附加的用户标识信息。

由于添加的用户标识信息和具体的交换机及接口相关，因此认证服务器可以将用户和交换机以及接口等信息绑定。从而有效防止账号共享和账号盗用问题，还可以更好的定位用户，以保证网络的安全性。

前提

无

8.4.3 PPPoE+功能的缺省配置

设备上 PPPoE+功能的缺省配置如下。

功能	缺省值
全局 PPPoE+功能状态	禁止
接口 PPPoE+功能状态	禁止
全局 PPPoE+Tag 的收包处理策略	replace
Circuit ID 的填充模式	SwitchCommon 模式
Circuit ID 信息	接口名/外层 VLAN 号/内层 VLAN 号
Circuit ID 的附加字符串	无
Remote ID 填充的 MAC 地址	交换机的 MAC 地址
Remote ID 填充形式	二进制形式
接口信任状态	非信任接口
PPPoE+ Tag 收包处理策略	替换

说明

缺省情况下，PPPoE 报文可以通过接口并且不会被附加任何信息。

8.4.4 配置 PPPoE+基本功能

注意

PPPoE+功能用于处理 PADI 和 PADR 报文，只针对 PPPoE 的客户端。一般只有连接客户端的接口使能 PPPoE+功能，而信任接口是指交换机与 PPPoE 服务器连接的接口，这两种接口角色是互斥的，即一个接口不能既使能 PPPoE+功能又是信任接口。

使能 PPPoE+功能

使能设备全局和接口的 PPPoE+功能后，发送到该接口的 PPPoE 认证报文会附加上用户信息，再发往信任接口。

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**pppoeplus start**	使能全局 PPPoE+功能。
3	JX(config)#**interface** *interface-type interface-number*	进入二层物理接口配置模式。
4	JX(config-ge-1/0/1)#**pppoeplus enable**	使能接口 PPPoE+功能。

配置 PPPoE+信任接口

配置 PPPoE+信任接口主要是为了降低 CPU 使用率，信任接口收到的 PPPoE 报文不会上送 CPU 处理，由交换芯片直接转发。通常配置在与 PPPoE 服务器相连的接口上。

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**interface** *interface-type interface-number*	进入二层物理接口配置模式。
3	JX(config-ge-1/0/1)#**pppoeplus trust**	配置 PPPoE+的信任接口。

8.4.5 配置 PPPoE+报文信息

PPPoE+功能主要是对 PPPoE 报文中的一个特定 Tag 进行处理，这个 Tag 包含 Circuit ID 和 Remote ID 两个字段。其中：

- Circuit ID 填充的是接收客户端请求报文接口的接口名、所属于的 VLAN ID（包括外层 VLAN ID 和内层 VLAN ID）
- Remote ID 填充的是接收客户端请求报文接口的 MAC 地址或者交换机的 MAC 地址。

配置 Circuit ID

Circuit ID 有三种填充模式：ascii 模式、default 模式和 user-define 模式。

各 Circuit ID 填充模式意义如下：

- ascii：即 Circuit ID 填充为命令行配置的字符串。
- default：即 Circuit ID 为默认值。
- user-define：即按照用户自定义格式填充 Circuit ID。

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**pppoeplus default circuit-id format** { **ascii** | **default** | **user-define** } *string*	配置交换机 Circuit ID。（default 模式不需要再配置字符串）
3	JX(config)#**interface** *interface-type interface-number*	进入二层物理接口配置模式。
4	JX(config-ge-1/0/1)#**pppoeplus circuit-id format** { **ascii** | default| **user-define** } *string*	在接口上配置 Circuit ID。 （default 模式不需要再配置字符串，如果接口和全局都配置了 circuit-id，则接口的配置优先生效）

配置 Remote ID

Remote ID 有三种填充模式：ascii 模式、default 模式和 user-define 模式。

各 Remote ID 填充模式意义如下：

- ascii：即 Circuit ID 填充为命令行配置的字符串。
- default：即 Circuit ID 为默认值。
- user-define：即按照用户自定义格式填充 Circuit ID。

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**pppoeplus default remote-id format** { **ascii** | **default** | **user-define** } *string*	配置交换机 Remote ID。（default 模式不需要再配置字符串）

步骤	配置	说明
3	JX(config)#**interface** *interface-type interface-number*	进入二层物理接口配置模式。
4	JX(config-ge-1/0/1)#**pppoeplus remote-id format** { **ascii** | default| **user-define** *string* }	在接口上配置 Remote ID。 （default 模式不需要再配置字符串，如果接口和全局都配置了 remote-id，则接口的配置优先生效）

配置 PPPoE+Tag 处理策略

由于某些原因，如某些信息字段的 Tag 可能是客户端伪造的，需要将报文原有的 Tag 替换掉。配置 PPPoE+Tag 收包处理策略为替换后，如果 PPPoE 报文已经携带信息字段 Tag，会将其替换；如果配置 PPPoE+Tag 收包处理策略为保持，则如果 PPPoE 报文已经携带信息字段 Tag，就保留原 tag；如果配置 PPPoE+Tag 收包处理策略为丢弃，则如果 PPPoE 报文已经携带信息字段 Tag，就会丢弃原 tag。

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**pppoeplus default policy** { **drop** | **keep** | **replace** }	配置全局的 PPPoE+Tag 的收包处理策略。
	JX(config-ge-1/0/1)# **pppoeplus policy** { **drop** | **keep** | **replace** }	配置接口上的 PPPoE+Tag 的收包处理策略。如果接口和全局都配置了收包处理策略，则接口的配置优先生效

8.4.6 检查配置

配置完成后，请在设备上执行以下命令检查配置结果。

序号	检查项	说明
1	JX#**show pppoeplus config**	查看 PPPoE+的配置信息。
2	JX#**show pppoeplus information**	查看 PPPoE+模块当前的全局配置信息，包括默认配置信息。

8.4.7 维护

用户可以通过以下命令，维护 PPPoE+特性的运行情况和配置情况。

命令	描述
JX(config)#**reset pppoeplus statistic [interface** *interface-type interface-number*]	清除 PPPoE+的统计信息，支持指定端口清除统计信息。
JX#**show pppoeplus interface** [*interface-type interface-number*]	查看 PPPoE+的接口上的配置和统计信息。

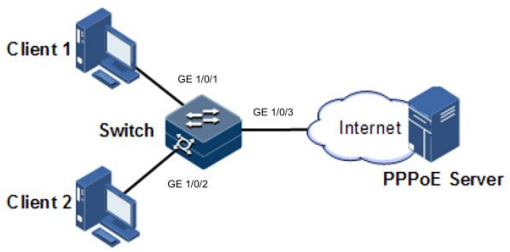
8.4.8 配置 PPPoE+示例

组网需求

如下图所示，为了防止在 PPPoE 认证过程中非法用户的接入，对用户上网进行控制和监管，可以在交换机上配置 PPPoE+功能，具体要求如下：

- 接口 GE 1/0/1 和 GE 1/0/2 分别连接 Client 1 和 Client 2，GE 1/0/3 连接 PPPoE 服务器；
- 使能全局 PPPoE+功能和 GE 1/0/1、GE 1/0/2、GE 1/0/3 的 PPPoE+功能，配置 GE 1/0/3 为信任接口；
- 配置 Circuit ID 类型为用户自定义，格式为接口名加外层 VLAN ID 加设备名；配置 Remote ID 类型为 ascii，内容为 01:02:03:04:05:06；
- 配置接口 GE 1/0/1 和 GE 1/0/2 的 PPPoE+Tag 的收包处理策略。

图 8-6 PPPoE+应用组网示意图



配置步骤

步骤 1 使能全局 PPPoE+功能，并使能 GE 1/0/1、GE 1/0/2 和 GE 1/0/3 的 PPPoE+功能。

```
JX(config)#pppoeplus start
JX(config)#interface ge 1/0/1
JX(config-ge-1/0/1)#pppoeplus enable
JX(config-ge-1/0/1)#exit
JX(config)#interface ge 1/0/2
JX(config-ge-1/0/2)#pppoeplus enable
JX(config-ge-1/0/2)#exit
JX(config)#interface ge 1/0/3
JX(config-ge-1/0/3)#pppoeplus enable
```

```
JX(config-ge-1/0/3)#exit
```

步骤 2 配置 GE 1/0/3 为信任接口。

```
JX#config
JX(config)#interface ge 1/0/3
JX(config-ge-1/0/3)#pppoeplus trust
JX(config-ge-1/0/3)#exit
```

步骤 3 配置 circuit-id 和 remote-id 的格式。

```
JX(config)#pppoeplus default circuit-id format user-
defined %portname:%svlan:%devicename
JX(config)#pppoeplus default remote-id format ascii 01:02:03:04:05:06
```

步骤 4 配置 GE 1/0/1 和 GE 1/0/2 的 PPPoE+Tag 的收包处理策略。

```
JX(config)#interface ge 1/0/1
JX(config-ge-1/0/1)#pppoeplus policy keep
JX(config-ge-1/0/1)#exit
JX(config)#interface ge 1/0/2
JX(config-ge-1/0/2)#pppoeplus policy drop
JX(config-ge-1/0/2)#exit
```

检查结果

通过 **show pppoeplus config** 命令查看设备上 PPPoE+功能的配置结果。

```
JX#show pppoeplus config
!
pppoeplus start
pppoeplus default remote-id format ascii 01:02:03:04:05:06
pppoeplus default circuit-id format user-
defined %portname:%svlan:%devicename
!
interface ge 1/0/1
 pppoeplus enable
 pppoeplus policy keep
!
interface ge 1/0/2
 pppoeplus enable
 pppoeplus policy drop
!
interface ge 1/0/3
 pppoeplus enable
 pppoeplus trust
```

8.5 安全 MAC

8.5.1 简介

接口安全 MAC 主要应用于网络边缘用户侧的交换设备上，用于保证某个接口接入数据的安全性，根据源 MAC 地址对输入的报文加以控制。用户可以启动接口安全功能

来限制和区分哪些用户可以通过安全接口来访问网络，只有接口安全 MAC 地址才能够访问网络，非安全 MAC 地址均按照用户配置的接口访问违例模式处理。

安全 MAC 地址分类

设备支持的安全 MAC 地址分为以下两类：

- 安全 MAC 地址

动态安全 MAC 地址是由设备自己学习得到的。用户可以在允许学习的 MAC 地址最大数目范围内，将学习到的 MAC 地址都设置为安全 MAC 地址。该类安全 MAC 地址不会被老化，但是不支持配置加载。

- Sticky 安全 MAC 地址

Sticky 安全 MAC 地址由用户在安全接口手动配置生成或者由安全 MAC 地址转化而来。与安全 MAC 地址不同，Sticky 安全 MAC 地址需要配合 Sticky 学习功能一起使用，支持配置加载：

说明

- 当 Sticky 学习功能使能时，接口下学习到的所有安全 MAC 地址均转换为 Sticky 安全 MAC 地址。
- 当 Sticky 学习功能禁止时，接口下所有 Sticky 安全 MAC 地址均转换为安全 MAC 地址。

安全 MAC 违例处理方式

当接口安全 MAC 的数目已经达到最大数目时，再有陌生源 MAC 报文输入则视为违规操作。对于非法的用户接入，根据安全 MAC 的违规策略配置交换机的不同处理方式如下：

- Protect 模式：对于非法接入的用户，安全接口直接丢弃该用户的报文。
- Restrict 模式：对于非法接入的用户，安全接口丢弃该用户的报文，同时在控制台打印 Syslog 信息，并发送告警信息至网管系统。
- Shutdown 模式：对于非法接入的用户，安全接口丢弃该用户的报文，同时在控制台打印 Syslog 信息、发送告警信息至网管系统并将该安全接口关闭。

注意

当发生 MAC 地址飘移，即安全接口 A 收到一个已经存在于安全接口 B 中的安全 MAC 所对应用户的访问时，安全接口 A 将其作为违例处理。

8.5.2 配置准备

场景

为了保证交换机接口接入数据的安全性，可以根据源 MAC 地址对输入的报文加以控制。通过安全 MAC 可以将接入接口配置成只允许特定的几个用户接入，也可以配置成允许特定数量的用户从该接口接入。但接入的用户超过限制时，接入的报文将按照安全 MAC 的违规策略进行处理。

前提

无

8.5.3 安全 MAC 功能的缺省配置

设备上安全 MAC 功能的缺省配置如下。

功能	缺省值
接口安全 MAC 功能状态	禁止
动态安全 MAC Sticky 学习功能状态	禁止
接口安全 MAC Trap 功能状态	禁止
接口安全 MAC 违规处理方式	**restrict**
接口安全 MAC 的最大数量	1024

8.5.4 配置安全 MAC 基本功能

注意

- 不建议用户在聚合组的单个成员接口上使能接口安全 MAC 功能。
- 不建议用户在同一接口上使能接口安全 MAC 功能的同时，使用 MAC 地址管理功能来配置静态 MAC 地址，会导致接口安全 MAC 功能失效。
- 当 802.1x 接口认证方式为基于 MAC 地址进行认证时，安全 MAC 功能与 802.1x 功能互斥，不建议用户在同一接口上同时进行配置。
- 安全 MAC 功能与基于接口和基于接口 VLAN 的 MAC 地址数目限制互斥，不能同时配置。

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。

步骤	配置	说明
2	JX(config)#**interface** *interface-type interface-number*	进入二层物理接口配置模式。
3	JX(config-ge-1/0/1)#**port-security enable**	使能接口安全 MAC 功能。
4	JX(config-ge-1/0/1)#**port-security maximum** *maximum*	配置接口安全 MAC 最大数目。
5	JX(config-ge-1/0/1)#port-security protect-action { **protect** | **restrict** | **shutdown** }	配置安全 MAC 违例模式。
6	JX(config-ge-1/0/1)#**shutdown** JX(config-ge-1/0/1)#**no shutdown**	将因违反接口安全 MAC 而被关闭的接口重新开启。
7	JX(config)# **port-security error-down recovery-interval** *second*	(可选)配置接口安全 MAC 恢复时间。

说明

当安全 MAC 违规策略为 Shutdown 模式时，可以使用该命令将因违反接口安全 MAC 而被关闭的接口重新开启。

当接口 Up 以后，配置的安全 MAC 违例模式继续保持。

8.5.5 配置接口 Sticky 安全 MAC 地址

注意

建议用户不要在 Sticky 安全 MAC 功能禁止的情况下配置 Sticky 安全 MAC 地址，否则可能导致功能异常。

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**interface** *interface-type interface-number*	进入二层物理接口配置模式。
3	JX(config-ge-1/0/1)#**port-security enable**	使能接口安全 MAC 功能。
4	JX(config-ge-1/0/1)#**port-security mac-address sticky enable**	使能 Sticky 安全 MAC 学习功能。
5	JX(config-ge-1/0/1)#**port-security mac-address sticky vlan** *vlan-id* **mac** *mac-address*	手动配置接口 Sticky 安全 MAC 地址。

说明

Sticky 安全 MAC 学习功能使能后，动态安全 MAC 地址转换为 Sticky 安全 MAC 地址；手动设置的 Sticky 安全 MAC 地址生效。

8.5.6 检查配置

配置完成后，请在设备上执行以下命令检查配置结果。

序号	检查项	说明
1	JX#**show mac-address config**	查看安全 MAC 的配置信息。
2	JX#**show mac-address** { **security** | **sticky** }	查看安全 MAC 地址表项信息。

8.5.7 维护

用户可以通过以下命令，维护设备安全 MAC 特性的运行情况和配置情况。

命令	描述
JX(config)#**no mac-address** { **security** | **sticky** } [*interface-type interface-number*]	清除指定类型的安全 MAC。

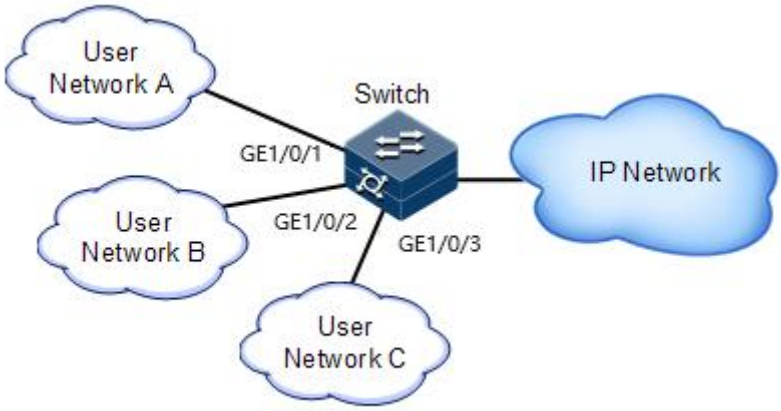
8.5.8 配置安全 MAC 示例

组网需求

如下图所示，交换机下联 3 个用户网络，为了保证交换机接口接入数据的安全性，要求配置如下：

- 接口 GE 1/0/1 最大允许 3 个用户接入网络。其中一个指定用户的 MAC 地址为 0000.0000.0001。其他 2 个用户为动态学习。违例模式采用 Protect 模式
- 接口 GE 1/0/2 最大允许 2 个用户接入网络。这 2 个用户的 MAC 地址通过学习确定。违例模式采用 Restrict 模式。
- 接口 GE 1/0/3 要求最大允许 1 个用户接入网络。该指定用户的 MAC 地址为 0000.0000.0002，违例模式采用 Shutdown 模式。

图 8-7 安全 MAC 应用组网示意图



配置步骤

步骤 1 配置 GE 1/0/1 接口安全 MAC。

```
JX#config
JX(config)#interface ge 1/0/1
JX(config-ge-1/0/1)#port-security enable
JX(config-ge-1/0/1)#port-security maximum 3
JX(config-ge-1/0/1)#port-security mac-aopddress sticky enable
JX(config-ge-1/0/1)#port-security mac-address sticky vlan 1
mac 00:00:00:00:00:01
JX(config-ge-1/0/1)#quit
```

步骤 2 配置 GE 1/0/2 的接口安全 MAC。

```
JX(config)#interface ge 1/0/2
JX(config-ge-1/0/2)#port-security enable
JX(config-ge-1/0/2)#port-security maximum 2
JX(config-ge-1/0/2)#port-security protect-action restrict
JX(config-ge-1/0/2)#quit
```

步骤 3 配置 GE 1/0/3 的接口安全 MAC。

```
JX(config)#interface ge 1/0/3
JX(config-ge-1/0/3)#port-security enable
JX(config-ge-1/0/3)#port-security maximum 1
JX(config-ge-1/0/3)#port-security protect-action shutdown
JX(config-ge-1/0/3)#port-security mac-address sticky enable
JX(config-ge-1/0/3)#port-security mac-address sticky vlan 1
mac 00:00:00:00:00:02
JX(config-ge-1/0/3)#quit
```

检查结果

通过 **show mac-address config** 查看安全 MAC 的接口配置是否正确。

```
JX#show mac-address config
!
```

```
mac-address aging-time 500
!
interface ge 1/0/1
 port-security enable
 port-security maximum 3
 port-security mac-address sticky enable
!
interface ge 1/0/2
 port-security enable
 port-security maximum 2
!
interface ge 1/0/3
 port-security enable
 port-security protect-action shutdown
 port-security mac-address sticky enable
```

通过 **show mac-address sticky** 查看设备上接口安全 MAC 地址配置及学习情况。

```
JX(config)#show mac-address sticky
  MacAddress         VLAN/VSI/BD Learned-From    Type       Valid
------------------------------------------------------------------------
  0000:0000:0001     1/--/--     ge-1/0/1        sticky     yes
  0000:0000:0002     1/--/--     ge-1/0/3        sticky     yes
------------------------------------------------------------------------
  Total:2       Static:0       Dynamic:0      Blackhole:0
  Sticky:2      Security:0      Snooping:0
```

8.6 风暴抑制与风暴控制

8.6.1 简介

二层网络是一个广播域，当接口接收到大量的广播、未知组播和未知单播报文时，就会产生广播风暴。如果不对广播风暴进行限制，就会耗费大量的网络带宽，造成网络速率下降，甚至造成通信中断，影响正常报文的转发。

风暴抑制与风暴控制能对网络中的广播流量进行限制，能在广播流量激增时抑制广播风暴的产生，从而保证正常报文的转发。

广播风暴产生

下面几种情况可能会产生广播风暴：

- 未知单播报文：目的 MAC 地址不在 MAC 地址表中的单播报文，即 DLF（Destination Lookup Failure，寻找目标失败）报文，如果某段时间内此种报文流量过多，进行大量的广播发送，可能会形成广播风暴。
- 未知组播报文：目的 MAC 地址不在 MAC 地址表中的组播报文，如果某段时间内此种报文流量过多，进行大量的广播发送，可能会形成广播风暴。
- 广播报文：目的 MAC 地址为广播的报文，如果某段时间内此种报文流量过多，可能会形成广播风暴。

风暴抑制原理

风暴抑制是对网络上可能形成广播风暴的广播、未知组播或未知单播报文进行过滤。当设备接收到的广播报文超过一定阈值时，将进行相应的动作。

风暴控制原理

风暴控制是对网络上可能形成广播风暴的广播、未知组播或未知单播报文进行过滤。当设备接收到的广播报文超过一定阈值时，将自动丢弃收到的广播报文。当未启用该功能或广播报文未达到一定阈值时，广播报文将被正常广播到设备的其它接口。

风暴抑制与风暴控制方式

方式有以下几种：

- BPS（Bits Per Second，每秒位数）：每秒允许通过的位数。
- PPS（Packets Per Second，每秒包数）：每秒允许通过的包数。
- Percent：以接口最大速率的百分比进行限速，仅物理口支持

8.6.2 配置准备

场景

在二层网络中配置风暴抑制功能，当网络中未知组播、未知单播和广播报文增多时可以抑制广播风暴的产生，从而保证正常报文的转发。

前提

无

8.6.3 风暴抑制和风暴控制的缺省配置

设备上风暴抑制的缺省配置如下。

功能	缺省值
广播流量风暴抑制状态	禁用
未知单播流量的风暴抑制状态	禁用
未知组播流量的风暴抑制状态	禁用
接口的风暴抑制动作	无
接口的恢复周期	30s
接口风暴抑制 Trap 功能	禁用
广播流量风暴控制状态	禁用
未知单播流量的风暴控制状态	禁用
未知组播流量的风暴控制状态	禁用

8.6.4 配置风暴抑制功能

注意

风暴抑制和基于 VLAN 的流量限速功能会相互影响，不建议用户在同一接口上同时开启这两个功能。

风暴抑制与风暴控制无法在同一接口上配置

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**interface** *interface-type interface-number*	进入二层物理接口节点。
3	JX(config-ge-1/0/1)#**storm-suppression** { **unknown-unicast** | **unknown-multicast** | **broadcast** } **min-rate** { **kbps** | **mbps** | **gbps** } *rate-value* **max-rate** { **kbps** | **mbps** | **gbps** } *rate-value*	使能接口下的风暴抑制功能，使用 BPS 配置风暴抑制的限速阈值。
	JX(config-ge-1/0/*)#**storm-suppression** { **unknown-unicast** | **unknown-multicast** | **broadcast** } **min-rate** *rate-value* **max-rate** *rate-value*	使能接口下的风暴抑制功能，使用 PPS 配置风暴抑制的限速阈值。
	JX(config-port-channel*)#**storm-suppression** { **unknown-unicast** | **unknown-multicast** | **broadcast** } **min-rate percent** *rate-value* **max-rate percent** *rate-value*	使能接口下的风暴抑制功能，使用比例配置风暴抑制的限速阈值。
4	JX(config-ge-1/0/1)#**storm-suppression action** { **block** | **error-down** | **none** }	配置接口的风暴抑制动作。
5	JX(config-ge-1/0/1)#**storm-suppression interval** *interval*	配置风暴抑制关闭接口后接口的恢复周期。
6	JX(config-ge-1/0/1)#**storm-suppression snmp-trap enable**	使能接口风暴抑制 Trap 功能。
7	JX(config-ge-1/0/1)#**exit**	返回全局配置模式。

8.6.5 检查风暴抑制配置

配置完成后，请在设备上执行以下命令检查配置结果。

序号	检查项	说明
1	JX#**show storm-suppression interface** [*interface-type interface-number*]	查看风暴抑制接口配置信息。
2	JX#**show storm-suppression information**	查看风暴抑制全局配置信息。

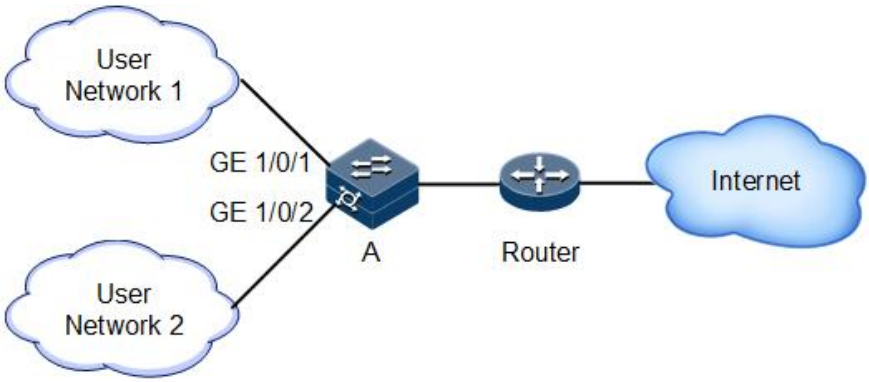
8.6.6 配置风暴抑制应用示例

组网需求

如下图所示，当 Switch A 的 GE 1/0/1 和 GE 1/0/2 接口接收到大量的未知单播或广播报文，Switch A 就会向 VLAN 内除了接收接口之外的所有接口转发这些报文，就可能会导致广播风暴，降低 Switch A 的转发性能。

为限制广播风暴对 Switch A 的影响，需要在 Switch A 的 GE 1/0/1 和 GE 1/0/2 接口上部署风暴抑制功能，分别限制来自用户网络 1 和用户网络 2 的广播报文，抑制阈值为 640kbit/s，若超限，则执行 shutdown 操作，低于 320kbit/s 才打开端口。

图 8-8 风暴抑制应用组网示意图



配置步骤

步骤 1 配置风暴抑制阈值。

```
JX(config)#interface ge 1/0/1
JX(config-ge-1/0/1)#storm-suppression broadcast min-rate kbps 320 max-rate kbps 640
JX(config-ge-1/0/1)#storm-suppression action error-down
JX(config-ge-1/0/1)#exit
JX(config)#interface ge 1/0/2
JX(config-ge-1/0/2)#storm-suppression broadcast min-rate kbps 320 max-rate kbps 640
JX(config-ge-1/0/2)#storm-suppression action error-down
```

检查结果

通过 **show storm-control** 查看风暴抑制配置是否正确。

```
JX# show storm-suppression interface
NOTE :
  UNMC: unknown multicast ; BC: broadcast ; UNC: unknown unicast
 Interface     Type    State   RateMode  Min/MaxRate          Action/Status
Interval
------------------------------------------------------------------------
----------------------
 ge-1/0/1      UNC    disable bps      n/a                  none/normal
5
              UNMC   disable bps      n/a                  none/normal
5
              BC     enable  bps      320/640 (kbit/s)      none/errordown
5
 ge-1//2      UNC    disable bps      n/a                  none/normal
5
              UNMC   disable bps      n/a                  none/normal
5
              BC     enable  bps      320/640 (kbit/s)      none/errordown
5
------------------------------------------------------------------------
----------------------
```

8.6.7 配置风暴控制功能

注意

风暴控制和基于 VLAN 的流量限速功能会相互影响，不建议用户在同一接口上同时开启这两个功能。

风暴抑制与风暴控制无法在同一接口上配置

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**interface** *interface-type interface-number*	进入二层物理接口节点。
3	JX(config-ge-1/0/*)#**storm-control** { **unknown-unicast** | **unknown-multicast** | **broadcast** | **unicast** | **multicast** } **cir** { **kbps** | **mbps** | **gbps** } *cir-value* **cbs** { **bytes** | **kbytes** | **mbytes** } *cbs-value*	使能接口下的风暴控制功能，使用 BPS 配置风暴控制的限速阈值。
	JX(config-ge-1/0/*)#**storm-control** { **unknown-unicast** | **unknown-multicast** | **broadcast** | **unicast** | **multicast** } **pps** *pps-value*	使能接口下的风暴控制功能，使用 PPS 配置风暴控制的限速阈值。

步骤	配置	说明
	JX(config-port-channel*)#**storm-control { unknown-unicast | unknown-multicast | broadcast | unicast | multicast } percent** *<1-100>*	使能接口下的风暴控制功能，使用比例配置风暴控制的限速阈值。
4	JX(config-ge-1/0/*)#**exit**	返回全局配置模式。

8.6.8 检查风暴控制配置

配置完成后，请在设备上执行以下命令检查配置结果。

序号	检查项	说明
1	JX#**show storm-control interface** [*interface-type interface-number*]	查看风暴控制接口配置信息。

8.6.9 配置风暴控制应用示例

组网需求

如图 8-7 所示，当 Switch A 的 GE 1/0/1 和 GE 1/0/2 接口接收到大量的未知单播或广播报文，Switch A 就会向 VLAN 内除了接收接口之外的所有接口转发这些报文，就可能会导致广播风暴，降低 Switch A 的转发性能。

为限制广播风暴对 Switch A 的影响，需要在 Switch A 的 GE 1/0/1 和 GE 1/0/2 接口上部署风暴控制功能，分别限制来自用户网络 1 和用户网络 2 的广播报文，承诺信息速率为 640kbit/s，承诺突发长度为 6400kbytes，超过的报文全丢弃。

配置步骤

步骤 1 配置风暴抑制阈值。

```
JX(config)#interface ge 1/0/1
JX(config-ge-1/0/1)#storm-control broadcast cir kbps 640 cbs kbytes 6400
JX(config-ge-1/0/1)#exit
JX(config)#interface ge 1/0/2
```

检查结果

通过 **show storm-control** 查看风暴控制配置是否正确。

```
JX# show storm-control interface
 NOTE :
  UNC: unknown unicast ; UNMC: unknown multicast ; NC: known unicast
  NMC: known multicast ; MC: all multicast ; UC: all unicast ; BC:
broadcast
 Interface         Type       State      RateMode    RateValue
-------------------------------------------------------------------------
-------
 ge-1/0/1          UNC        disable    pps         n/a
```

```
                    UNMC        disable     pps         n/a
                    BC          enable      bps         cir: 640(kbit/s),cbs:
6555600(byte)
                    UC          disable     pps         n/a
                    MC          disable     pps         n/a
 ge-1/0/2             UNC         disable     pps         n/a
                    UNMC        disable     pps         n/a
                    BC          enable      bps         cir: 640(kbit/s),cbs:
6555600(byte)
                    UC          disable     pps         n/a
                    MC          disable     pps         n/a
```

8.7 ARP 防攻击

8.7.1 配置准备

场景

ARP 协议有简单、易用的优点，但是也因为其没有任何安全机制而容易被攻击发起者利用。

攻击者可以仿冒用户、仿冒网关发送伪造的 ARP 报文，使网关或主机的 ARP 表被篡改。如果网络中有主机通过向设备发送大量目标 IP 地址不能解析的 IP 报文来攻击设备，则会造成下面的危害：

- 设备向目的网段发送大量 ARP 请求报文，加重目的网段的负载。
- 设备会试图反复地对目标 IP 地址进行解析，增加了 CPU 的负担。

为避免这种 IP 报文攻击所带来的危害，设备提供了下 ARP 防攻击功能。

前提

无

8.7.2 ARP 防攻击缺省配置

缺省没有配置任何 ARP 防攻击配置

8.7.3 配置 ARP 防攻击功能

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。

步骤	配置	说明
2	JX(config)# **arp-antiattack src-ip enable**	针对 ARP 源 IP 地址的 ARP 冲突检测。通过 ARP 报文中的源 IP 地址，查找 ARP 表，如果找到了对应的 ARP 表项，并且 ARP 表项中对应的 MAC 地址和 ARP 报文中的源 MAC 地址不一致，则认为是 ARP 冲突。会丢弃该 ARP 报文，防止攻击者篡改 ARP 表中的 MAC 地址。
3	JX(config)# **arp-antiattack src-mac enable**	针对 ARP 源 MAC 地址的 ARP 冲突检测。通过 ARP 报文中的源 MAC 地址，查找 ARP 表，如果找到了对应的 ARP 表项，并且 ARP 表项中对应的 IP 地址和 ARP 报文中的源 IP 地址不一致，则认为是 ARP 冲突。会丢弃该 ARP 报文，防止攻击者篡改 ARP 表中的 IP 地址。
4	JX(config)# **arp-antiattack arp-cheat enable**	仿冒本设备的 ARP 欺骗检测。检测设备收到 ARP 报文里的源或者目的 IP 地址，以及源 MAC 地址和 ARP 报文类型，判断 ARP 报文是否是仿冒本设备的攻击包，如果判断为攻击包，则会丢弃该 ARP 报文，并发送一个本设备的免费 ARP。
5	JX(config)# **arp-antiattack gratuitous-arp enable**	免费 ARP 报文主动丢弃。使能免费 ARP 报文主动丢弃功能后，设备直接丢弃免费 ARP 报文，可以防止设备因处理大量免费 ARP 报文，导致 CPU 负荷过重而无法处理其他业务。
6	JX(config)# **arp-antiattack gateway-cheat enable**	ARP 防网关冲突。通过 ARP 防网关冲突功能，可以防止用户仿冒网关发送 ARP 报文，非法修改网络内其他用户的 ARP 表项。ARP 防网关冲突和动态 ARP 检测不能同时配置。
7	JX(config-vlan*)# **arp-antiattack check user-bind enable** **或** JX(config-ge-*/*/*)# **arp-antiattack check user-bind enable**	动态 ARP 检测，即 DAI(Dynamic ARP Inspection)。使能动态 ARP 检测 DAI(Dynamic ARP Inspection)功能后，当设备收到 ARP 报文时，将此 ARP 报文的源 IP、源 MAC、收到 ARP 报文的接口及 VLAN 信息和 DHCP Snooping 绑定表的信息进行比较，如果信息匹配，则认为是合法用户，允许此用户的 ARP 报文通过，否则认为是攻击，丢弃该 ARP 报文。本功能仅适用于 DHCP Snooping 场景。

步骤	配置	说明
8	JX(config-vlan*)# **arp-antiattack check user-bind check-item { interfce | ip-address | mac-address }** **或** JX(config-ge-*/*/*)# **arp-antiattack check user-bind check-item { vlan | ip-address | mac-address }**	动态 ARP 检测的检测项，三个检查项可自由组合。默认是三个检查项都检查。

8.7.4 检查配置

配置完成后，请在设备上执行以下命令检查配置结果。

序号	检查项	说明
1	JX#**show arp-antiattack config**	查看 ARP 防攻击配置信息。

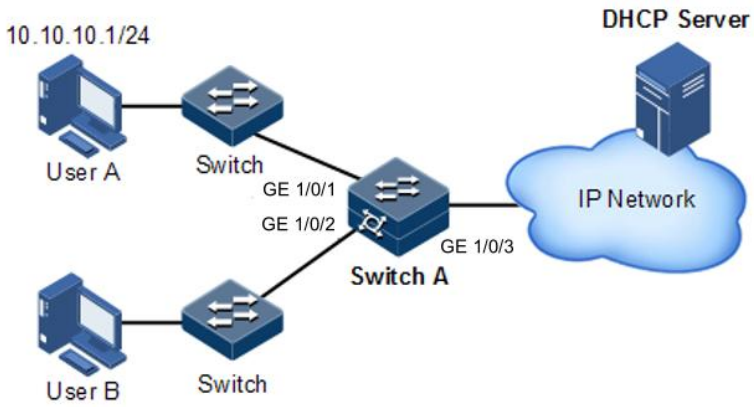
8.7.5 配置 ARP 防攻击示例

组网需求

为了防止 ARP 攻击，如下图所示，需要在 Switch A 设备上配置 ARP 防攻击功能。要求如下：

- 免费 ARP 报文不允许通过，使能针对 ARP 源 IP 地址和源 mac 地址的 ARP 冲突检测，以及仿冒本设备的 ARP 欺骗检测功能。
- Switch A 作为 User A 的网关，使能 ARP 防网关冲突。
- 清除 Switch A 上的 ARP 防网关冲突配置，Switch A 提供 User B 到 DHCP Server 的二层可达性，在 GE 1/0/2 上配置动态 ARP 检测功能。

图 8-9 动态 ARP 检测应用组网示意图



配置步骤

步骤 1 配置针对 ARP 源 IP 地址的 ARP 冲突检测。

```
JX#config
JX(config)#arp-antiattack src-ip enable
```

步骤 2 配置针对 ARP 源 MAC 地址的 ARP 冲突检测。

```
JX(config)# arp-antiattack src-mac enable
```

步骤 3 使能仿冒本设备的 ARP 欺骗功能。

```
JX(config)#arp-antiattack arp-cheat enable
```

步骤 4 使能免费 ARP 报文主动丢弃。

```
JX(config)#arp-antiattack gratuitous-arp enable
```

步骤 5 使能 ARP 防网关冲突。

```
JX(config)#arp-antiattack gateway-cheat enable
```

步骤 6 使能动态 ARP 检测，要先去使能 ARP 防网关冲突。

```
JX(config)#arp-antiattack gateway-cheat disable
JX(config)#interface ge 1/0/2
JX(config-ge-1/0/2)#arp-antiattack check user-bind enable
JX(config-ge-1/0/2)#exit
JX(config)#
```

检查结果

通过 **show arp-antiattack config** 命令查看设备上 ARP 防攻击的配置。

```
JX#show arp-antiattack config

Version: ANTIARP_VL2.10.00.00
!
```

```
arp-antiattack src-ip enable
arp-antiattack src-mac enable
arp-antiattack arp-cheat enable
arp-antiattack gratuitous-arp enable
arp-antiattack gateway-cheat enable

JX#show arp-antiattack config
Version: ANTIARP_VL2.10.00.00
!
arp-antiattack src-ip enable
arp-antiattack src-mac enable
arp-antiattack arp-cheat enable
arp-antiattack gratuitous-arp enable
!
interface ge 1/0/2
 arp-antiattack check user-bind enable
```

8.8 ND Snooping

8.8.1 简介

ND（Neighbor Discovery，邻居发现）是确定邻居节点之间关系的一组消息和进程。邻居发现协议替代了 IPv4 的 ARP（Address Resolution Protocol）、ICMP 路由器发现（Router Discovery）和 ICMP 重定向（Redirect）消息，并提供了地址冲突检测、邻居地址解析、确定邻居可达性以及进行主机地址配置等功能。

ND Snooping 功能主要应用于接入设备上，检查用户的合法性。对于合法用户的 ND 报文进行正常转发，否则直接丢弃，从而防止仿冒用户、仿冒网关的攻击。

用户合法性检查是根据 ND 报文中源 IPv6 地址和源 MAC 地址，检查用户是否是报文收到端口所属 VLAN 上的合法用户。

ND Snooping 功能将接入设备上的端口分为两种：ND 信任端口、ND 非信任端口。

- 对于 ND 信任端口：不进行用户合法性检查。从 ND 信任端口接收到的 ND 报文，设备可以进行正常转发。
- 对于 ND 非信任端口：从 ND 非信任端口接收到的 RA 报文，设备会认为是非法报文直接丢弃；从 ND 非信任端口接收到的 NA/RS/NS 报文，则会通过 ND 报文合法性检查功能进行绑定表匹配检查，当不符合绑定表关系时，设备会认为是非法报文直接丢弃；从 ND 非信任端口接收到的其他类型报文，设备进行正常转发。

8.8.2 配置准备

场景

ND Snooping 用来防止网络中常见的 ND 欺骗攻击，实现了对不安全来源的 ND 报文进行隔离。是否对 ND 报文信任通过配置接口的信任状态实现，而是否符合要求则通过配置绑定表实现。

前提

无

8.8.3 ND Snooping 的缺省配置

设备上 ND Snooping 的缺省配置如下。

功能	缺省值
ND Snooping 接口信任状态	不信任
全局 ND Snooping 功能	未启动
端口 ND Snooping 功能	未使能
NS、NA、RS 报文检查功能	未使能

8.8.4 配置 ND Snooping

请在需要的设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**nd-snooping start**	开启全局 ND Snooping 功能。
3	JX(config)#**interface** *interface-type interface-number*	进入二层物理接口或 VLAN 接口配置模式。
4	JX(config-ge-1/0/*)#**nd-snooping enable**	配置连接网关的接口开启 ND Snooping 功能。
5	JX(config-ge-1/0/*)#**nd-snooping trust**	配置连接网关的接口为信任接口。
6	JX(config)#**nd-snooping check { na | ns | rs } enable**	开启 ND Snooping 对 ns、na、rs 报文的合法性检查。

8.8.5 检查配置

配置完成后，请在设备上执行以下命令检查配置结果。

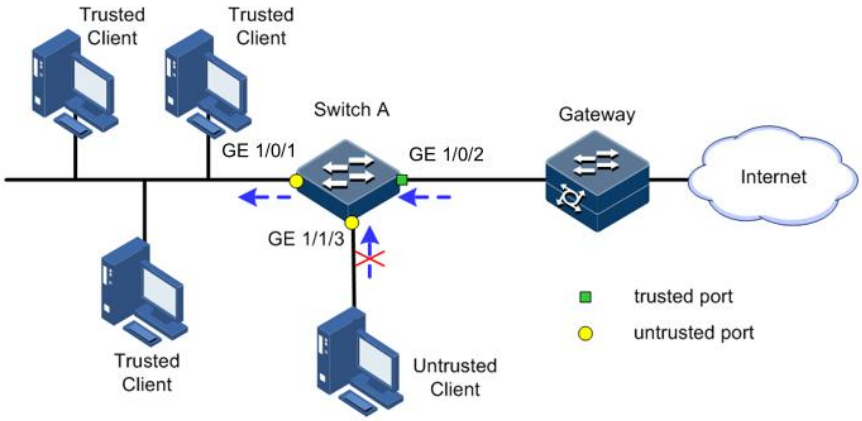
序号	检查项	说明
1	JX#**show nd-snooping config**	查看 ND Snooping 功能配置信息。
2	JX#**show nd-snooping user-bind**	查看 ND Snooping 用户绑定表。
3	JX#**show nd-snooping prefix**	查看 ND Snooping 前缀表。

8.8.6 配置 ND Snooping 示例

组网需求

如下图所示，某局域网络用户主机通过 Switch A 连接网关设备，由于网络中未部署 DHCPv6 服务器，这些主机需要根据网关分配给用户网络前缀信息，通过无状态地址自动配置方式获取 IPv6 地址。为防止非法用户发送 NA/NS/RS/RA 报文，导致合法主机无法获取 IPv6 地址，需要在 Switch 上开启 ND Snooping 功能，对非法的报文进行拦截。

图 8-10 配置 ND Snooping 组网示意图



配置步骤

步骤 1 在 Switch 上创建 VLAN 10 并激活。

配置 Switch。

```
JX#config
JX(config)#hostname SwitchA
SwitchA(config)#vlan 10
```

步骤 2 将 Switch A 的接口 GE 1/0/2 以 Access 模式加入 VLAN 10，设置接口 GE 1/0/1 为 Trunk 模式并允许 VLAN 10 通过。

```
SwitchA(config)#interface ge 1/0/2
SwitchA(config-ge-1/0/2)#port link-type access
SwitchA(config-ge-1/0/2)#port default vlan 10
SwitchA(config-ge-1/0/2)#exit
SwitchA(config)#interface ge 1/0/1
SwitchA(config-ge-1/0/1)#port link-type trunk
SwitchA(config-ge-1/0/1)#port trunk allow-pass vlan 10
SwitchB(config-ge-1/0/1)#exit
```

步骤 3 全局和 VLAN 10 分别使能 ND Snooping 功能，配置 GE 1/0/1 为信任接口。

```
SwitchA(config)#nd-snooping start
SwitchA(config)#vlan 10
SwitchA(config-vlan-10)#nd-snooping enable
SwitchA(config-vlan-10)#exit
SwitchA(config)#interface ge 1/0/1
SwitchA(config-ge-1/0/1)#nd-snooping enable
SwitchA(config-ge-1/0/1)#nd-snooping trust
SwitchA(config-ge-1/0/1)#exit
```

步骤 4 使能 ND 协议报文合法性检查功能。

```
SwitchA(config)#nd-snooping check na enable
SwitchA(config)#nd-snooping check ns enable
SwitchA(config)#nd-snooping check rs enable
```

检查结果

通过 **show nd-snooping config** 命令查看配置是否正确。

```
JX#show nd-snooping config
!
nd-snooping start
nd-snooping check na enable
nd-snooping check ns enable
nd-snooping check rs enable
!
vlan 10
 nd-snooping enable
!
interface ge 1/0/1
nd-snooping enable
nd-snooping trust
```

8.9 DHCP Snooping

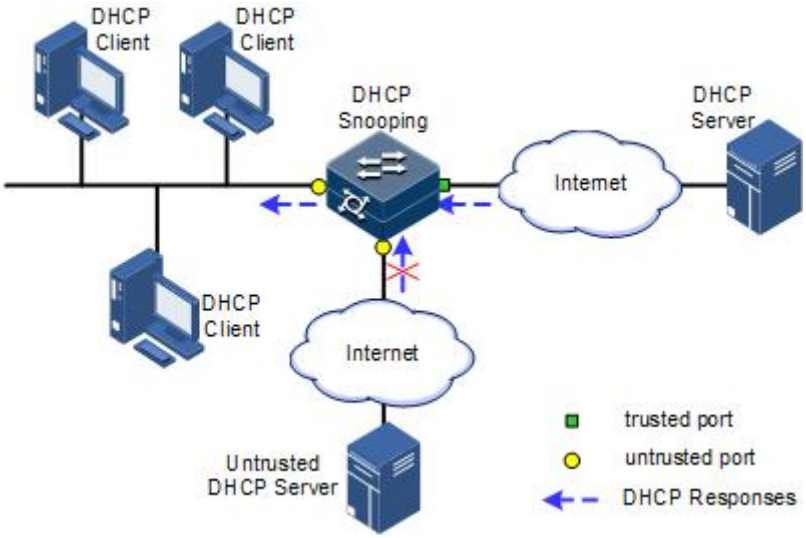
8.9.1 简介

DHCP Snooping 是 DHCP 的一种安全特性，具有如下功能：

- 保证 DHCP 客户端从合法的 DHCP 服务器获取 IP 地址；

网络中如果存在私自架设的伪 DHCP 服务器，则可能导致 DHCP 客户端获取错误的 IP 地址和网络配置参数，无法正常通信。如图 8-11 所示，为了使 DHCP 客户端能通过合法的 DHCP 服务器获取 IP 地址，DHCP Snooping 安全机制允许将接口设置为信任接口和不信任接口：信任接口正常转发接收到的 DHCP 报文；不信任接口接收到来自 DHCP 服务器的回应报文后将其丢弃。

图 8-11 DHCP Snooping 组网示意图



- 记录 DHCP 客户端 IP 地址与 MAC 地址的对应关系。

DHCP Snooping 通过监听请求和信任接口收到的回应报文，记录 DHCP Snooping 表项，其中包括客户端的 MAC 地址、获取到的 IP 地址、与 DHCP 客户端连接的接口及该接口所属的 VLAN 等信息。利用这些信息可以实现：

- ARP Detection：根据 DHCP Snooping 表项来判断发送 ARP 报文的用户是否合法，从而防止非法用户的 ARP 攻击。
- IP Source Guard：通过动态获取 DHCP Snooping 表项对接口转发的报文进行过滤，防止非法报文通过该接口。
- VLAN 映射：发送给用户的报文通过查找映射 VLAN 对应的 DHCP Snooping 表项中的 DHCP 客户端 IP 地址、MAC 地址和原始 VLAN 的信息，将报文的映射 VLAN 修改为原始 VLAN。

DHCP 报文中的 Option 字段记录了 DHCP 客户端的位置信息。管理员可以利用该选项定位 DHCP 客户端，实现对客户端的安全和计费等控制。

如果设备配置了 DHCP Snooping 支持 Option 功能：

- 当设备接收到 DHCP 请求报文时，将根据报文中是否包含 Option 字段以及用户配置的处理策略及填充模式等对报文进行相应的处理，并将处理后的报文转发给 DHCP 服务器；
- 当设备接收到 DHCP 回应报文时，如果报文中含有 Option 字段，则删除该字段，并转发给 DHCP 客户端；如果报文中不含有 Option 字段，则直接转发。

8.9.2 配置准备

场景

DHCP Snooping 作为 DHCP 的一种安全特性，用于保证 DHCP 客户端从合法的 DHCP 服务器获取 IP 地址，并记录 DHCP 客户端 IP 地址与 MAC 地址的对应关系。

DHCP 报文中的 Option 字段记录了 DHCP 客户端的位置信息。管理员可以利用该选项定位 DHCP 客户端，实现对客户端的安全和计费等控制。配置了 DHCP Snooping 支持 Option 功能的交换机设备可以根据报文中是否包含 Option 字段，对其进行相应的处理。

前提

无

8.9.3 DHCP Snooping 的缺省配置

设备上 DHCP Snooping 的缺省配置如下。

功能	缺省值
全局 DHCP Snooping 状态	禁止
接口 DHCP Snooping 状态	去使能
接口信任状态	不信任
DHCP Snooping 支持 Option 82	禁止

8.9.4 配置 DHCP Snooping

通常情况下，需要确保设备连接 DHCP 服务器侧的接口为信任状态，连接用户侧的接口为不信任状态。

对于启动了 DHCP Snooping 的设备，如果没有配置 DHCP Snooping 支持 Option 功能，则设备将不会对报文的 Option 字段进行任何处理。对于没有携带 Option 字段的报文，设备也不会进行插入处理。

缺省情况下设备所有接口的 DHCP Snooping 功能均已使能，但只有在使能全局 DHCP Snooping 功能后，接口的 DHCP Snooping 功能才会生效。

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**dhcp-snooping start**	使能全局 DHCP Snooping 功能。
3	JX(config)#**interface** *interface-type interface-number*	进入物理接口配置模式。
4	JX(config-ge-1/0/*)#**dhcp-snooping enable**	使能接口的 DHCP Snooping 功能，支持 QinQ 接口。
5	JX(config-ge-1/0/*)#**dhcp-snooping trust**	配置 DHCP Snooping 信任接口。
6	JX(config-ge-1/0/*)#**dhcp-snooping max-user-number** { *number* | **default** }	配置 DHCP Snooping 绑定表容量。

步骤	配置	说明
7	JX(config-ge-1/0/*)#**dhcp-snooping server-filter enable**	配置 DHCP Snooping 信任 server 配置功能
8	JX(config-ge-1/0/*)#**dhcp-snooping check user-bind enable**	配置 DHCP 报文进行绑定表匹配检测功能
9	JX(config-ge-1/0/*)#**dhcp-snooping check mac-address enable**	配置 DHCP 用户上送的请求报文头中的 MAC 地址是否合法功能
10	JX(config-ge-1/0/*)#**dhcp-snooping option82 enable**	配置 DHCP Snooping 支持 Option82 功能。
11	JX(config)#**exit**	返回全局配置模式。

8.9.5 配置 DHCP Snooping 支持 Option 82 功能

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**dhcp-snooping start**	使能全局 DHCP Snooping 功能。
3	JX(config)#**interface** *interface-type interface-number*	进入物理接口配置模式。
4	JX(config-ge-1/0/*)#**dhcp-snooping enable**	使能接口的 DHCP Snooping 功能。
5	JX(config-ge-1/0/*)#**dhcp-snooping option82 enable**	配置全局开启 DHCP Snooping 支持 Option 82 功能。
6	JX(config-ge-1/0/*)#**dhcp-snooping option82 circuit-id format** { **default** | **user-defined** format-string }	配置 DHCP Snooping Option82 的 circuitID 字段内容。
7	JX(config-ge-1/0/*)#**dhcp-snooping option82 remote-id format** { **default** | **user-defined** format-string }	配置 DHCP Snooping Option82 的 remoteID 字段内容。
8	JX(config-ge-1/0/*)#**dhcp-snooping option82** { **drop** | **keep** | **append** }	配置 DHCP Snooping 对含 Option 82 的 DHCP 请求报文处理策略。
9	JX(config-ge-1/0/*)#**exit**	返回全局配置模式。

8.9.6 配置 DHCPv6 Snooping

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。

步骤	配置	说明
2	JX(config)#**dhcp-snooping start**	使能全局 DHCP Snooping 功能。
3	JX(config)#**interface** *interface-type interface-number*	进入物理接口配置模式。
4	JX(config-ge-1/0/*)#**dhcpv6-snooping enable**	使能接口的 DHCPv6 Snooping 功能。
5	JX(config-ge-1/0/*)#**dhcpv6-snooping trust**	配置 DHCPv6 Snooping 信任接口。
6	JX(config-ge-1/0/*)#**dhcpv6-snooping max-user-number** { *number* | **default** }	配置 DHCPv6 Snooping 绑定表容量。
7	JX(config-ge-1/0/*)#**dhcpv6-snooping option18 enable**	使能 DHCPv6 Snooping 支持 Option18 功能。
8	JX(config-ge-1/0/*)#**dhcpv6-snooping option18 format** { **default** | **user-defined** format-string }	配置 DHCPv6 Snooping Option18 的内容。
9	JX(config-ge-1/0/*)#**dhcpv6-snooping option37 enable**	使能 DHCPv6 Snooping 支持 Option37 功能。
10	JX(config-ge-1/0/*)#**dhcpv6-snooping option37 format** { **default** | **user-defined** format-string }	配置 DHCPv6 Snooping Option37 的内容。

8.9.7 检查配置

配置完成后，请在设备上执行以下命令检查配置结果。

序号	检查项	说明
1	JX#**show dhcp-snooping config**	查看 DHCP Snooping 功能配置信息。
2	JX#**show dhcp-snooping user-bind**	查看 DHCP Snooping 绑定表信息。
3	JX#**show dhcp-snooping statistics**	查看 DHCP Snooping 的报文统计信息。
4	JX#**show dhcp-snooping interface**	查看 DHCP Snooping 的接口配置信息。
5	JX#**show dhcp-snooping vlan**	查看 DHCP Snooping 的 vlan 配置信息。
6	JX#**show dhcpv6-snooping config**	查看基于 ipv6 的 DHCP Snooping 功能配置信息。
7	JX#**show dhcpv6-snooping user-bind**	查看基于 ipv6 的 DHCP Snooping 绑定表信息。
8	JX#**show dhcpv6-snooping statistics**	查看基于 ipv6 的 DHCP Snooping 的报文统计信息。

序号	检查项	说明
9	JX#show dhcpv6-snooping interface	查看基于 ipv6 的 DHCP Snooping 的接口配置信息。
10	JX#show dhcpv6-snooping vlan	查看基于 ipv6 的 DHCP Snooping 的 vlan 配置信息。

8.9.8 维护

用户可以通过以下命令，维护设备 DHCP Snooping 特性的运行情况和配置情况。

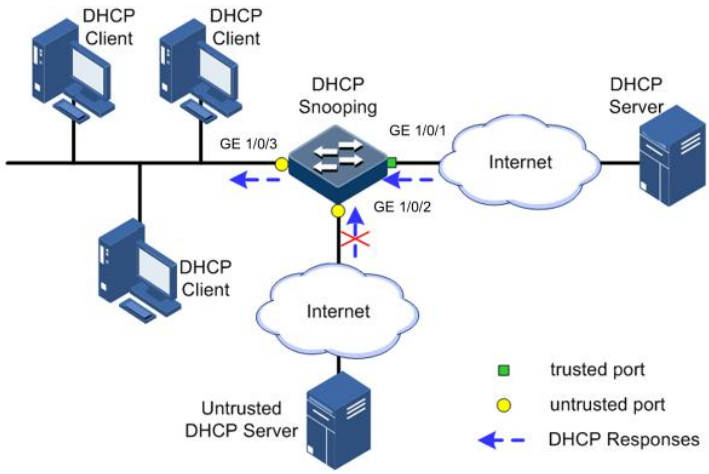
命令	描述
JX(config)#reset dhcp-snooping user-bind	清除 dhcpsnoop ipv4 协议下的用户绑定表信息
JX(config)#reset dhcpv6-snooping user-bind	清除 dhcpsnoop ipv6 协议下的用户绑定表信息
JX(config)#reset dhcp-snooping statistics	清除 dhcpsnoop ipv4 协议下的用户接口配置统计信息
JX(config)#reset dhcpv6-snooping statistics	清除 dhcpsnoop ipv6 协议下的用户接口配置统计信息

8.9.9 配置 DHCP Snooping 示例

组网需求

如图 8-12 所示，Switch 作为 DHCP Snooping 设备，需要保证 DHCP 客户端从合法的 DHCP 服务器获取 IP 地址，此外为了便于对客户端的管理，还需要设备支持 Option82 功能，在接口 GE 1/01/3 上配置电路 ID 子选项信息填充内容为 JX，远程 ID 子选项信息填充内容为 user01。

图 8-12 配置 DHCP Snooping 组网示意图



配置步骤

步骤 1 配置全局 DHCP Snooping 功能。

```
JX#config
JX(config)#dhcp-snooping start
```

步骤 2 配置信任接口。

```
JX(config)#interface ge 1/0/1
JX(config-ge-1/0/1)#dhcp-snooping enable
JX(config-ge-1/0/1)#dhcp-snooping trust
JX(config-ge-1/0/1)#exit
```

步骤 3 配置 DHCP Snooping 支持 Option82 功能并配置 Option82 字段。

```
JX(config)#interface ge 1/0/3
JX(config-ge-1/0/3)#dhcp-snooping enable
JX(config-ge-1/0/3)#dhcp-snooping option82 enable
JX(config-ge-1/0/3)#dhcp-snooping option82 remote-id format user-defined
'user01'
JX(config-ge-1/0/3)#dhcp-snooping option82 circuit-id format user-defined
'JX'
JX(config-ge-1/0/3)#exit
```

检查结果

通过 **show dhcp-snooping config** 命令查看 DHCP 服务器配置是否正确。

```
JX#show dhcp-snooping config
 Version : DHCPSNOOP_V7.00.03.00
!
dhcp-snooping start
!
interface ge 1/0/1
 dhcp-snooping enable
```

```
 dhcp-snooping trust
!
interface ge 1/0/3
 dhcp-snooping enable
 dhcp-snooping option82 enable
 dhcp-snooping option82 remote-id format user-defined 'user01'
 dhcp-snooping option82 circuit-id format user-defined 'JX'
……
```

8.10 IP Source Guard

8.10.1 简介

IP Source Guard 功能用于对接口收到的报文进行过滤控制，通常配置在接入用户侧的接口上，以防止非法用户报文通过，从而限制了对网络资源的非法使用（比如非法主机仿冒合法用户 IP 接入网络），提高了接口的安全性。

IP Source Guard 绑定表项

IP Source Guard 用于匹配报文的特征项包括源 IP 地址、源 MAC 地址和 VLAN 标签，并且可支持接口与以下特征项的组合（以下简称绑定表项）：

- 接口＋IP
- 接口＋IP＋MAC
- 接口＋IP＋VLAN
- 接口＋IP＋MAC＋VLAN

按照绑定表项的产生方式，IP Source Guard 分为以下两种：

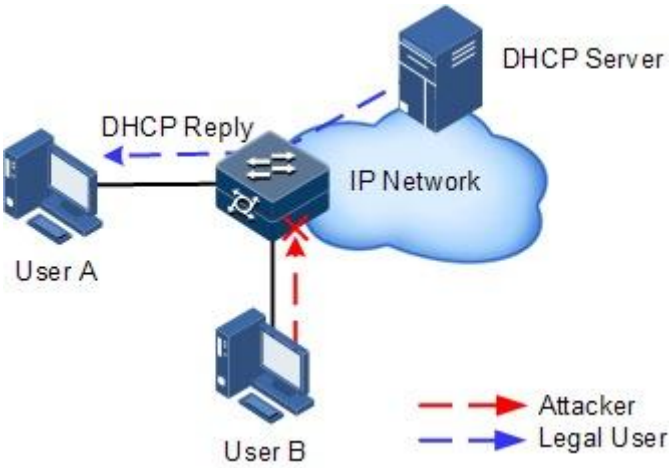
- 静态绑定：通过手工配置绑定信息，产生绑定表项来完成接口的控制功能，适用于主机数较少或者需要对某台主机进行单独绑定的情况。
- 动态绑定：通过 DHCP Snooping 自动获取绑定信息来完成接口的控制功能，适用于主机数较多并且采用 DHCP 进行动态主机配置的情况，可有效防止 IP 地址冲突和盗用等问题。

IP Source Guard 原理

IP Source Guard 的基本原理是在设备内部构建一个 IP 源绑定表，作为每个接口对接收数据包的检验依据。IP Source Guard 原理如下图所示，其转发原则如下：

- 所接收到的 IP 报文满足 IP 源绑定表中 Port/IP/MAC/VLAN 绑定表项的对应关系，则转发；
- 所接收到的 IP 报文为 DHCP 数据包，则转发；
- 所接收到的 IP 报文为其他情况，则丢弃。

图 8-13 IP Source Guard 功能示意图



当设备在转发 IP 报文时，将此 IP 报文中的源 IP、源 MAC、接口、VLAN 信息和绑定表的信息进行比较，如果信息匹配，说明是合法用户，则允许此报文正常转发；否则认为是攻击者，丢弃该用户发送的 IP 报文。

8.10.2 配置准备

场景

网络中常常存在针对 IP 源进行欺骗的攻击行为，如攻击者仿冒合法用户发送 IP 报文给服务器，或者伪造其他用户的源 IP 地址进行通信，从而导致合法用户不能正常获得网络服务。

通过 IP Source Guard 绑定功能，可以对接口转发的报文进行过滤控制，防止非法报文通过接口，从而限制了对网络资源的非法使用，如非法主机仿冒合法用户 IP 接入网络等，提高了接口的安全性。

前提

配置 IP Source Guard 之前，需要完成以下任务：

- 如果存在 DHCP 用户，则需要使能 DHCP Snooping 功能。

8.10.3 IP Source Guard 功能的缺省配置

设备上 IP Source Guard 功能的缺省配置如下。

功能	缺省值
接口 IP Source Guard 状态	禁止

8.10.4 配置 IP Source Guard 绑定功能

配置静态绑定功能

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**user-bind static ip** *ip-address/any* ***mac*** *mac-address/any* [**interface** *interface-type interface-number*] **vlan** *vlan-id/any*	配置静态绑定关系。
	JX(config)#**user-bind static ip6** *ipv6-address/any* ***mac*** *mac-address/any* [**interface** *interface-type interface-number***vlan** *vlan-id/any*	配置 IPv6 静态绑定关系。

8.10.5 配置 IP Source Guard 接口信任状态

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**interface** *interface-type interface-number*	进入物理接口配置模式。
3	JX(config-ge-1/0/*)#**ip source check user-bind enable**	使能接口 IP Source Guard 功能。

8.10.6 检查配置

配置完成后，请在设备上执行以下命令检查配置结果。

序号	检查项	说明
1	JX#**show ip source check config**	查看 IP Source Guard 功能配置信息。
2	JX#**show ip source check interface**	查看已使能 IP Source Guard 的接口信息

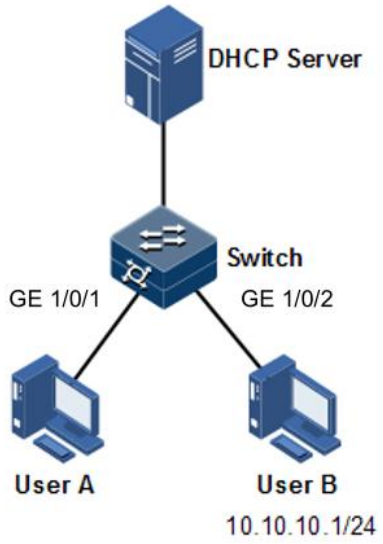
8.10.7 配置 IP Source Guard 示例

组网需求

如下图所示，为了防止 IP 地址盗用，需要在交换机上配置 IP Source Guard 功能，要求如下：

- 交换机允许接口 GE 1/0/1 上的所有 IP 报文通过；
- 接口 GE 1/0/2 允许指定 IP 地址为 10.10.10.1，子网掩码为 255.255.255.0 的 IP 报文以及符合 DHCP Snooping 学习到的动态绑定关系的报文通过；
- 其它接口仅允许通过 DHCP Snooping 学习的动态绑定关系的报文通过。

图 8-14 IP Source Guard 应用组网示意图



配置步骤

步骤 1 配置 IP Source Guard 功能。

```
JX#config
JX(config)#int ge 1/0/2
JX(config-ge-1/0/2)#ip source check user-bind enable
JX(config-ge-1/0/2)#exit
JX(config)#int ge 1/0/1
JX(config-ge-1/0/1)#ip source check user-bind enable
JX(config-ge-1/0/1)#exit
```

步骤 2 配置静态绑定表项。

```
JX(config)# user-bind static ip 10.10.10.1 mac any interface ge 1/0/2
vlan any
```

步骤 3 接口 GE 1/0/2 和其他接口配置 DHCP snooping 功能。

检查结果

通过 **show ip source check config** 命令查看 IP Source Guard 配置结果。

```
JX#show ip source check config
!
user-bind static ip 10.10.10.1 mac any interface ge 1/0/2 vlan any
!
interface ge 1/0/2
ip source check user-bind enable
!
interface ge 1/0/1
ip source check user-bind enable
```

通过 **show ip source check interface** 命令查看已使能 IP Source Guard 的接口信息。

```
JX#show ip source check interface
Interface         Check-Item        Alarm    Limit      DropPkts
------------------------------------------------------------------------
ge-1/0/2          ip,mac,vlan       disable  100        0
ge-1/0/1          ip,mac,vlan       disable  100        0
------------------------------------------------------------------------
```

8.11 CPU 防攻击

8.11.1 配置准备

场景

当设备短时间内接收到大量的攻击报文，导致 CPU 满负荷运转，利用率达到 100%，会导致设备的正常功能无法运行。使用 CPU 防攻击功能可以有效限制进入 CPU 的报文的速率。

前提

无

8.11.2 配置 CPU 防攻击限速功能

注意

CPU 防攻击的配置对各协议模块的功能有重要的影响，建议不要轻易修改 CPU 保护的参数配置，只有专家才能修改相关配置。

请在设备上进行以下配置。

步骤	配置	说明
1	JX#config	进入全局配置模式。

步骤	配置	说明
2	JX(config)# **cpu-defend policy test**	配置 CPU 防攻击的策略，设备在启动的时候，会下发并绑定一个默认策略，默认策略不能修改，但是设备可以更改绑定的策略。
3	JX(config-cpudefend-policy-test)# **car packet-type** { **arp** | **bfd** | **dhcpreply** | **dhcprequest** | **igmp** | **lacp** | **lldp** | **mld** | **stp-customer** | **telnet** | **...** } **pps** *pps-value*	配置该策略的指定协议包类型的限速值。设备支持哪些协议包类型，取决于设备支持哪些协议。
4	JX(config-cpudefend-policy-test)# **packet-type** { **arp** | **bfd** | **dhcpreply** | **dhcprequest** | **igmp** | **lacp** | **lldp** | **mld** | **stp-customer** | **telnet** | **priority** } *priority-value*	配置该策略的指定协议包类型的优先级。设备支持哪些协议包类型，取决于设备支持哪些协议。
5	JX(config-cpudefend-policy-test)# **deny packet-type** { **arp** | **bfd** | **dhcpreply** | **dhcprequest** | **igmp** | **lacp** | **lldp** | **mld** | **stp-customer** | **telnet** | **...** }	配置该策略的指定协议包类型的丢弃。设备支持哪些协议包类型，取决于设备支持哪些协议。
6	JX(config-cpudefend-policy-test)# **session-car packet-type** { **bgp** | **ftp** | **http** | **isis** | **ospf** | **ospfv3** | **ssh** | **telnet** | **tftp** | **...** } **pps** *pps-value*	配置该策略的指定协议包类型的动态链路保护功能的协议会话的限速值。
7	JX(config-cpudefend-policy-test)# **filter** *<1-8>* { **acl-ipv4** *acl-listid* | **acl-ipv6** *acl-listid* }	配置该策略的过滤器。
8	JX(config)# **cpu-defend bind-policy test**	配置 CPU 防攻击的绑定策略。绑定后的策略不能修改，只能先解绑定后再修改。在基于 CPU 利用率动态调节模式下，该命令不可配置。
9	JX(config)# **cpu-defend mode** { **cpuratio-control** | **fixed** }	配置 CPU 防攻击的绑定策略的模式。分为基于 CPU 利用率动态调节和固定模式。默认是固定模式。
10	JX(config)# **cpu-defend** { **level1-policy** *policy-name* | **level2-policy** *policy-name* | **level3-policy** *policy-name* }	配置用于基于 CPU 利用率动态调节绑定策略模式下的各 level 的策略。
11	JX(config)# **cpu-defend** { **level1-cpuratio** *<1-100>* | **level2-cpuratio** *<1-100>* }	配置用于基于 CPU 利用率动态调节绑定策略模式下的 level-1 或者 level-2 的最大 CPU 利用率。

8.11.3 配置 CPU 防攻击的攻击溯源功能

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)# **cpu-defend policy test**	配置 CPU 防攻击的策略，设备在启动的时候，会下发并绑定一个默认策略，默认策略不能修改，但是设备可以更改绑定的策略。
3	JX(config-cpudefend-policy-test)# **auto-defend enable**	配置该策略的攻击溯源功能使能，缺省情况下，该功能是使能的。
4	JX(config-cpudefend-policy-test)# **auto-defend threshold** *threshold-value*	配置该策略的攻击溯源门限值。缺省值为 128pps。
5	JX(config-cpudefend-policy-test)# **auto-defend attack-packet sample** *sample-value*	配置该策略的攻击溯源采样比。缺省值为 8。
6	JX(config-cpudefend-policy-test)# **auto-defend alarm enable**	配置该策略的攻击溯源的告警功能使能。缺省为未使能。
7	JX(config-cpudefend-policy-test)# **auto-defend alarm threshold** *threshold-value*	配置该策略的攻击溯源的告警阈值。缺省值为每秒触发攻击溯源 128 次。
8	JX(config-cpudefend-policy-test)# **auto-defend action** { **deny** | **error-down** }	配置该策略的攻击溯源的惩罚动作。缺省为没有惩罚动作。
9	JX(config)# **cpu-defend bind-policy test**	配置 CPU 防攻击的应用。绑定后的策略不能修改，只能先解绑定后再修改。

8.11.4 检查配置

配置完成后，请在设备上执行以下命令检查配置结果。

序号	检查项	说明
1	JX#**show cpu-defend config**	查看 CPU 防攻击配置信息。
	JX#**show cpu-defend statistics**	查看 CPU 防攻击统计信息。
3	JX#**show auto-defend attack-source**	查看 CPU 防攻击攻击溯源信息

8.11.5 维护

命令	说明
JX(config)#**reset cpu-defend statistics**	清除全局的 CPU 防攻击统计信息。

8.11.6 MAC 认证

8.11.7 简介

MAC 地址认证是一种基于端口和 MAC 地址对用户的网络访问权限进行控制的认证方法，无需安装客户端软件。设备在启动了 MAC 地址认证的端口上首次检测到用户的 MAC 地址以后，启动对该用户的认证操作。认证过程中，不需要用户手动输入用户名或密码。若该用户认证成功，则允许其通过端口访问网络资源，否则该用户的 MAC 地址就被设置为静默 MAC。在静默时间内，来自此 MAC 地址的用户报文到达时，设备直接做丢弃处理，以防止非法 MAC 短时间内的重复认证。

MAC 认证用户的账号格式：

- MAC 地址账号：设备使用源 MAC 地址作为用户认证时的用户名和密码，或者使用 MAC 地址作为用户名，并配置密码。
- 固定用户名账号：所有 MAC 地址认证用户均使用设备上指定的一个固定用户名和密码替代用户的 MAC 地址作为身份信息进行认证。

认证方式：

- RADIUS 服务器认证方式进行 MAC 地址认证：当选用 RADIUS 服务器认证方式进行 MAC 地址认证时，设备作为 RADIUS 客户端，与 RADIUS 服务器配合完成 MAC 地址认证操作。
- 本地认证方式进行 MAC 地址认证：当选用本地认证方式进行 MAC 地址认证时，直接在设备上完成对用户的认证。需要在设备上配置本地用户名和密码。

重认证：

- MAC 地址重认证是指设备周期性对端口上在线的 MAC 地址认证用户发起重认证，以检测用户连接状态的变化、确保用户的正常在线。

8.11.8 缺省配置

设备上 MAC 认证的缺省配置如下：

功能	缺省值
全局 MAC 认证功能状态	禁止
接口 MAC 认证功能状态	禁止
全局认证方式	**pap**
802.1x 重认证功能状态	允许

功能	缺省值
802.1x 重认证定时器时间	1800s
MAC 认证静默定时器时间	60s
最大用户数	256

8.11.9 配置 MAC 认证

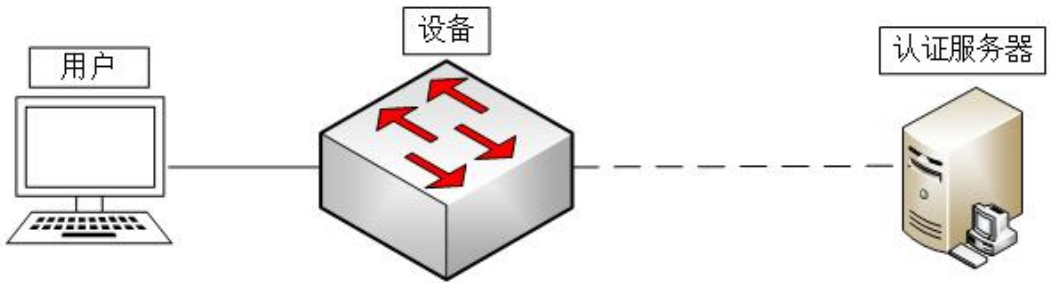
步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**mac-authen** { **start** | **stop** }	使能或去使能全局 MAC 认证功能。
3	JX(config)#**mac-authen aaa authentication method** *method-name*	绑定全局 MAC 认证时的 AAA 方法。
4	JX(config)#**mac-authen max-user** *user-number*	配置全局 MAC 认证最大用户数。
5	JX(config)#**interface** *interface-type interface-number*	进入二层物理接口配置模式。
6	JX(config-ge-1/0/*)#**mac-authen enable**	使能接口 MAC 认证功能。
7	JX(config-ge-1/0/*)#**mac-authen max-user** *user-number*	配置端口允许 MAC 认证的最大用户数。
8	JX(config-ge-1/0/*)#**authentication guest-vlan** *vlan-id*	配置指定端口的 Guest VLAN，对 802.1x 协议和 mac 认证都生效。
9	JX(config-ge-1/0/*)#**mac-authen critical-vlan** *vlan-id*	配置指定端口的 MAC 认证 Critical VLAN。
10	JX(config-ge-1/0/*)#**mac-authen restrict-vlan** *vlan-id*	配置指定端口的 MAC 认证 Restrict VLAN。
11	JX(config-ge-1/0/*)#**mac-authen reauthenticate all user**	手动触发指定端口下 802.1x 用户进行重认证。
12	JX(config-ge-1/0/*)#**mac-authen delete all user**	强制将端口下 MAC 认证用户下线。
13	JX(config-ge-1/0/*)#**mac-authen quiet** { **disable** | **enable** }	使能或去使能端口 MAC 认证用户静默功能。
14	JX(config-ge-1/0/*)#**mac-authen quiet-times** *times*	配置端口 MAC 认证用户触发静默功能的认证失败次数，默认 3 次。
15	JX(config-ge-1/0/*)#**mac-authen critical-vlan user aging** { **disable** | **enable** }	使能或去使能端口 MAC 认证超时用户老化功能。

步骤	配置	说明
16	JX(config-ge-1/0/*)#**mac-authen critical-vlan user reauthenticate { disable | enable }**	使能或去使能端口 MAC 认证超时用户重认证功能。
17	JX(config-ge-1/0/*)#**mac-authen restrict-vlan user aging { disable | enable }**	使能或去使能端口 MAC 认证失败用户重认证功能。
18	JX(config-ge-1/0/*)#**mac-authen restrict-vlan user reauthenticate { disable | enable }**	使能或去使能端口 MAC 认证失败用户重认证功能。
19	JX(config)#**show mac-authen config**	查看 MAC 认证向配置。
20	JX(config)#**show mac-authen information**	查看 MAC 认证的全局信息。
21	JX(config)#**show mac-authen** *interface-type interface-number*	查看指定接口的 MAC 认证相关信息。
22	JX(config)#**show mac-authen user**	查看当前 MAC 认证的用户。

8.11.10 配置举例

如下图所示，用户需要进行 MAC 认证，用户直接跟设备相连，设备可以访问认证服务器。

图 8-15 MAC 认证组网示意图



配置步骤

设备配置如下：

```
JX(config)#aaa
JX(config-aaa)#radius-server host server1 ip-address 192.168.5.66 key
12345
JX(config-aaa)#server-group grp1 radius-server server1
JX(config-aaa)#aaa authentication mac-authen method m1 first grp1
JX(config-aaa)#quit
JX(config)#mac-authen start
JX(config)#mac-authen aaa authentication method m1
JX(config)#mac-authen mode fixed-user username wwf password plain 12345
JX(config)#interface ge 1/0/1
```

```
JX(config-ge-1/0/1)#mac-authen enable
```

通过 show mac-authen information 命令查看 MAC 认证相关信息：

```
Max User Number                 : 256
  Default Max User Number          : 256
  Current User Number              : 1
  Auth Success User Number         : 1
  Auth Fail User Number            : 0
  Auth Timeout User Number         : 0
  UserName Passwor Format          : Fixed
  MacAdress UpperCase              : Disable
  MacAdress With Hypen             : Disable
  Mac Auth Method                  : Pap
  AAA Authentication Method        : m1
  AAA Accounting Method            : n/a
  Fixed User Name                  : wwf
  Fixed Password                   : 12345
```

检查结果

通过命令 **show mac-authen user** 查看当前 MAC 认证的用户：

```
JX(config)# show mac-authen user
 S:Success, F:Fail, T:Timeout, J:Join, A:Authenticate, O:Origin,
C:Current, L:Last
 Interface    Mac-Address      AuthNum(S/F/T)  Vlan(J/A/O)    Result(C/L)
 ge 1/0/1     0010:9400:0001    1/0/0          11/0/11        Success/None
```

8.12 URPF

8.12.1 简介

通常情况下，路由器收到数据报文后，获取到数据包中的目的 IP 地址，针对目的 IP 地址查找本地路由转发表，如果有对应转发表项则转发数据报文；否则，将报文丢弃。由此看来，路由器转发报文时，并不关心数据包的源地址。这就给源地址欺骗攻击有了可乘之机。

源地址欺骗攻击为入侵者构造出一系列带有伪造源地址的报文，频繁访问目的地址所在设备或者主机；即使响应报文不能到达攻击者，也会对被攻击对象造成一定程度的破坏。

URPF（Unicast Reverse Path Forwarding，单播逆向路径转发）的主要功能是用于防止基于源地址欺骗的网络攻击行为。路由器接口一旦使能 URPF 功能，当该接口收到数据报文时，首先会对数据报文的源地址进行合法性检查，对于源地址合法性检查通过的报文，才会进一步查找去往目的地址的转发表项，进入报文转发流程；否则，将丢弃报文。

严格（strict）型 URPF 检测功能

不但要求路由器转发表中，存在去往报文源地址的路由；而且还要求报文的入接口与转发表中去往源地址路由的出接口一致，只有同时满足上述两个条件的报文，才被认为是合法报文。在一些特殊情况下（如存在非对称路径），严格型检查会错误的丢弃非攻击报文。

松散（loose）型 URPF 检测功能

仅要求路由器的转发表中，存在去往报文的源地址路由即可，不再检查报文的入接口与转发表中去往源地址的路由的出接口是否一致。当用户网络中存在非对称路径等无法保证报文入接口和设备去往源地址路由出接口一致的情况下，可以配置松散型 URPF 检查。

诸如 TCP Syn Flood、UDP flood 和 ICMP flood 等攻击，都可能通过借助源地址欺骗的方式攻击目标设备或者主机，造成被攻击者系统性能严重的降低，甚至导致系统崩溃。URPF 就是网络设备为了防范此类攻击而使用的一种常用技术。

8.12.2 配置准备

场景

Switch 通过上行接口与 ISP（Internet Service Provider）的路由器连接，下行接口接用户网络。管理员希望 Switch 能够防范下行接口源 IP 地址欺骗攻击，避免非法用户伪造源 IP 地址攻击合法用户。

前提

无

8.12.3 缺省配置

功能	缺省值
接口 URPF 功能	不使能

8.12.4 配置 URPF

步骤	配置	说明
1	JX#**configure**	进入全局配置模式。
2	JX(config)#**interface interface-type** *interface-number*	进入接口配置模式。

步骤	配置	说明
3	JX(config-ge-1/0/1)# **urpf { loose | strict } { allow-default-route }**	使能 URPF 功能 loose：松散模式 strict：严格模式 （可选）allow-default-route：允许匹配默认路由

8.12.5 检查配置

序号	检查项	说明
1	JX#**show urpf interface** {*interface-type interface-number* }	显示接口的 URPF 配置模式

8.13 Timerange

8.13.1 简介

部分功能需要定时启动，或者在某些周期内启动，就可以利用 timerange 命令加以限制。在配置可以搭配 timerange 的功能（如 ACL，POE,NQA 等）时，通过时间段索引引用这个时间范围

8.13.2 配置准备

场景

当设备需要在不同时间段执行不同的安全策略时，可以配置 timerange。

前提

无

8.13.3 缺省配置

功能	缺省值
timerange 功能	不使能

8.13.4 配置 timerange

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**time-range list** *id*	创建时间段列表

步骤	配置	说明
3	JX(config-timerange-*)#**time-range** *timerange-id* **absolute from** *hour:minute:second year/month/monthday* **to** *hour:minute:second year/month/monthday* JX(config-timerange-*)#**time-range** *timerange-id* **everyhour** *minute:second* **to** *minute:second* JX(config-timerange-*)#**time-range** *timerange-id* **everyday** *hour:minute:second* **to** *hour:minute:second* JX(config-timerange-*)#**time-range** *timerange-id* **everyweek** *hour:minute:second* { **mon** | **tue** | **wed** | **thu** | **fri** | **sat** | **sun** } **to** { **mon** | **tue** | **wed** | **thu** | **fri** | **sat** | **sun** } JX(config-timerange-*)#**time-range** *timerange-id* **everymonth** *hour:minute:second monthday* **to** *hour:minute:second monthday* JX(config-timerange-*)#**time-range** *timerange-id* **everyweekend** *hour:minute:second* **to** *hour:minute:second* JX(config-timerange-*)#**time-range** *timerange-id* **everyyear** *hour:minute:second month/monthday* **to** *hour:minute:second month/monthday*	创建时间段，可应用于 ACL 规则。 对时间段（周期时间段）：是指 everyxxx 描述的以一周为间隔的期性时间，同时可以依靠绝对时段指明时间范围有效日期。 对时间段：是指由 absolute 指定特定的时间范围。可以通过绝对间段指定的时间范围来限制相对间段（周期时间段）在什么时间围生效。 果该时间段配置了多个生效时，生效原则为：周期时间段之间“或”，周期时间段和绝对时间段间取“与”
4	JX(configure-acl-l2-*)#**rule** *rule-id* **time-range** *timerange-list-id*	在 acl 中应用时间段
5	JX(config)#**nqa schedule** *admin-name operate-tag* **time-range** *timer-ange-list*	在 nqa 中应用时间段

8.13.5 检查配置

序号	检查项	说明
1	JX#**show timerange list**	查看所有 timerange

8.14 DOS 防攻击

8.14.1 简介

畸形报文攻击防范

畸形报文攻击是通过向目标设备发送有缺陷的 IP 报文，使得目标设备在处理这样的 IP 报文时出错和崩溃，给目标设备带来损失。畸形报文攻击防范是指设备实时检测出畸形报文并予以丢弃，实现对本设备的保护。

分片报文攻击防范

分片报文攻击是通过向目标设备发送分片出错的报文，使得目标设备在处理分片错误的报文时崩溃、重启或消耗大量的 CPU 资源，给目标设备带来损失。分片报文攻击防范是指设备实时检测出分片报文并予以丢弃或者限速处理，实现对本设备的保护。

泛洪攻击防范

泛洪攻击是指攻击者在短时间内向目标设备发送大量的虚假报文，导致目标设备忙于应付无用报文，而无法为用户提供正常服务。

泛洪攻击防范是指设备实时检测出泛洪报文并予以丢弃或者限速处理，实现对本设备的保护。

泛洪攻击主要分为 TCP SYN 泛洪攻击、UDP 泛洪攻击和 ICMP 泛洪攻击。

(1) TCP SYN 泛洪攻击

TCP SYN 攻击利用了 TCP 三次握手的漏洞。在 TCP 的 3 次握手期间，当接收端收到来自发送端的初始 SYN 报文时，向发送端返回一个 SYN+ACK 报文。接收端在等待发送端的最终 ACK 报文时，该连接一直处于半连接状态。如果接收端最终没有收到 ACK 报文包，则重新发送一个 SYN+ACK 到发送端。如果经过多次重试，发送端始终没有返回 ACK 报文，则接收端关闭会话并从内存中刷新会话，从传输第一个 SYN+ACK 到会话关闭大约需要 30 秒。

在这段时间内，攻击者可能将数十万个 SYN 报文发送到开放的端口，并且不回应接收端的 SYN+ACK 报文。接收端内存很快就会超过负荷，且无法再接受任何新的连接，并将现有的连接断开。

设备对 TCP SYN 攻击处理的方法是在使能了 TCP SYN 泛洪攻击防范后对 TCP SYN 报文进行速率限制，保证受到攻击时设备资源不被耗尽。

(2) UDP 泛洪攻击

UDP 泛洪攻击是指攻击者在短时间内向目标设备发送大量的 UDP 报文，导致目标设备负担过重而不能处理正常的业务。UDP 泛洪攻击分为以下两类：

•Fraggle 攻击

Fraggle 攻击的原理是攻击者发送源地址为目标主机地址，目的地址为广播地址，目的端口号为 7 的 UDP 报文。如果该广播网络中有很多主机都起用了 UDP 响应请求服务，目的主机将收到很多回复报文，造成系统繁忙，达到攻击效果。

使能泛洪攻击防范功能后，设备默认 UDP 端口号为 7 的报文是攻击报文，直接将其丢弃。

•UDP 诊断端口攻击

攻击者对 UDP 诊断端口（7-echo，13-daytime，19-Chargen 等 UDP 端口）发送报文，如果同时发送的数据包数量很大，造成泛洪，可能影响网络设备的正常工作。

使能泛洪攻击防范功能后，设备将 UDP 端口为 7、13 和 19 的报文认为是攻击报文，直接丢弃。

(3) ICMP 泛洪攻击

通常情况下，网络管理员会用 Ping 程序对网络进行监控和故障排除，大概过程如下：

1.源设备向接收设备发出 ICMP 响应请求报文；

2.接收设备接收到 ICMP 响应请求报文后，会向源设备回应一个 ICMP 应答报文。

如果攻击者向目标设备发送大量的 ICMP 响应请求报文，则目标设备会忙于处理这些请求，而无法继续处理其他的数据报文，造成对正常业务的冲击。

设备针对 ICMP 泛洪攻击进行 CAR（Committed Access Rate）限速，保证 CPU 不被攻击，保证网络的正常运行。

8.14.2 配置准备

场景

设备经常会受到不同类型的网络攻击，这样会导致设备资源使用率过高，影响网络服务。为保障给用户提供安全的网络服务，在设备上部署 Dosantiattack 功能，主要防范的攻击类型包括：

畸形报文攻击防范，防止畸形报文攻击。

分片报文攻击防范，限制分片报文的速率，防止分片报文对 CPU 造成攻击，占用过多 CPU 和设备资源。

泛洪攻击防范，包括以下三种：

TCP SYN 泛洪攻击防范，限制 TCP SYN 报文的速率，防止 CPU 处理 TCP SYN 报文占用过多资源；

UDP 泛洪攻击防范，对特定端口发送的 UDP 报文直接丢弃；

ICMP 泛洪攻击防范，限制 ICMP 泛洪攻击报文的上送速率，防止 CPU 处理 ICMP 泛洪攻击报文占用过多资源。

前提

无。

8.14.3 DOS 防攻击功能的缺省配置

设备上 DOS 防攻击功能的缺省配置如下。

功能	缺省值
Dosantiattack 全局功能状态	禁用
畸形报文攻击防范功能	禁用
分片报文攻击防范功能	禁用
分片报文发送速率	155000000bit/s
TCP Syn 攻击防范功能	禁用
TCP Syn 泛洪报文发送速率	155000000bit/s
UDP 泛洪攻击防范功能	禁用
ICMP 泛洪攻击防范功能	禁用
ICMP 泛洪报文发送速率	155000000bit/s

8.14.4 配置畸形报文攻击防范

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**dos-antiattack pkt-limit abnormal enable**	使能畸形报文攻击防范功能。

8.14.5 配置分片报文攻击防范

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**dos-antiattack pkt-limit fragment enable**	使能畸形报文攻击防范功能。
3	JX(config)#**dos-antiattack pkt-limit fragment cir** { **kbps**|**mbps**|**gbps** } *cir-number* JX(config)#**dos-antiattack pkt-limit fragment pps** *pps-number*	限制分片报文接收的速率。

8.14.6 配置 TCP SYN 泛洪攻击防范

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**dos-antiattack pkt-limit tcp-syn enable**	使能 TCP SYN 泛洪攻击防范功能。
3	JX(config)#**dos-antiattack pkt-limit tcp-syn cir** { **kbps**|**mbps**|**gbps** } *cir-number* JX(config)#**dos-antiattack pkt-limit tcp-syn pps** *pps-number*	限制 TCP SYN 报文接收的速率。

8.14.7 配置 UDP 泛洪攻击防范

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**dos-antiattack pkt-limit udp-flood enable**	使能 UDP 泛洪攻击防范功能。

8.14.8 配置 ICMP 泛洪攻击防范

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**dos-antiattack pkt-limit icmp-flood enable**	使能 ICMP 泛洪攻击防范功能。
3	JX(config)#**dos-antiattack pkt-limit icmp-flood cir** { **kbps**|**mbps**|**gbps** } *cir-number* JX(config)#**dos-antiattack pkt-limit icmp-flood pps** *pps-number*	限制 ICMP 泛洪攻击报文接收的速率。

8.14.9 检查配置

配置完成后，请在设备上执行以下命令检查配置结果。

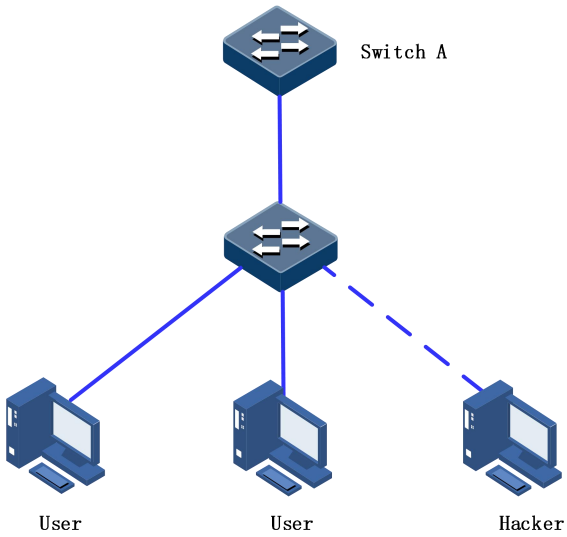
序号	检查项	说明
1	JX#show dos-antiattack config	查看 Dos 防攻击配置信息。
2	JX#show dos-antiattack statistics	查看 Dos 防攻击统计数据。

8.14.10 配置 DOS 防攻击示例

组网需求

如果局域网内存在 Hacker 向 SwitchA 发起畸形报文攻击、分片报文攻击和泛洪攻击，将会造成 SwitchA 瘫痪。为了预防这种情况，管理员希望通过在 SwitchA 上部署各种攻击防范措施来为用户提供安全的网络环境，保障正常的网络服务。

图 8-16 DOS 防攻击应用组网示意图



配置步骤

步骤 1 配置 DOS 防攻击功能。

```
JX#config
JX(config)#dos-antiattack pkt-limit enable
```

步骤 2 配置畸形报文攻击防范。

```
JX(config)#dos-antiattack pkt-limit abnormal enable
```

步骤 3 配置分片报文攻击防范，并限制分片报文接收的速率为 15kbit/s。

```
JX(config)#dos-antiattack pkt-limit fragment enable
JX(config)#dos-antiattack pkt-limit fragment cir kbps 15
```

步骤 4 配置 TCP SYN 攻击防范，并限制 TCP SYN 报文接收的速率为 15kbit/s。

```
JX(config)#dos-antiattack pkt-limit tcp-syn enable
```

```
JX(config)#dos-antiattack pkt-limit tcp-syn cir kbps 15
```

步骤 5 配置 UDP 泛洪攻击防范。

```
JX(config)#dos-antiattack pkt-limit udp-flood enable
```

步骤 6 配置 ICMP 泛洪攻击防范，并限制 ICMP 泛洪报文接收的速率为 15kbit/s。

```
JX(config)#dos-antiattack pkt-limit icmp-flood enable
JX(config)#dos-antiattack pkt-limit icmp-flood cir kbps 15
```

检查结果

通过 **show dos-antiattack config** 命令查看 DOS 防攻击配置结果。

```
JX#show dos-antiattack config
!
!
dos-antiattack pkt-limit enable
!
dos-antiattack pkt-limit abnormal enable
dos-antiattack pkt-limit fragment enable
dos-antiattack pkt-limit fragment cir kbps 15
dos-antiattack pkt-limit tcp-syn enable
dos-antiattack pkt-limit tcp-syn cir kbps 15
dos-antiattack pkt-limit udp-flood enable
dos-antiattack pkt-limit icmp-flood enable
dos-antiattack pkt-limit icmp-flood cir kbps 15
```

9 可靠性

本章介绍了网络可靠性的基本原理和配置过程，并提供相关的配置案例。

- 链路聚合
- G.8031
- G.8032
- STP/RSTP
- MSTP
- 环路检测
- 接口隔离
- L2CP
- BFD
- 链路震荡保护

9.1 链路聚合

9.1.1 简介

链路聚合通过将多个物理以太网接口聚合在一起形成一个逻辑上的聚合组，并把同一聚合组内的多条物理链路视为一条逻辑链路。链路聚合可以实现流量在聚合组各成员接口之间负载分担，在有效提高设备之间链路可靠性的同时，还在不进行硬件升级的条件下增大了带宽。

按照聚合方式的不同，设备支持以下四种链路聚合方式：

- 手工聚合方式

聚合组中所有接口都参与数据转发，平均分担负载流量，适用于两个直连设备，且一端设备无法使用 LACP 协议的情况。

- 手工主备方式链路聚合方式

聚合组中共有 2 个接口，2 个接口之间形成备份，一个处于 Active 状态，另外一个处于 Shutdown 状态。适用于一端设备无法使用 LACP 协议的情况。

- 静态 LACP 聚合方式

聚合组通过 LACP 协议来选择主动端和活动接口。活动接口用于转发数据，而非活动接口用于备份链路。适用于两端设备均支持 LACP 协议的情况。

- 静态 LACP 主备方式链路聚合方式

聚合组中共有 2 个接口，2 个接口之间形成备份，一个处于 Active 状态，另外一个处于 Shutdown 状态。适用于两端设备均支持 LACP 协议的情况。

9.1.2 配置准备

场景

当需要为 2 台设备之间的链路提供更高的通讯带宽和更高的可靠性时，可以配置选择手工模式或者静态 LACP 链路聚合功能。

前提

- 在配置链路聚合之前，需配置接口的物理参数，使接口的物理层状态为 Up。
- 在同一个链路聚合组中，参与负载分担的成员接口必须有一致的配置，否则数据转发存在问题。这些配置主要包括 QoS、QinQ、VLAN、接口属性、MAC 地址学习几个方面：
  - QoS 配置一致：流量监管、流量整形、拥塞避免、接口限速、SP 队列、WRR 队列调度、WFQ 队列、接口优先级、接口信任模式。
  - QinQ 配置一致：接口的 QinQ 功能使能/禁用状态、添加的外层 VLAN Tag、不同内层 VLAN ID 添加外层 VLAN Tag 的策略。
  - VLAN 配置一致：接口上允许通过的 VLAN、接口缺省 VLAN ID、接口的链路类型（即 Trunk、Hybrid、Access 类型）、子网 VLAN 配置、协议 VLAN 配置、VLAN 报文是否带 Tag 配置。
  - 接口属性配置一致：接口是否加入隔离组、接口速率、双工模式、链路 up/down 状态。
  - MAC 地址学习配置一致：是否使能 MAC 地址学习功能、接口是否具有最大学习 MAC 地址个数的限制。

9.1.3 缺省配置

设备上链路聚合的缺省配置如下。

功能	缺省值
负载均衡模式	**src-dst-mac** 模式
LACP 系统优先级	32768
LACP 接口优先级	32768
LACP 接口模式	**active**
LACP 超时模式	**slow**

功能	缺省值
最小活动接口数	1
最大活动接口数	8

9.1.4 配置手工链路聚合

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**int eth-trunk** *trunk-number*	进入聚合组配置模式。
3	JX(config-eth-trunk-*)#**mode manual**	配置链路聚合组的工作模式为手工链路聚合。
4	JX(config-eth-trunk-*)#**active-linknumber** { **max** | **min** } {*<1-8>* | *default* }	配置 LACP 链路聚合组最大或最小的活跃链路数量。 缺省情况下，最大活跃链路数为 8，最小活跃链路数为 1。
5	JX(config-eth-trunk-*)#**add interface** *interface-type interface-number*	将接口接入聚合
6	JX(config-eth-trunk-*)#**exit**	返回全局配置模式。

9.1.5 配置静态 LACP 链路聚合

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**lacp system-priority** *system-priority*	配置 LACP 协议优先级，优先级高的一端为主动端，LACP 按主动端的配置情况选择活动接口和备份接口。数值越小优先级越高，系统 LACP 优先级相同时，选择系统 MAC 地址小的作为主动端。 缺省情况下，系统的 LACP 优先级为 32768。
3	JX(config)#**interface eth-trunk** *trunk-number*	进入聚合组配置模式。
4	JX(config-eth-trunk-*)#**mode lacp-static**	配置链路聚合组的工作模式为静态 LACP 链路聚合。

步骤	配置	说明
5	JX(config-eth-trunk-*)#**lacp timeout** { **fast** | **slow** }	配置 LACP 超时模式。 缺省情况下，LACP 超时模式为慢速模式。
6	JX(config-eth-trunk-*)#**active-linknumber** { **max** | **min** } { *number* | **default** }	配置 LACP 链路聚合组最大或最小的活跃链路数量。 缺省情况下，最大活跃链路数为 8，最小活跃链路数为 1。
7	JX(config-eth-trunk-*)#**lacp preempt enable**	使能链路聚合组的优先级抢占功能。
8	JX(config-eth-trunk-*)#**lacp preempt delay** *time*	配置端口延时抢占。
9	JX(config-eth-trunk-*)#**add interface** *interface-type interface-number*	将接口接入聚合
10	JX(config-ge-1/0/*)#**lacp priority** *priority*	配置接口的 LACP 协议优先级，接口协议优先级影响 LACP 协议的缺省接口的选举，数值越小优先级越高。 缺省情况下，系统的 LACP 优先级为 32768。
11	JX(config-ge-1/0/*)#**exit**	返回全局配置模式。

说明

- 在静态 LACP 链路聚合组中，成员接口可以处于 Active 或 Standby 两种状态。Active 接口和 Standby 接口都能收发 LACP 协议报文，但 Standby 接口不能转发用户报文。
- 系统按照是否发现邻居、接口速率最大、接口 LACP 协议优先级最高、接口号最小的顺序选择缺省接口，缺省接口处于 Active 状态，与缺省接口具有相同速率、相同对端设备和对端设备操作 key 的接口也处于 Active 状态，其他接口则处于 Standby 状态。

9.1.6 配置主备方式链路聚合

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**int eth-trunk** *trunk-number*	进入聚合组配置模式。
3	JX(config-eth-trunk-*)#**mode** { **manual** | **lacp-static** } **active-standby**	配置链路聚合组的工作模式为手工主备或静态 LACP 主备方式链路聚合。

步骤	配置	说明
4	JX(config-eth-trunk-*)#**add primary interface** *interface-type interface-number*	配置链路聚合的主接口。
4	JX(config-eth-trunk-*)#**add secondary interface** *interface-type interface-number*	配置链路聚合的备接口。
5	JX(config-eth-trunk-*)#**revert enable** JX(config-eth-trunk-*)#**wait-to-store** *time*	配置链路聚合组恢复模式及延迟恢复时间。 缺省情况下，聚合组恢复模式为非返回模式。
6	JX(config-eth-trunk-*)#**exit**	返回全局配置模式。

说明

在配置链路聚合组故障返回模式为非返回模式时，必须先通过 **master-port** 命令进行主接口的配置。

9.1.7 配置聚合组负载分担算法

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**int eth-trunk** *trunk-number*	进入聚合组配置模式。
3	JX(config-eth-trunk-*)#**load-balance { src-mac | dst-mac | srcdst-mac | src-ip | dst-ip | srcdst-ip | default }**	基于全局配置链路聚合组的负载均衡模式。 缺省情况下，系统采用 **src-dst-mac** 模式，即依据源和目的 MAC 地址逻辑或的结果选择转发接口。

9.1.8 检查配置

配置完成后，请在设备上执行以下命令检查配置结果。

序号	检查项	说明
1	JX#**show lacp eth-trunk** *trunk-number*	查看本端系统 LACP 协议接口状态、标志、接口优先级、管理 key、操作 key 和接口状态机状态。邻居 LACP 协议信息，包括标志、接口优先级、设备 ID、Age、操作键值、接口号、接口状态机状态。

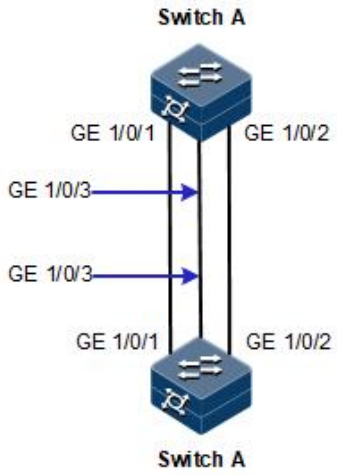
序号	检查项	说明
2	JX#**show lacp statistics interface eth-trunk** *trunk-number*	查看接口 LACP 协议统计信息，包括 LACP 报文的总收发数、Marker 报文的收发数、Marker Response 报文的收发数和错误报文数。
3	JX#**show lacp information**	查看本端系统的 LACP 系统优先级，MAC 地址，各定时器时间
4	JX#**show** interface **eth-trunk** *trunk-number*	查看当前系统是否使能聚合链路、链路聚合负载均衡模式、当前所有聚合组设置的组成员接口列表和当前生效的成员接口列表等信息。 **说明** 当前生效的成员接口是指组成员接口中接口状态为 Up 的接口列表。

9.1.9 配置静态 LACP 方式的链路聚合示例

组网需求

如下图所示，为提高 Switch A 与 Switch B 之间链路的可靠性，在 Switch A 与 Switch B 之间配置静态 LACP 方式的链路聚合，将 GE 1/0/1、GE 1/0/2 和 GE 1/0/3 加入链路聚合组。

图 9-1 静态 LACP 方式链路聚合应用组网示意图



配置步骤

步骤 1 在 Switch A 上配置静态 LACP 链路聚合组，并将 Switch A 配置成主动端。

```
JX#hostname SwitchA
```

```
SwitchA#config
SwitchA(config)#lacp system-priority 1000
SwitchA(config)#int eth-trunk 1
SwitchA(config-eth-trunk-1)#mode lacp-static
SwitchA(config-eth-trunk-1)#add interface ge 1/0/1
SwitchA(config-eth-trunk-1)#add interface ge 1/0/2
SwitchA(config-eth-trunk-1)#add interface ge 1/0/3
SwitchA(config-eth-trunk-1)#exit
```

步骤 2 在 Switch B 上配置静态 LACP 链路聚合组。

```
JX#hostname SwitchB
SwitchB#config
SwitchB(config)#int eth-trunk 1
SwitchB(config-eth-trunk-1)#mode lacp-static
SwitchB(config-eth-trunk-1)#add interface ge 1/0/1
SwitchB(config-eth-trunk-1)#add interface ge 1/0/2
SwitchB(config-eth-trunk-1)#add interface ge 1/0/3
SwitchB(config-eth-trunk-1)#exit
```

检查结果

在 Switch A 上通过 **show lacp eth-trunk 1** 查看静态 LACP 方式链路聚合全局配置是否正确。

```
SwitchA#show lacp eth-trunk 1
-------------------------------------------------------------------------------
------------
  LACP Status                 : master
  System Priority             : 1000
  System Mac Address          : f0:f1:f2:f3:01:01
  Member ports number         : 3
  Max Active ports number      : 8
  LACP timeout                : slow
  Preempt state               : disable
  Preempt delay               : 30(s)

 ge-1/0/1 :
   Port Status  : Up and bind
   Local information:
     Mode     Flags   PortPri   AdminKey  OperKey   PortId  State   Status
     active   slow    32768     1         1         449     0x3d    selected
   Partner information:
     SysPri    Flags   PortPri   AdminKey  OperKey   PortId  State   DeviceID
     32768     slow    32768     0         1         449     0x3d
0xf0f1f2f30201

 ge-1/0/2 :
   Port Status  : Up and bind
   Local information:
     Mode     Flags   PortPri   AdminKey  OperKey   PortId  State   Status
     active   slow    32768     1         1         450     0x3d    selected
   Partner information:
     SysPri    Flags   PortPri   AdminKey  OperKey   PortId  State   DeviceID
```

```
     32768    slow    32768    0        1        450    0x3d
0xf0f1f2f30201

 ge-1/0/3 :
   Port Status  : Up and bind
   Local information:
     Mode     Flags   PortPri   AdminKey  OperKey   PortId  State   Status
     active   slow    32768     1         1         451     0x3d    selected
   Partner information:
     SysPri    Flags   PortPri   AdminKey  OperKey   PortId  State   DeviceID
     32768     slow    32768     0         1         451     0x3d
0xf0f1f2f30201
-------------------------------------------------------------------------
------------
```

9.2 G.8031

9.2.1 简介

G.8031 是 ITU-T 基于 VLAN 的以太网技术定义的线性保护倒换标准。在保护切换机制中，对工作资源都分配相应的保护资源，如路径和带宽等。相对于 IEEE 定义的生成树保护技术，G.8031 定义的保护技术简单快速，以一种可预测的方式实现网络资源切换，更易于运营商有效地规划网络及明了网络的活动状态，实现电信级的运营。

G.8031 定义了 1+1 和 1:1 两种保护结构，在 1+1 结构中每一个保护资源都对应着一个工作资源，在保护域内，1+1 结构采用双发单收的保护机制；1:1 结构采用保护资源与工作资源彼此切换的机制。

相关概念

- 故障检测机制

G.8031 采用 Y.1731 或 IEEE 802.1ag 中定义的连续性检测(CC)进行链路双向转发检测，能够定位故障点并检测故障是单向还是双向的，并且用于保护转换时，CC 帧默认的传输周期是 3.33 ms（即每秒 300 帧的传输速率）。

两个相邻节点间周期性的从物理端口发送连续性检测(CC)帧以检测故障，当一个节点在特定的时间内检测到 CC 帧丢失，即检测到了一个故障。

节点从检测到故障的端口发送 RDI(Remote Defect Indication)帧，如果是单向故障，链路下行的节点将检测到该 RDI 帧。

- 1+1 保护结构

在 1+1 结构中，保护连接是每条工作连接专用的，工作连接与保护连接在保护域的源端进行桥接。业务在工作和保护连接上同时发向保护域的宿端，在宿端，选择器基于缺陷指示来选择接收来自工作或保护连接上的业务。

1+1 以太网线性保护的倒换类型包括单向倒换和双向倒换。单向倒换时只有受影响的连接方向倒换至保护路径，两端的选择器是独立的，不需要 APS 信令支持。双向倒换的机制与单向类似，通常需要 APS 信令在两端协调。单向保护可以防止在两个独立方向上的单通故障。

1+1 以太网线性保护的操作类型可以是不可恢复或可恢复的。在可恢复模式下，故障链路恢复时，启动 WTR 定时器，WTR 超时后，选择器切换到工作链路；在不可恢复模式下，即使故障链路恢复，选择器仍然连接到保护链路。

- 1:1 保护结构

在 1:1 结构中，保护连接是每条工作连接专用的，被保护的工作业务由工作或保护连接进行传送。工作和保护连接的选择方法基于缺陷指示机制。

1:1 以太网线性保护的倒换类型也包括单向倒换和双向倒换，操作类型可以是可恢复的，也可以是不可恢复的，双向倒换时受影响的和未受影响的连接方向均倒换至保护路径，而单向倒换仅受影响的的连接方向均倒换至保护路径。倒换时源端和宿端连接器需要切换到同一个连接，因此需要自动保护倒换协议 APS 用于协调连接的两端。

在 1:1 保护倒换模式下，基于本地或近端信息和来自另一端或远端的 APS 协议信息，保护倒换由保护域源端选择器桥接和宿端选择器共同来完成。

使用连接性检查包检测工作和保护连接故障，当工作连接故障时，检测到故障的一端选择器切换到保护连接上，同时发送 APS 通知对端，源端收到 APS 同步倒换。

9.2.2 配置准备

场景

无

前提

在配置 G.8031 之前，需完成以下任务：

- 连接接口并配置接口的物理参数，使接口的物理层状态为 Up；
- 创建 VLAN；
- 将接口加入 VLAN。

9.2.3 G.8031 的缺省配置

设备上 G.8031 的缺省配置如下。

功能	缺省值
G.8031 模式	返回模式
WTR 定时器	5min
HOLDOFF 定时器	0

9.2.4 创建 G.8031 保护组

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**configure**	进入全局配置模式。
2	JX(config)#**g8031 instance** *instance-id*	创建 G8031 实例并进入配置节点
3	JX(config-g8031-instance-*)#**control-vlan** *vlan-id*	指定控制 vlan，如果配置了 **revertive dsiable**，则保护组变为非返回模式。非返回模式与返回模式的区别在于，返回模式下工作链路故障恢复时，流量由保护链路切换回工作链路，非返回模式下不切换。缺省情况下，保护组处于返回模式。
4	JX(config-g8031-instance-*)#**data-vlan** *vlan-id*	指定数据 vlan
5	JX(config-g8031-instance-*)# **working-port interface** *interface-type interface-number*	配置端口并指定为工作端口
6	JX(config-g8031-instance-*)# **protection-port interface** *interface-type interface-number*	配置端口并指定为保护端口
7	JX(config-g8032-instance-*)#**wtr-timer** *wtr-time*	配置 WTR 定时器。在返回模式下当工作链路故障恢复时，等待 WTR 定时器超时之后，才会恢复到工作链路上工作。
8	JX(config-g8032-instance-*)#**holdoff-timer** *holdoff-time*	配置环 HOLDOFF 定时器后，当工作链路故障时，系统会延时上报故障，即延时一段时间后再倒换到保护链路，可以防止工作链路震荡引起的频繁倒换。 **说明** HOLDOFF 定时器配置值较大时会影响 50ms 倒换性能，所以推荐使用缺省值 0。

9.2.5 配置 G.8031 倒换控制

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config-g8031-instance-*)#**protection-switch force-switch**	配置流量强制倒换。 强制倒换可配置在多个环节点的多个接口上。

步骤	配置	说明
3	JX(config-g8031-instance-*)#**protection-switch manual-switch**	配置流量手工倒换，优先级低于强制倒换和工作链路故障时产生的自动倒换。 手工倒换只能配置在同一个环节点的一个接口上。
4	JX(config-g8031-instance-*)#**protection-switch clear**	配置流量倒换清除。

9.2.6 检查配置

配置完成后，请在设备上执行以下命令检查配置结果。

序号	检查项	说明
1	**JX#show g8031 interface**	查看 G.8031 接口状态信息。
2	**JX#show g8031 instance [*instanceid*]**	查看 G.8031 状态信息。

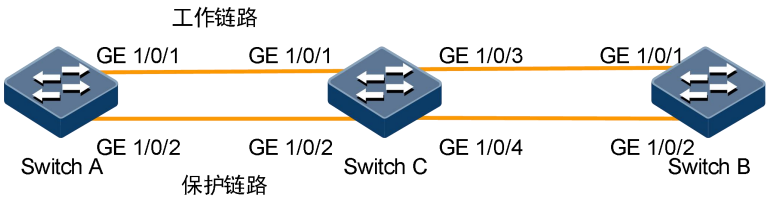
9.2.7 配置 G.8031 保护示例

组网需求

如图 9-2 所示，为提高以太网络的可靠性，Switch A、Switch B、Switch C 三台设备组成 G.8031 保护链路。

协议控制 VLAN 为 2，阻塞的 VLAN 范围为缺省值 2～10。

图 9-2 G.8031 组网示意图



配置步骤

步骤 1 配置接口加入 VLAN 2～VLAN 10。

配置 Switch A。

```
JX#hostname SwitchA
SwitchA#configure
SwitchA(config)#interface ge 1/0/1 to ge 1/0/2
SwitchA(config-ge-1/0/1->ge-1/0/2)#port link-type trunk
SwitchA(config-ge-1/0/1->ge-1/0/2)#port trunk allow-pass vlan 2-10
SwitchA(config-ge-1/0/1->ge-1/0/2)#exit
```

Switch B 配置同 Switch A

配置 Switch C。

```
JX#hostname SwitchC
SwitchC#config
SwitchC(config)#interface ge 1/0/1 to ge 1/0/4
SwitchC(config-ge-1/0/1->ge-1/0/4)#port link-type trunk
SwitchC(config-ge-1/0/1->ge-1/0/4)#port trunk allow-pass vlan 2-10
SwitchC(config-ge-1/0/1->ge-1/0/4)#exit
```

步骤 2 创建保护组。

配置 Switch A。

```
SwitchA(config)#g8031 instance 1
SwitchA(config-g8031-instance-1)#control-vlan 2
SwitchA(config-g8031-instance-1)#data-vlan 2-10
SwitchA(config-g8031-instance-1)#working-port interface ge 1/0/1
SwitchA(config-g8031-instance-1)#protection-port interface ge 1/0/2
```

配置 Switch B。

```
SwitchC(config)#g8031 instance 1
SwitchC(config-g8031-instance-1)#control-vlan 2
SwitchC(config-g8031-instance-1)#data-vlan 2-10
SwitchC(config-g8031-instance-1)#working-port interface ge 1/0/1
SwitchC(config-g8031-instance-1)#protection-port interface ge 1/0/2
```

检查结果

在设备上通过 **show g8031 interface** 查看 G.8031 保护是否生效。

以 Switch A 为例，状态信息如下：

```
SwitchA#show g8031 interface
Instance Interface      Role         Active  Operate   Forward    RxCount
TxCount
-------------------------------------------------------------------------
--------------------------
1       ge-1/0/1       working      active  working   forwarding  0        0
1       ge-1/0/2       protection    standby working   blocking    0        6
-------------------------------------------------------------------------
--------------------------
```

手动断开链路模拟故障，在 Switch A 上再次使用命令查看 G.8031 保护状态

```
SwitchA#show g8031 interface
Instance Interface      Role         Active  Operate   Forward    RxCount
TxCount
-------------------------------------------------------------------------
--------------------------
1       ge-1/0/1       working      standby failed    blocking  0        0
1       ge-1/0/2       protection    active  working   forwarding   0      6
```

9.3 G.8032

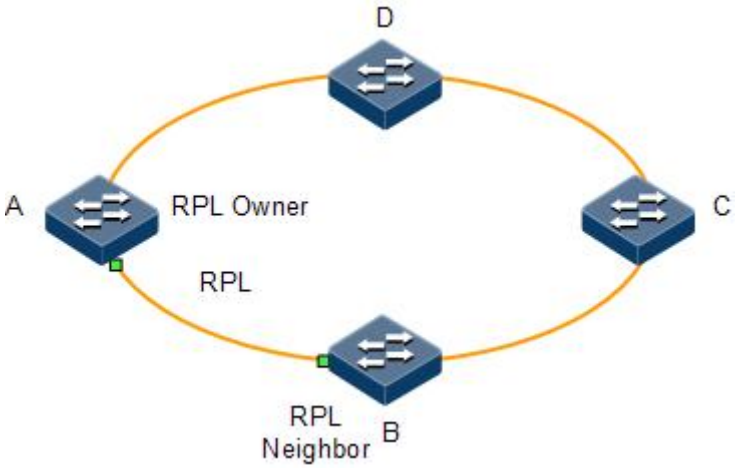
9.3.1 简介

G.8032 以太网环网保护倒换（Ethernet Ring Protection Switching，ERPS）是基于 ITU-T G.8032 标准的 APS 协议，是一种专门应用于以太网环的 链路层协议。正常情况下，它在以太网环中能够防止数据环路引起的广播风暴。当以太网环上链路或设备故障时，能迅速切换到备份链路，保证业务快速恢复。

G.8032 利用环网内专用的控制 VLAN 传递环网控制信息，同时结合环网本身的拓扑特点，在网络发生故障时快速发现，并启用备份链路从而做到快速恢复。

相关概念

图 9-3 G.8032 环网示意图



如图 9-3 所示，G.8032 环网的一些基本概念如下。

- RPL：RPL（Ring Protection Link，环网保护链路）是 RPL 节点间的链路，正常状态下阻塞该链路的相应接口以避免形成环路。一个环网只有一条 RPL。
- RPL Owner：与 RPL 相连的一个环网节点称为 RPL Owner，由用户指定，用于控制 RPL 接口的阻塞或解除阻塞。在正常状态下，RPL Owner 阻塞其 RPL 接口以防止业务形成环路。
- RPL Neighbor：与 RPL 相连的另一个环网节点称为 RPL Neighbor，配合 RPL Owner 完成保护倒换。

- 协议 VLAN：协议 VLAN 是 G.8032 采用的用于承载 R-APS 报文的独立 VLAN 通道。
- 阻塞 VLAN：业务 VLAN，不同于协议 VLAN 承载 R-APS 报文，是进行业务信息传递的 VLAN。
- R-APS 消息：定义在 G.8032 标准中的快速倒换协议报文。包括如下消息类型：
  - FS（Forced Switching）：FS 节点定期发送的消息用于执行强制倒换。
  - SF（Signal Failed）：故障节点定期发送的消息用于上报故障消息。
  - MS（Manual Switching）：MS 节点定期发送的消息用于执行手工倒换。
  - NR，RB（No Request Request Block）：RPL Owner 节点在无故障和无手动命令情况下定期发送通知环上其他节点的消息；用于 RPL Owner 节点阻塞 RPL 后定期发送此消息。
  - NR（No Request）：用于故障或管理命令清除时发送的消息。

在环网保护过程中，会使用 Guard Timer、WTR（Wait To Restore，等待恢复） Timer、WTB（Wait To Block，等待阻塞）Timer 和 Holdoff Timer 四个定时器。

- Guard Timer：用于过滤掉无效的 R-APS 报文，避免环上接收到无效的 R-APS 报文而产生的环上节点的错误保护倒换动作。尤其是在较大的环网络中，节点故障后如果立即恢复，可能会收到从环上传来的邻居节点发送的故障通知，从而再次陷于 Down 状态，而这个通知却是由本节点引起的。配置环 Guard Timer 可以解决这个问题。
- WTR Timer：当工作路径恢复正常时，RPL Owner 上的 WTR Timer 开始计时。当 WTR Timer 计时器超时才恢复业务到工作路径。WTR Timer 用于避免工作路径不稳定而引起频繁倒换。
- WTB Timer：在返回模式下，当手动命令被清除时用于 RPL Owner 上延迟 RPL 端口阻塞的定时器，避免重复阻塞端口带来的震荡。
- Holdoff Timer：当检测到一个或多个新的故障后，如果设置的 Holdoff Timer 值不为 0，则启动 Holdoff Timer。在 Holdoff Timer 未超时时间内，系统会延时发送故障消息，即延时一段时间后再倒换到保护链路，可以防止链路震荡引起的频繁倒换。当 Holdoff Timer 超时，不论触发该定时器启动的这个故障是否仍然存在，都将检查链路状态，若存在故障则发送故障消息至保护倒换。

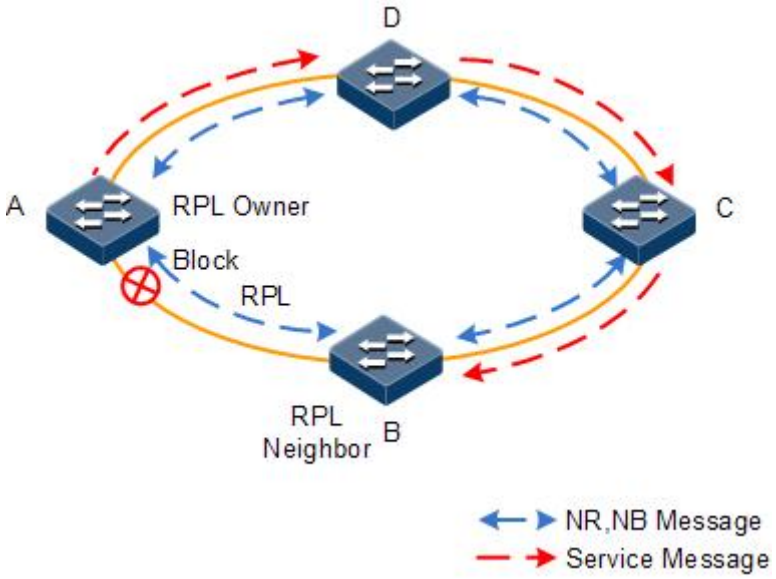
环状态

G.8032 定义了环网节点的 5 种状态。

- 空闲状态（Idle State）：无故障时的正常工作状态。
- 保护状态（Protecting State）：检测到链路故障后切换到的保护状态，该自动倒换过程由以太网 OAM（Operation, Administration and Maintenance，操作、管理与维护）的 CCM（Continuity Check Message，故障检测报文）检测到故障而触发。
- 挂起状态（Pending State）：故障修复前的悬而未决状态。
- 强制倒换状态（FS State）：下发强制倒换命令时的状态。
- 手工倒换状态（MS State）：下发手工倒换命令时的状态。

当系统无故障或故障恢复时，G.8032 以太网环处于空闲状态，空闲状态示意图如图 9-4 所示。

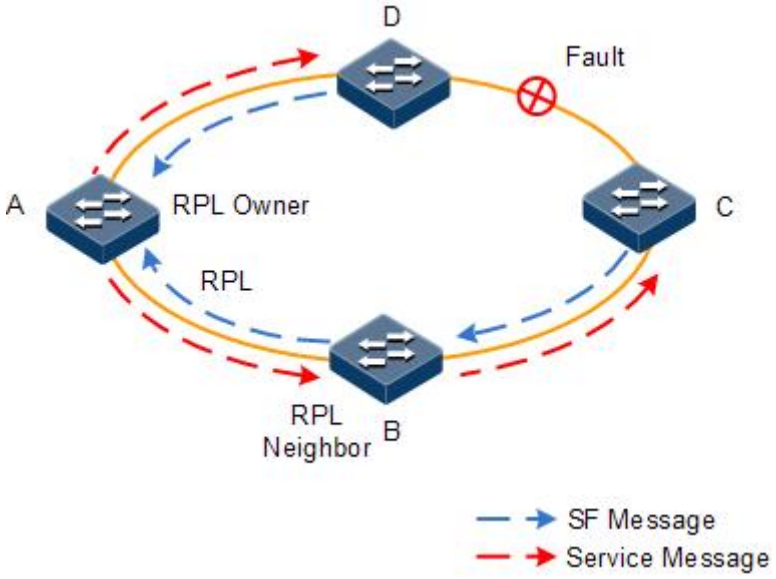
图 9-4 环网空闲状态



如图 9-4 所示，当环网处于空闲状态时，链路有如下的特征：

- 所有的节点在物理拓扑上以环的方式连接。
- G.8032 协议通过 RPL Owner 不断发送（NR，RB）信号，表示无故障，阻塞 RPL 链路，以确保不会成环。
- 相邻节点对每条链路使用以太网 OAM 中的 CCM 报文对链路进行监测。
- 当环路中出现故障时，G.8032 协议采用信号故障（SF）类型触发环路保护倒换。

图 9-5 环网保护状态



如图 9-5 所示当检测到环网故障时，启动自动保护倒换，进入保护状态：

- 在 Holdoff Timer 超时后，RPL Owner 触发与故障链路相邻的节点对故障链路进行阻塞，并发送 SF 信号向链路中其它节点报告该故障。如图 9-5 中的 C、D 节点间的故障发生后，D 节点和 C 节点分别向其它节点发送 SF 信号。
- SF 信号触发 RPL Owner 打开阻塞接口，并触发所有节点进行 FDB（Forwarding Database，转发数据库）清空，进入保护状态。

当故障恢复时，链路进行故障恢复倒换：

- 故障相邻的节点继续保持阻塞状态，Guard Timer 超时后，C、D 节点向其它节点发送 R-APS 的 NR 信号，表示没有本地故障请求。
- 当 RPL Owner 收到第一个 NR 信号后，立即启动 WTR Timer。
- 当 WTR 超时后，RPL Owner 阻塞 RPL，并发送 R-APS 的（NR，RB）信号，表示无本地故障请求，RPL 链路为阻塞状态。
- 其他节点收到该信号后，刷新 MAC 地址转发 FDB，发送 NR 信号的节点停止周期性发送报文，并打开原因故障阻塞的接口。
- 链路中各节点返回空闲状态。

子环

G.8032 的修订版增加了以太网多环的保护方案。子环是现存环网的附属环，通过互联节点（连接多个环的节点）与其它环或网络连接构成环。子环并不闭合，互联节点也不属于子环。

图 9-6 子环模型

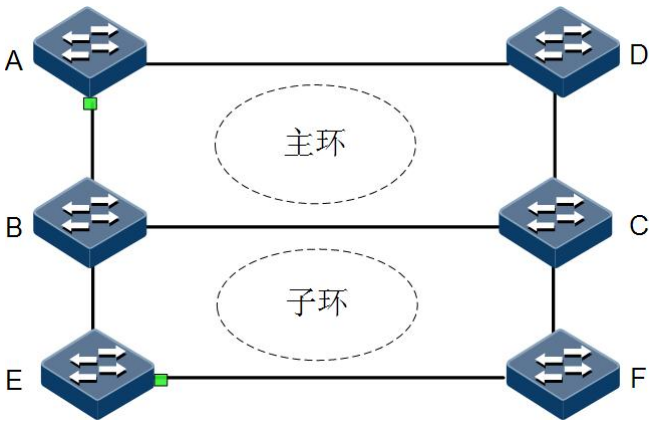


图 9-6 中，B 和 C 为互联节点，连接两个互联节点之间的通道称为 R-APS 虚通道，R-APS 虚通道用于相交环拓扑中的相交节点，如果相交环带有 R-APS 虚通道，则主环为子环 APS 报文提供虚拟通路，即子环 APS 报文会被传送到主环；如果不带，则主环不为子环提供虚拟通路，即子环 APS 报文在相交节点处被终结。主环和子环各自成环，每个环设定自己的 RPL Owner。多环的保护和单环类似，各自处理自己环内的故障。当互联节点间的共享链路出现故障时，主环切换到保护状态，子环不做处理。

因为子环的数据需要通过主环转发，所以主环设备上会存在子环的 MAC 地址列表，在子环出现故障时需要通过 Propagate 开关及时通知主环刷新 FDB，避免流量丢失。

环模式

返回模式与非返回模式的区别如下：

- 返回模式：等待 WTR 定时器超时后，流量切换到未发生故障前的链路进行转发。
- 非返回模式：等待 WTR 定时器超时后，流量不切换到未发生故障前的链路。缺省情况下，保护环处于返回模式。

子环的虚通路模式如下：

- **with** 模式：子环使用 R-APS 虚通道。主环为子环 APS 报文提供通路，子环相交节点收到子环 APS 报文会传送到主环，利用主环完成子环相交节点间的通信。
- **without** 模式：子环不使用 R-APS 虚通道。相交节点上子环 APS 报文要求终结，不会传送到主环。这种方式要求子环不能阻塞子环协议 VLAN（以保证子环报文可以通过 Owner）。

9.3.2 配置准备

场景

随着以太网向电信级网络的发展，语音、视频组播业务对以太网的冗余保护和故障恢复时间提出了更高的要求。现有的 STP 机制对故障恢复的收敛时间都在秒级，远远达不到要求。G.8032 技术通过定义环上节点的不同角色，在正常情况下阻断环路防止产生广播风暴，在环上链路或节点故障的情况下迅速切换到备份链路，从而实现消除环路、故障保护倒换和自动故障恢复等功能，并且故障保护倒换时间低于 50ms，支持单环、相交环和相切环三种组网方式。

G.8032 提供基于物理接口状态来检测故障，能够快速获知链路故障达到快速倒换的目的，适用于相邻设备之间。

前提

在配置 G.8032 之前，需完成以下任务：

- 连接接口并配置接口的物理参数，使接口的物理层状态为 Up；
- 创建 VLAN；
- 将接口加入 VLAN。

9.3.3 G.8032 的缺省配置

设备上 G.8032 的缺省配置如下。

功能	缺省值
保护环模式	返回模式
环 WTR 定时器	5min
环协议版本	2

功能	缺省值
Guard 定时器	500ms
环 HOLDOFF 定时器	0
相交节点上子环虚通路模式	with 模式
相交节点上环 Propagate 开关	禁止

9.3.4 创建 G.8032 保护环

注意

环上只允许一台设备配置为 RPL（Ring Protection Link，环保护链路）Owner，一台设备配置为 RPL Neighbour，其他设备只能配置为环转发节点。

相切环实际为两个独立的单环，配置与普通单环相同；相交环分为主环和子环，主环与单环配置相同，子环配置请参见“9.3.5 创建 G.8032 保护子环”。

ERPS 环接口需要工作在 switch 交换模式。

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**g8032 instance** *instance-id*	创建 G8032 实例并进入配置节点
3	JX(config-g8032-instance-*)#**control-vlan** *vlan-id*	指定控制 vlan，如果配置了 **revertive dsiable**，则保护环变为非返回模式。非返回模式与返回模式的区别在于，返回模式下工作链路故障恢复时，流量由环保护链路切换回工作链路，非返回模式下不切换。缺省情况下，保护环处于返回模式。
4	JX(config-g8032-instance-*)#**data-vlan** *vlan-id*	指定数据 vlan
5	JX(config-g8032-instance-*)# **add interface** *interface-type interface-number* [**rpl** { **owner** | **neighbor** }]	配置端口并指定 rpl 角色
6	JX(config-g8032-instance-*)#**version** { **v1** | **v2** }	配置协议版本。同一个环上所有节点协议版本应一致，版本 1 通过协议 VLAN 区分不同环，因此不同环需配置不同的协议 VLAN，即使使用协议版本 2 也建议不同环配置不同的协议 VLAN。

步骤	配置	说明
7	JX(config-g8032-instance-*)#**guard-timer** *guard-time*	配置环 Guard 定时器后，到时之前的时间内不转发收到的任何协议报文。在节点发生恢复事件时 guard 定时器启动。在较大的环网络中，节点故障后如果立即恢复，可能会收到从环上传来的邻居节点发送的过时的故障通知，从而再次陷于 Down 状态，而这个通知却是由本节点引起的。
8	JX(config-g8032-instance-*)#**wtr-timer** *wtr-time*	配置环 WTR 定时器。在返回模式下当工作链路故障恢复时，等待 WTR 定时器超时之后，才会恢复到工作链路上工作。
9	JX(config-g8032-instance-*)#**holdoff-timer** *holdoff-time*	配置环 HOLDOFF 定时器后，当工作链路故障时，系统会延时上报故障，即延时一段时间后再倒换到保护链路，可以防止工作链路震荡引起的频繁倒换。 **说明** HOLDOFF 定时器配置值较大时会影响 50ms 倒换性能，所以推荐使用缺省值 0。

9.3.5 创建 G.8032 保护子环

注意

- 只有相交环存在主环和子环之分。
- 相交环主环的配置与单环或相切环相同，具体配置请参见“创建 G.8032 保护环”。
- 配置相交环时应先配置主环，再配置子环，否则子环找不到主环接口，将无法建立子环虚通路。
- 子环的实例号必须大于主环的实例号。
- 相交环子环上的非相交节点配置与单环或相切环相同，具体配置请参见“9.3.4 创建 G.8032 保护环”。
- ERPS 环接口需要工作在 switch 交换模式。

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。

步骤	配置	说明
2	JX(config)# **g8032 instance** *instance-id*	创建实例
3	JX(config-g8032-instance-*)#**control-vlan** *vlan-id*	指定控制 vlan，如果配置了 **revertive dsiable**，则保护环变为非返回模式。非返回模式与返回模式的区别在于，返回模式下工作链路故障恢复时，流量由环保护链路切换回工作链路，非返回模式下不切换。缺省情况下，保护环处于返回模式。 **说明** 相交环两个相交节点之间的链路属于主环，所以在相交节点上配置子环时只能配置 port1 或 port2，不能同时配置。
4	JX(config-g8032-instance-*)#**data-vlan** *vlan-id*	指定数据 vlan
5	JX(config-g8032-instance-*)#**virtual-control-vlan** *vlan-id*	指定虚通道 vlan
5	JX(config-g8032-instance-*)# **add interface** *interface-type interface-number* [**rpl** { **owner** | **neighbor** }]	配置端口并指定 rpl 角色

步骤	配置	说明
6	JX(config-g8032-instance-*)# **add interface** *interface-type interface-number* **vc-mep**	配置端口并指定为 vc-mep

9.3.6 配置 G.8032 倒换控制

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config-g8032-instance-*)#**force-switch** *interface-type interface-number*	配置环上的流量强制倒换。 强制倒换可配置在多个环节点的多个接口上。
3	JX(config-g8032-instance-*)#**manual-switch** *interface-type interface-number*	配置环上的流量手工倒换，优先级低于强制倒换和工作链路故障时产生的自动倒换。 手工倒换只能配置在同一个环节点的一个接口上。

说明

缺省情况下，工作链路故障时流量会自动倒换到保护链路。所以只在某些特殊情况下才需要配置 ERPS 倒换控制。

9.3.7 检查配置

配置完成后，请在设备上执行以下命令检查配置结果。

序号	检查项	说明
1	JX#**show g8032 interface**	查看 G.8032 环接口状态信息。
2	JX#**show g8032 instance**	查看 G.8032 环状态信息。

9.3.8 维护

用户可以通过以下命令，维护设备 ERPS 特性的运行情况和配置情况。

命令	描述
JX(config-g8032-instance-*)#clear	清除环倒换控制命令（force-switch、manual-switch、WTR 定时器超时和 WTB 定时器超时）的作用。

9.3.9 配置单环 G.8032 保护示例

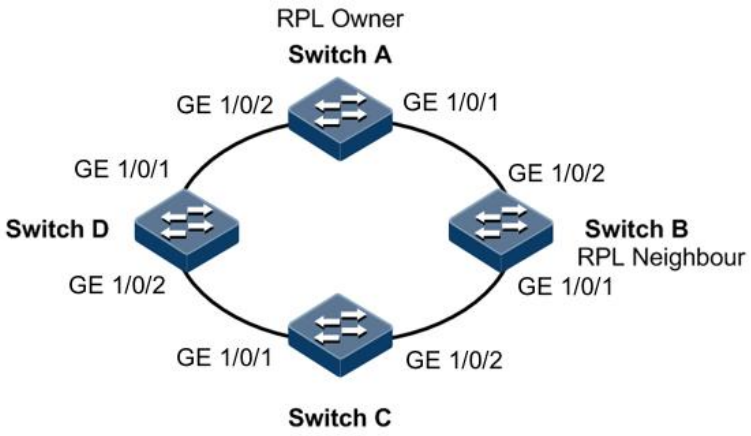
组网需求

如所示，为提高以太网络的可靠性，Switch A、Switch B、Switch C 和 Switch D 四台设备组成 G.8032 单环。

Switch A 设备作为 RPLOwner，Switch B 为 RPL Neighbour，Switch A 和 Switch B 之间的 RPL 链路阻塞。

协议控制 VLAN 为 1，阻塞的 VLAN 范围为缺省值 2～10。

图 9-7 单环 G.8032 应用组网示意图



配置步骤

步骤 1 配置接口加入 VLAN 2～VLAN 10。

配置 Switch A。

```
JX#hostname SwitchA
SwitchA#config
SwitchA(config)#interface ge 1/0/1 to ge 1/0/2
SwitchA(config-ge-1/0/1->ge-1/0/2)#port link-type trunk
SwitchA(config-ge-1/0/1->ge-1/0/2)#port trunk allow-pass vlan 2-10
SwitchA(config-ge-1/0/1->ge-1/0/2)#exit
```

Switch B，Switch C，Switch D 配置同 Switch A

步骤 2 创建 ERPS 保护环。

配置 Switch A。

```
SwitchA(config)#g8032 instance 1
SwitchA(config-g8032-instance-1)#control-vlan 1
SwitchA(config-g8032-instance-1)#data-vlan 2-10
SwitchA(config-g8032-instance-1)#add interface ge 1/0/1 rpl owner
SwitchA(config-g8032-instance-1)#add interface ge 1/0/2
```

配置 Switch B。

```
SwitchB(config)#g8032 instance 1
SwitchB(config-g8032-instance-1)#control-vlan 1
SwitchB(config-g8032-instance-1)#data-vlan 2-10
SwitchA(config-g8032-instance-1)#add interface ge 1/0/1 rpl neighbor
SwitchA(config-g8032-instance-1)#add interface ge 1/0/2
```

配置 Switch C。

```
SwitchC(config)#g8032 instance 1
SwitchC(config-g8032-instance-1)#control-vlan 1
SwitchC(config-g8032-instance-1)#data-vlan 2-10
SwitchC(config-g8032-instance-1)#add interface ge 1/0/1
SwitchC(config-g8032-instance-1)#add interface ge 1/0/2
```

配置 Switch D。

```
SwitchD(config)#g8032 instance 1
SwitchD(config-g8032-instance-1)#control-vlan 1
SwitchD(config-g8032-instance-1)#data-vlan 2-10
SwitchD(config-g8032-instance-1)#add interface ge 1/0/1
SwitchD(config-g8032-instance-1)#add interface ge 1/0/2
```

检查结果

在设备上通过 **show g8032 interface** 查看 G.8032 保护环是否生效。

以 Switch A 为例，RPL 链路被阻断防止环路产生，WTR 定时器超时后环状态信息如下：

```
SwitchA#show g8032 interface

Instance Interface      Role   Type   Operate   Forward    Rx-Count  Tx-
Count

-------------------------------------------------------------------------
--------------------------
1       ge 1/0/1       port1   rpl    working   blocking   0        15
1       ge 1/0/2       port2   normal  working   forwarding  0        11
-------------------------------------------------------------------------
--------------------------
```

手动断开 Switch B 和 Switch C 之间的链路模拟故障，在 Switch A 上再次使用命令查看 G.8032 保护环状态，RPL 链路切换为转发状态。

```
SwitchA#show g8032 interface
Instance Interface      Role   Type   Operate   Forward    Rx-Count  Tx-
Count
```

```
------------------------------------------------------------------------------
--------------------------
1       ge 1/0/1       port1   rpl     working   forwarding  16       41
1       ge 1/0/2       port2   normal  failed    blocking    18       42
------------------------------------------------------------------------------
--------------------------
```

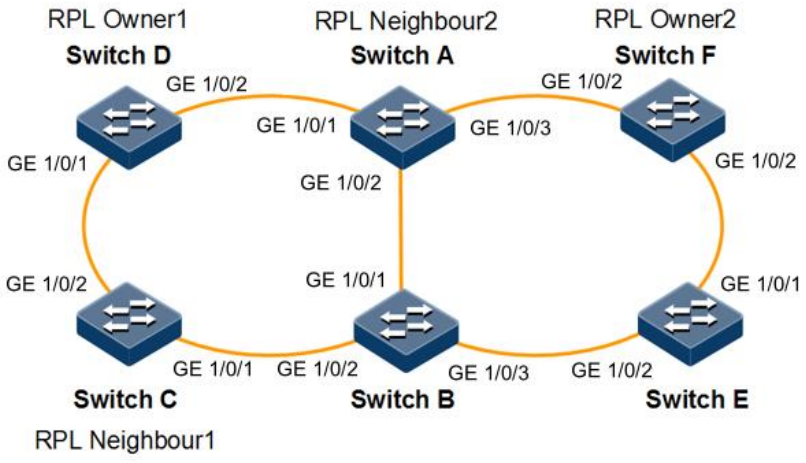
9.3.10 配置相交环 G.8032 保护示例

组网需求

如图 9-8 所示，为提高以太网络的可靠性，Switch A、Switch B、Switch C、Switch D、Switch E 和 Switch F 六台设备以 ERPS 相交环的形式组网。

- Switch A、Switch B、Switch C 和 Switch D 组成主环，Switch D 为主环 RPL Owner，Switch C 为主环 RPL Neighbour，阻断端口为 Switch D 的 GE 1/0/1，协议控制 VLAN 采用 VLAN 1。
- Switch A、Switch B、Switch E 和 Switch F 组成子环，Switch F 为子环 RPL Owner，Switch A 为子环 RPL Neighbour，阻断端口为 Switch F 的 GE 1/0/1，协议控制 VLAN 为 VLAN 4094。
- 主环和子环的阻塞 VLAN 范围均为缺省的 1～4094。

图 9-8 相交环 G.8032 应用组网示意图



配置步骤

步骤 1 创建 VLAN，并配置接口加入 VLAN。

配置 Switch A & B。

```
Switch#configure
Switch(config)#vlan 1-4094
Switch(config)#interface ge 1/0/1 to ge 1/0/3
Switch(config-ge-1/0/1->ge-1/0/3)#port link-type trunk
Switch(config-ge-1/0/1->ge-1/0/3)#port trunk allow-pass vlan all
Switch(config-ge-1/0/1->ge-1/0/3)#exit
```

配置 Switch C & D & E & F。

```
Switch#config
Switch(config)#vlan 1-4094
Switch(config)#interface ge 1/0/1 to GE 1/0/2
Switch(config-ge-1/0/1->ge-1/0/2)#port link-type trunk
Switch(config-ge-1/0/1->ge-1/0/2)#port trunk allow-pass vlan all
Switch(config-ge-1/0/1->ge-1/0/2)#exit
```

步骤 2 创建 G.8032 保护环主环。

配置 Switch A & B。

```
Switch(config)#g8032 instance 1
Switch(config-g8032-instance-1)#control-vlan 1
Switch(config-g8032-instance-1)#data-vlan 1-4094
Switch(config-g8032-instance-1)#add interface ge 1/0/1
Switch(config-g8032-instance-1)#add interface ge 1/0/2
```

配置 Switch C。

```
Switch(config)#g8032 instance 1
Switch(config-g8032-instance-1)#control-vlan 1
Switch(config-g8032-instance-1)#data-vlan 1-4094
Switch(config-g8032-instance-1)#add interface ge 1/0/1
Switch(config-g8032-instance-1)#add interface ge 1/0/2 rpl neighbour
```

配置 Switch D。

```
Switch(config)#g8032 instance 1
Switch(config-g8032-instance-1)#control-vlan 1
Switch(config-g8032-instance-1)#data-vlan 1-4094
Switch(config-g8032-instance-1)#add interface ge 1/0/1 rpl owner
Switch(config-g8032-instance-1)#add interface ge 1/0/2
```

步骤 3 配置 G.8032 保护环子环。

配置 Switch A。

```
Switch(config)#g8032 instance 2
Switch(config-g8032-instance-2)#control-vlan 4094
Switch(config-g8032-instance-2)#data-vlan 1-4094
Switch(config-g8032-instance-2)#add interface ge 1/0/3 rpl owner
Switch(config-g8032-instance-2)#virtual-control-vlan 3
```

配置 Switch B。

```
Switch(config)#g8032 instance 2
Switch(config-g8032-instance-2)#control-vlan 4094
Switch(config-g8032-instance-2)#data-vlan 1-4094
Switch(config-g8032-instance-2)#add interface ge 1/0/3
Switch(config-g8032-instance-2)virtual-control-vlan 3
```

配置 Switch E。

```
Switch(config)#g8032 instance 2
Switch(config-g8032-instance-2)#control-vlan 4094
Switch(config-g8032-instance-2)#data-vlan 1-4094
Switch(config-g8032-instance-2)#add interface ge 1/0/1
Switch(config-g8032-instance-2)#add interface ge 1/0/2
```

配置 Switch F。

```
Switch(config)#g8032 instance 2
Switch(config-g8032-instance-2)#control-vlan 4094
Switch(config-g8032-instance-2)#data-vlan 1-4094
Switch(config-g8032-instance-2)#add interface ge 1/0/1 rpl neighbour
Switch(config-g8032-instance-2)#add interface ge 1/0/2
```

检查结果

在设备上通过 **show g8032 interface** 查看 G.8032 保护环是否生效。

分别在 Switch A、Switch D 和 Switch F 上检查，等待 WTR 定时器超时后，结果如下。

```
SwitchD#show g8032 interface

Instance Interface      Role   Type   Operate   Forward    Rx-Count  Tx-
Count

-------------------------------------------------------------------------
---------------------------
1       ge 1/0/1       port1   rpl    working   blocking    0        15
1       ge 1/0/2       port2   normal  working   forwarding  0        11
-------------------------------------------------------------------------
---------------------------
```

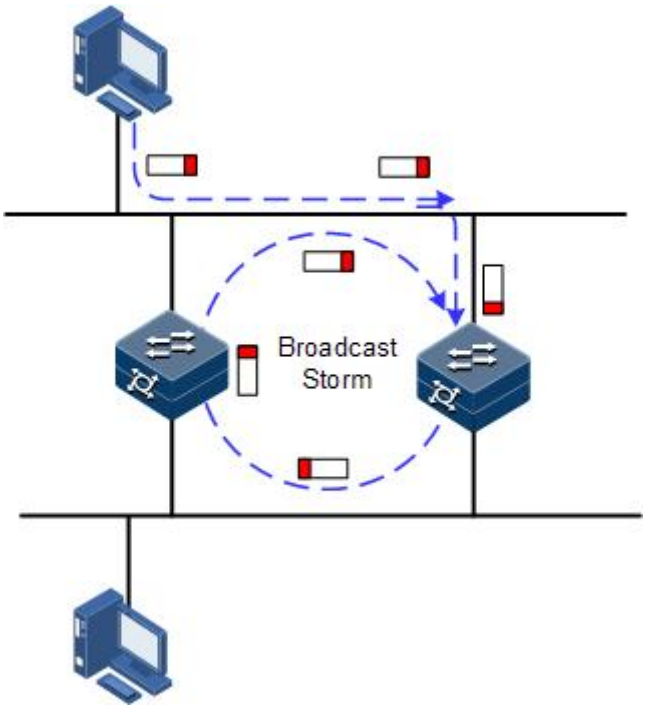
9.4 STP/RSTP

9.4.1 简介

STP

随着网络结构的日益复杂和网络中交换机数量的增多，网络环路成为以太网中最突出的问题。由于交换机对报文的广播机制，网络环路会使得网络中产生网络风暴，耗尽网络资源，对正常的数据转发产生严重的影响。由于网络环路产生的网络风暴示意图如下图所示。

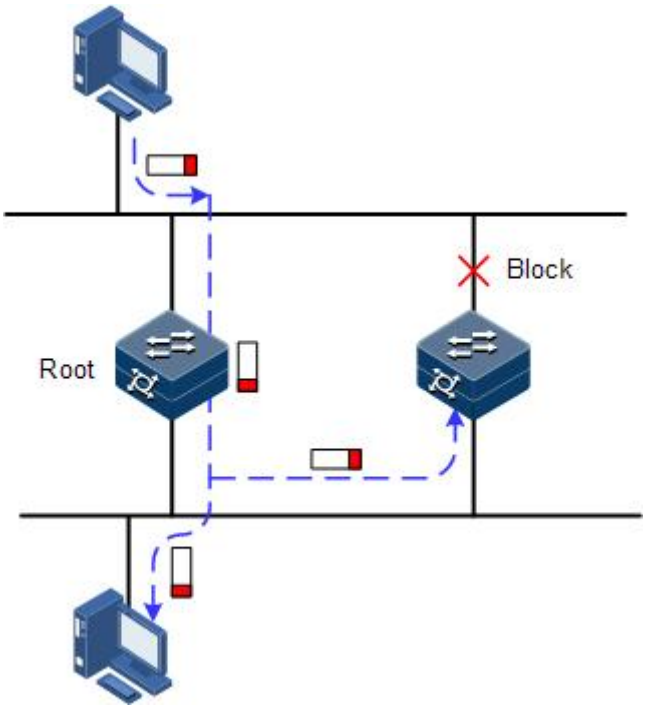
图 9-9 网络环路造成网络风暴示意图



STP（Spanning Tree Protocol，生成树协议）是根据 IEEE 802.1d 标准建立的，用于在局域网中消除数据链路层物理环路的协议。

运行 STP 的设备可以通过彼此交互 BPDU（Bridge Protocol Data Unit，桥协议数据单元）报文进行根交换机的选举、根端口和指定端口的选择，并根据选择结果对设备中存在环路的接口进行逻辑上的阻塞，最终将环路网络结构修剪成以某一台设备为根（Root）的无环路的树型网络结构，从而防止报文在环路网络中不断增生和无限循环而导致广播风暴，并避免主机由于重复接收相同的报文造成的报文处理能力下降的问题发生。运行了 STP 协议的环网示意图如下图所示。

图 9-10 运行 STP 协议的环网示意图



虽然 STP 协议能够很好的消除环网，防止广播风暴的产生，但是随着应用的深入和网络技术的发展，STP 协议的缺点也逐渐暴露了出来。

STP 协议的主要缺点表现在收敛速度较慢。

RSTP

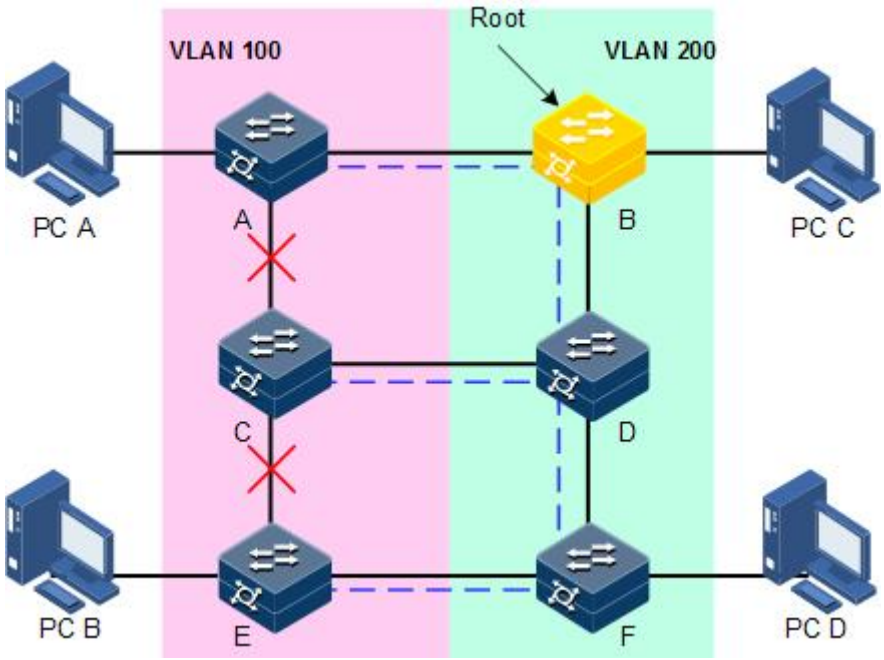
为了弥补 STP 协议收敛速度较慢的不足，IEEE 802.1w 制定了 RSTP（Rapid Spanning Tree Protocol，快速生成树协议）。在普通 STP 协议的基础上，增加了接口可以快速由阻塞状态转变为转发状态的机制，加快了拓扑的收敛速度。

STP/RSTP 协议的目的都是将一个桥接的局域网，修剪成为逻辑拓扑上的一棵单一的生成树，从而避免广播风暴的产生。

由于 VLAN 技术快速发展，STP/RSTP 的局限性逐渐暴露出来。STP/RSTP 协议将网络拓扑修剪为单一生成树，会导致以下问题：

- 整个交换网络只有一个生成树，在网络规模比较大的时候会导致较长的收敛时间。
- 链路被阻塞后将不承载任何流量，造成带宽的浪费。
- 在网络结构不对称时，可能造成部分 VLAN 的报文无法转发。如下图所示，由于 RSTP 协议，选举 Switch B 为根交换机，且逻辑上阻断了 Switch A 和 Switch C 之间的链路，造成 VLAN 100 中的 VLAN 报文无法转发，Switch A 和 Switch C 无法通信。

图 9-11 RSTP 协议造成 VLAN 报文无法转发示意图



9.4.2 配置准备

场景

在较大型的局域网中，有多台设备进行级连满足多主机相互访问的需求，为防止设备之间组成环路造成 MAC 地址学习错误，并导致数据帧快速在环路中进行复制转发造成广播风暴、网络瘫痪，需要在这些设备上开启 STP。通过 STP 协议计算，阻塞掉环路当中的其中一个接口，保证每一个数据流去往目的主机的路径只有一条，并且被 STP 协议计算为最优路径。

前提

无

9.4.3 STP 的缺省配置

设备上 STP 的缺省配置如下。

功能	缺省值
全局 STP 功能状态	禁止
接口 STP 功能状态	使能
设备的 STP 优先级	32768
接口的 STP 优先级	128

功能	缺省值
接口的路径开销	0
Max Age 定时器	20s
Hello Time 定时器	2s
Forward Delay 定时器	15s

9.4.4 使能 STP 功能

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**stp mode** { **stp** | **rstp** | **mstp** | **default** }	配置生成树的运行模式。
3	JX(config)#**interface** *interface-type interface-number*	进入物理接口配置模式或聚合组配置模式。以下步骤以物理接口配置模式为例。
4	JX(config-ge-1/0/*)#**stp enable**	使能接口生成树协议。

9.4.5 配置 STP 参数

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**stp priority** *priority-value*	配置设备优先级。
3	JX(config)#**interface** *interface-type interface-number* JX(config-ge-1/0/*)#stp **priority** *priority-value*	配置设备接口优先级。
4	JX(config-ge-1/0/*)#**stp path-cost** *cost-value*	配置设备接口路径开销。
5	JX(config)#**stp hello-time** *value*	配置 Hello Time 的值。
6	JX(config)#**stp forward-delay** *value*	配置 Forward Delay 的值。
7	JX(config)#**stp max-age** *value*	配置 Max Age 的值。
8	JX(config)#**stp pathcost-standard** { **dot1d-1998** | **dot1t** }	配置生成树路径开销计算标准。

9.4.6 配置 RSTP 边缘接口

边缘接口是指不直接与任何设备连接，也不通过接口所连接的网络间接与任何设备相连的接口。

设置为边缘接口能够使该接口的状态迅速转变为转发状态，而不需要时间等待，对于直接与用户终端相连的以太网接口，为能使其快速迁移到转发状态，应将其设置为边缘接口。

当某个接口设置为边缘接口（enable）时，当接口收到 BPDU 后实际运行值会变为非边缘接口。当某个接口设置为非边缘接口（disable）时，同样，无论其实际情况下为边缘或非边缘接口，此接口会保持为非边缘接口，直到配置改变。

缺省情况下，以太网设备中所有接口的均设置为 disable。

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**interface** *interface-type interface-number*	进入物理接口配置模式或聚合组配置模式。以下步骤以物理接口配置模式为例。
3	JX(config-ge-1/0/*)#**stp edge-port { enable | disable }**	配置 RSTP 边缘接口属性。

9.4.7 配置 RSTP 链路类型

点对点链路相连的两个接口可以通过传送同步报文快速迁移到转发状态，减少了不必要的转发延迟时间。缺省情况下，根据双工状态设定接口的链路类型。全双工接口被认为是点到点链路，半双工被认作共享链路。

用户可以手工强行配置当前以太网接口与点对点链路相连，但是如果该链路不是点到点链路会使系统出现问题，一般情况下建议用户将此配置项设为自动状态，由系统自动发现接口是否与点到点链路相连。

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**interface** *interface-type interface-number*	进入物理接口配置模式或聚合组配置模式。以下步骤以物理接口配置模式为例。
3	JX(config-ge-1/0/*)#**stp point-to-point { force-true | force-false | auto }**	配置接口的链路类型。

9.4.8 检查配置

配置完成后，请在设备上执行以下命令检查配置结果。

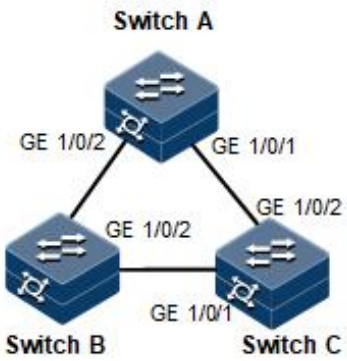
序号	检查项	说明
1	JX#show stp infomation	查看 STP 基本配置信息。
2	JX#show stp interface	查看接口下生成树配置信息。
3	JX#show stp bridge	查看 stp 根桥信息

9.4.9 配置 STP 示例

组网需求

如下图所示，三台设备 Switch A、Switch B 和 Switch C 组网成一个环，需在物理链路成环情况下解决环路问题，三台设备上需开启 STP，并设置 Switch A 的优先级为 0，Switch B 到 Switch A 的开销改为 10。

图 9-12 STP 应用组网示意图



配置步骤

步骤 1 在三台设备上均开启 STP 功能。

配置 Switch A。

```
JX#hostname SwitchA
SwitchA#configure
SwitchA(config)#stp mode stp
```

配置 Switch B。

```
JX#hostname SwitchB
SwitchB#configure
SwitchB(config)#stp mode stp
```

配置 Switch C。

```
JX#hostname SwitchC
SwitchC#configure
SwitchC(config)#stp mode stp
```

步骤 2 配置三台设备的接口模式。

配置 Switch A。

```
SwitchA(config)#interface ge 1/0/1
SwitchA(config-ge-1/0/1)#port link-type trunk
SwitchA(config-ge-1/0/1)#stp enable
SwitchA(config-ge-1/0/1)#exit
SwitchA(config)#interface ge 1/0/2
SwitchA(config-ge-1/0/2)#port link-type trunk
SwitchA(config-ge-1/0/2)#stp enable
SwitchA(config-ge-1/0/2)#exit
```

配置 Switch B。

```
SwitchB(config)#interface ge 1/0/1
SwitchB(config-ge-1/0/1)#port link-type trunk
SwitchB(config-ge-1/0/1)#stp enable
SwitchB(config-ge-1/0/1)#exit
SwitchB(config)#interface ge 1/0/2
SwitchB(config-ge-1/0/2)#port link-type trunk
SwitchB(config-ge-1/0/2)#stp enable
SwitchB(config-ge-1/0/2)#exit
```

配置 Switch C。

```
SwitchC(config)#interface ge 1/0/1
SwitchC(config-ge-1/0/1)#port link-type trunk
SwitchC(config-ge-1/0/1)#stp enable
SwitchC(config-ge-1/0/1)#exit
SwitchC(config)#interface ge 1/0/2
SwitchC(config-ge-1/0/2)#port link-type trunk
SwitchC(config-ge-1/0/2)#stp enable
SwitchC(config-ge-1/0/2)#exit
```

步骤 3 配置生成树优先级及接口路径开销。

配置 Switch A。

```
SwitchA(config)#stp priority 0
SwitchA(config)#interface ge 1/0/2
SwitchA(config-ge-1/0/2)#stp path-cost 10
```

配置 Switch B。

```
SwitchB(config)#interface ge 1/0/1
SwitchB(config-ge-1/0/1)#stp path-cost 10
```

检查结果

通过 **show stp** 命令查看桥状态，以 Switch A 为例。

```
SwitchA#show stp infomation
```

```
--------------------------------------------------------------------------
-------
  Mode                              : stp
  Trap state                         : disable
  Bridge type                        : customer
  BPDU Guard state                    : disable
  TC protection state                  : disable
  TC protection threshold               : 2
  Hello time                         : 2
  Max age                           : 20
  Forward delay                      : 15
  Max hops                          : 20
  Time factor                        : 6
  Format selector                     : 0
  Revision level                     : 0
  Config name                        : region
  TC flush arp state                  : disable
  Migration time                     : 3
  Pathcost standard                   : dot1t
  TC holdoff time                    : 10
  Transmit limit                     : 6 (packets/s)
  Link detection state                : enable
  Edge default state                  : disable
--------------------------------------------------------------------------
-------
```

通过 **show stp interface** 查看接口状态，以 Switch A 为例。

```
SwitchA#show stp interface
  MSTID Port         Role         State        Protection    Region
--------------------------------------------------------------------------
------------
  0    ge-1/0/1     designated    forward       --           different
  0    ge-1/0/2     designated    forward       --           different
```

9.5 MSTP

9.5.1 简介

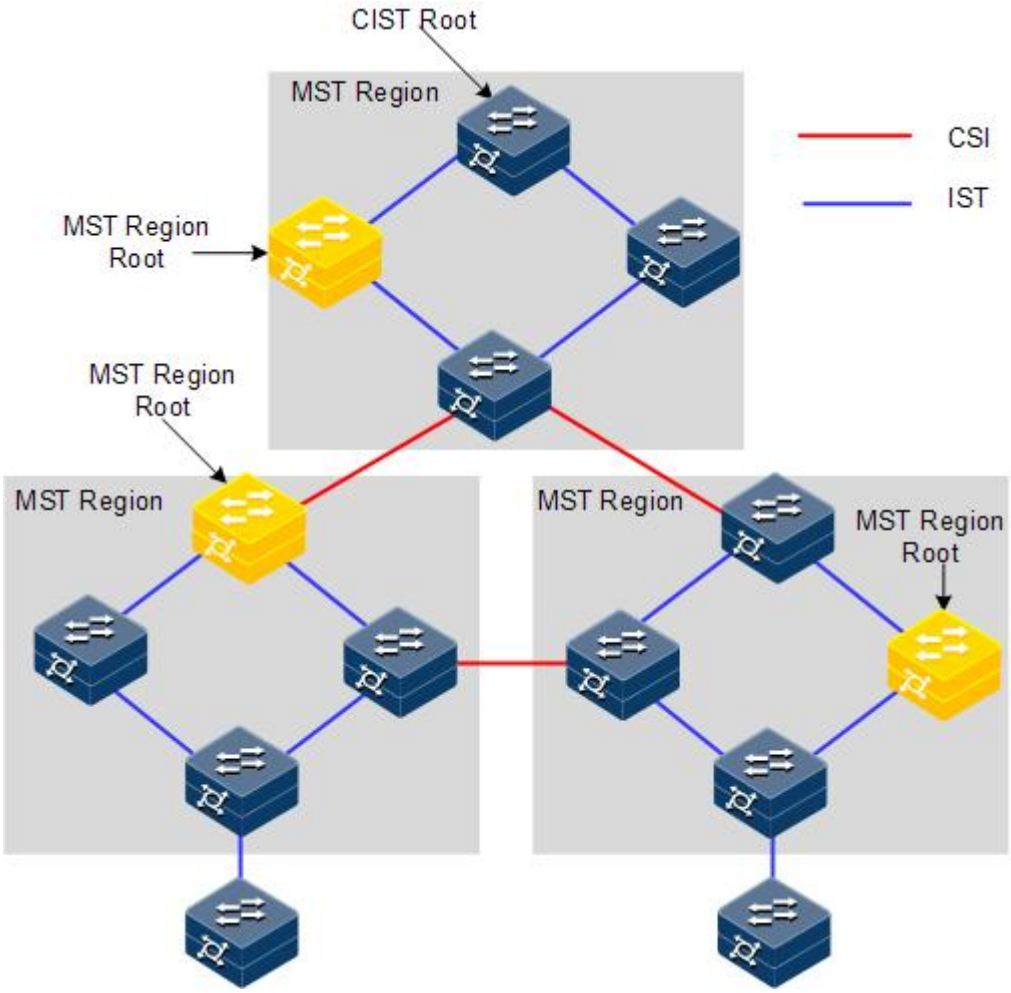
IEEE 802.1s 标准定义了 MSTP（Multiple Spanning Tree Protocol，多生成树协议）。MSTP 可以弥补 STP 和 RSTP 的缺陷，既可以快速收敛，也能使不同 VLAN 的流量沿各自的路径分发，从而提供了很好的负荷分担机制。

MSTP 把一个交换网络划分成多个域，每个域叫做一个 MST 域。每个域内形成多棵生成树，生成树之间彼此独立。每棵生成树叫做一个 MSTI（Multiple Spanning Tree Instance，多生成树实例）。

MSTP 协议引入了 CST（Common Spanning Tree，公共生成树）和 IST（Internal Spanning Tree，内部生成树）的概念。其中 CST 是指把 MST 域当成一个整体时，计算生成的一棵生成树。而 IST 是指在 MST 域内部生成的生成树。

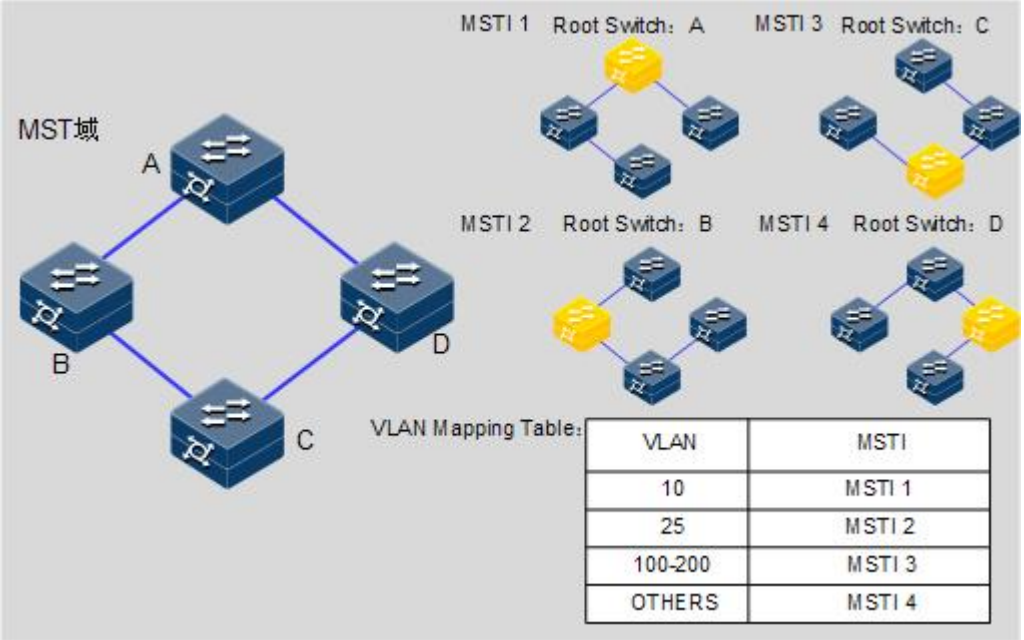
与 STP 和 RSTP 相比，MSTP 中还引入了总根（CIST Root）和域根（MST Region Root）的概念。总根是一个全局概念，对于所有运行的 STP/RSTP/MSTP 的交换机只能有 1 个总根，也即是 CIST 的根。而域根是一个局部概念，是相对于某个域的某个实例而言。如下图所示，所有相连的设备，总根只有 1 个，而每个域包含的域根数目与实例个数相关。

图 9-13 MSTP 网络基本概念示意图



在每个 MST 域中，可以有不同的 MST 实例，通过设置 VLAN 映射表（即 VLAN 和 MSTI 的对应关系表），把 VLAN 和 MSTI 联系起来。MSTI 概念示意图如下图所示。

图 9-14 MSTI 概念示意图

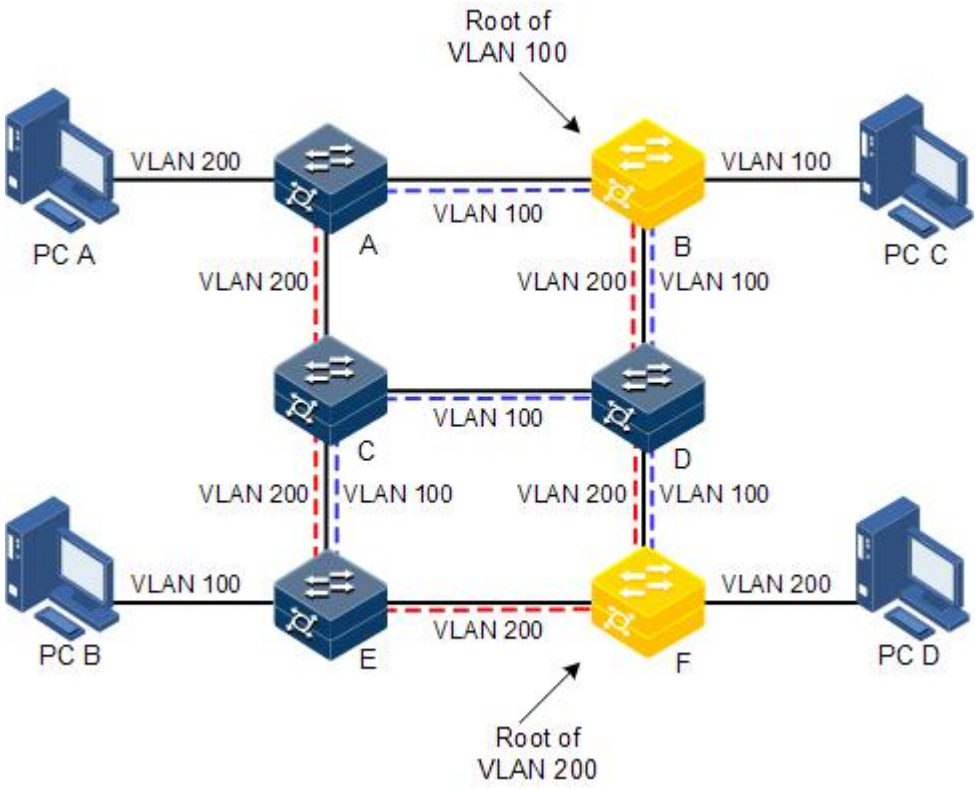


说明

每个 VLAN 只能对应一个 MSTI，即同一 VLAN 的数据只能在一个 MSTI 中传输；而一个 MSTI 可能对应多个 VLAN。

MSTP 协议相对于之前的 STP 协议和 RSTP 协议，优势非常明显。MSTP 具有 VLAN 认知能力，可以实现负载均衡分担，可以实现类似 RSTP 的端口状态快速切换，可以捆绑多个 VLAN 到一个 MST 实例中，以降低资源占用率。此外，网络中运行 MSTP 协议的设备可以很好的与运行 STP 协议和 RSTP 协议的设备兼容。

图 9-15 MST 域内多生成树实例组网示意图



将 MSTP 应用于如上图所示的网络，经计算最终生成两棵生成树（也即 2 个 MST 实例）：

- MSTI1 以 B 为根交换设备，转发 VLAN100 的报文；
- MSTI2 以 F 为根交换设备，转发 VLAN200 的报文。

这样所有 VLAN 内部可以互通，同时不同 VLAN 的报文沿不同的路径转发，实现了负荷分担。

9.5.2 配置准备

场景

大型局域网或小区汇聚时，汇聚设备之间组成一个环作为线路的备份，在实现线路备份的同时，需要防止环路以及实现业务的负载分担，MSTP 协议可以为每一个或一组 VLAN 选择不同且唯一的转发路径。

前提

无

9.5.3 MSTP 的缺省配置

设备上 MSTP 的缺省配置如下。

功能	缺省值
全局 MSTP 功能状态	禁止
接口 MSTP 功能状态	使能
MST 域的最大跳数	20
设备的 MSTP 优先级	32768
接口的 MSTP 优先级	128
接口的路径开销	0
每个 Hello time 内的最大发送报文数量	3
Max Age 定时器	20s
Hello Time 定时器	2s
Forward Delay 定时器	15s
MST 域的修订级别	0
TC 保护	禁用
TC 保护功能的阈值	1

9.5.4 使能 MSTP 功能

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**stp mode mstp**	配置生成树模式为 MSTP。
3	JX(config)#**interface** *interface-type interface-number*	进入物理接口配置模式或聚合组配置模式。以下步骤以物理接口配置模式为例。
4	JX(config-ge-1/0/*)#**stp enable**	使能接口生成树协议。

9.5.5 配置 MST 域和 MST 域最大跳数

当设备的运行模式为 MSTP 时，可为设备设置其归属的域信息。设备属于哪个 MST 域，是由域名，VLAN 映射表，MSTP 修订级别配置决定的。用户可以通过下面的配置过程将当前设备划分在一个特定的 MST 域内。

MST 域的最大跳数限制了 MST 域的规模。从域内的生成树的根桥开始，域内的配置消息（BPDU）每经过一台设备的转发跳数就被减 1，设备将丢弃收到的跳数为 0 的配置消息。使处于最大跳数外的设备无法参与生成树的计算，从而限制了 MST 域的规模。

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**stp instance** *instance-id* **vlan** *vlan-list*	配置 MST 域的 VLAN 到实例映射关系。
3	JX(config)#**stp config-name** *name*	配置 MST 域名。
4	JX(config)#**stp revision-level** *level-value*	配置 MST 域的修订级别。
6	JX(config)#**stp max-hops** *hops-value*	配置设备 MST 域最大跳数。

说明

当且仅当配置的设备为域根时，配置的最大跳数才作为 MST 域的最大跳数，其他非域根桥配置此项无效。

9.5.6 配置根桥/备份根桥

MSTP 根桥的选举，一方面可以通过配置设备的优先级，然后经过生成树计算，来确定生成树的根桥或备份根桥；另一方面，用户也可以通过此命令来直接指定。当根桥出现故障或被关机时，备份根桥可以取代根桥成为相应实例的根桥。此时如果用户设置了新的根桥，则备份根桥将不会成为根桥。如果用户为一棵生成树实例配置了多个备份根桥，当根桥失效时，MSTP 将选择 MAC 地址最小的那个备份根桥作为根桥。

注意

如果采用这种直接指定根桥的方式，建议用户不要再修改网络中任何设备的优先级，否则，可能会造成指定根桥或备份根桥无效。

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**spanning-tree** [**instance** *instance-id*] **root** { **primary** | **secondary** }	为某个生成树实例，设置设备为根桥或备份根桥。

- 用户可以通过参数 **instance** *instance-id* 确定根桥或备份根桥生效的实例。如果 *instance-id* 取值为 0，或者省去参数 **instance** *instance-id* 时，当前设备将被指定为 CIST 的根桥或备份根桥。
- 设备在各实例中的根类型是互相独立的，即它既可以作为一个实例的根桥或备份根桥，同时又可以作为其他生成树实例的根桥或备份根桥。但在同一棵生成树实例中，同一台设备不能既作为根桥，又作为备份根桥。
- 用户不能同时为一棵生成树实例指定两个或两个以上的根桥。相反，用户可以给同一棵生成树指定多个备份树根。一般情况下，建议用户给一棵生成树指定一个树根和多个备份树根。

9.5.7 配置设备接口和系统的优先级

接口是否被选为根接口需要根据接口优先级进行判断。同等条件下，接口优先级值越小，接口越优先被选为根接口。接口可在不同的实例中具有不同的接口优先级，也可以在不同实例中充当不同的角色。

设备 Bridge ID 的大小决定了这台设备是否能够被选作生成树的根。通过配置较小的优先级，可以得到较小的设备 Bridge ID，达到指定某台设备成为生成树树根的目的。优先级相同的情况下，MAC 地址小的为树根。

与配置根与备份根相同，优先级在不同实例中的配置相互独立。用户可以通过参数 **instance** *instance-id* 确定的配置优先级的实例。如果 *instance-id* 取值为 0，或者省去参数 **instance** *instance-id* 时，则是为 CIST 配置的桥优先级。

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**interface** *interface-type interface-number*	进入物理接口配置模式或聚合组配置模式。以下步骤以物理接口配置模式为例。
3	JX(config-ge-1/0/*)#**stp** [**instance** *instance-id*] **priority** *priority-value* JX(config-ge-1/0/*)#**exit**	配置某个生成树实例的接口优先级。
4	JX(config)#**stp** [**instance** *instance-id*] **priority** *priority-value*	配置某个生成树实例的系统优先级。

优先级取值必须为 4096 的倍数，如 0、4096、8192 等，缺省值为 32768。

9.5.8 配置接口的路径开销

在选举根接口（root port）和指定接口（designated port）时，路径开销越小的接口越容易被选举为根接口或者指定接口。接口的路径开销在不同实例中的配置相互独立。用户可以通过参数 **instance** *instance-id* 确定的配置接口的内部路径开销的实例。如果 *instance-id* 取值为 0，或者省去参数 **instance** *instance-id* 时，则是为 CIST 配置的接口内部路径开销。

接口的开销一般依据其物理特性，缺省情况如下：

- 10Mbit/s 为 2000000
- 100Mbit/s 为 200000
- 1000Mbit/s 为 20000
- 10Gbit/s 为 2000

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**interface** *interface-type interface-number*	进入物理接口配置模式或聚合组配置模式。以下步骤以物理接口配置模式为例。
3	JX(config-ge-1/0/*)#**stp** [**instance** *instance-id*] **path-cost** *cost-value*	配置接口的路径开销。

9.5.9 配置接口最大发送速率

MSTP 每 Hello Time 时间内允许发送的最大 BPDU 数量。此参数是一个相对值，没有单位，该参数被配置得越大，则每个 Hello Time 内允许发送的报文个数就越多，同时也会占用会更多的设备资源。与时间参数相同，只有根设备的此项配置生效。

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**stp transmit-limit** *value*	配置接口最大发送速率。

9.5.10 配置 MSTP 定时器

- Hello Time：设备定期发送桥配置信息（BPDU）的时间间隔，用于设备检测链路是否存在故障。设备每隔 Hello Time 时间，会向周围的设备发送 Hello 报文，以确认链路是否存在故障。缺省值为 2s，用户可以根据网络情况对此值进行调整。

当网络中链路出现频繁变化时，可以适当缩短该值，来增强生成树协议的健壮性。相反，增大此值则可以降低生成树协议对系统 CPU 资源的占用率。

- Forward Delay：保证设备状态安全迁移的时间参数。链路故障会引发网络重新进行生成树的计算，不过重新计算得到的新配置消息无法立刻传遍整个网络。如果新选出的根接口和指定接口立刻开始数据转发，可能会造成暂时性的路径回环。为此协议采用了一种状态迁移的机制：根接口和指定接口重新开始数据转发之前，要经历一个中间状态（学习状态），中间状态经过 Forward Delay 时间的延时后，才能进入转发状态。这个延时保证了新的配置消息已经传遍整个网络。用户可以根据实际情况调整该值，当网络拓扑不频繁变化时可以将该值减小，反之增大。
- Max Age：生成树协议所使用的桥配置信息有生存周期，用来判断配置消息是否过时。设备会将过时的配置消息丢弃。当桥配置信息过期后，生成树协议将重新计算生成树。缺省值为 20s，该值过小会导致生成树重计算过于频繁，过大则会导致生成树协议不能及时适应网络拓扑结构的变化。

整个交换网络中所有的设备采用 CIST 根设备上的三个时间参数，因此只有在根设备上的配置生效。

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**stp hello-time** *value*	配置 HelloTime 的值。
3	JX(config)#**stp forward-delay** *value*	配置 Forward Delay 的值。
4	JX(config)#**stp max-age** *value*	配置 MaxAge 的值。

9.5.11 配置边缘接口

边缘接口是指不直接与任何设备连接，也不通过接口所连接的网络间接与任何设备相连的接口。

设置为边缘接口能够使该接口的状态迅速转变为转发状态，而不需要时间等待，对于直接与用户终端相连的以太网接口，为能使其快速迁移到转发状态，应将其设置为边缘接口。

当某个接口设置为边缘接口自动检测（auto）则边缘接口的属性是由实际情况决定的。当某个接口设置为边缘接口（force-true）时，当接口收到 BPDU 后实际运行值会变为非边缘接口。当某个接口设置为非边缘接口（force-false）时，同样，无论其实际情况下为边缘或非边缘接口，此接口会保持为非边缘接口，直到配置改变。

缺省情况下，以太网设备中所有接口的均设置为自动检测属性。

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。

步骤	配置	说明
2	JX(config)#**interface** *interface-type interface-number*	进入物理接口配置模式或聚合组配置模式。以下步骤以物理接口配置模式为例。
3	JX(config-ge-1/0/*)#**stp edge-port** { **enable** | **disable** }	配置接口边缘接口属性。

9.5.12 配置 BPDU 过滤

用户使能边缘接口的 BPDU 过滤功能后，边缘接口不会发送 BPDU 报文，也不会处理收到的 BPDU 报文。

请在设备上进行以下配置。

步骤	配置	说明
1	JX#config	进入全局配置模式。
2	JX(config)#**interface** *interface-type interface-number*	进入物理接口配置模式或聚合组配置模式。以下步骤以物理接口配置模式为例。
3	JX(config-ge-1/0/*)#**stp bpdu-filter** { **enable** | **disable** }	配置边缘接口的 BPDU 过滤功能。

9.5.13 配置 BPDU 保护

在交换机上，通常将直接与用户终端（如 PC 机）或文件服务器等非交换机设备相连的接口配置为边缘接口，以实现这些接口的快速迁移。

正常情况下，这些边缘接口不会收到 BPDU。如果有人伪造 BPDU 恶意攻击交换机，当这些接口接收到 BPDU 时，会自动将这些接口设置为非边缘接口，并重新进行生成树计算，从而引起网络震荡。

MSTP 提供 BPDU 保护功能来防止这种攻击。启动 BPDU 保护功能后，可以防止伪造 BPDU 恶意攻击。

如果使能 BPDU 保护功能，则边缘接口收到了 BPDU，设备将关闭这些接口，同时通知网管系统。被关闭的接口只能由网络管理人员通过命令手动恢复。

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**stp bpdu-protection** { **enable** | **disable** }	配置 BPDU 保护功能。

步骤	配置	说明
3	JX(config)#**interface** *interface-type interface-number*	进入物理接口配置模式或聚合组配置模式。以下步骤以物理接口配置模式为例。
4	JX(config-ge-1/0/*)#**stp bpdu-protection error-down recovery-interval** *interval*	配置 BPDU 保护的恢复周期

说明

当边缘接口使能了 BPDU 过滤功能，同时设备使能了 BPDU 保护功能，则 BPDU 保护功能优先生效，即边缘接口接收到 BPDU 报文时，接口被关闭。

9.5.14 配置 STP/RSTP/MSTP 模式切换

当生成树协议开启时，支持三种生成树运行模式，分别为 STP 兼容模式、RSTP 模式和 MSTP 模式。

- STP 兼容模式：不执行替换接口到根接口的快速转换和指定接口快速 Forwarding。只发送 STP 配置报文（STP Configuration BPDU）和拓扑变化通知（STP TCN BPDU）。收到 MST BPDU 将丢弃不识别部分。
- RSTP 模式：执行替换接口到根接口的快速转换和指定接口快速 Forwarding。只发送 RST BPDU。收到 MST BPDU 将丢弃不识别部分；如果本交换机接口的对端运行 STP 协议，接口将转移到 STP 兼容模式下。
- MSTP 模式：发送 MST BPDU。如果本交换机接口的对端运行 STP 协议，接口将转移到 STP 兼容模式下。

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**stp mode { stp | rstp | mstp }**	配置生成树的运行模式。
3	JX(config-ge-1/0/*)#**stp mcheck**	(可选)强制端口变为 MSTP 模式。

9.5.15 配置链路类型

点对点链路相连的两个接口可以通过传送同步报文快速迁移到转发状态，减少了不必要的转发延迟时间。缺省情况下，MSTP 根据双工状态设定接口的链路类型。全双工接口被认为是点到点链路，半双工被认作共享链路。

用户可以手工强行配置当前以太网接口与点对点链路相连，但是如果该链路不是点到点链路会使系统出现问题，一般情况下建议用户将此配置项设为自动状态，由系统自动发现接口是否与点到点链路相连。

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**interface** *interface-type interface-number*	进入物理接口配置模式或聚合组配置模式。以下步骤以物理接口配置模式为例。
3	JX(config-ge-1/0/*)#**stp point-to-point** { **force-true** | **force-false** | **auto** }	配置接口的链路类型。

9.5.16 配置根接口保护

当桥收到更高优先级的报文的时候就需要重新选举，重新选举一个是会影响网络的连通性，二来会消耗 CPU 资源。对于开启了 MSTP 功能的网络，如果有人发送高优先级的 BPDU 报文进行攻击，网络就会由于不断的选举而导致不稳定。而一般而言，各个桥的优先级是在网络规划阶段就已经配置好，越是靠近边缘的桥优先级越低，因此下行接口一般不会收到比桥优先级高的报文，除非有人恶意攻击。对于这些接口，可以通过开启根接口保护功能，拒绝处理比桥优先级高的报文，并在收到高优先级报文的时候阻塞接口一段时间，防止攻击源的其他攻击损害更上层的链路。

请在需要的设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**interface** *interface-type interface-number*	进入物理接口配置模式。
3	JX(config-ge-1/0/*)#**stp root-guard** { **enable** | **disable** }	配置接口的根接口保护功能。

9.5.17 配置接口环路保护

生成树主要作用有两个：防止环路和链路备份。防止环路就要求必须将拓扑裁剪成树状结构，而如果需要进行链路备份，拓扑中必须有冗余的链路。生成树就是通过阻塞冗余链路来达到防止环路的功能，而在链路发生故障的时候放开冗余链路从而达到链路备份的功能。

生成树模块会周期性交换报文，如果一定时间内没有收到报文即认为发生了链路故障。然后选举，放开备份接口。而在实际应用中，导致收不到报文的原因可能并不是链路故障，如果在这种情况下放开备份接口就有可能导致环路。

环路保护的目的是当接口在一定时间内收不到报文的时候，不进行重新选举，保持接口原来的状态不变。注意：环路保护的功能和链路备份的功能是对立的，也即环路保护是以失去链路备份功能的代价来实现环路避免。

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**interface** *interface-type interface-number*	进入物理接口配置模式或聚合组配置模式。以下步骤以物理接口配置模式为例。
3	JX(config-ge-1/0/*)#**stp loop-guard** { **enable** | **disable** }	配置接口环路保护功能。

9.5.18 配置端口 TC 报文抑制功能

用户接入网络的拓扑改变会引起核心网络的转发地址更新，当用户接入网络的拓扑因某种原因而不稳定时，就会对核心网络形成冲击。为了避免这种情况，可以在端口上使能 TC 报文抑制功能，此后当该端口收到 TC 报文时，不会再向其他端口传播。

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**interface** *interface-type interface-number*	进入物理接口配置模式或聚合组配置模式。以下步骤以物理接口配置模式为例。
3	JX(config-ge-1/0/*)#**stp tc-restriction** { **enable** | **disable** }	配置端口 TC 抑制功能。

9.5.19 配置 TC 保护功能

TC 保护功能可避免拓扑变化相关的 BPDU 报文攻击，提高设备和网络的安全性。

- 使能 TC 保护功能后，在 STP 的 Hello Time 时间内，仅接收阈值数量以内的 TC 报文。该 Hello Time 时间内的、超过阈值数量的 TC 报文会被丢弃，直到下一个 Hello Time 时间再重新开始计算接收的 TC 报文数量。
- 禁用 TC 保护功能后，设备会处理所有 TC 报文。当设备受到 TC 报文攻击时，可能会导致业务中断、设备 CPU 利用率过高甚至无法正常工作。

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**stp tc-protection threshold** *threshold-value*	配置 TC 保护功能的阈值。

步骤	配置	说明
3	JX(config)#**stp tc-protection { enable | disable }**	配置 TC 保护功能。

9.5.20 检查配置

配置完成后，请在设备上执行以下命令检查配置结果。

序号	检查项	说明
1	JX#**show stp infomation**	查看 STP 基本配置信息。
2	JX#**show stp interface**	查看接口下生成树配置信息。
3	JX#**show stp bridge**	查看 STP 根桥信息
4	JX#**show stp instance**	查看 STP 实例与 VLAN 映射关系。

9.5.21 维护

用户可以通过以下命令维护 MSTP 特性。

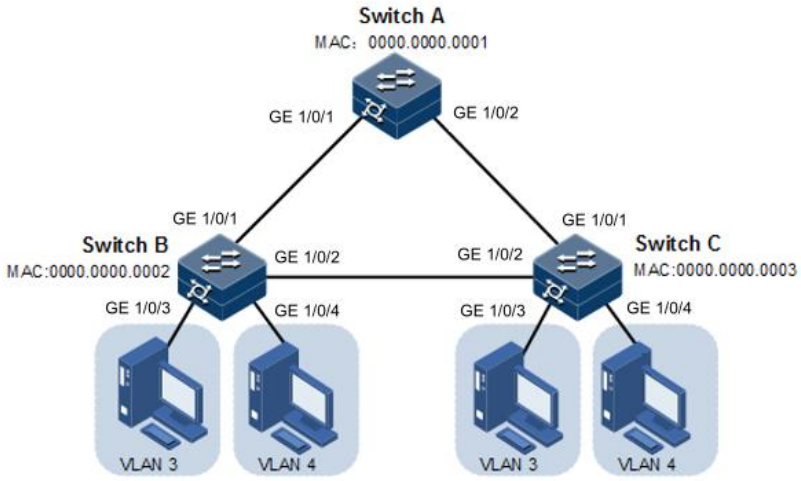
命令	描述
JX(config-ge-1/0/*)#**reset stp statistics**	清除接口生成树统计信息。

9.5.22 配置 MSTP 示例

组网需求

如下图所示，三台交换机设备组成环形网络运行 MSTP 协议，域名 aaa。Switch B 和 Switch C 上分别各连接两台 PC，分别属于 VLAN 3 和 VLAN 4。实例 3 关联 VLAN 3，实例 4 关联 VLAN 4。配置优先级，使得实例 3 的根桥为 Switch A，实例 4 的根桥为 SwicthB,这样可以使两个 VLAN 的报文分别在两条路径进行转发，消除了环路的同时实现了负载分担。

图 9-16 MSTP 应用组网示意图



配置步骤

步骤 1 在三台交换机上分别创建 VLAN 3 和 VLAN 4 并激活。

配置 Switch A。

```
JX#hostname SwitchA
SwitchA#config
SwitchA(config)#vlan 3,4
```

配置 Switch B。

```
JX#hostname SwitchB
SwitchB#config
SwitchB(config)#vlan 3,4
```

配置 Switch C。

```
JX#hostname SwitchC
SwitchC#config
SwitchC(config)#vlan 3,4
```

步骤 2 Switch A 的接口 GE 1/0/1、GE 1/0/2 以 Trunk 模式允许所有 VLAN 通过，Switch B 的接口 GE 1/0/1、GE 1/0/2 以 Trunk 模式允许所有 VLAN 通过，Switch C 的接口 GE 1/0/1、GE 1/0/2 以 Trunk 模式允许所有 VLAN 通过。Switch B、Switch C 的接口 GE 1/0/3、GE 1/0/4 以 Access 模式分别允许 VLAN 3、VLAN 4 通过。

配置 Switch A。

```
SwitchA(config)#interface ge 1/0/1
SwitchA(config-ge-1/0/1)#port link-type trunk
SwitchA(config-ge-1/0/1)#port trunk allow-pass vlan all
SwitchA(config-ge-1/0/1)#exit
SwitchA(config)#interface ge 1/0/2
SwitchA(config-ge-1/0/2)#port link-type trunk
SwitchA(config-ge-1/0/1)#port trunk allow-pass vlan all
SwitchA(config-ge-1/0/2)#exit
```

配置 Switch B。

```
SwitchB(config)#interface ge 1/0/1
SwitchB(config-ge-1/0/1)#port link-type trunk
SwitchB(config-ge-1/0/1)#port trunk allow-pass vlan all
SwitchB(config-ge-1/0/1)#exit
SwitchB(config)#interface ge 1/0/2
SwitchB(config-ge-1/0/2)#switchport mode trunk
SwitchB(config-ge-1/0/2)#port trunk allow-pass vlan all
SwitchB(config-ge-1/0/2)#exit
SwitchB(config)#interface ge 1/0/3
SwitchB(config-ge-1/0/3)#switchport access vlan 3
SwitchB(config-ge-1/0/3)#exit
SwitchB(config)#interface ge 1/0/4
SwitchB(config-ge-1/0/4)#switchport access vlan 4
SwitchB(config-ge-1/0/4)#exit
```

配置 Switch C。

```
SwitchC(config)#interface ge 1/0/1
SwitchC(config-ge-1/0/1)#port link-type trunk
SwitchC(config-ge-1/0/1)#port trunk allow-pass vlan all
SwitchC(config-ge-1/0/1)#exit
SwitchC(config)#interface ge t 1/0/2
SwitchC(config-ge-1/0/2)#switchport mode trunk
SwitchC(config-ge-1/0/2)#port trunk allow-pass vlan all
SwitchC(config-ge-1/0/2)#exit
SwitchC(config)#interface ge 1/0/3
SwitchC(config-ge-1/0/3)#switchport access vlan 3
SwitchC(config-ge-1/0/3)#exit
SwitchC(config)#interface ge 1/0/4
SwitchC(config-ge-1/0/4)#switchport access vlan 4
SwitchC(config-ge-1/0/4)#exit
```

步骤 3 Switch A、Switch B、Switch C 设置生成树模式为 MSTP，并开启生成树协议。进入 MSTP 配置模式设置域名为 aaa，修正版本为 0，instance 3 映射 VLAN 3、instance 4 映射 VLAN 4，退出 mst 配置模式。

配置 Switch A。

```
SwitchA(config)#stp mode mstp
SwitchA(config)#stp config-name aaa
SwitchA(config)#stp revision-level 0
SwitchA(config)#stp instance 3 vlan 3
SwitchA(config)#stp instance 4 vlan 4
SwitchA(config)#stp instance 3 priority 0
SwitchA(config)#interface ge 1/0/1
SwitchA(config-ge-1/0/1)#stp enable
SwitchA(config-ge-1/0/1)#exit
SwitchA(config)#interface ge 1/0/2
SwitchA(config-ge-1/0/2)#stp enable
```

配置 Switch B。

```
SwitchB(config)#stp mode mstp
SwitchB(config)#stp config-name aaa
```

```
SwitchB(config)#stp revision-level 0
SwitchB(config)#stp instance 3 vlan 3
SwitchB(config)#stp instance 4 vlan 4
SwitchA(config)#stp instance 4 priority 0
SwitchB(config)#interface ge 1/0/1
SwitchB(config-ge-1/0/1)#stp enable
SwitchB(config-ge-1/0/1)#exit
SwitchB(config)#interface ge 1/0/2
SwitchB(config-ge-1/0/2)#stp enable
```

配置 Switch C。

```
SwitchC(config)#stp mode mstp
SwitchC(config)#stp config-name aaa
SwitchC(config)#stp revision-level 0
SwitchC(config)#stp instance 3 vlan 3
SwitchC(config)#stp instance 4 vlan 4
SwitchC(config)#interface ge 1/0/1
SwitchC(config-ge-1/0/1)#stp enable
SwitchC(config-ge-1/0/1)#exit
SwitchC(config)#interface ge 1/0/2
SwitchC(config-ge-1/0/2)#stp enable
```

步骤 4 Switch B 修改生成树实例 3 接口 GE 1/0/1 的路径开销为 500000。

```
SwitchB(config)#interface ge 1/0/1
SwitchB(config-ge-1/0/1)#stp instance 3 path-cost 500000
```

检查结果

通过 **show stp interface** 命令查看 MST 域的接口状态信息，以 Switch C 为例。

```
SwitchA#show stp interface
  MSTID Port         Role         State        Protection    Region
-------------------------------------------------------------------------------
------------
  0    ge-1/0/1      root         forward      --           same
  0    ge-1/0/2      alternate     discarding    --           same
  3    ge-1/0/1      root         forward      --           same
  3    ge-1/0/2      alternate     discarding    --           same
  4    ge-1/0/1      alternate     discarding    --           same
  4    ge-1/0/2      root         forward      --           same
-------------------------------------------------------------------------------
------------------------------
```

9.6 环路检测

9.6.1 简介

环路检测功能可以消除因环路对网络造成的影响，提高网络的自检错性、容错性和健壮性。

环路检测过程中每个开启环路检测功能的接口周期性地发送环路检测报文（Hello 报文），由于环路检测功能主要应用于网络边缘接口，因此，正常情况下边缘接口是不

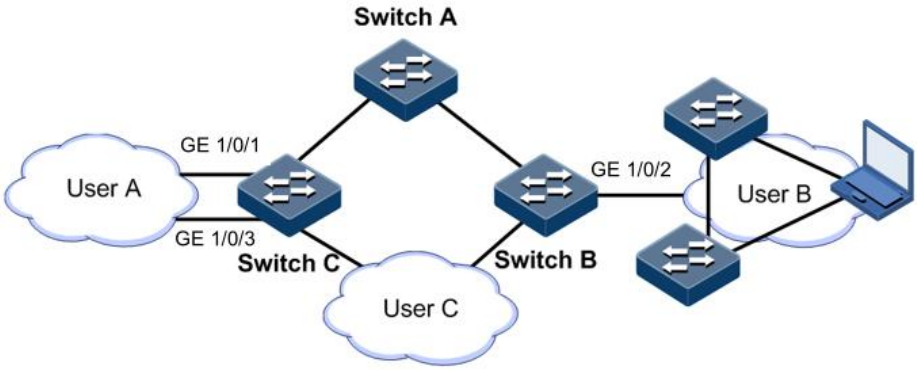
应该收到任何环路检测报文的。如果边缘设备接收到了环路检测报文，则认为网络出现了环路。接收环路检测报文分为两种情况：接收到了自身发送的报文和接收到了其他设备发送的报文，主要依据设备 MAC 地址和报文携带的 MAC 地址进行比较区分。

环路类型

常见的环路类型有自环、内环两种。如下图所示，Switch B 和 Switch C 是连接用户网络的边缘交换机。

- 自环：同一设备上同一以太网接口下存在的用户环路，例如用户网络 B 自身存在环路使 Switch B 的 GE 1/0/2 接口形成自环。
- 内环：同一设备上不同以太网接口之间形成的环路，例如 Switch C 的接口 GE 1/0/1 和接口 GE 1/0/3 与用户网络 A 形成内环。

图 9-17 环路类型示意图



环路处理机制

设备遵循如下原则进行环路处理：

- 如果接收和发送环路检测报文的设备是同一设备，但不是同一接口，则处理接口号小的接口消除环路（内环）。
- 如果接收和发送环路检测报文的设备是同一设备且是同一接口，则处理该接口消除环路（自环）。

在下图中，假设 Switch B 和 Switch C 连接用户网络的接口都使能了环路检测功能。环路检测针对不同环路类型的处理机制如下：

- 对于自环，Switch B 收发报文接口号相同，将在 GE 1/0/2 接口采取配置的环路检测动作来消除自环。
- 对于内环，Switch C 会收到自身发出的环路检测报文，而且收发报文接口号不同，因此会在接口号小的 GE 1/0/1 采取配置的环路检测动作来消除内环。

环路处理动作

环路处理动作即设备检测到接口出现环路时的处理方式，用户可以根据实际情况在指定接口上配置不同的环路处理动作。包括以下几种：

- Block：阻塞指定接口并发送 Trap 信息。
- Trap-only：只发送 Trap 信息。
- Shutdown：关闭指定接口并发送 Trap 信息。
- Shutdown-restore：发送 trap，关闭端口并等待恢复

环路检测模式

环路检测模式为 Port 模式。

Port 模式：出现环路时，若环路处理方式为 Block，则阻塞接口并发送 Trap；若环路处理方式为 Shutdown 方式，则关闭物理接口并发送 Trap。

若环路处理方式为 Trap-only，则只发送 Trap 信息。

环路恢复

接口由于环路被关闭之后，可以根据用户配置于指定时间后自动恢复。

9.6.2 配置准备

场景

在网络中，所有接入设备下连的主机或二层设备都可能存在有意、无意的连接而引起的环路，在每台接入设备上开启环路检测功能，可以避免网络环路造成数据流无限制复制而形成的网络拥塞状况。

前提

设备上环路检测、接口备份、STP、G.8032 功能之间可能会相互影响，建议不要同一接口上同时开启这些功能。

9.6.3 环路检测的缺省配置

设备上环路检测的缺省配置如下。

功能	缺省值
环路检测功能状态	禁止
接口阻塞后的自动恢复时间	30s
环路检测处理方式	block
环路检测报文发送周期	5s
环路检测模式	vlan 模式

9.6.4 配置环路检测功能

请在设备上进行以下配置。

说明

环路检测与生成树协议冲突，二者不能同时开启。

直连设备两端不能同时开启环路检测。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**interface** *interface-type interface-number*	进入二层物理接口配置模式。
3	JX(config-ge-1/0/*)#**loopback-detect mode vlan**	配置基于 VLAN 的环路检测功能。
4	JX(config-ge-1/0/*)#**loopback-detect vlan** *vlan-list*	配置发包 vlan
5	JX(config-ge-1/0/*)#**loopback-detect action block**	配置环路动作
6	JX(config)#**loopback-detect interval** *interval*	配置发包间隔，单位：秒
7	JX(config)#**loopback-detect recovery-interval** *interval*	配置恢复时间，单位：秒

9.6.5 检查配置

配置完成后，请在设备上执行以下命令检查配置结果。

序号	检查项	说明
1	JX#**show loopback-detect interface**	查看接口环路检测配置。

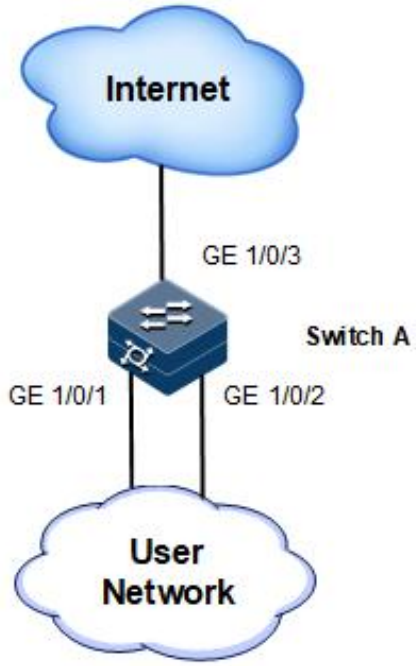
9.6.6 配置环路检测内环应用示例

组网需求

如下图所示，Switch A 通过 GE 1/0/1 和 GE 1/0/2 接口连接 VLAN 3 用户网络，为了避免用户网络中存在环路，需要在 Switch A 上开启环路检测功能，及时检测用户网络中的环路并进行处理。具体要求如下：

- 使能 GE 1/0/1 和 GE 1/0/2 接口的环路检测功能。
- 配置环路检测报文发送间隔为 3s。
- 配置环路检测模 VLAN 为 VLAN3。
- 配置 GE 1/0/2 接口的环路检测处理动作为 Block，发送告警信息并阻塞接口。

图 9-18 环路检测内环应用组网示意图



配置步骤

步骤 1 创建 VLAN，并将接口加入 VLAN。

```
JX#configure
JX(config)#vlan 3
JX(config)#interface ge 1/0/1
JX(config-ge-1/0/1)#port link-type hybrid
JX(config-ge-1/0/1)#port hybrid vlan 3 untagged
JX(config-ge-1/0/1)#exit
JX(config)#interface ge 1/0/2
JX(config-ge-1/0/1)#port link-type hybrid
JX(config-ge-1/0/2)#port hybrid vlan 3 untagged
JX(config-ge-1/0/2)#exit
```

步骤 2 配置环路检测 VLAN、环路检测处理动作和环路检测报文发送周期。

```
JX(config)#interface ge 1/0/1
JX(config-ge-1/0/1)#loopback-detect enable
JX(config-ge-1/0/1)#loopback-detect action block
JX(config-ge-1/0/1)#loopback-detect vlan 3
JX(config-ge-1/0/1)#loopback-detect interval 3
JX(config-ge-1/0/1)#exit
JX(config)#interface ge 1/0/2
```

```
JX(config-ge-1/0/2)#loopback-detect enable
JX(config-ge-1/0/2)#loopback-detect action block
JX(config-ge-1/0/2)#loopback-detect vlan 3
JX(config-ge-1/0/2)#loopback-detect interval 3
```

检查结果

通过 **show loopback-detect interface** 命令查看接口环路检测状态，此时没有环路。

```
JX#show loopback-detect interface
 Interface     Enable  Action        Loop-Status
------------------------------------------------------------
 ge-1/0/1      enable  block         none-loop
 ge-1/0/2      enable  block         none-loop
------------------------------------------------------------
```

9.7 接口备份

9.7.1 简介

双上行组网是目前常用的应用组网之一，该组网下常通过生成树协议（STP，Spanning Tree Protocol）阻塞冗余链路，起备份作用。虽然这种方案从功能上可以实现客户冗余备份的需求，但是在性能上却不能达到很多用户的要求，即使采用快速生成树协议的快速迁移，也只能是秒级的收敛速度。这对于应用于电信级网络核心的高端以太网交换机，是非常不利的一个性能参数。

接口备份解决方案针对双上行组网，实现主备链路冗余备份及快速迁移。该方案为双上行组网量身定做，既保证了性能，又简化了配置。

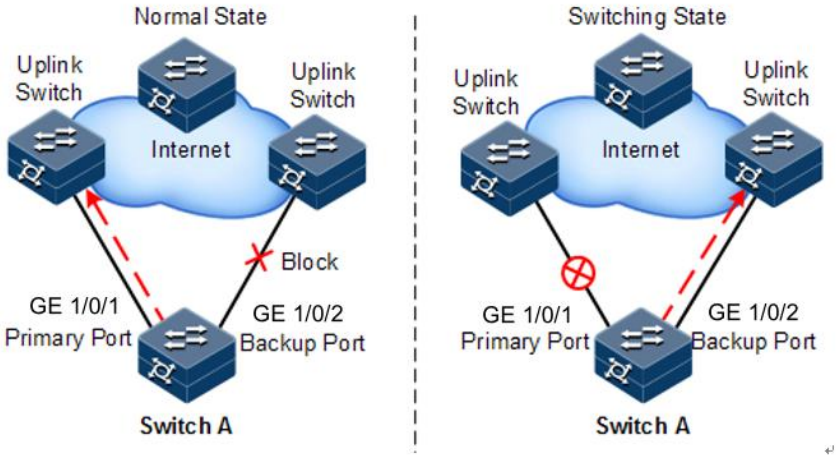
接口备份功能是 STP 协议的另一个解决方案，用户可以在关闭 STP 功能的情况下，通过手动设置接口实现基本的链路冗余。如果交换机已经开启 STP，就需要禁止接口备份功能，因为 STP 已经提供了类似的功能。

接口备份原理

接口备份功能需要设置接口备份组来实现。接口备份组包括一对接口，其中一个接口是主接口，另一个接口是备份接口。主接口所在的链路称为主链路，备份接口所在的链路称为备份链路。接口备份组的成员接口支持物理接口和链路聚合组，不支持三层接口。

在接口备份组中，当一个接口处于转发（Forward）状态时，另一个接口则处于阻塞状态（Block）。任何时刻，两个接口中只有一个接口处于转发状态。当处于转发状态的接口发生链路故障时，处于待命状态的接口才会切换到转发状态，以保持链路正常。

图 9-19 接口备份原理示意图



接口备份原理如上图所示。Switch A 上的 GE 1/0/1、GE 1/0/2 分别与上行交换机相连，接口转发状态如下：

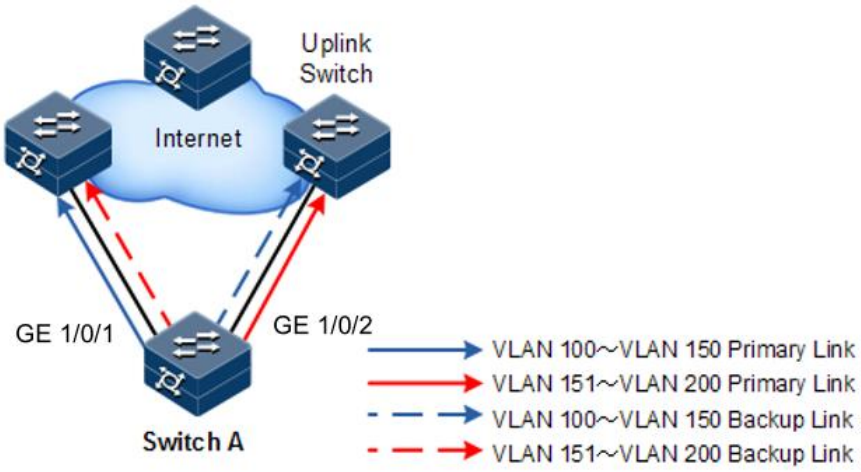
- 正常状态下，Switch A 的 GE 1/0/1 为主接口，GE 1/0/2 为备份接口，GE 1/0/1 和上行交换机之间转发报文，GE 1/0/2 和上行交换机之间不转发报文。
- 当 GE 1/0/1 与上行交换机之间出现链路故障时，备份接口 GE 1/0/2 和上行交换机之间转发报文。
- 当 GE 1/0/1 链路故障恢复并保持一段时间（恢复延时）后，GE 1/0/1 变为转发状态，GE 1/0/2 变为待命状态。

如果主接口和备份接口之间发生切换，交换机会发送一个 Trap 上报网管系统。

接口备份在不同 VLAN 上的应用

接口备份通过在 VLAN 上进行应用，还可以实现两个接口在不同 VLAN 上同时进行转发，如下图所示。

图 9-20 接口备份在不同 VLAN 上的应用原理示意图



在不同的 VLAN 上，接口的转发状态如下：

- 正常情况下，配置 Switch A 在 VLAN 100～VLAN 150 上，GE 1/0/1 为主接口，GE 1/0/2 为备份接口；在 VLAN 151～VLAN 200 上，GE 1/0/2 为主接口，GE 1/0/1 为备份接口。那么，GE 1/0/1 在 VLAN 100～VLAN 150 转发流量，GE 1/0/2 在 VLAN 151～VLAN 200 上转发流量。
- 当 GE 1/0/1 发生链路故障时，GE 1/0/2 负责转发 VLAN 100～VLAN 200 上的流量。
- 当 GE 1/0/1 恢复正常并保持一段时间（恢复延时）后，GE 1/0/1 在 VLAN 100～VLAN 150 上转发流量，GE 1/0/2 在 VLAN 151～VLAN 200 上转发流量。

利用这种方法，接口备份可以用于负载均衡。同时，这种应用不依赖于上联交换机的配置，便于用户操作。

9.7.2 配置准备

场景

在双上行网络中，通过配置端口备份功能，可以实现主备链路的冗余备份及其快速倒换。通过在不同 VLAN 中应用端口备份，还可以实现各端口之间的负载分担。

与 STP 功能相比，端口备份功能既保证了毫秒级的倒换性能，又简化了配置。

前提

无

9.7.3 接口备份的缺省配置

设备上接口备份的缺省配置如下。

功能	缺省值
接口备份组	无
故障恢复延时时间	15s
恢复模式	不返回模式

9.7.4 配置接口备份基本功能

注意

设备上接口备份与 STP、环路检测和 ERPS 功能之间可能会相互影响，建议不要在同一接口上同时开启这些功能。

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**configure**	进入全局配置模式。
2	JX(config)#**protect-link group** *group-id*	创建备份组
3	JX(config-protectlink-*)#**protect-vlan** *vlan-list*	配置指定的 VLAN 列表
4	JX(config-protectlink-*)#**add interface** *interface-type interface-number* **role master**	指定主接口
5	JX(config-protectlink-*)#**add interface** *interface-type interface-number* **role slave**	指定备份接口
6	JX(config-protectlink-*)#**port backup fault-detect lldp**	配置 LLDP 故障探测。
7	JX(config-protectlink-*)#**reverse** { **enable** | **disable** }	配置恢复模式。
8	JX(config-protectlink-*)#**reverse time** *interval*	配置恢复时间

说明

- 在一个接口备份组中，一个接口不能既是主接口，又是备份接口。
- 在同一 VLAN 上，一个接口/链路聚合组不能同时充当两个接口备份组的成员。

9.7.5 配置接口强制倒换

注意

- 配置强制倒换成功后，主备链路将进行倒换，工作链路被强制倒换到备份链路上。

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**configure**	进入全局配置模式。
2	JX(config)#**protect-link group** *group-id*	进入物理接口配置模式或聚合组配置模式。以下步骤以物理接口配置模式为例。
3	JX(config-protectlink-*)#**manual-switch**	配置接口强制倒换。

9.7.6 检查配置

配置完成后，请在设备上执行以下命令检查配置结果。

序号	检查项	说明
1	JX#**show protect-link interface**	查看接口备份相关的状态信息。
2	JX#**show protect-link group**	查看接口备份组相关的配置信息。

9.7.7 配置接口备份示例

组网需求

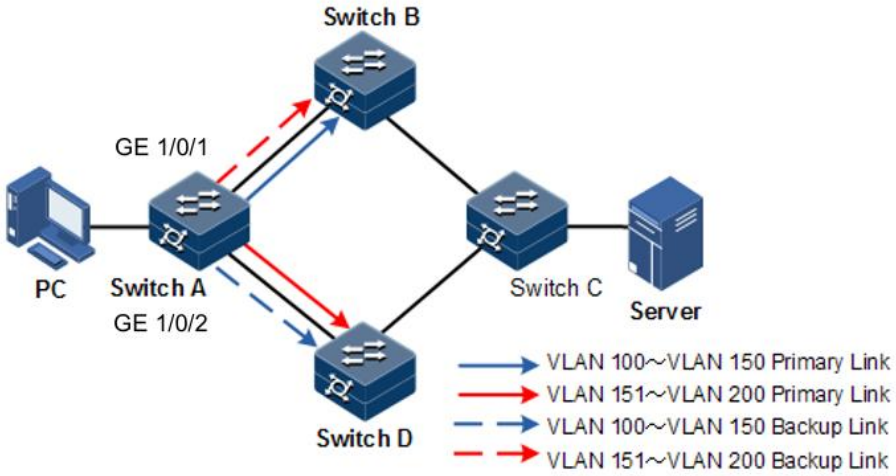
如下图所示，为实现远程 PC 到服务器的可靠访问，需要在 Switch A 上配置接口备份组，并指定 VLAN 列表，实现接口链路保护和负载分担。其要求如下：

- 配置 Switch A 在 VLAN 100～VLAN 150 上，GE 1/0/1 为主接口，GE 1/0/2 为备份接口；
- 配置 Switch A 在 VLAN 151～VLAN 200 上，GE 1/0/2 为主接口，GE 1/0/1 为备份接口。

当 GE 1/0/1 发生链路故障时，切换到备份接口 GE 1/0/2，以保持链路正常。

Switch A 需要支持接口备份功能，Switch B、Switch C、Switch D 无需支持接口备份功能。

图 9-21 接口备份应用组网示意图



配置步骤

步骤 1 创建 VLAN 100～VLAN 200，并将 GE 1/0/1 和 GE 1/0/2 加入到 VLAN 100～VLAN 200 中。

```
JX#configure
JX(config)#vlan 100-200
JX(config)#interface ge 1/0/1
JX(config-ge-1/0/1)#port link-type trunk
JX(config-ge-1/0/1)#port trunk allow-pass vlan 100-200
JX(config-ge-1/0/1)#exit
JX(config)#interface ge 1/0/2
JX(config-ge-1/0/2)#port link-type trunk
JX(config-ge-1/0/2)#port trunk allow-pass vlan 100-200
JX(config-ge-1/0/2)#exit
```

步骤 2 在 VLAN 100～VLAN 150 上配置 GE 1/0/1 为主接口，GE 1/0/2 为备份接口。

```
JX(config)#protect-link group 1
JX(config-protectlink-1)#protect-vlan 100-150
JX(config-protectlink-1)#add interface ge 1/0/1 role master
JX(config-protectlink-1)#add interface ge 1/0/2 role slave
```

步骤 3 在 VLAN 151～VLAN 200 上配置 GE 1/0/2 为主接口，GE 1/0/1 为备份接口。

```
JX(config)#protect-link group 2
JX(config-protectlink-1)#protect-vlan 151-200
JX(config-protectlink-1)#add interface ge 1/0/2 role master
```

```
JX(config-protectlink-1)#add interface ge 1/0/1 role slave
```

检查结果

通过 **show protect-link interface** 命令，分别在正常状态和链路故障情况下查看接口备份相关的状态信息。

当 GE 1/0/1 和 GE 1/0/2 链路均为 Forward 时，GE 1/0/1 在 VLAN 100～VLAN 150 上转发流量，GE 1/0/2 在 VLAN 151～VLAN 200 上转发流量。

```
JX#show protect-link interface
 Interface         Group  Role    State     Status    Linkstate
--------------------------------------------------------------------------
-------
 ge-1/0/1          1      master  forward   active    up/up
 ge-1/0/2          1      slave   block     active    up/up
 ge-1/0/1          2      slave   block     active    up/up
 ge-1/0/2          2      master  forward   active    up/up
--------------------------------------------------------------------------
-------
```

手动断开 Switch A 和 Switch B 之间的链路来模拟故障，此时 GE 1/0/1 变为 Down，则 GE 1/0/2 负责转发 VLAN 100～VLAN 200 上的流量。

```
JX#show protect-link interface
  Interface         Group  Role    State     Status    Linkstate
--------------------------------------------------------------------------
-------
 ge-1/0/1          1      master  block     active    up/down
 ge-1/0/2          1      slave   forward   active    up/up
 ge-1/0/1          2      slave   block     active    up/down
 ge-1/0/2          2      master  forward   active    up/up
--------------------------------------------------------------------------
-------
```

9.8 接口隔离

9.8.1 简介

通过接口隔离特性，用户可以将需要进行控制的接口使能隔离特性，实现接口之间二层数据的隔离，达到类似于接口之间物理隔离效果，既增强了网络的安全性，也为用户提供了灵活的组网方案。

配置接口保护后，在使能接口保护的接口之间报文不能互通，使能接口保护的接口和未使能接口保护的接口之间的通信不会受影响。

9.8.2 配置准备

场景

通过配置接口隔离功能，可以实现接口之间互相隔离，能增强网络的安全性，也为用户提供了灵活的组网方案。

前提

无

9.8.3 接口隔离的缺省配置

设备上接口隔离的缺省配置如下。

功能	缺省值
各接口的接口隔离功能状态	禁止

9.8.4 配置接口隔离

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**configure**	进入全局配置模式。
2	JX(config)#**port-isolate group** *group-id*	创建隔离组。
3	JX(config-isolate-group-1)#**add interface** *interface-type interface-number*	将接口加入隔离组

9.8.5 检查配置

配置完成后，请在设备上执行以下命令检查配置结果。

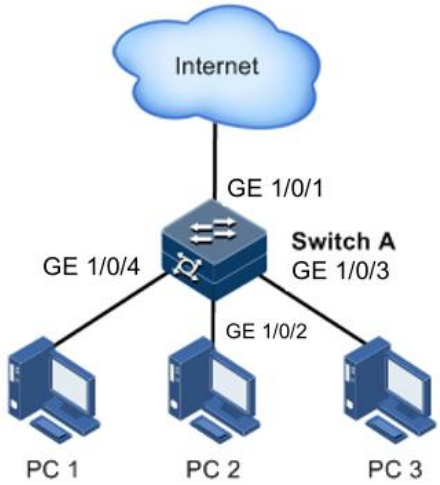
序号	检查项	说明
1	JX#**show port-isolate information**	查看接口保护配置。
2	JX#**show port-isolate group**	查看接口隔离配置信息。

9.8.6 配置接口保护示例

组网需求

如下图所示，为了使 PC1 和 PC2 之间不能互通，但 PC 1 和 PC 2 可以分别与 PC 3 通信，可以在 Switch A 的 GE 1/0/4 和 GE 1/0/2 接口上开启接口保护功能。

图 9-22 接口保护组网示意图



配置步骤

步骤 1 创建接口隔离组。

```
JX#configure
JX(config)#port-isolate group 1
```

步骤 2 将 GE 1/0/1 和 GE 1/0/2 加入隔离组。

```
JX(config-isolate-group-1)##add interface ge 1/0/4
JX(config-isolate-group-1)##add interface ge 1/0/2
```

检查结果

通过 **show port-isolate group** 查看接口保护配置是否正确。

```
JX#show port-isolate group
  GroupId       Ports
-------------------------------------------------------------------------------
-------
  1           ge-1/0/2 ge-1/0/4
-------------------------------------------------------------------------------
-------
```

通过 PC 1 ping PC 3、PC 2 ping PC 3 是否能够 ping 通。

- PC 1 ping PC 3，可以 ping 通。
- PC 2 ping PC 3，可以 ping 通。

通过 PC 1 ping PC 2 是否能够 ping 通，查看接口保护功能是否正确。

PC 1 ping PC 2，不能 ping 通，接口保护功能生效。

9.9 L2CP

9.9.1 简介

MEF（Metro Ethernet Forum，城域以太网论坛）引入了服务的概念，如 EPL、EVPL、EP-LAN 和 EVP-LAN 等六种服务类型，每种服务对 L2CP（Layer 2 Control Protocol，二层控制协议）报文的处理方式不同。

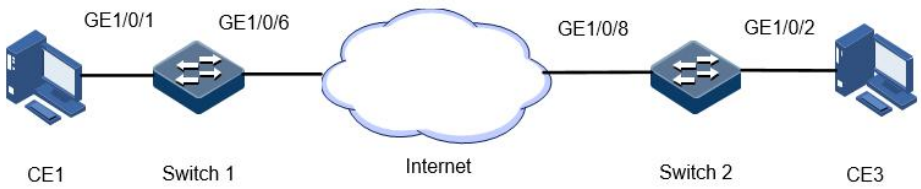
在 MEF6.1 中定义了 L2CP 报文的处理方式，分别为：

- discard：丢弃报文。
- peer：上交到 cpu。
- tunnel：传递到城域网，处理方式配置相比 discard 和 peer 较为复杂。需通过将用户网络侧接口的匹配规则和运营商侧接口 tunnel 终端的配合使用，才能使报文穿过运营商网络。

9.9.2 配置准备

场景

图 9-23 L2CP 拓扑图



switch1,switch2 交换机作为运营商网络接入设备，CE1，CE3 作为用户网络接入设备，CE1，CE3 分别与 switch1 的 ge 1/0/1,switch2 的 ge 1/0/2 相连。

通过配置不同用户网络的 L2PROTOCOL 报文透明传输，使得：

用户网络中的设备可以共同完成生成树功能

前提

无

9.9.3 缺省配置

功能	缺省值
L2CP 功能	未使能

9.9.4 配置 L2CP

步骤	配置	说明
1	JX#**configure**	进入全局配置模式。
2	JX(config)#**interface interface-type** *interface-number*	进入接口配置模式。
3	JX(config-ge-1/0/1)#**l2cp { uni | nni | disable }**	使能 l2cp 功能 uni：用户侧接口 nni：网络侧接口 disable：去使能端口角色 缺省为没有角色
4	JX(config-ge-1/0/1)#**l2cp known-protocol {cdp|eoam3ah|gmrp |gvrp|hgmp|lacp|lldp|pagp|stp |udld|vtp|lamp|esmc |dot1x |elmi|pvst} [vlan** *vlan-id* **] action tunnel group-mac** *mac-address*	在 uni 接口上配置替换规则 known-protocol：匹配的协议名称 mac-address：替换的组播 mac 地址 vlan-id：匹配 vlan id
5	JX(config-ge-1/0/1)#**l2cp protocol-mac** *dst-mac-address* **[vlan** *vlan-id* **] action tunnel group-mac** *mac-address*	在 uni 接口上配置替换规则 dst-mac-address：匹配的目的 mac 地址 mac-address：替换的组播 mac 地址 vlan-id：匹配 vlan id

步骤	配置	说明
6	JX(config-ge-1/0/1)#**l2cp known-protocol {cdp|eoam3ah|gmrp |gvrp|hgmp|lacp|lldp|pagp|stp |udld|vtp|lamp|esmc |dot1x |elmi|pvst} action { peer | discard }**	在 uni 接口上配置上送或丢弃规则 known-protocol：匹配的协议名称 peer：上送 CPU discard：丢弃
7	JX(config-ge-1/0/1)#**l2cp protocol-mac** *dst-mac-address* **action { peer | discard }**	在 uni 接口上配置上送或丢弃规则 dst-mac-address：匹配的目的 mac 地址 peer：上送 CPU discard：丢弃

9.9.5 检查配置

步骤	配置	说明
1	JX#**show l2cp**	显示 l2cp 全局信息。
2	JX#**show l2cp config**	显示 l2cp 配置信息。
3	JX#**show protocol statistics rx with-value**	显示 l2cp 上送 CPU 的报文统计

9.9.6 配置 L2CP 示例

配置步骤

步骤 1 配置 CE-1：

```
JX(config)#stp enable
```

步骤 2 配置 CE-3：

```
JX(config)#stp enable
```

步骤 3 配置 switch1：

```
JX(config-ge-1/0/1)#l2cp uni
JX(config-ge-1/0/1)#l2cp known-protocol stp action tunnel group-mac
01:00:0c:cd:cd:d0
JX(config-ge-1/0/6)#l2cp nni
```

步骤 4 配置 switch2：

```
JX(config-ge-1/0/2)#l2cp uni
JX(config-ge-1/0/2)#l2cp known-protocol stp action tunnel group-mac
01:00:0c:cd:cd:d0
JX(config-ge-1/0/8)#l2cp nni
```

检查配置

CE1 和 CE3 上分别可以看到对方的生成树信息

9.10 BFD

9.10.1 简介

双向转发检测 BFD（Bidirectional Forwarding Detection）是一种全网统一的检测机制，用于快速检测、监控网络中链路或者 IP 路由的转发连通状况。

9.10.2 配置准备

场景

BFD 在两台网络设备上建立会话，用来检测网络设备间的双向转发路径，为上层应用服务。BFD 本身并没有邻居发现机制，而是靠被服务的上层应用通知其邻居信息以建立会话。会话建立后会周期性地快速发送 BFD 报文，如果在检测时间内没有收到 BFD 报文则认为该双向转发路径发生了故障，通知被服务的上层应用进行相应的处理。

前提

无

9.10.3 BFD 的缺省配置

设备上接口隔离的缺省配置如下。

功能	缺省值
全局 BFD 功能	未使能
发送间隔	1000 毫秒
接收间隔	1000 毫秒
本地检测倍数	3

9.10.4 配置 BFD 静态会话检测

请在设备上进行以下配置。

步骤	配置	说明
1	`JX#config`	进入全局配置模式。

步骤	配置	说明
2	JX(config)#**bfd start**	全局使能 bfd。
3	JX(config)#bfd track 1 remote-ip 10.0.0.2 local-ip 10.0.0.1 interface vlan 1	配置 track，产生 bfd 静态会话。
4	JX(config)#bfd track 1 remote-ip6 2001::2 local-ip6 2001::1 interface vlan 1	配置 track，产生 ipv6 bfd 静态会话。

9.10.5 配置 BFD 单臂回声会话

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**bfd start**	全局使能 bfd。
3	JX(config)#bfd track 1 remote-ip 10.0.0.2 local-ip 10.0.0.1 interface vlan 1 one-arm-echo	配置单臂 ECHO 功能的 BFD 会话。
4	JX(config)#bfd track 1 remote-ip6 2001::2 local-ip6 2001::1 interface vlan 1 one-arm-echo	配置单臂 ECHO 功能的 BFD 会话。

9.10.6 配置 BFD 检测三层 eth-trunk

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**bfd start**	全局使能 bfd。
3	JX(config)#bfd track 1 link-bundle remote-ip 10.0.0.2 local-ip 10.0.0.1 interface eth-trunk 1	配置 track 绑定聚合口，产生主会话和子会话。
4	JX(config)#bfd track 1 link-bundle remote-ip6 2001::2 local-ip 2001::1 interface eth-trunk 1	配置 track 绑定聚合口，产生主会话和子会话。

9.10.7 配置 BFD 会话参数

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**bfd start**	全局使能 bfd。

步骤	配置	说明
3	JX(config)#bfd track 1 remote-ip 10.0.0.2 local-ip 10.0.0.1	创建会话。
4	JX(config)#bfd track 1 min-tx 1000 min-rx 1000 multiplier 3	配置会话参数。

9.10.8 检查配置

配置完成后，请在设备上执行以下命令检查配置结果。

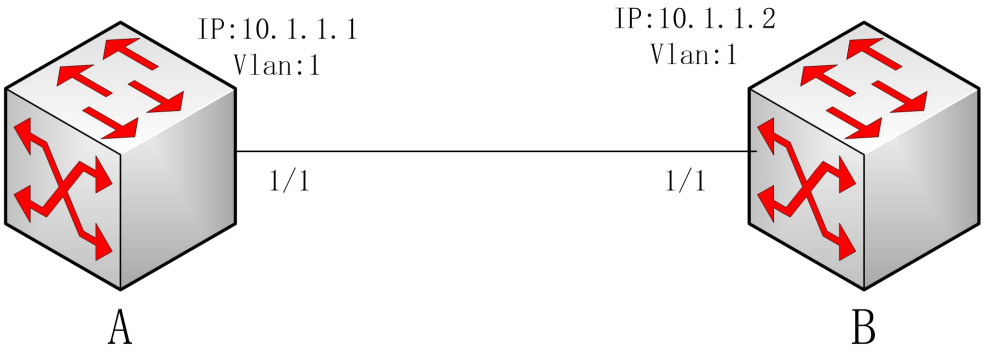
序号	检查项	说明
1	JX#**show bfd session**	查看 bfd 会话。
2	JX#**show bfd track**	查看 bfd track。
3	JX#**show bfd link-bundle session**	查看 bfd link-bundle 会话。

9.10.9 配置 BFD 单跳应用示例

组网需求

如下图所示，配置 BFD 单跳会话检测设备 A、B 之间的链路状态。

图 9-24 BFD 单跳应用



配置步骤

步骤 1 配置 A 设备 IP 和会话参数。

```
JX#config
JX(config)#interface vlan 1
JX(config)#ip address 10.1.1.1/24
```

```
JX(config)#bfd track 1 remote-ip 10.1.1.2 local-ip 10.1.1.1 interface
vlan 1
JX(config)#bfd track 1 min-tx 500 min-rx 600 multiplier 3
```

步骤 2 配 A 设备 IP 和会话参数。

```
JX#config
JX(config)#interface vlan 1
JX(config)#ip address 10.1.1.1/24
JX(config)#bfd track 1 remote-ip 10.1.1.1 local-ip 10.1.1.2 interface
vlan 1
JX(config)#bfd track 1 min-tx 500 min-rx 600 multiplier 4
```

检查结果

通过 **show bfd session** 查看会话状态

```
JX>show bfd session
  Interface      State Local-Discr Remote-Discr local-addr         remote-addr
-------------------------------------------------------------------------------------------
  vlan-1         up    1           1            10.1.1.1           10.1.1.2
-------------------------------------------------------------------------------------------
-----------------
```

9.11 链路震荡保护

9.11.1 简介

链路震荡保护（Link-flap Protection）是一种将物理状态频繁 Up/Down 变化的接口关闭，使之处于 Down 状态，防止网络拓扑结构频繁变化的技术。

9.11.2 配置准备

场景

网络抖动或者链路线路故障等原因会引起本端设备接口物理状态频繁 Up/Down 变化，导致链路震荡，致使网络拓扑结构频繁变化，影响用户通信。为了解决上述问题，用户可以配置链路震荡保护，将物理状态频繁 Up/Down 的接口关闭，使之处于 Down 状态，使网络拓扑结构停止频繁变化。

链路震荡次数：接口状态 Up/Down 切换一次，记为一次震荡。

链路震荡检测时间间隔：系统需要统计指定时间间隔内链路震荡的次数。

如果在链路震荡检测时间间隔内，链路震荡次数达到了门限值，则将该端口关闭。

前提

无

9.11.3 链路震荡保护的缺省配置

设备上接口隔离的缺省配置如下。

功能	缺省值
使能链路震荡保护	未使能
链路检测检测时间间隔	10 秒
链路震荡次数	5 次

9.11.4 配置链路震荡保护

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**configure**	进入全局配置模式。
2	JX(config)#**interface** *interface-type interface-number*	进入物理接口配置模式。
3	JX(config-ge-1/0/*)#**port link-flap protection** { **enable** | **disable** }	配置链路震荡保护使能或去使能。
4	JX(config)#**port link-flap interval** { *60* | **default**}	配置链路震荡检测时间间隔。
5	JX(config)#**port link-flap threshold** { *7* | **default**}	配置链路震荡次数门限值。

9.11.5 检查配置

配置完成后，请在设备上执行以下命令检查配置结果。

序号	检查项	说明
1	JX#**show link-flap config**	查看链路震荡保护相关配置。
2	JX#**show link-flap interface**	查看链路震荡保护相关接口信息。

9.11.6 配置链路震荡保护应用示例

配置步骤

步骤 1 配置 A 设备 IP 和会话参数。

```
JX#configure
JX(config)#interface ge 1/0/1
JX(config-ge-1/0/1)#port link-flap protection enable
JX(config-ge-1/0/1)#port link-flap interval 60
JX(config-ge-1/0/1)#port link-flap threshold 7
```

检查结果

通过 **show link-flap interface** 查看链路震荡保护接口状态

```
JX#show link-flap interface
 Error-down recovery interval   : 0
 Interface          Status     Interval    Threshold
--------------------------------------------------------------------------
 ge-1/0/1           enable     60          7
--------------------------------------------------------------------------
```

10 系统管理

本章介绍系统管理与维护特性的基本原理和配置过程，并提供相关的配置案例。

- SNMP
- RMON
- LLDP
- 端口镜像
- 电缆诊断
- UDLD
- 光模块数字诊断
- 系统日志
- 告警管理
- CPU 监控
- 内存监控
- Ping
- Trace
- 硬件监控
- 风扇监控
- 设备堆叠
- MAD
- NQA
- PATCH 补丁功能
- 定时备份配置功能

10.1 SNMP

10.1.1 简介

SNMP（Simple Network Management Protocol，简单网络管理协议）是由 IETF（Internet Engineering Task Force，互联网工程任务组）为了解决 Internet 中网络设备的

管理问题而提出的一套网络管理协议。SNMP 可以使一个网管系统远程管理所有支持这种协议的网络设备，包括监视网络状态、修改网络设备配置、接收网络事件告警等。它是目前 TCP/IP 网络中应用最广泛的网络管理协议。

工作机制

SNMP 的结构分为代理 Agent 和网管系统两部分。代理和网管系统之间是通过 UDP 传送 SNMP 报文进行通信。SNMP 工作机制如图 10-1。

图 10-1 SNMP 工作机制示意图

甲信 NView NNM 网管系统能够提供友好的人机交互界面，方便网络管理员完成网络管理工作，主要可以实现以下功能：

- 向被管设备发送请求报文。
- 接收来自被管设备的响应报文和 Trap 报文，并显示结果。

代理是驻留在被管设备上的一个进程，主要实现以下功能：

- 接收、响应来自 NView NNM 网管系统的请求报文。
- 根据报文类型，对其进行读或写操作，并生成响应报文，返回给 NView NNM 网管系统。
- 根据各协议模块对触发条件进行定义，在达到触发条件后进入系统、退出系统、设备重新启动等，响应的模块通过代理向 NView NNM 网管系统发送 Trap 报文，报告设备的当前状态。

说明

代理可以同时配置多个版本，采用不同的版本与不同的网管系统交互。但是当代理和某个网管系统通信时，代理和该网管系统上的 SNMP 版本配置必须相同，才能正确互通。

协议版本

目前，SNMP 协议共有 v1、v2c 和 v3 三个版本。

- SNMPv1 采用共同体名（Community Name）认证机制。共同体名用来定义 SNMP 网管系统和 SNMP 代理之间的关系，起到了类似于密码的作用，用来限制 SNMP

网管系统对 SNMP 代理的访问。如果 SNMP 报文携带的共同体名在设备上没有认证通过，该报文将被丢弃。

- SNMPv2c 也采用共同体名认证机制。它在兼容 SNMP v1 的同时又扩充了 SNMP v1 的功能：支持更多的操作类型、数据类型和错误代码、能够更细致地区分错误。
- SNMPv3 采用了 USM（User-Based Security Model，基于用户的安全模型）和 VACM（View-based Access Control Model，基于视图的访问控制模型）安全机制。用户可以设置认证和加密功能，通过有无认证和有无加密等功能组合，可以为 SNMP 网管系统和 SNMP 代理之间的通信提供更高的安全性。认证用于验证报文发送方的合法性，避免非法用户的访问；加密则是对网管系统和代理之间的传输报文进行加密，以免被窃听。

设备同时支持 SNMP 的 v1、v2c、v3 三个版本。

MIB

MIB（Management Information Base，管理信息库）是网管系统能够管理的所有的对象的集合。它定义了被管理对象的一些属性：

- 名字
- 访问权限
- 数据类型

通过对这些数据项目的存取访问，就可以得到与设备相关的统计内容。每个代理都有自己的 MIB。MIB 可以看成是网管系统和代理之间的一个接口，通过这个接口，网管系统可以对代理中的每一个被管对象进行读/写操作，从而达到管理和监控设备的目的。

MIB 采用树形结构进行存储，它的根在最上面，没有名字。树的节点表示被管理对象，它可以从根开始的一条路径唯一地识别（OID）。SNMP 协议报文通过遍历 MIB 树形目录中的节点来访问网络中的设备。

设备支持标准的 MIB 库和甲信自定义的 MIB 库。

10.1.2 配置准备

场景

当用户需要通过网管系统登录交换机设备时，应首先对设备配置 SNMP 基本功能。

前提

在配置 SNMP 之前，需完成以下任务：

- 配置路由协议，使设备和网管系统之间路由可达。

10.1.3 SNMP 的缺省配置

设备上 SNMP 的缺省配置如下。

功能	缺省值
SNMP 服务	缺省关闭，需要手动配置开启

功能	缺省值
SNMP 视图	缺省存在：internet 视图
SNMP 共同体	缺省存在：public、private 共同体
SNMP 访问组	缺省不存在
SNMP 用户	缺省不存在
SNMP 用户和访问组的映射关系	无
网管人员的标识和联系方法	support@JX.com
设备放置的物理位置	World China JX
Trap 状态	使能
SNMP 目标主机地址	无
SNMP 引擎 ID	800022B603000000111233

10.1.4 配置 SNMP v1/v2c 基本功能

SNMP Agent 为了保护自身及其管理的 MIB 不被非法访问，提出了共同体的概念。在某一个共同体内的管理站必须在所有对 Agent 的操作中使用该共同体的名字，否则其请求不被受理。

共同体名是用不同的字符串来标识不同的 SNMP 团体。不同的共同体可以具有只读（read-only）或读写（read-write）访问权限。具有只读权限的团体只能对设备信息进行查询，具有读写权限的团体除了可以对设备信息进行查询之外还可以对设备进行配置。

SNMP v1/v2c 采用共同体名认证方案，与设备认可的共同体名不符合的 SNMP 报文将被丢弃。

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**configure**	进入全局配置模式。
2	JX(config)#**snmp server start**	启动 SNMP 服务端功能。
3	JX(config)#**snmp mib-view** *view-name* { **included** | **excluded** } *oid-tree* [**mask** *subtree-mask*]	创建 SNMP 视图，并配置访问的 MIB 变量范围。 缺省视图是 internet，范围包括 MIB 树中“1.3.6”节点以下的所有 MIB 变量。

步骤	配置	说明
4	JX(config)#**snmp community** { **read** | **write** } { **cipher** | **plain** } *community-name* [**mib-view** *view-name* |**acl-ipv4** *acl-ipv4-number* |**acl-ipv6** acl-ipv6-number]*	创建共同体名并配置对应的视图和访问权限。 如果没有填写 **mib-view** *view-name* 选项，则采用缺省视图 internet。

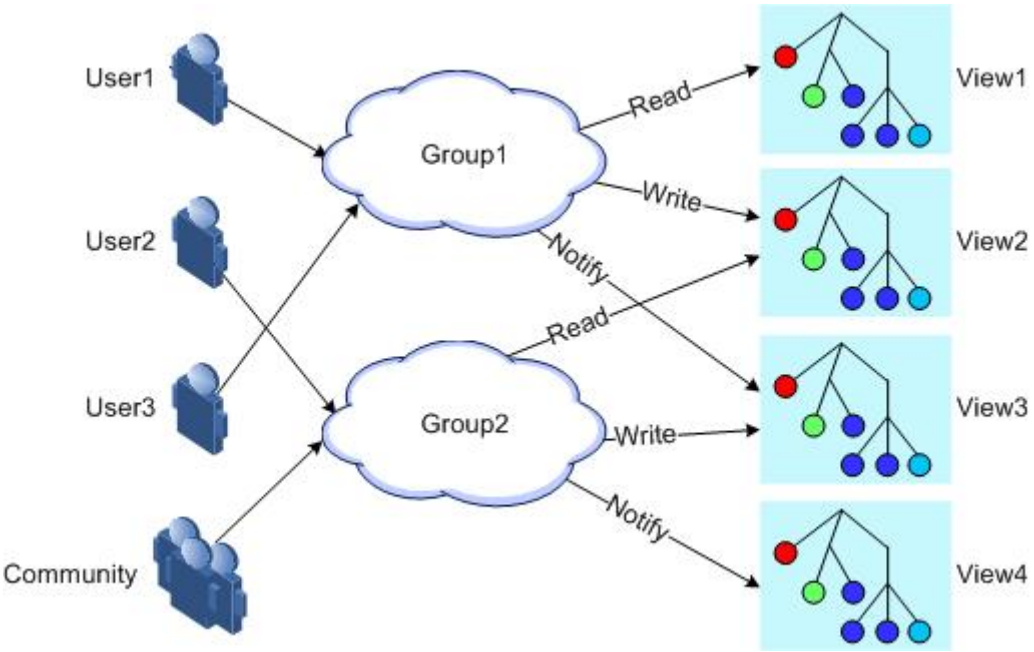
10.1.5 配置 SNMP v3 基本功能

SNMPV3 采用 USM 基于用户的认证机制。USM 提出了访问组（Group）的概念：一个或多个用户对应于一个访问组，每个访问组设定相应的读、写、通告视图，访问组中的用户拥有在该视图内的权限。发送 Get 和 Set 等请求的用户所在的访问组必须具有和其请求相应的权限，否则请求不被受理。

如图 10-2 所示，网管站采用 SNMP v3 对交换机正常的访问，需要进行的配置如下：

- 配置用户
- 确定用户属于哪个访问组
- 配置访问组拥有的视图权限
- 创建视图

图 10-2 SNMP v3 认证机制示意图



请在设备上进行以下配置。

步骤	配置	说明
1	JX#**configure**	进入全局配置模式。
2	JX(config)#**snmp mib-view** *view-name* { **included** | **excluded** } *oid-tree* [**mask** *subtree-mask*]	创建 SNMP 视图，并配置访问的 MIB 变量范围。 缺省视图是 internet，范围包括 MIB 树中“1.3.6”节点以下的所有 MIB 变量。
3	JX(config)#**snmp group** *group-name* { **authentication** | **privacy** | **noauthentication** } [**read-view** *read-view-name* | **write-view** *write-view-name* | **notify-view** *notify-view-name*]*	配置访问组相关。
4	JX(config)#**snmp user** *user-name* **group** *group-name* **authentication** { **md5** | **sha** } *authpassword* **privacy** { **des** | **aes** } *privkeypassword*	创建用户并绑定组，并配置认证和加密。
5	JX(config)#**snmp user** *user-name* **group** *group-name* **authentication** { **md5** | **sha** } *authpassword*	创建用户并绑定组，并配置认证方式。
6	JX(config)#**snmp user** *user-name* **group** *group-name* [**acl-ipv4** *acl-ipv4-number* | **acl-ipv6** *acl-ipv6-number*]*	创建用户并绑定访问组。

10.1.6 配置 SNMP 其他信息

SNMP v1、v2c、v3 均支持以下信息的配置。

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**configure**	进入全局配置模式。
2	JX(config)#**snmp contact** *string*	配置联系方法。
3	JX(config)#**snmp location** *location-string*	配置设备当前所在地址信息。
4	JX(config)#**snmp auth-fail-reply** { **enable** | **disable** }	配置在认证失败时是否发送 SNMP 响应报文。
5	JX(config)#**snmp auth-trap** { **enable** | **disable** }	配置是否使能认证 Trap。
6	JX(config)#**snmp auth-fail-count** *fail-count-value*	配置 SNMP 认证失败次数。

步骤	配置	说明
7	JX(config)#**snmp reauth-interval** *reauth-interval-value*	配置 SNMP 重认证时间。
8	JX(config)#**snmp packet max-size** { *size-value*|**default** }	配置 SNMP 收发包最大长度。
9	JX(config)#**snmp reply-source-ip** { **enable** | **disable** }	配置将所输入的 IP 地址作为应答报文的源地址
10	JX(config)#**snmp udp-port** { *port-number* | **default** }	配置 SNMP 服务端口号。

10.1.7 配置 Trap

说明

SNMP v1、v2c 和 v3 的 Trap 配置步骤除了目标主机的配置，其余都是一样的，请根据需要进行选择。

Trap 是设备主动向网管系统发送的未经请求的信息，用于报告一些紧急的重要事件。

在配置 Trap 功能之前，需完成以下任务：

- 配置 SNMP 的基本功能。如果使用 SNMP v1 和 v2c 版本需要配置共同体名；如果使用 SNMP v3 版本需要配置用户名和 SNMP 视图。
- 配置路由协议，使设备和网管系统之间路由可达。

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**configure**	进入全局配置模式。
2	JX(config)#**snmp server** { **start** | **stop** }	启动 SNMP 服务端功能。
3	JX(config)#**snmp trap-host** *ip-address* { **v1** | **v2** } [**port** *port-id* | **securityname** *security-name* | **vpn-instance** *vpn-name*]*	配置基于 SNMP v1 和 SNMP v2c 的 Trap 目标主机(IPV4)。
4	JX(config)#**snmp-ipv6 trap-host** *ipv6-address* { **v1** | **v2** } [**port** *port-id* | **securityname** *security-name* | **vpn-instance** *vpn-name*]*	配置基于 SNMP v1 和 SNMP v2c 的 Trap 目标主机(IPV6)。
5	JX(config)#**snmp trap-host** *ip-address* **v3** { **authentication** | **privacy** } **securityname** *security-name* [**port** *port-id* | **vpn-instance** *vpn-name*]	配置基于 SNMP v3 的 Trap 目标主机(IPV4)。

步骤	配置	说明
6	JX(config)#**snmp-ipv6 trap-host** *ipv6-address* **v3 { authentication | privacy } securityname** *security-name* **[port** *port-id* **| vpn-instance** *vpn-name* **]***	配置基于 SNMP v3 的 Trap 目标主机 IPV6)。

10.1.8 检查配置

配置完成后，请在设备上执行以下命令检查配置结果。

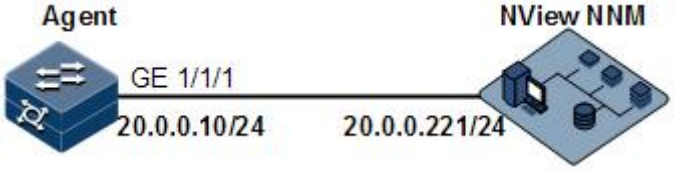
序号	检查项	说明
1	JX#**show snmp information**	查看 SNMP 全局的信息。 包含本地 SNMP 引擎 ID，网管人员的标识及联系方法，设备所在位置，Trap 开关状态信息，SNMP 服务开启状态，服务端口号。
2	JX#**show snmp group**	查看 SNMP 访问组的信息。
3	JX#**show snmp community**	查看 SNMP 共同体的配置信息。
4	JX#**show snmp config**	查看 SNMP 的基本配置信息。
5	JX#**show snmp trap-host**	查看 Trap 目标主机信息。
6	JX#**show snmp statistics**	查看 SNMP 的统计信息。
7	JX#**show snmp user**	查看 SNMP 用户信息。
8	JX#**show snmp mib-view**	查看 SNMP 的视图信息。

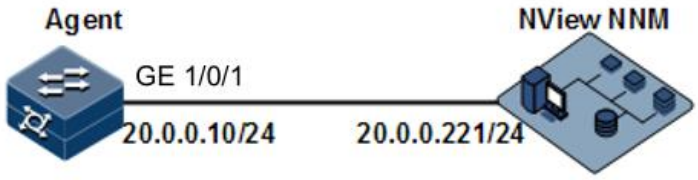
10.1.9 配置 SNMP v1/v2c 和 Trap 示例

组网需求

如图 10-3 所示，NView NNM 网管系统与交换机设备之间路由可达，NView NNM 通过 SNMP v1/v2c 可以查看远程交换机的对应视图下的 MIB，交换机出现紧急情况时可以主动向 NView NNM 发送 Trap。

图 10-3 SNMP v1/v2c 组网示意图





配置步骤

步骤 1 配置交换机设备的 IP 地址。

```
JX#config
JX(config)#interface meth 0/0/0
JX(config-meth-0/0/0)#ip address 192.168.62.100/24
JX(config-meth-0/0/0)#quit
```

步骤 2 启动 SNMP 服务端功能。

```
JX(config)#snmp server start
```

步骤 3 配置 Trap 告警。

```
JX(config)#snmp trap-host 192.168.62.1 v2
```

检查结果

通过 **show ip interface** 查看 IP 地址配置是否正确。

```
JX#show ip interface
Total number: 2
 Interface         State(a/o) Addr/Prefix       Role     Type      Vpn-
instance
------------------------------------------------------------------------------
---------------------------
 loopback-0        up/up     127.0.0.1/8       primary   auto      N/A
 meth-0/0/0        up/up     192.168.62.100/24  primary   static    N/A
------------------------------------------------------------------------------
---------------------------
```

通过 **show snmp config** 查看 SNMP 配置是否正确。

```
JX#show snmp config
Software Version: SNMP_VL3.00.00.00
!
snmp server start
snmp trap-host 192.168.62.1 securityname
AJ35Grrnpx3xRv_UdbJBiGsl3Dwxp932il5pfsKvs8X-GN6R49ihEydRaxBXkM5V-w
```

通过 **show snmp trap-host** 查看目标主机配置是否正确。

```
JX#show snmp trap-host
Trap-host : 192.168.62.1
------------------------------------------------------------------------------
-------
Status            : ACTIVE
```

```
Udp Port           : 162
MP Modle           : V2
Security Level      : None
Security Name       : AJ35Grrnpx3xRv_UdbJBiGsl3Dwxp932il5pfsKvs8X-
GN6R49ihEydRaxBXkM5V-w
Vpn Instance Name    : public
------------------------------------------------------------------------
-------
```

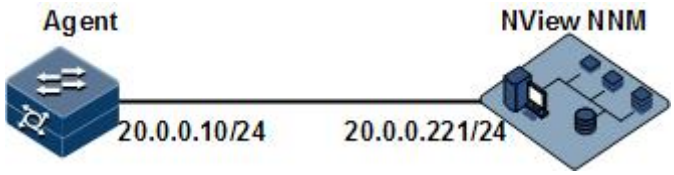
10.1.10 配置 SNMP v3 和 Trap 示例

组网需求

如图 10-4 所示，NView NNM 网管系统与 Agent 之间有可达路由，NView NNM 通过 SNMP v3 对 Agent 进行监控，Agent 出现紧急情况时可以主动向 NView NNM 发送 Trap。

缺省情况下，交换机设备上存在 VLAN 1，所有物理接口属于 VLAN 1。

图 10-4 SNMP v3 和 Trap 组网示意图



配置步骤

步骤 1 配置交换机设备的 IP 地址。

```
JX#config
JX(config)#interface meth 0/0/0
JX(config-meth-0/0/0)#ip address 192.168.62.100/24
JX(config-meth-0/0/0)#quit
```

步骤 2 配置 SNMP v3 访问。

启动 SNMP 服务端功能。

```
JX(config)#snmp server start
```

创建 g1 的访问组，安全等级为认证但不加密。

```
JX(config)#snmp group g1 authentication
```

创建用户 u1 为认证但不加密，绑定访问组 g1，采用 md5 鉴别算法，口令为 JX。

```
JX(config)#snmp user u1 group g1 authentication md5 JX
```

步骤 3 配置 Trap 告警，配置的 trap-host 类型必须与 user 的认证类型一致，不存在包含关系。

```
JX(config)#snmp trap-host 192.168.62.1 v3 authentication securityname u1
```

检查结果

通过 **show snmp group** 查看 SNMP 访问组信息配置是否正确。

```
JX#show snmp group
Group                          Security     ReadView
WriteView                       NotifyView
-------------------------------------------------------------------------
-------------------------------------------------
g1                            authNoPriv   internet                      -
-
-------------------------------------------------------------------------
-------------------------------------------------
```

通过 **show snmp user** 查看用户和访问组的映射关系配置是否正确。

```
JX#show snmp user
User                           Group                          Auth    Priv
Filter
-------------------------------------------------------------------------
---------------------------
u1                            g1                            MD5    no-priv  N/A
-------------------------------------------------------------------------
---------------------------
```

通过 **show snmp trap-host** 查看 Trap 目标主机配置是否正确。

```
JX#show snmp trap-host
Trap-host : 192.168.62.1
-------------------------------------------------------------------------
-------
Status             : ACTIVE
Udp Port           : 162
MP Modle           : V3
Security Level      : Auth
Security Name       : u1
Vpn Instance Name    : public
-------------------------------------------------------------------------
-------
```

通过 **show snmp config** 查看所有配置信息是否正确。

```
JX#show snmp config
Software Version: SNMP_VL3.00.00.00
!
snmp server start
snmp trap-host 192.168.62.1 v3 authentication securityname u1
snmp group g1 authentication
snmp user u1 group g1 authentication md5
ACs87nMCx4RbzvvPS5lo3zDtbzu5xyzSZaDGmHOuqsc0
```

10.2 RMON

10.2.1 简介

RMON（Remote Network Monitoring，远端网络监控）是 IETF 制定的一种可以通过不同的网络代理 Agent 和网管中心进行网络数据监控的标准。

RMON 基于 SNMP 体系结构实现，包括网管中心和运行在各网络设备上的代理 Agent 两部分。在 SNMP 基础上增加了子网流量、统计、分析能力，可以实现对一个网段乃至整个网络的监控，而 SNMP 只能监控单个设备局部信息，对一个网段的监控非常困难。

RMON 中代理 Agent 一般称为探测程序，RMON Probe（RMON 探测器）能够统计和分析子网的通信性能指标，无论何时发现网络故障，RMON Probe 都能够上报网管中心，并描述不正常情况的捕获信息，网管中心无需对设备进行不停的轮询。RMON 可以比 SNMP 更主动、有效地监测远程设备，网络管理员可以更快地跟踪网络、网段或设备出现的故障。该方法减少了网管中心同代理 Agent 间的数据流量，使简单而有力地管理大型网络成为可能，弥补了 SNMP 在日益扩大的分布式互联中所面临的局限性。

RMON Probe 收集数据的方式：

- 分布式 RMON：利用专用的 RMON Probe 收集数据，网管中心直接从 RMON Probe 获取管理信息并控制网络资源。
- 嵌入式 RMON：将 RMON Agent 直接嵌入网络设备（如交换机）中，使它们成为带 RMON Probe 功能的网络设备。网管中心使用 SNMP 的基本操作与 RMON Agent 交换数据信息，收集网络管理信息。

设备采用嵌入式 RMON。如图 10-5 所示，在设备上实现了 RMON Agent 功能。通过该功能，管理站可以获得与被管网络设备接口相连的网段上的整体流量、错误统计和性能统计等信息，从而实现对一个网段的监控。

图 10-5 RMON 应用示意图

RMON MIB 中按照功能分成 9 个组，目前实现了 RMON 的四个功能组：即统计组、历史组、告警组和事件组。

- 统计组：负责收集在一个接口上的统计信息，包括接收到的报文计数和大小分布统计。
- 历史统计组：类似于统计组，但它是在一个指定的检测周期内收集统计信息。
- 告警组：在指定的时间间隔内，监视一个指定的管理信息库（MIB）对象，并且设定上升阈值和下降阈值，若监视对象达到阈值则触发一个事件。

- 事件组：配合告警组使用，当告警触发一个事件时，用来记录相应的事件信息，如发送 Trap 信息，写入日志等操作。

10.2.2 配置准备

场景

当用户需要对某一网段进行监控或流量统计时，可以配置 RMON。

RMON 是一种比 SNMP 更高效的监控手段。用户只需要指定告警阈值，超出该阈值时设备将会主动发送告警信息，而不用去获取变量信息。减少管理设备和被管理设备的通信量，对网络进行简单有效的管理。

前提

设备和网管系统之间链路可达。

10.2.3 RMON 的缺省配置

设备上 RMON 的缺省配置如下。

功能	缺省值
统计组	无
历史统计组	禁止
告警组	无
事件组	无

10.2.4 配置 RMON 统计功能

RMON 统计功能可以设置对接口的统计，包括接口收发报文、过小或过大包、冲突、循环冗余校验和错误数、丢弃报文，接收报文长度、碎片、广播、多播、单播消息等。

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**configure**	进入全局配置节点。
2	JX(config)#**interface** *interface-type interface-number*	进入接口节点下。
3	JX(config-ge-1/0/*)#**rmon statistics** *index-number* [**owner** *owner-name*]	添加该接口的 RMON 统计功能并配置相关参数。

说明

当在接口下使用 **no rmon statistics** *index-number* 命令关闭该接口的统计功能时，是指用户不能继续获取该接口的统计数据了，而不是接口不再进行数据统计了。

10.2.5 配置 RMON 历史统计功能

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**configure**	进入全局配置节点。
2	JX(config)#**interface** *interface-type interface-number*	进入接口节点下。
2	JX(config-ge-1/0/*)#**rmon history** *index-number* **bucket** *buckets-number* **interval** *sampling-interval* [**owner** *owner-name*]	添加该接口的 RMON 历史统计功能并配置相关参数。

说明

当在接口下使用 **no rmon history** *index-number* 命令禁止该接口的历史组统计功能时，不再进行数据收集统计，并且将以前收集的所有历史数据清除。

10.2.6 配置 RMON 告警组

可通过设置 RMON 一个告警组实例（*alarm-id*），监控一个 MIB 变量（*object-id*）。当被监控数据的值越过定义的阈值时会产生告警事件，再按照告警事件的定义进行记录日志或向网管站发送 Trap 信息。

所监控的 MIB 变量必须是真实存在的，并且数据值类型设置正确。

- 如果在设置时，变量不存在或值类型不正确，则返回错误。
- 在已经设置成功的告警中，如果后期该变量无法被采集，则该告警就被关掉，若想重新监控该变量，必须重新设置。

只要事件表中配置了上限或下限其中一个事件，符合条件便会触发相应的告警。如果告警上限和下限所对应事件（*rising-event-id*、*falling-event-id*）在事件表中均没有配置，即使达到了告警条件也不会产生告警。

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**configure**	进入全局配置节点。
2	JX(config)#**rmon alarm** *alarm-id object-id*	在 RMON 告警组中添加告警实例，并

步骤	配置	说明
	query-interval { **absolute** | **delta** } **rising-threshold** *rising-threshold rising-event* **falling-threshold** *falling-threshold falling-event* [**startup-alarm** { **rising** | **falling** | **risingorfalling** } | **owner** *owner-name*]*	配置相关参数。

10.2.7 配置 RMON 事件组

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**configure**	进入全局配置节点。
2	JX(config)#**rmon event** *event-id* { **log** | **trap** | **both** } [**description** *string* | **owner** *owner-name*]*	在 RMON 事件组中添加事件，并配置事件的处理方式。

10.2.8 检查配置

配置完成后，请在设备上执行以下命令检查配置结果。

序号	检查项	说明
1	JX#**show rmon config**	查看 RMON 配置信息。
2	JX#**show rmon alarm**	查看 RMON 告警组信息。
3	JX#**show rmon event**	查看 RMON 事件组信息。
4	JX#**show rmon statistics**	查看 RMON 统计组信息。
5	JX#**show rmon history**	查看 RMON 历史统计组配置信息
6	JX#**show rmon history** *index-number*	查看指定 RMON 历史统计组信息。
7	JX#**show rmon history statistics**	查看指定 RMON 历史统计组统计信息。
8	JX#**show rmon log**	查看 RMON 事件组的日志信息。

10.2.9 维护

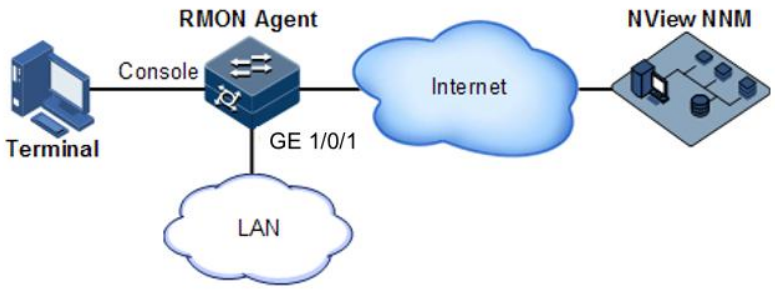
无

10.2.10 配置 RMON 告警组应用示例

组网需求

如图 10-6 所示，交换机设备作为 Agent，通过 Console 口连接配置终端，通过 Internet 连接远端 NNM 系统。使能 RMON 统计功能并对 ge 1/0/1 接口进行性能统计，当接口在一段时间内收到的报文数量超过设置的阈值后，记录日志并发送 Trap 告警。

图 10-6 RMON 典型应用组网示意图



配置步骤

步骤 1 创建索引号为 1 的事件，该事件用于记录并发送,描述字符串为 Falling-etherStatsBroadcastPkts 的日志信息，该日志信息的所有者为 **system**。

```
JX#configure
JX(config)#rmon event 1 both description Falling-etherStatsBroadcastPkts
owner system
```

步骤 2 在接口 ge 1/0/1 下创建统计表，该统计表的所有者也为 **system**

```
JX(config-ge-1/0/1)#rmon statistics 1 owner system
```

步骤 3 创建索引号为 10 的告警项，该告警项用于监控 MIB 变量 etherStatsBroadcastPkts.1 即 1.3.6.1.2.1.16.1.1.1.6.1，每 20 秒检查一次，如果该变量的取值升到 100 以上或者降到 15 以下，便触发 Trap 告警，该告警信息的所有者也为 **system**。

```
JX(config)#rmon alarm 10 1.3.6.1.2.1.16.1.1.1.6.1 20 absolute rising-
threshold 100 1 falling-threshold 15 1 owner system
```

检查结果

通过 **show rmon alarm** 命令查看设备上是否有告警组信息。

```
JX#show rmon alarm
  RMON Alarm:10
       Interval:20
       Source OID:1.3.6.1.2.1.16.1.1.1.6.1
       Sample Type:absolute value
       Alarm Value:0
       Startup Alarm:risingOrFallingAlarm
```

```
        Rising Threshold:100
        Rising Event:1
        Falling Threshold:15
        Falling Event:1
        Owner:system
        Status:valid
```

通过 **show rmon event** 命令查看设备上是否有事件组信息。

```
JX#show rmon event
  RMON Event:1
        Type:trap&log
        Status:valid
        Lastsent time:0 days 12 hours 6 minutes 1 seconds
        Description:Falling-etherStatsBroadcastPkts
        Owner:system
```

通过 **show rmon log** 命令查看设备上是否有事件记录的日志信息。

```
JX#show rmon log
  RMON Log:1/1
        Time:0 days 12 hours 6 minutes 1 seconds
        Description:alarm falling  10,1.3.6.1.2.1.16.1.1.1.6.1,1,0,15
```

当告警事件被触发时，在 NNM 系统的告警管理部分也可以查看相应的记录。

10.3 LLDP

10.3.1 简介

随着网络规模的扩大，网络设备的增多，网络拓扑日趋复杂，对网络的管理变得尤为重要。为了跟踪网络拓扑信息的变化，许多网络管理软件都采用“自动发现”功能来跟踪网络拓扑的变化，但大多数网络管理软件只能分析到网络层拓扑结构，无法确定设备通过哪些接口与其他设备相连。

LLDP（Link Layer Discovery Protocol，链路层发现协议）是由 IEEE 802.1AB 定义的一种链路层发现协议。网络管理系统可以通过该协议快速掌握二层网络的拓扑及其变化情况。

LLDP 将本地设备的信息组织成不同的 TLV（Type Length Value，类型/长度/值单元），并封装在 LLDPDU（Link Layer Discovery Protocol Data Unit，链路层发现协议数据单元）中发送给直连的邻居，同时将邻居发来的信息以标准 MIB（Management Information Base，管理信息库）的形式保存起来，以供网管系统查询及判断链路的通信状况。

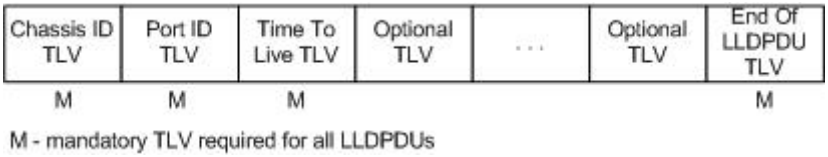
基本概念

LLDP 报文是指在数据单元封装了 LLDPDU 的以太网报文。

LLDPDU 是 LLDP 报文的数据单元。在组成 LLDPDU 之前，设备先将本地信息封装成 TLV，再由若干 TLV 组合成一个 LLDPDU，封装在以太网数据部分进行传送。

如图 10-7 所示，LLDPDU 由若干个 TLV 组合而成，其中包含四个必选的 TLV 和若干个可选的 TLV。

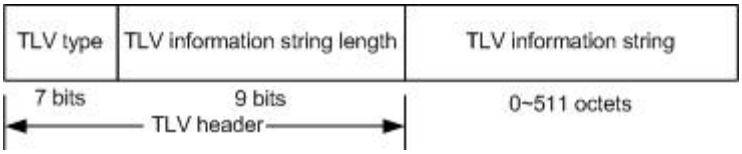
图 10-7 LLDPDU 结构图



TLV 是组成 LLDPDU 的单元，表示一个对象的类型、长度和信息的单元。

TLV 的结构如图 10-8 所示，每个 TLV 代表一个本端信息。例如设备 ID，接口 ID 等各自对应 Chassis ID TLV，Port ID TLV 固定的 TLV。

图 10-8 基本 TLV 结构图



TLV 类型值对应如表 10-1 所示，目前只用到其中的 0～8 种类型。

表 10-1 TLV 类型

TLV 类型	说明	是否必选
0	End Of LLDPDU，表示 LLDP 报文结束	必选
1	Chassis Id，发送设备的 MAC 地址	必选
2	Port Id，LLDP 报文发送端的接口	必选
3	Time To Live，本设备信息在邻居设备上的老化时间	必选
4	Port Description，以太网接口的描述信息	可选
5	System Name，设备名称	可选
6	System Description，系统描述	可选
7	System Capabilities，系统的主要功能以及已使用的功能项	可选
8	Management Address，管理地址	可选

组织定义 TLV 属于可选的 TLV 集合，根据用户的实际需要在 LLDPDU 中发布。目前比较常见的组织定义 TLV 如下。

表 10-2 IEEE 802.1 组织定义的 TLV

TLV 类型	TLV 说明
Port VLAN ID TLV	端口的 VLAN 标识符
Port And Protocol VLAN ID TLV	端口的协议 VLAN 标识符
VLAN Name TLV	端口的 VLAN 名称
Protocol Identity TLV	端口支持的协议类型

表 10-3 IEEE 802.3 组织定义的 TLV

TLV 类型	TLV 说明
MAC/PHY Configuration//Status TLV	端口的速率双工状态、是否支持并使能自动协商功能
Power Via MDI TLV	端口的供电能力
Link Aggregation TLV	端口的链路聚合能力及当前的聚合状态
Maximum Frame Size TLV	端口所能传输的最大的帧的大小

LLDP 工作原理

LLDP 是一种点对点单向发布协议，通过本机向对端周期性的发送 LLDP 报文（或者本端信息有变化时发送 LLDP 报文），通知对端本机的链路状态。

其数据流如下：

- 发送时，设备从 NMS 获取所选择 TLV 需要的系统信息，以及从 LLDP MIB 中获得配置信息，生成 TLV，组成 LLDPDU，封装成 LLDP 报文发送给对端。
- 对端接收到 LLDP 报文后，对端设备会解析获得的各个 TLV 信息，如果有变更，将信息更新至 LLDP 的邻居 MIB 表中，并通知 NMS。

本端设备信息在邻居节点中老化时间 TTL（Time to live），可通过修改老化系数参数值调整，向邻居节点发送 LLDP 报文，邻居节点收到 LLDP 报文后，调整其邻居节点（即发送端）信息的老化时间。老化时间计算公式，TTL=Min{65535，(interval×hold-multiplier)}，其中：

- interval 表示设备向邻居节点发送 LLDP 报文的时间周期。
- hold-multiplier 表示设备信息在邻居节点的老化系数。

10.3.2 配置准备

场景

当用户通过 NView NNM 系统获取设备之间的连接信息，进行拓扑发现时，需要在设备之间使能 LLDP 功能，向邻居互相通告自己的信息，以及存储邻居信息，方便 NView NNM 系统查询。

前提

无

10.3.3 LLDP 的缺省配置

设备上 LLDP 的缺省配置如下。

功能	缺省值
LLDP 全局使能或禁止	禁止
接口 LLDP 功能状态	使能
延迟发送定时器	2s
周期发送定时器	30s
老化系数	4
重启定时器	2s
告警使能或禁止	去使能
告警通知定时器	5s
LLDP 报文目的 MAC 地址	0180.c200.000e

10.3.4 使能全局 LLDP 功能

注意

禁止全局 LLDP 功能后，不能立即再使能，必须等重启定时器超时后才能再次使能。

通过 NView NNM 系统获取设备之间的连接信息，进行拓扑发现时，需要在设备之间使能 LLDP 功能，向邻居互相通告自己的信息，以及存储邻居信息，方便 NView NNM 系统查询。

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**configure**	进入全局配置模式。
2	JX(config)#**lldp { start | stop }**	使能全局 LLDP 功能，使用 stop 格式禁用该功能。 start：使能 LLDP 功能 stop：禁用 LLDP 功能

10.3.5 使能接口 LLDP 功能

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**configure**	进入全局配置模式。
2	JX(config)#**interface** *interface-type interface-number*	进入物理接口配置模式。
3	JX(config-ge-1/0/1)#**lldp admin-status { tx-only | rx-only | rx-tx | disable }** Example: JX(config-ge-1/0/1)#**lldp admin-status tx-only**	使能接口 LLDP 功能，使用 disable 格式禁用该功能。 tx-only：只发 LLDP 报文 rx-only：只收 LLDP 报文 rx-tx：收发 LLDP 报文 disable：禁用 LLDP 功能

10.3.6 全局配置 LLDP 基本功能

注意

配置延时发送定时器和周期发送定时器时，延时发送定时器的取值要小于或等于周期发送定时器取值的四分之一。

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**configure**	进入全局配置模式。
2	JX(config)#**lldp tx-interval** { <5-32768> | **default** } Example: JX(config)#**lldp tx-interval 10**	（可选）配置 LLDP 报文的周期发送定时器。 second：LLDP 报文的发送周期，整数形式，取值范围是 5～32768，单位是秒 default：2 秒

步骤	配置	说明
3	JX(config)#**lldp tx-delay** { <1-8192> | **default** } Example: JX(config)#**lldp tx-delay 5**	可选）配置 LLDP 报文的延迟发送定时器。 second：发送延迟时间，整数形式，取值范围是 1～8192，单位是秒 default：2 秒。
4	JX(config)#**lldp reinit-delay** { <1-10> | **default** } Example: JX(config)#**lldp reinit-delay 5**	可选）配置重启定时器。即设备禁止全局 LLDP 功能后，需要等待重启定时器设定的时间后才能重新使能全局 LLDP 功能。 second：重启延迟时间值，整数形式，取值范围是 1～10，单位是秒 default：2 秒
5	JX(config)#**lldp tx-hold** { <2-10> | **default** } Example: JX(config)#**lldp tx-hold 2**	（可选）配置报文发送间隔的倍数 multiple：倍数，整数形式，取值范围是 2～10，单位是倍 default：4。

10.3.7 接口配置 LLDP 基本功能

步骤	配置	说明
1	JX#**configure**	进入全局配置模式。
2	JX(config)#**interface** *interface-type interface-number* Example: JX(config)#interface **ge** 1/0/1	进入物理接口配置模式。
3	JX(config-ge-1/0/1)#**lldp management-address** { A.B.C.D } { **enable** | **disable** }	（可选）在端口下配置 LLDP 的管理地址 A.B.C.D 管理地址；Enable 使能；disable 去使能

10.3.8 配置 LLDP 告警功能

当网络自身发生变化时，需要使能 LLDP 告警通知功能，及时向 NView NNM 系统发送拓扑信息更新告警。

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**configure**	进入全局配置模式。

步骤	配置	说明
2	JX(config)#**lldp trap-interval** { <5-5600> | **default** } Example: JX(config)#**lldp notification-interval 10**	（可选）配置 lldp 邻居变化时间通告间隔定时器 second：重启延迟时间值，整数形式，取值范围是 5～5600，单位是秒 default：5 秒
3	JX(config)#**interface** *interface-type interface-number* Example: JX(config)#interface ge-1/0/1	进入物理接口配置模式。
4	JX(config-ge-1/0/1)#**lldp trap** { **enable** | **disable** }	配置 LLDP 告警 Trap 使能/去使能

10.3.9 配置 TLV

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**configure**	进入全局配置模式。
2	JX(config)#**interface** *interface-type interface-number*	进入物理接口配置模式。
3	JX(config-ge-1/0/*)#**lldp basic-tlv-tx** { **port-description** | **system-name** | **system-description** | **system-capability** | **all** } { **enable** | **disable** }	（可选）在端口下配置 LLDP 报文的基本 TLV port-description 端口描述；system-name 系统名称；system-description 系统描述；all 所有；s enable 使能；disable 去使能
4	JX(config-ge-1/0/*)#**lldp dot1-tlv- tx port-vid** { **enable** | **disable** } JX(config-ge-1/0/1)#**lldp dot1-tlv-tx protocol-vid** *vlan-list* { **enable** | **disable** } JX(config-ge-1/0/1)#**lldp dot1-tlv-tx vlan-name** *vlan-list* { **enable** | **disable** }	可选）配置端口下设置关于 802.1 组织定义的 TLV 的相关配置。 port-vid 端口 vlan；protocol-vid 协议 vlan；vlan-namevlan 名称；enable 使能；disable 去使能
5	JX(config-ge-1/0/*)#**lldp dot3-tlv-tx** { **mac-phy** | **power** | **link-aggregation** | **max-frame-size** | **all** } { **enable** | **disable** }	（可选）配置端口下关于 802.3 组织定义的 TLV 的相关配置 mac-phy 端口的速率；power 端口的供电能力；link-aggregation 链路聚合；max-frame-size 最大帧长度；all 所有；enable 使能；disable 去使能

步骤	配置	说明
6	JX(config-ge-1/0/1)#**lldp med-tlv- tx { capabilities | network-policy | location | extended-pse | extended-pd | inventory | all } { enable | disable }**	（可选）配置端口下与 MED 相关的配置 Capabilities 能力级；network-policy 支持的应用；location 端口位置标识信息；extended-pse 供电能力；extended-pd 供电能力；inventory 详细目录； all 所有；enable 使能；disable 去使能
7	JX(config-ge-1/0/1)# **lldp voice-vlan { untagged | vlan** *vlan-id* **[cos {***cos-value* **| default} | dscp {***dscp-value* **| default}] }**	（可选）配置端口下与 network-policy tlv 封装 voice vlan 相关的配置 Untagged 配置终端设备发送语音流量时不带 VLAN ID；vlan-id 配置 voice-vlan 的 VLAN ID；cos-value 配置 CoS 优先级，默认值为 5；dscp-value 配置 DSCP 优先级，默认值为 46

10.3.10 检查配置

配置完成后，请在设备上执行以下命令检查配置结果。

序号	检查项	说明
1	JX# **show lldp config**	显示 lldp 配置
2	JX#**show lldp information**	查看 LLDP 本地系统信息。
3	JX#**show lldp remote** [*interface-type interface-number*]	查看 LLDP 邻居信息。
4	JX#**show lldp statistics**	查看 LLDP 统计信息。
5	JX#**show lldp interface** [*interface-type interface-number*]	查看 LLDP 端口信息。

10.3.11 维护

用户可以通过以下命令维护 LLDP 特性。

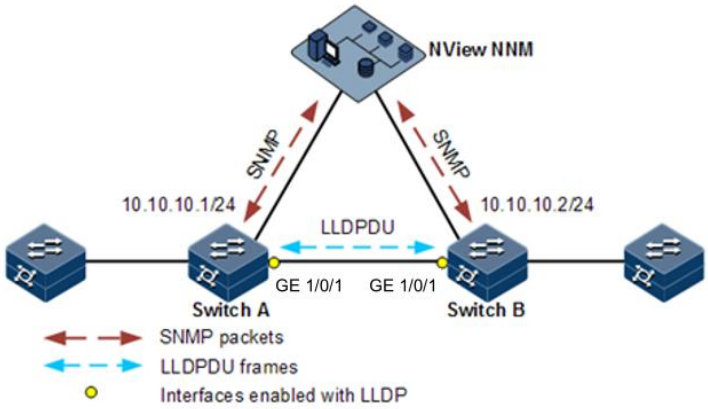
命令	说明
JX(config-ge-1/0/1)# **lldp port statistics counter**	接口下清除 LLDP 统计信息。

10.3.12 配置 LLDP 基本功能示例

组网需求

如图 10-9 所示，交换机和 NView NNM 系统相连，在 Switch A 和 Switch B 之间使能 LLDP 协议，则两设备之间二层链路的变化情况，可以通过 NView NNM 系统查询。如果邻居老化、新增邻居、邻居信息变化时会向 NView NNM 系统上报 LLDP 告警。

图 10-9 配置 LLDP 基本功能组网示意图



配置步骤

步骤 1 配置全局使能 LLDP 并使能 LLDP 告警。

配置 Switch A。

```
JX#hostname SwitchA
SwitchA#config
SwitchA(config)#lldp start
```

配置 Switch B。

```
JX#hostname SwitchB
SwitchB#config
SwitchB(config)#lldp start
```

步骤 2 配置管理 IP 地址。

配置 Switch A。

```
SwitchA(config)# vlan 10
SwitchA(config)# interface vlan 10
SwitchA(config)# interface GE 1/0/1
SwitchA(config-ge-1/0/1)# port hybrid vlan 10 tagged
SwitchA(config-ge-1/0/1)# port hybrid pvid 10
SwitchA(config-ge-1/0/1)#exit
SwitchA(config)#interface vlan 10
```

```
SwitchA(config-vlan1024)# ip address 10.0.0.1/24
SwitchA(config-vlan1024)#exit
```

配置 Switch B。

```
SwitchB(config)# vlan 10
SwitchB (config)# interface vlan 10
SwitchB (config)# interface GE 1/0/1
SwitchB (config-ge-1/0/1)# port hybrid vlan 10 tagged
SwitchB (config-ge-1/0/1)# port hybrid pvid 10
SwitchB (config-ge-1/0/1)#exit
SwitchB (config)#interface vlan 10
SwitchB (config-vlan1024)# ip address 10.0.0.2/24
SwitchB (config-vlan1024)#exit
```

步骤 3 配置 LLDP 属性。

配置 Switch A。

```
SwitchA(config)# lldp tx-interval 60
SwitchA(config)# lldp tx-delay 9
SwitchA(config)# lldp notification-interval 10
```

配置 Switch B。

```
SwitchA(config)# lldp tx-interval 60
SwitchA(config)# lldp tx-delay 9
SwitchA(config)# lldp notification-interval 10
```

检查结果

通过 **show lldp local config** 命令查看本地配置是否正确。

```
SwitchA# show lldp local
  LLDP local:
       Message tx-interval:60(s)
       Message tx-hold:4
       Reinit delay:2(s)
       Tx delay:9(s)
       Notification interval:10(s)
       Chassis type:MAC Address
       Chassis ID:f0f1:f2f3:0101
       System name:SIM-MPU
       System desc:JX-SWITCH-SIM
       System supported:Bridge/Switch,Router
       System capenabled:Bridge/Switch,Router

    Port ge-1/0/1:
       Admin status:TxRx
       Trap enable:no
       Support tlv:port-description,system-name,system-description,system-
capability
       Enabled tlv:port-description,system-name,system-description,system-
capability
       Port type:interface name
       Port ID:GE1/0/1
       Port description:GE1/0/1 SNMP-Index:537397249
```

```
        Number of remote system:1
        Number of MED remote system:0
……
```

通过 **show lldp remote** 命令查看邻居信息是否建立。

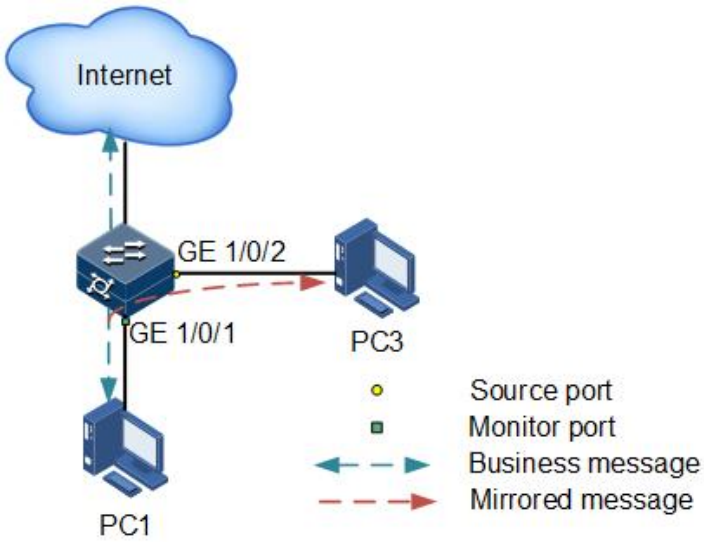
```
SwitchA#show lldp remote
Interface      Index TTL(s)  ChassId                    PortId
 SysName Vlan
ge-1/0/1       1     116     f0f1:f2f3:0201             GE1/0/1
 SIM-MPU 10
ge-1/0/2       2     116     f0f1:f2f3:0201             GE1/0/2
 SIM-MPU --
……
```

10.4 端口镜像

10.4.1 简介

接口镜像功能是指将指定源接口的某些报文镜像到目的接口，即监视接口，而不影响正常报文转发的功能。交换机设备用户使用该功能可以监控某个接口的报文接收和发送情况，并分析相关网络状况。

图 10-10 接口镜像功能原理示意图



接口镜像功能基本原理如上图所示。PC 1 通过交换机的 GE 1/0/1 与外网连接，PC 3 为监测 PC，通过交换机的 GE 1/0/2 与外网连接。

当要监测从 PC 1 发出的报文时，需要将 PC 1 所连接设备的 GE 1/0/1 指定为镜像源接口，并使能对入接口报文的镜像功能，而将 GE 1/0/2 指定为监视接口，即镜像目的接口。

当 GE 1/0/1 发出的业务报文进入交换机时，交换机将对入报文进行转发，并复制一份到监视接口（GE 1/0/2）。连接在监视接口的监控设备可以接收这些被镜像的报文，并进行相关的分析工作。

设备支持基于入接口和出接口的数据流镜像。镜像功能生效后，出/入镜像接口的报文会被复制一份到监视接口。监视接口与镜像接口不能为同一个接口。

10.4.2 配置准备

场景

接口镜像功能主要用在网络管理人员定期监控网络内数据类型及流量。

接口镜像功能是将需要被监控接口的流量复制到一个监视接口或者 CPU，从而抓取入/出接口出现故障或异常时的数据流，用来分析、发现问题根源并及时解决。

前提

无

10.4.3 接口镜像的缺省配置

设备上接口镜像的缺省配置如下。

功能	缺省值
接口镜像功能状态	禁止
镜像源接口	无

10.4.4 配置接口镜像功能

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**configure**	进入全局配置模式。

步骤	配置	说明
2	JX(config)#**mirror group** group-number *interface-type interface-number* { **rspan** *vlan-id* **tpid** { **standard** | protocol-id } }	group-number：指定镜像组号 取值范围是 1~4 interface-type：接口类型 interface-number：接口号为 unit/slot/port 形式，取值范围由接口类型决定 （可选）vlan-id：指定远程镜像 VLAN ID 取值范围是 1~4094 （可选）standard ：指定为标准值 0x8100 （可选）protocol-id：当前接口的外层 Tag 的标签协议标识 取值范围是 <0x0-0xffff>。
3	JX(config)#**interface** *interface-type interface-number*	进入物理接口配置模式
4	JX(config-ge-1/0/1)#**mirror** { **inbound** | **outbound** | **both** } **group** *group-number*	配置接口镜像规则 { **inbound** | **outbound** | **both** }：指定镜像方向，入方向、出方向、入/出同时 group-number：指定镜像组号 取值范围是 1~4。

10.4.5 检查配置

配置完成后，请在设备上执行以下命令检查配置结果。

序号	检查项	说明
1	JX#**show mirror group**	显示镜像组配置信息。
2	JX#**show mirror interface**	显示接口上的镜像配置信息

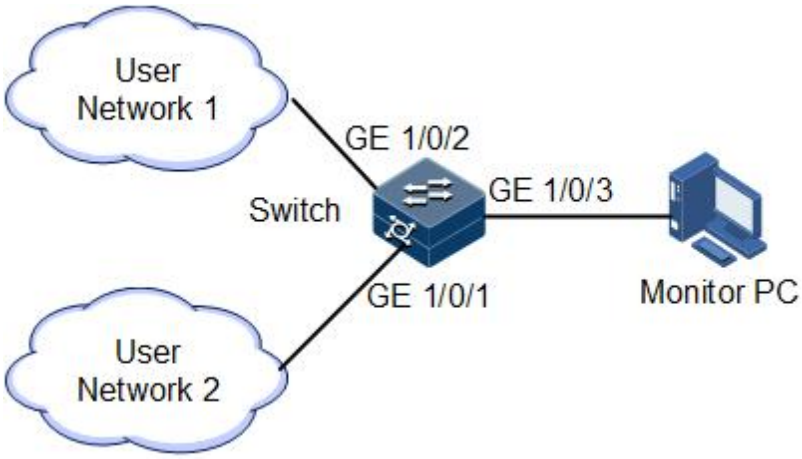
10.4.6 配置接口镜像应用示例

组网需求

如下图所示，网络管理员希望通过数据监测设备仅对用户网络 1 的报文进行监控，从而抓取出现故障或异常时的数据流，来分析、发现问题根源并及时解决。

交换机禁止所有自发包功能和风暴抑制功能。用户网络 1 通过 GE 1/0/2 接入交换机；用户网络 2 通过 GE 1/0/1 接入交换机；数据监测设备连接在交换机的 GE 1/0/3 上。

图 10-11 接口镜像应用组网示意图



配置步骤

步骤 1　在 Switch 上使能接口镜像功能。

```
JX#config
JX(config)#mirror group 1 ge 1/0/3
JX(config)#interface ge 1/0/2
JX(config-ge1/0/2)#mirror ingress group 1
```

检查结果

通过 **show mirror** 查看接口镜像信息配置是否正确。

```
JX# show mirror group
Mirror ---------------------------------------------------------------------
----
  1         ge-1/0/3         n/a
---------------------------------------------------------------------------
```

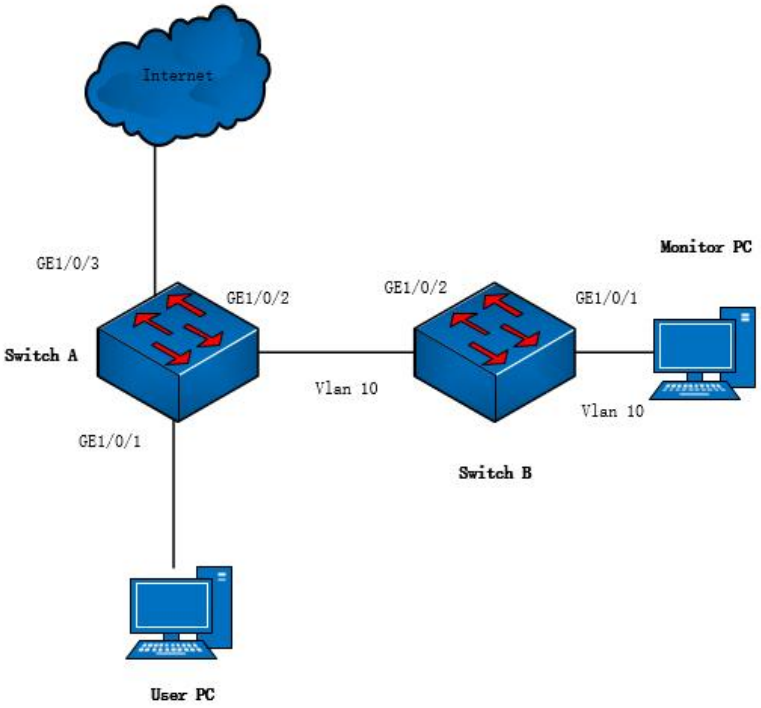
10.4.7 配置远端接口镜像应用示例

组网需求

如下图所示，网络管理员希望通过远端数据监测设备仅对用户 PC 的报文进行监控，从而抓取出现故障或异常时的数据流，来分析、发现问题根源并及时解决。

交换机禁止所有自发包功能和风暴抑制功能。用户通过 GE 1/0/1 接入 switch A；数据监测设备连接在 switch B 的 GE 1/0/1 上。

图 10-12 接口远端镜像应用组网示意图



配置步骤

步骤 1 在 Switch A 上使能接口镜像功能。

```
JX#config
JX(config)#mirror group 1 ge 1/0/2 rspan 10 tpid 0x8100
JX(config)#interface ge1/0/1
JX(config-ge1/0/1)#mirror inbound group 1
```

步骤 2 在 Switch B 上配置 VLAN 转发。

```
JX#config
JX(config)#interface ge 1/0/1
JX(config-ge1/0/1)#port trunk allow-pass vlan 10
JX(config)#interface ge 1/0/2
JX(config-ge1/0/2)#port trunk allow-pass vlan 10
```

10.5 电缆诊断

10.5.1 简介

设备支持电缆诊断功能，可以对线路进行检测。

电缆诊断可查询的结果如下：

- 发送线缆的检测结果；
- 发送线缆的错误位置；

- 接收线缆的检测结果；
- 接收线缆的错误位置。

10.5.2 配置准备

场景

使能设备的电缆诊断功能，可及时了解设备电缆线路的运行状态，及早定位并排除设备电缆故障。

前提

无

10.5.3 配置电缆诊断功能

请在需要配置电缆诊断的设备上进行以下配置。

步骤	配置	说明
1	JX(config)#**virtual-cable-test force-detect**	全局使能电缆诊断，对设备上所有支持电缆诊断的端口进行一次电缆诊断。
2	JX(config-ge-1/0/5)#**virtual-cable-test**	使能接口的电缆诊断功能。

说明

使能接口电缆诊断不重启接口的功能时，如果接口状态为 Up，接口会重启一次，获取电缆诊断数据。配置此功能后，执行电缆诊断时，如果接口状态为 Up，接口不会再次重启，直接读取缓存中上次电缆诊断的数据；如果接口状态为 Down，接口执行一次实际电缆诊断获取实际故障点的长度；新插入的接口会自动执行电缆诊断并将结果存入缓存。

10.5.4 检查配置

配置完成后，请在设备上执行以下命令检查配置结果。

序号	检查项	说明
1	JX(config)#**show virtual-cable-test config**	查看电缆诊断相关配置信息。
2	JX(config)#**show virtual-cable-test**	查看所有接口的电缆诊断的结果。

序号	检查项	说明
3	JX(config)#**show virtual-cable-test interface** *interface-type primary-interface-number*	查看指定接口的电缆诊断的结果。
4	JX(config)#**show virtual-cable-test link-down**	查看所有是 down 状态接口的电缆诊断的结果。

10.6 UDLD

10.6.1 简介

UDLD（UniDirectional Link Detection，单向链路检测）用于监听利用光纤或以太网线连接的物理配置，当出现单向链路（只能向一个方向传输）时，UDLD 可以检测出这一状况，关闭相应接口并发送警告信息。单向链路可能引起很多问题，尤其是生成树，可能会造成回环。

10.6.2 配置准备

场景

当出现单向链路（只能向一个方向传输）时，UDLD 可以检测出这一状况，关闭相应接口并发送警告信息。

原理

UDLD 协议通过与对方交互协议报文（DLDPDU）来识别对端设备、检测单向链路。UDLD 协议有七种状态：Init（初始化）、Linkdown（未连通）、Linkup（活动）、Advertisement（通告）、Detect（探测）和 Disable（单通）状态。

前提

UDLD 需要链路两端设备都支持才能正常运行。

10.6.3 UDLD 功能的缺省配置

设备上故障转移功能的缺省配置如下。

功能	缺省值
UDLD 功能	禁用

10.6.4 配置 UDLD

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局模式。
2	JX(config)#**udld error-down recovery** { **disable** | **enable** }	配置单向链路恢复使能或去使能。
3	JX(config)#**udld error-down recovery-interval** *interval*	配置单向链路恢复时间间隔，默认值为 45 秒。
4	JX(config)#**udld work-mode** { **aggressive** | **normal** }	配置 UDLD 协议工作模式，分为普通模式和激进模式。
5	JX(config)#**udld advertise-interval** *interval*	配置 UDLD 协议发送 advertise 报文事件间隔，默认事件间隔为 7 秒。
6	JX(config)#**udld global up-delay** *time*	配置全局的端口单通恢复时端口延迟 up 的时间。
7	JX(config)#**udld snmp-trap** { **disable** | **enable** }	使能或去使能 UDLD 协议发送 TRAP 功能。
8	JX(config)#**udld uni-shutdown** { **auto** | **manual** }	配置当 UDLD 协议检测到端口单通时是自动还是自动将端口关闭。
9	JX(config)#**interface** *interface-type primary-interface-number* JX(config-ge-1/0/*)#**udld** { **disable** | **enable** }	端口下使能或去使能 UDLD 功能。
10	JX(config-ge-1/0/*)#**udld aggressive** { **disable** | **enable** }	使能或去使能端口的 UDLD 协议工作模式为激进模式。
12	JX(config-ge-1/0/*)#**udld rx-mode** { **normal** | **rxloss** }	配置端口是否关注光模块 RXLOS 等相关消息进而进行接口 error-down 和 up 处理。
13	JX(config-ge-1/0/*)#**udld up-delay** *time*	配置端口的单通恢复时端口延迟 up 的时间。

10.6.5 检查配置

配置完成后，请在设备上执行以下命令检查配置结果。

序号	检查项	说明
1	JX#**show udld config**	查看 UDLD 配置信息。
2	JX#**show udld interface**	查看所有使能了 UDLD 协议的端口的协议状态
3	JX#**show udld local**	查看 UDLD 协议本地信息。

序号	检查项	说明
4	JX#show udld peer	查看 UDLD 协议邻居信息。

10.7 光模块数字诊断

10.7.1 简介

设备上光模块数字诊断支持对 SFP（Small Form-factor Pluggables，小型封装可插拔）光模块的诊断。

光模块数字诊断功能为系统提供一种性能监测手段，网络管理员通过分析该模块提供的监测数据，可以预测收发模块的寿命、隔离系统故障并在现场安装中验证模块的兼容性。

光模块数字诊断功能监控光模块的性能参数包括：

- 模块温度
- 内部供电电压
- 发送偏置电流
- 发送光功率
- 接收光功率

当性能参数达到了告警阈值或状态信息发生变化，会产生相应的 Trap 告警信息。

10.7.2 配置准备

场景

光模块故障诊断功能为用户提供一种对 SFP 光模块性能参数的检测手段，用户通过分析光模块的监测数据，可以预测其寿命、隔离系统故障并在现场安装中验证光模块的兼容性。

前提

设备所使用光模块要求为甲信认证光模块，如使用其他厂家光模块，可能导致业务不稳定、不支持诊断或诊断信息不准确等问题。

10.7.3 光模块数字诊断的缺省配置

设备上光模块数字诊断的缺省配置如下。

功能	缺省值
接口光模块数字诊断告警发送 Trap 功能	禁止

10.7.4 配置使能光模块数字诊断

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**transceiver monitor interval** *interval*	配置光模块数字诊断轮询间隔时间。

10.7.5 配置光模块数字诊断告警发送 Trap

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**interface** *interface-type interface-number*	进入物理层接口配置模式。
3	JX(config 10ge 1/0/*)#**transceiver snmp-trap** { **enable** | **disable** }	使能接口光模块数字诊断告警发送 Trap。
4	JX(config 10ge 1/0/*)#**transceive rx-power low-threshold** *low-threshold-value* **high-threshold** *high-threshold-value*	设置光模块收光功率的高低门限值，在光模块收光功率低于低门限值或者高于高门限值时，光模块可以发出相应告警
5	JX(config 10ge 1/0/*)#**transceive tx-power low-threshold** *low-threshold-value* **high-threshold** *high-threshold-value*	设置光模块发光功率的高低门限值，在光模块发光功率低于低门限值或者高于高门限值时，光模块可以发出相应告警
6	JX(config 10ge 1/0/*)#**transceive temperature low-threshold** *low-threshold-value* **high-threshold** *high-threshold-value*	设置光模块温度的高低门限值，在光模块温度低于低门限值或者高于高门限值时，光模块可以发出相应告警
7	JX(config 10ge 1/0/*)#**transceive voltage low-threshold** *low-threshold-value* **high-threshold** *high-threshold-value*	设置光模块电压的高低门限值，在光模块电压低于低门限值或者高于高门限值时，光模块可以发出相应告警
8	JX(config 10ge 1/0/*)#**transceive bias-current low-threshold** *low-threshold-value* **high-threshold** *high-threshold-value*	设置光模块偏执电流的高低门限值，在光模块偏执电流低于低门限值或者高于高门限值时，光模块可以发出相应告警
9	JX(config 10ge 1/0/*)#**transceive tec-current low-threshold** *low-threshold-value* **high-threshold** *high-threshold-value*	设置光模块制冷电流的高低门限值，在光模块制冷电流低于低门限值或者高于高门限值时，光模块可以发出相应告警

10.7.6 检查配置

配置完成后，请在设备上进行以下命令检查配置结果。

序号	检查项	说明
1	JX#**show transceiver config**	查看当前配置的光模块监控信息的相关配置。
2	JX#**show transceiver interface**	查看光模块的大概数字诊断信息，展示光模块的收发光功率、温度、电压、偏执电流等实时信息。
3	JX#**show transceiver** *interface-type primary-interface-number*	查看指定端口的光模块的相关监控信息。
4	JX#**show transceiver** *interface-type primary-interface-number* **threshold**	查看指定端口上配置的光模块相关监控信息的门限值。

10.8 系统日志

10.8.1 简介

系统日志功能是指设备将系统信息和调试信息等内容以日志的形式记录并输出到指定的目的地，在设备发生故障时，方便用户查看和定位故障。

设备的系统消息和一些调试输出会被送至系统日志处理。系统日志根据用户的配置将信息送往不同的目的地，接收系统日志的目的地有以下几类。

- Console 控制台：将日志信息通过 Console 接口输出到本地控制台。
- Monitor 监控台：将日志信息输出到监控台，如 Telnet 终端。
- Logfile 日志文件：将日志信息以日志文件形式输出到设备的 Flash 中。
- Buffer 缓冲区：将日志信息输出到 log 缓冲区中。
- Trapbuffer 缓冲区：将日志信息输出到 trap 缓冲区中。
- SNMP 服务器：将日志信息转化为 Trap 输出到 SNMP 服务器。
- Syslog 服务器：将日志信息输出到 syslog 服务器。
- Smtp 电子邮件：将日志信息输出到 smtp 电子邮件。

系统日志信息级别，根据严重程度分为 8 个等级，如表 10-4 所示。

表 10-4 信息级别

严重等级	级别	说明
emergencies	0	系统不可以使用

严重等级	级别	说明
alerts	1	需要立即处理
critical	2	严重状态
error	3	错误状态
warning	4	警告状态
notification	5	正常但是很重要的状态
information	6	通告事件
debugging	7	调试信息

说明

输出信息的严重等级是可手动设置的。根据配置的严重等级输出信息时，仅输出级别小于或等于所配置的严重等级的信息。比如，配置输出级别指定为 3（也可直接指定严重等级 error）的信息输出，则级别为 0～3 的信息，即严重等级为 emergencies～error 的信息均可以输出。

10.8.2 配置准备

场景

设备会将系统的登录成功失败、关键信息、调试信息、错误信息等生成系统日志，输出为日志文件或传送到日志主机、Console 接口或监控台，以便用户查看并定位故障。

前提

无

10.8.3 系统日志的缺省配置

设备上系统日志的缺省配置如下。

功能	缺省值
系统日志功能状态	使能
日志消息输出到 console 控制台功能	使能，缺省级别为 warning（4）
日志信息输出到 file 文件功能	使能，缺省级别为 debugging(7)
日志输出到 monitor 监控台功能	使能，缺省级别为 warning（4）
日志输出到 buffer 缓冲区功能	使能，缺省级别为 debugging（7）

功能	缺省值
日志输出到 SNMP 服务器功能	禁止，缺省级别为 debugging（7）
日志输出到 Trapbuffer 缓冲区功能	禁止，缺省级别为 information（6）
日志输出到 syslog 服务器功能	禁止，缺省级别为 information（6）
日志输出到 smtp 电子邮件功能	禁止，缺省级别为 warning（4）

10.8.4 配置系统日志基本信息

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**configure**	进入全局配置模式。
2	JX(config)#**logging** { **start** | **stop** }	配置使能系统日志功能。
3	JX(config)#**logging file max-number** { *max-num* | **default** }	配置系统日志的日志文件最大个数。
4	JX(config)#**logging file size kbytes** { *size-value* | **default** }	配置系统日志的日志文件大小。

10.8.5 配置系统日志输出

步骤	配置	说明
1	JX#**configure**	进入全局配置模式。
2	JX(config)#**logging module** *module-name* **action** { **console** | **monitor** | **logfile** | **buffer** | **trap** | **trapbuffer** | **syslog** | **smtp** } { **log** | **debug** | **trap** } [**state** { **enable** | **disable** | **default** } | **level** { **emergencies** | **alert** | **critical** | **error** | **warning** | **notification** | **information** | **debugging** | **default** }]	配置模块日志输出方向，输出使能以及输出级别

10.8.6 配置系统日志输出 TELNET/SSH 终端

步骤	配置	说明
1	JX#**terminal monitor** JX#**no terminal monitor**	打开关闭信息输出到 TELNET/SSH 终端。
2	JX#**terminal log** JX#**no terminal log**	打开关闭日志信息输出到 TELNET/SSH 终端。缺省为开启状态。

步骤	配置	说明
3	JX#terminal trap JX#no terminal trap	打开关闭 Trap 信息输出到 TELNET/SSH 终端。缺省为开启状态。
4	JX#terminal debug JX#no terminal debug	打开关闭 Trap 信息输出到 TELNET/SSH 终端。缺省为关闭状态。

10.8.7 检查配置

在设备上执行以下命令可以查看日志相关信息与配置。

序号	检查项	说明
1	JX#show logging information	查看日志配置的全局信息。
2	JX#show logging { buffer | trap }	查看日志或者 trap 缓冲区信息。
3	JX#show logging { buffer | trap } { include | exclude | begin } string *string*	查看日志或者 trap 缓冲区信息，指定字符串过滤条件。
4	JX#show logging { buffer | trap } size *size-number*	查看日志或者 trap 缓冲区信息，指定条目数。
5	JX#show logging { buffer | trap } module *module-name*	查看指定模块的日志或者 trap 缓冲区信息。
6	JX#show logging { buffer | trap } size *size-number* { include | exclude | begin } string *string*	查看日志或者 trap 缓冲区信息，指定条目数和字符串过滤条件。
7	JX#show logging { buffer | trap } start-time *start-time* [end-time *end-time*]	查看日志或者 trap 缓冲区信息，指定日志产生的时间范围。
8	JX#show logging action	查看日志所有模块的动作信息。
9	JX#show logging action { console | monitor|logfile | buffer | trap | trapbuffer | syslog | smtp }	查看日志所有模块指定动作信息。
10	JX#show logging error	查看所有模块日志的错误码信息。
11	JX#show logging event	查看所有模块日志的事件信息。
12	JX#show logging file *file-name*	查看指定日志文件的内容
13	JX#show logging file *file-name* { include | exclude | begin } string *string*	查看指定日志文件的指定内容，指定字符串过滤条件。
14	JX#show logging module	查看所有模块日志的信息，包括模块名、模块 ID、日志文件名、事件数目和错误码条目数。

序号	检查项	说明
15	JX#**show logging statistics**	查看日志模块的统计信息。

10.8.8 维护

用户可以通过以下命令，维护系统日志特性的运行情况和配置情况。

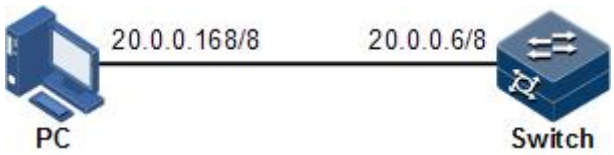
命令	描述
JX(config)#**clear logging** { **buffer** | **trap** }	清除 log 或者 trap 缓冲区中的日志信息。
JX(config)#**clear logging statistics**	清除日志的统计信息。
JX(config)#**clear logging file all**	清除所有日志文件。
JX(config)#**clear logging module** *module-name*	清除指定模块日志的统计信息。

10.8.9 配置系统日志输出到当前终端示例

组网需求

如图 10-13 所示，配置系统日志功能，将设备上的日志信息输出到日志主机，以便用户随时查看。

图 10-13 系统日志输出到日志主机组网示意图



配置步骤

步骤 1 配置设备的 IP 地址。

```
JX#config
JX(config)#interface vlan 1
JX(config-vlanif-1)#ip address 20.0.0.6 255.0.0.0
JX(config-vlanif-1)#quit
```

步骤 2 配置系统日志输出到终端上。

```
JX(config)#terminal monitor
```

步骤 3 配置调试信息输出到终端上。

```
JX(config)#terminal debug
```

检查结果

通过 **show logging information** 命令查看系统日志全局配置信息是否正确。

```
JX#show logging information
------------------------------------------------------------
  Logging                : on
  Module number           : 125
  Logfile path            : "/ram/log"
  Logfile max size        : 3072 Kb
  Logfile max number      : 3
  Logbuffer Max number    : 2000
  Logbuffer Current number : 201
  Logbuffer history number : 216
------------------------------------------------------------
```

10.9 告警管理

10.9.1 简介

告警是指当设备出现故障或某些工作状态发生变化时，系统能够根据不同故障类型、不同告警来源产生的信息。

告警信息用于报告一些紧急且重要的事件，及时通知网络管理人员，为监控设备运行和进行故障诊断提供有力支持。

告警信息保存在设备的告警缓存区中，同时将生成日志信息。若配置了网管系统，则该告警信息会通过 SNMP 向网管系统发送，发送给网管系统的信息称为 Trap 信息。

10.9.2 配置准备

场景

设备发生故障时，由告警管理模块来收集设备故障信息，以日志等形式输出告警的发生时间、告警的名称和描述信息等，帮助用户快速进行问题定位。

如果设备上配置网管系统，告警信息可以直接上报网管系统，给出告警产生的可能原因和处理建议，帮助用户及时处理故障。

如果设备上配置了硬件监控功能。当设备运行环境出现异常时，会记录硬件监控告警表、产生 Syslog 系统日志或发送 Trap 等告警信息，通知用户进行相应处理，防止故障发生。

告警管理方便用户直接在设备上进行告警抑制，告警自动上报，告警监控，告警反转，告警延迟，告警存储模式，清除告警，查看告警等配置。

前提

如果设备上需要配置硬件监控功能时：

- 以 Syslog 方式输出时，告警信息会生成系统日志。当需要把告警信息发送到系统日志主机时，设备上需要配置系统日志主机的 IP 地址等信息。
- 需要把告警信息以 Trap 方式发送到网管中心时，设备上需要配置网管中心的 IP 地址等信息。

10.9.3 配置告警基本功能

无。

10.9.4 检查配置

配置完成后，请在设备上执行以下命令检查配置结果。

序号	检查项	说明
1	JX#**show alarm information**	查看告警管理模块的表项数量信息。
2	JX#**show alarm description**	查看告警管理模块的所有告警信息。 包括告警 id、告警名称、告警级别、对应 Trap-OID 和描述信息。
3	JX#**show alarm** { **active** | **history** | **all** } **time-order** [**object** *object-name*]	查看指定告警信息，并且按时序排列（活动告警、历史告警、全部告警）。
4	JX#**show alarm** { **active** | **history** | **all** } [**object** *object-name*]	查看指定告警信息（活动告警、历史告警、全部告警）。
6	JX#**show alarm object**	查看告警对象信息。
7	JX#**show event description**	查看告警管理模块的所有事件信息。 包括事件 id、事件名称、事件级别、对应 Trap-OID 和描述信息。
8	JX#**show event all** [**object** *object-name*]	查看告警管理模块的所有当前事件。

10.10 CPU 监控

10.10.1 简介

设备支持 CPU 监控功能，可以实时监控系统中各任务的状态、CPU 利用率和堆栈使用情况，帮助网管人员快速定位故障。

CPU 监控可以提供以下功能：

- 查看 CPU 利用率

提供查看各周期（5 秒，1 分钟，10 分钟，2 小时）内各任务的 CPU 占用时间和利用率。可以静态显示，也可以动态显示各周期内 CPU 总的利用率。

提供查看所有任务的运行状态信息和指定任务的详细运行状态信息。

提供查看各周期内 CPU 历史利用率。

提供查看死亡任务信息。

- CPU 利用率门限告警

在指定的采样周期内，系统的 CPU 利用率从低于下限阈值上升到高于上限阈值或者从高于上限阈值下降到低于下限阈值时，会产生告警并发送 Trap，Trap 信息会提供最近某个周期（5 秒，1 分钟，10 分钟）内 CPU 利用率最高的 5 个任务序号及其 CPU 利用率。

10.10.2 配置准备

场景

CPU 监控功能可以实时监控系统中各任务的状态、CPU 利用率和堆栈使用情况，提供 CPU 利用率门限值告警，方便及时发现并消除隐患，或帮助网管人员进行故障定位。

前提

在配置 CPU 监控之前，需完成以下任务：

- 当需要把 CPU 监控告警信息以 Trap 方式输出时，需在设备上配置 Trap 输出目标主机地址，即网管中心的 IP 地址等信息。

10.10.3 CPU 监控的缺省配置

设备上 CPU 监控的缺省配置如下。

功能	缺省值
CPU 利用率告警 Trap 输出	使能
CPU 利用率告警的上限阈值	80%
CPU 利用率采样周期	10s

10.10.4 配置 CPU 监控告警

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**cpu** { **all** | *cpu-index* } **high-threshold** *high-threshold-value*	设置所有 CPU 表项或指定 CPU 表项的告警的上限阈值。

步骤	配置	说明
3	JX(config)#**cpu** { **all** | *cpu-index* } **description** *description-string*	设置所有 CPU 表项或指定 CPU 表项的描述信息。
4	JX(config)#**cpu** { **all** | *cpu-index* } **snmp-trap** { **disable** | **enable** }	设置所有 CPU 表项或指定 CPU 表项的发送 TRAP 使能或去使能。
5	JX(config)#**cpu monitor interval** *interval-value*	配置 CPU 利用率采样时间间隔。
6	JX(config)#**cpu monitor** { **disable** | **enable** }	使能或去使能 CPU 利用率监控。

10.10.5 检查配置

配置完成后，请在设备上执行以下命令检查配置结果。

序号	检查项	说明
1	JX#**show cpu**	查看 CPU 利用率。
2	JX#**show cpu verbose**	查看 CPU 利用率的相关详细信息。
3	JX#**show cpu task**	查看各任务的 CPU 利用率。

10.11 内存监控

10.11.1 配置准备

场景

内存利用率监控功能可以实时监控系统的内存利用率，提供内存利用率阈值告警，方便及时发现并消除隐患，或帮助网管人员进行故障定位。

前提

当用户需要把内存利用率监控告警信息以 Trap 方式输出时，应首先在设备上配置 Trap 输出目标主机地址，即网管中心的 IP 地址等信息。

10.11.2 配置内存监控

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**memory** { **all** | *memory-index* } **high-threshold** *high-threshold-value*	配置所有内存表项或者指定内存表项的监控告警门限阈值。
3	JX(config)#**memory** { **all** | *memory-index* } **description** *description-string*	配置所有内存表项或者指定内存表项的描述信息。
4	JX(config)#**memory** { **all** | *memory-index* } **snmp-trap** { **disable** | **enable** }	配置所有内存表项或者指定内存表项的 TRAP 使能或去使能。
5	JX(config)#**memory monitor interval** *interval-value*	配置内存利用率的监控周期。
6	JX(config)#**memory monitor** { **disable** | **enable** }	配置内存利用率监控使能或去使能。

10.11.3 检查配置

配置完成后，请在设备上执行以下命令检查配置结果。

序号	检查项	说明
1	JX#**show memory**	查看系统内存利用率。
2	JX#**show memory verbose**	查看系统内存利用率的详细信息。
4	JX#**show memory task**	查看系统内每个进程的内存占用信息。

10.12 Ping

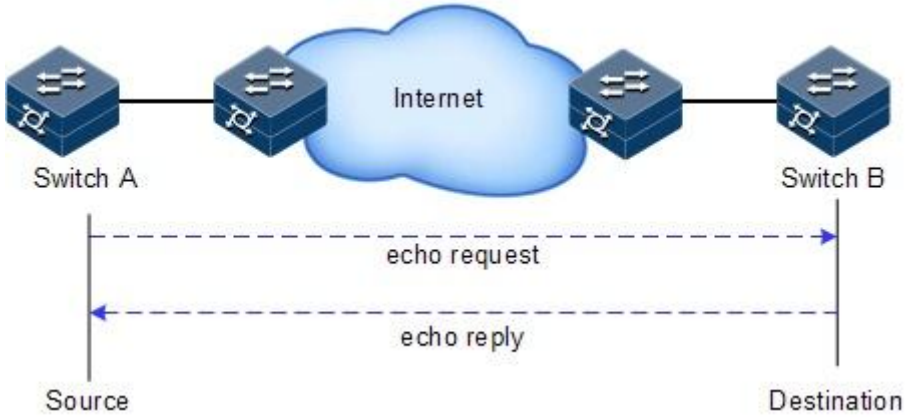
10.12.1 简介

Ping 的名字源于声纳定位操作，用于检测网络连接是否正常。

Ping 功能一般借助 ICMP echo 报文来实现。首先发送 echo request 报文到某个地址，然后等待该地址对应的设备响应 echo reply 报文，当 echo request 到达目标地址以后，在一个有效的时间内返回 echo reply 报文给源地址，则说明目的地可达。如在有效时间内没有收到回应，则在发送端显示超时，并表明目的地不可达。

Ping 功能实现原理如图 10-14 所示。

图 10-14 Ping 功能实现原理组网示意图



10.12.2 配置 Ping 功能

请在设备上进行以下操作。

步骤	配置	说明
1	JX#**ping** { *ipv4-address* | *hostname* } [**-n** *nvalue*|**-system-time**|**-t**|**-q**|**-f**|**-h** *hvalue*|**-l** *lengthvalue*|**-w** *waitvalue*|**-tos** *tosvalue*|**-m** *mvalue*|**-dscp** *dscpvalue*|**-8021p** *8021pvalue*|**-range** *rangevalue*|**-s** *source-ipv4-address*|**-nexthop** *next-ipv4-address*|**-vpn-instance** *vpn-name*|**-eth-trunk** *trunk-number*|**-bridge-domain** *bridge-domain-number*|**-loopback** *loopback-number*|**-vlan** *vlan-id*|**-ge** *interface-number1*|**-10ge** *interface-number2*|**-25ge** *interface-number3*|**-40ge** *interface-number4*|**-100ge** *interface-number5*|**-400ge** *interface-number6*|**-meth** *meth-number*]	通过 **ping** 命令测试 IPv4 网络的连通性。
2	JX#**ping-ipv6** {*ipv6-address* | *hostname* } [**-n** *nvalue* |**-system-time**|**-t**|**-q**|**-f**|**-h** *hvalue*|**-l** *lengthvalue*|**-w** *waitvalue*|**-m** *mvalue*|**-tc** *tcvalue*|**-8021p** *8021pvalue*|**-range** *rangevalue*|**-s** *source-ipv6-adress*|**-nexthop** *next-ipv6-address*|**-vpn-instance** *vpn-name*|**-eth-trunk** *trunk-number*|**-bridge-domain** *bridge-domain-number*|**-loopback** *loopback-number*|**-vlan** *vlan-id*|**-ge** *interface-number1*|**-10ge** *interface-number2*|**-25ge** *interface-number3*|**-40ge** *interface-number4*|**-100ge** *interface-number5*|**-400ge** *interface-number6*|**-meth** *meth-number*]	通过 **ping** 命令测试 IPv6 网络的连通性。

说明

在 **ping** 命令执行的过程中，无法对设备进行其他操作，只有等待执行过程结束或者通过“Ctrl＋C”键强制中断执行过程后才能进行其他操作。

10.13 Trace

10.13.1 简介

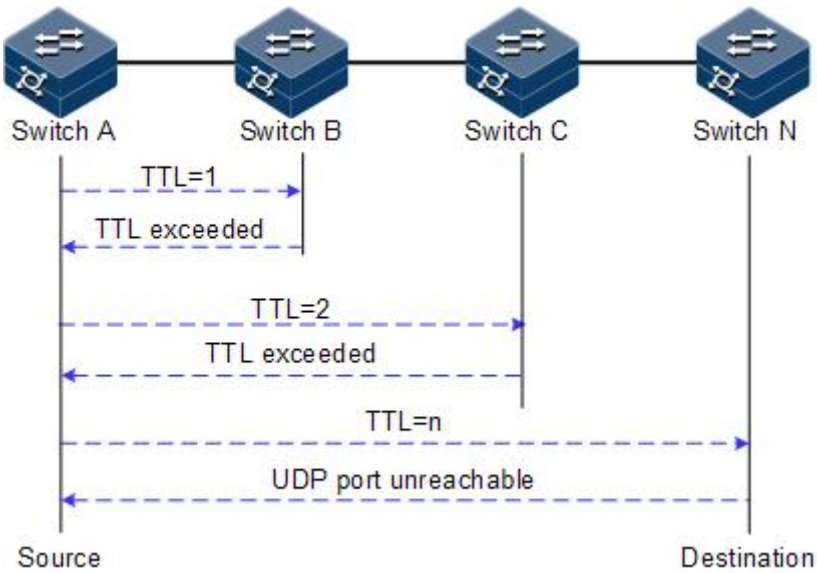
Trace 和 Ping 一样，是网络管理中常用的维护手段。Trace 功能常用于测试报文从发送端到目的端所经过的网络节点，检测网络连接是否可达，并分析网络中的故障点。

Trac 的执行过程如下：

- 首先发送一份 TTL 为 1 的嗅探报文（其中报文的 UDP 端口号是目的端的任何一个应用程序都不可能使用的端口号）。
- 到达第 1 跳时 TTL 减 1，由于 TTL 的值为 0，第 1 跳设备发回一个 ICMP 超时报文，指明此报文不能被发送。
- 发送主机将 TTL 加 1 后重新发送此报文。
- 到达第 2 跳时由于 TTL 的值被减为 0，第 2 跳设备发回一个 ICMP 超时报文，指明此报文不能被发送。

以上步骤循环进行，直到到达目的主机，目的主机并不会送回 ICMP 超时报文，由于目的主机的端口号没有被使用，目的主机会发送端口不可达报文，测试结束。这样，发送主机就能够记录每一个 ICMP TTL 超时报文的源地址，根据得到的回应报文分析出到达目的地所经历的路径。Trace 功能实现原理如图 10-15 所示。

图 10-15 Trace 功能实现原理组网示意图



10.13.2 配置 IPv4 Trace 功能

请在设备上进行以下操作。

步骤	配置	说明
1	JX#**trace** *ip-address* [**-n** *number*|**-r**|**-q**|**-f**|**-fh** *first-ttl-value*|**-h** *max-ttl-value*|**-l** *length-value*|**-w** *wait-for-each-value*|**-tos** *tosvalue*|**-m** *wait-for-send-value*|**-dscp** *dscpvalue*|**-8021p** *8021p-value*|**-range** *range-value*|**-s** *source-ip-address*|**-nexthop** *next-ip-address*|**-vpn-instance** *vpn-name*|**-eth-trunk** *trunk-number*|**-bridge-domain** *bridge-domain-id*|**-loopback** *loopback-number*|**-vlan** *vlan-id*|**-ge** *interface-number1*|**-10ge** *interface-number2*|**-25ge** *interface-number3*|**-40ge** *interface-number4*|**-100ge** *interface-number5*|**-400ge** *interface-number6*|**-meth** *meth-number*]	通过 **trace** 命令测试 IPv4 网络的连通性并查看报文经过的网络节点。

10.13.3 配置 IPv6 Trace 功能

请在设备上进行以下操作。

步骤	配置	说明
1	JX#**trace-ipv6** *ipv6-address* [**-n** *number*|**-r**|**-q**|**-f**|**-fh** *first-ttl-value*|**-h** *max-ttl-value*|**-l** *length-value*|**-w** *wait-for-each-value*|**-m** *wait-for-send-value*|**-tc** *traffic-class-value*|**-8021p** *8021p-value*|**-range** *pkt-length-value*|**-s** *source-ipv6-address*|**-nexthop** *next-ipv6-address*|**-vpn-instance** *vpn-name*|**-eth-trunk** *trunk-number*|**-bridge-domain** *bridge-domain-id*|**-loopback** *loopback-number*|**-vlan** *vlan-id*|**-ge** *interface-number1*|**-10ge** *interface-number2*|**-25ge** *interface-number3*|**-40ge** *interface-number4*|**-100ge** *interface-number5*|**-400ge** *interface-number6*|**-meth** *meth-number*]	通过 **trace** 命令测试 IPv6 网络的连通性并查看报文经过的网络节点。

10.14 硬件监控

说明

不是所有设备都支持温度告警。请以具体设备支持情况为准。

10.14.1 简介

硬件环境监控主要是对设备的运行环境进行监控，监控的对象主要包括设备的温度和电源。

10.14.2 温度监控

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**temperature monitor** { **disable** | **enable** }	使能或去使能温度监控。
3	JX(config)#**temperature** { **all** | *temperature-index* } **description** *description-string*	设置所有或指定温度监控表项的描述信息。
4	JX(config)#**temperature** { **all** | *temperature-index* } **low-threshold** *low-threshold-value* **high-threshold** *high-threshold-value*	设置所有或指定温度监控表项的高低门限值，当温度高于高门限值或低于低门限值时都会产生告警。
5	JX(config)#**temperature** { **all** | *temperature-index* } **snmp-trap** { **disable** | **enable** }	使能或去使能所有温度监控表项或指定温度监控表项发送 TRAP 信息的功能。。

10.14.3 电源监控

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**power monitor** { **disable** | **enable** }	使能或去使能电源信息监控。
3	JX(config)#**power monitor interval** *interval-value*	设置电源信息监控的周期，默认值是 25 秒。
4	JX(config)#**power** { **all** | *power-index* } **description** *description-string*	设置所有电源表项或指定电源表项的描述信息。
5	JX(config)#**power** { **all** | *power-index* } **snmp-trap** { **disable** | **enable** }	使能或去使能所有电源监控表项或指定电源监控表项发送 TRAP 信息的功能。

10.14.4 检查配置

配置完成后，请在设备上执行以下命令检查配置结果。

步骤	配置	说明
1	JX#show temperature config	显示对温度监控所作的相关配置信息。
2	JX#show power config	显示对电源监控所作的相关配置信息。

10.15 风扇监控

说明

本节仅适用于带风扇款型。

10.15.1 简介

设备支持风扇监控功能，可以对风扇的转速和温度进行监控，当设备监控到风扇的转速和温度出现异常时，会产生告警并发送 Trap 信息。

设备对风扇的监控模式有两种：

- 固定转速等级模式，即强制设定风扇的转速等级；
- 温控模式，即风扇根据温度的变化自动调节转速等级。

在温控模式下，当检测到风扇的温度超出高门限值时，就会将风扇的转速等级提升一档，如果检测到风扇的温度低于低门限值时，就会将风扇的转速等级下调一档。

10.15.2 配置准备

场景

当设备安放于比较炎热的环境时，过高的温度会影响设备的散热性能，此时需要配置风扇监控功能，使设备的风扇能够依据周围的环境温度自动调节，以维护设备的正常运转。

前提

无

10.15.3 配置风扇监控功能

请在需要配置风扇监控功能的设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**fan monitor** { **disable** | **enable** }	使能或去使能风扇监控。
3	JX(config)#**fan monitor interval** *interval-value*	配置风扇监控周期。
4	JX(config)#**fan** { **all** | *fan-index* } **mode** { **speed-level** | **temperature-control** }	配置所有风扇或指定风扇的模式，speed-level 为固定转速等级模式，temperature-control 为温控模式。
5	JX(config)#**fan** { **all** | *fan-index* } **description** *description-string*	配置所有风扇或指定风扇的描述信息。
6	JX(config)#**fan** { **all** | *fan-index* } **level** *level-value*	配置所有风扇或指定风扇的转速等级。
7	JX(config)#**fan** { **all** | *fan-index* } **snmp-trap** { **disable** | **enable** }	使能或去使能所有风扇或指定风扇的 TRAP 发送功能。

10.15.4 检查配置

配置完成后，请在设备上执行以下命令检查配置结果。

序号	检查项	说明
1	JX#**show fan**	查看风扇监控的相关信息，当前转速、转速等级等信息。
2	JX#**show fan verbose**	查看风扇的基本信息。
3	JX#**show fan config**	查看风扇监控配置的相关信息。

10.16 设备堆叠

10.16.1 简介

堆叠，是指将两台及以上支持堆叠特性的交换机设备组合在一起，从逻辑上形成一台交换设备。使用这种虚拟化技术可以集合多台设备的硬件资源能力和软件处理能力，实现多台设备的协同工作、统一管理和不间断维护。

堆叠基本概念

- 运行模式

设备支持两种运行模式：

普通模式：该模式下，设备不能与其他设备形成堆叠。

堆叠模式：该模式下，进行堆叠配置后可与其他堆叠设备形成堆叠。

当两种堆叠模式切换时，将会引起设备重启。

- 成员设备角色

堆叠中的每台设备都称为成员设备，每个成员设备根据协商的结果，可能处于不同的角色：

Master：主用设备，负责管理和控制整个堆叠系统。

Slave：处理业务、转发报文的同时作为主用设备的备份设备运行。当主用设备故障时，堆叠系统会自动从备设备中选举一个新的主设备接替原主用设备的工作。

- 成员设备编号

堆叠系统使用成员设备编号来标识和管理成员设备。在进行堆叠配置时，必须保证设备的成员设备编号是唯一的，如果两台设备的成员编号是一样的，则在协商时将会出现问题，导致堆叠系统建立出现问题。

- 成员优先级

成员优先级是成员设备的一个属性，默认情况下，设备的成员优先级是 1。在堆叠协商的过程中，成员优先级的大小将影响成员设备是否能被选举成员堆叠主用设备。如果想让某台设备称为堆叠系统的主用设备，可以将设备的成员优先级设置为一个较大的值。

- 堆叠端口

堆叠端口是一个逻辑端口，专门用于堆叠成员设备之间的连接。每台成员设备可以配置两个堆叠端口，堆叠端口需要和物理端口绑定之后才能生效。

- 堆叠域

堆叠域是一个逻辑概念，一个堆叠系统对应一个堆叠域。只有堆叠域 ID 一致的成员设备之间才能建立堆叠系统，不同堆叠域之间的成员设备即使能够收到对端设备的堆叠报文，也无法建立堆叠系统。

10.16.2 堆叠拓扑

本系列交换机可以支持 3 台设备组成堆叠，堆叠设备之间组成链型或环型拓扑的堆叠系统。

图 10-16 3 台成员设备组成链型拓扑

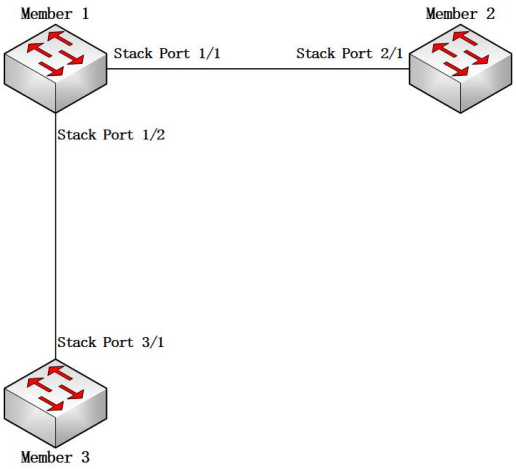
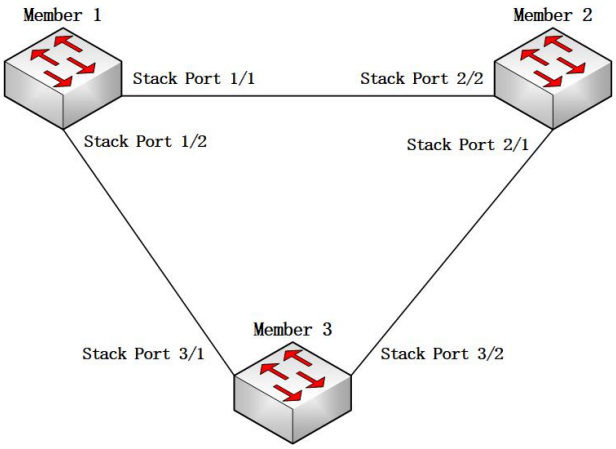


图 10-17 3 台成员设备组成环形型拓扑



注意

- 成员设备组成环型堆叠拓扑时，相邻两台堆叠成员设备之间的堆叠端口号是不一样的。如图 10-16 所示，堆叠成员设备 1 的堆叠端口 1/1 和堆叠成员设备 2 的堆叠端口的 2/2 相连。如果出现两台相连设备之间相同堆叠端口号一样，在主备切换时将会导致堆叠系统出现同步失败。
- 堆叠配置时，需要在普通模式下先规划好堆叠系统拓扑，然后按照拓扑设计配置好对应的堆叠端口，最后配置堆叠模式。否则设备没有保存堆叠端口的配置，设备重启后堆叠端口未创建，不能进行堆叠协商。

堆叠角色选举

当堆叠系统出现如下情况时都将进行角色选举：

- 堆叠系统建立。
- 堆叠系统分裂，即堆叠链路断开。

- 两个独立运行的堆叠系统合并。

角色选举时将按照如下优先级选举主备设备：

- 当前的主设备优先，如果是两个独立堆叠系统合并，此时将在两个对立堆叠系统的主用设备中选举出新的主用设备。
- 成员设备配置了强制为主的设备。
- 成员设备优先级大的设备。
- 系统运行时间长的设备，如果设备启动时间间隔超过 1 分钟，则运行时间长的设备优先。
- 成员设备的 mac 地址，mac 地址小的设备优先。

通过如上规则选举出来的最优设备即为主用设备，其他设备为从设备。

10.16.3 堆叠缺省值

堆叠功能相关缺省值如下：

功能	缺省值
堆叠功能	禁用
堆叠成员优先级	1
堆叠成员编号	1
堆叠成员所属域 ID	1
堆叠成员发生堆叠协议报文时间间隔	3（单位秒）

10.16.4 配置堆叠

请在设备上进行以下配置：

步骤	配置	说明
1	JX(config)#**interface stack-port** *number*	创建堆叠端口。
2	JX(config-stack-port-*number*)#**add interface** ge 1/0/1	将物理端口添加值堆叠端口。
3	JX(config)#**hvs hello-interval** *time*	配置堆叠协议发生报文的时间间隔，单位秒。
4	JX(config)#**hvs master**	配置堆叠成员在堆叠协商时强制为主用设备。
5	JX(config)#**hvs member-id** *id*	配置堆叠设备的成员编号。
6	JX(config)#**hvs priority** *value*	配置堆叠成员设备的优先级。
7	JX(config)#**hvs switch-master member** { **myself** | *id* }	配置堆叠系统进行主备切换。

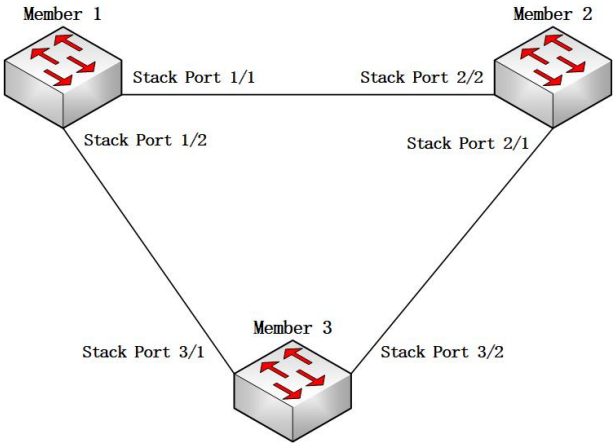
步骤	配置	说明
8	JX(config)#**hvs mode { normal | stack }**	配置堆叠成员设备进行模式切换，模式切换后设备将自动重启。
9	JX(config)#**show hvs member**	查看当前堆叠系统中成员以及主备角色
10	JX(config)#**show hvs topo**	查看整个堆叠系统中的拓扑信息
11	JX(config)#**show hvs interface**	查看本设备堆叠端口的信息
12	JX(config)#**show hvs config**	查看本设备堆叠相关配置

10.16.5 配置举例

如网需求

如下图的拓扑结构，如果 3 台设备需要组建链型拓扑结构的堆叠系统，需要进行如下配置。

图 10-18 3 台堆叠拓扑图



配置步骤

设备 1 的配置：

```
JX(config)#interface stack-port 1
JX(config-stack-port-1)#add interface ge 1/0/1
JX(config-stack-port-1)#add interface ge 1/0/2
JX(config-stack-port-1)#quit
JX(config)#interface stack-port 2
JX(config-stack-port-2)#add interface ge 1/0/3
JX(config-stack-port-2)#quit
JX(config)#hvs member-id 1
JX(config)#hvs mode stack
```

设备 2 的配置：

```
JX(config)#interface stack-port 1
JX(config-stack-port-1)#add interface ge 1/0/3
JX(config-stack-port-1)#quit
JX(config)#interface stack-port 2
JX(config-stack-port-2)#add interface ge 1/0/1
JX(config-stack-port-2)#add interface ge 1/0/2
JX(config-stack-port-2)#quit
JX(config)#hvs member-id 2
JX(config)#hvs mode stack
```

设备 3 的配置：

```
JX(config)#interface stack-port 1
JX(config-stack-port-1)#add interface ge 1/0/1
JX(config-stack-port-1)#quit
JX(config)#interface stack-port 2
JX(config-stack-port-2)#add interface ge 1/0/2
JX(config-stack-port-2)#quit
JX(config)#hvs member-id 3
JX(config)#hvs mode stack
```

待到 3 台设备重启完成之后，它们之间就会开始协商，然后可以通过命令查看堆叠是否建立完成。

此时在设备 3 上通过 **show hvs member** 的命令的结果如下：

```
JX(config)#show hvs member
SysId Role/ConfRole  State/Pri     Mac             Port0   Port0Mac
Port1   Port1Mac         Uptime
   1    master/no     done/1        f0f1:f2f3:0101    up     f0f1:f2f3:0101
up     f0f1:f2f3:0101    05h:37m:34s
   2    slave/no      done/1        f0f1:f2f3:0201    up     f0f1:f2f3:0201
up     f0f1:f2f3:0201    05h:37m:34s
 * 3    slave/no      done/1        f0f1:f2f3:0301    up     f0f1:f2f3:0301
up     f0f1:f2f3:0301    05h:37m:34s
```

通过 **show hvs topo** 的命令的结果如下：

```
JX(config)#show hvs topo
 Interface: stack-port-3/1,  State: up,  MAC: f0:f1:f2:f3:03:01
 Hop   SysId    Port0  State   MAC             Port1  State   MAC
 0     3        1      up      f0:f1:f2:f3:03:01  2     up
f0:f1:f2:f3:03:01
 1     1        1      up      f0:f1:f2:f3:01:01  2     up
f0:f1:f2:f3:01:01
 2     2        1      up      f0:f1:f2:f3:02:01  2     up
f0:f1:f2:f3:02:01
 Interface: stack-port-3/2,  State: up,  MAC: f0:f1:f2:f3:03:01
 Hop   SysId    Port0  State   MAC             Port1  State   MAC
```

```
 0    3        1     up     f0:f1:f2:f3:03:01  2     up
f0:f1:f2:f3:03:01
 1    2        1     up     f0:f1:f2:f3:02:01  2     up
f0:f1:f2:f3:02:01
 2    1        1     up     f0:f1:f2:f3:01:01  2     up
f0:f1:f2:f3:01:01
```

10.17 MAD

10.17.1 简介

当堆叠系统中成员设备之间链路端口，可能会导致堆叠系统分裂称为多个新的堆叠系统，这些新的堆叠系统有相同的 IP 地址等三层配置，会引起地址冲突，导致网络故障。MAD(Multi-Active Detection，多主冲突检测)协议用来进行堆叠分裂检测、冲突处理和故障恢复，以提高系统的可用性。

MAD 协议检测冲突主要有几种工作模式。

- 直连模式

直连模式下，需要在要进行多主冲突检测的堆叠成员设备之间分配一个额外的物理端口，用于收发 MAD 协议报文，可如图 10-18 建立拓扑。

- 代理模式

代理模式下，需要另外一个设备启用 MAD 协议，然后该设备与堆叠的各个成员设备之间建立跨设备聚合链路，然后在该聚合口上使能 MAD 协议，可如图 10-19 建立拓扑。

带外口检测模式

该模式下，只需要在带外口使能 MAD 协议，并且将所有堆叠成员设备的带外口连接到同一交换机，且保证所有带外口之间能正常通信即可。

图 10-19 MAD 直连模式

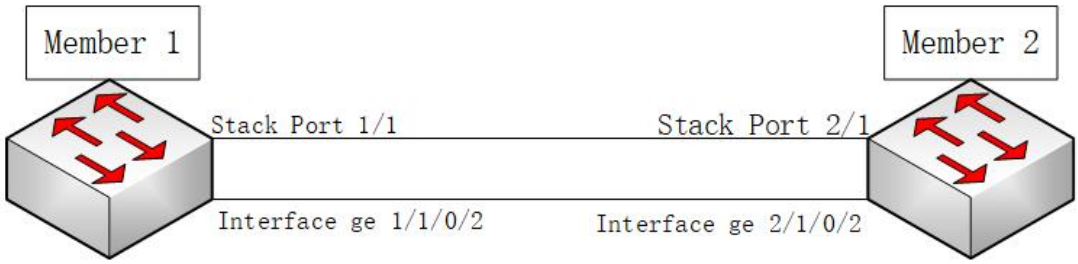
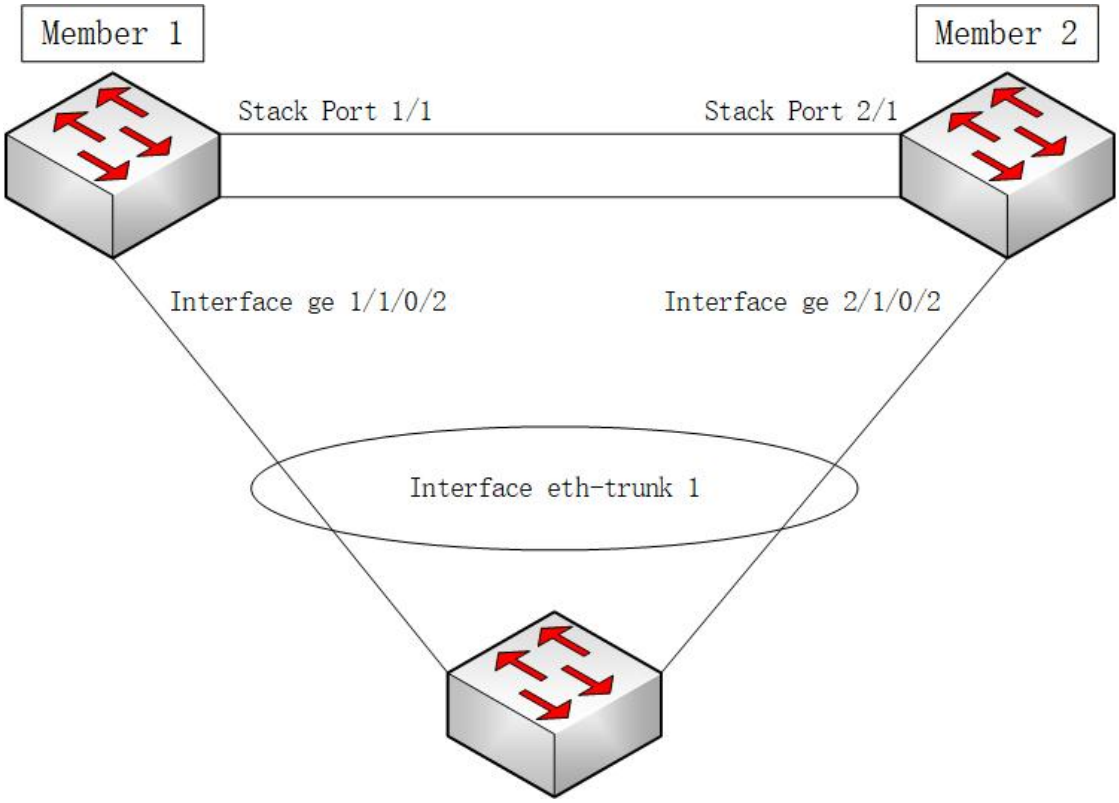


图 10-20 MAD 代理模式



10.17.2 配置准备

场景

- 备用管理 IP 地址

MAD 协议除了可以配置端口的工作模式外，可以为每个堆叠成员配置一个备份 IP 地址，该 IP 在 MAD 协议检测到多主时，自动将设备的带外口设置为该 IP 地址。

- 保留端口

MAD 协议在检测到多主时，进行比较之后，如果发现自己不是优选设备，则会将设备自己的端口进行 shutdown 操作。如果用户不想某些端口在 MAD 检测到多主时被 shutdown，可以进行保留端口的配置。

前提

无

10.17.3 配置 MAD

请在设备上进行如下配置：

步骤	配置	说明
1	JX(config)#**multi-active-detect backup ip address** *ip-address* **member** { **all** | *member-id*}	为指定的堆叠成员设备设置备份管理 IPv4 地址。
2	JX(config)#**multi-active-detect backup ipv6 address** *ipv6-address* **member** { **all** | *member-id*}	为指定的堆叠成员设备设置备份管理 IPv6 地址。
3	JX(config)#**interface GE 1/0/0/*** JX(config-ge-1/1/0/*)#**multi-active-detect mode direct**	配置该端口下 MAD 协议的工作模式。
4	JX(config)#**interface eth-trunk 1** JX(config-eth-trunk-1)#**multi-active-detect mode relay**	配置该聚合组端口下 MAD 协议的工作模式。
5	JX(config)#**interface meth 0/0/0** JX(config-meth-0/0/0)#**multi-active-detect** { **disable** | **enable** } JX(config)#**interface GE 1/0/0/*** JX(config-ge-1/1/0/*)#**multi-active-detect** { **disable** | **enable** }	在端口下使能或去使能 MAD 协议。
6	JX(config-meth-0/0/0)#**multi-active-exclude** { **disable** | **enable** }	在端口下使能 MAD 保留端口功能。
7	JX(config)#**show multi-active-detect config**	查看当前设备上的 MAD 相关配置。
8	JX(config)#**show multi-active-detect information**	查看你当前设备上 MAD 协议相关状态。

10.17.4 配置举例

组网需求

如图 10-19，首先需要进行堆叠向配置，让两台设备建立堆叠系统，然后在配置 MAD。

配置步骤

步骤 1 设备 1 的配置：

JX(config)#**interface stack-port 1**

```
JX(config-stack-port-1)#add interface ge 1/0/1
JX(config-stack-port-1)#quit
JX(config)#hvs member-id 1
JX(config)#hvs mode stack
```

步骤 2 设备 2 的配置：

```
JX(config)#interface stack-port 1
JX(config-stack-port-1)#add interface ge 1/0/1
JX(config-stack-port-1)#quit
JX(config)#hvs member-id 2
JX(config)#hvs mode stack
```

步骤 3 待到两台设备建立堆叠系统后，开始进行 MAD 协议相关配置。

```
JX(config)#interface GE 1/0/0/2
JX(config-ge-1/1/0/2))#stp disable
JX(config-ge-1/1/0/2))#multi-active-detect mode direct
JX(config-ge-1/1/0/2)#quit
JX(config)#interface ge 2/1/0/2
JX(config-ge-1/1/0/2))#stp disable
JX(config-ge-2/1/0/2))#multi-active-detect mode direct
```

检查配置

此时将堆叠端口端口，然后通过命令 **show multi-active-detect information** 查看 MAD 协议状态。

设备 1 状态：

```
JX(config)#show multi-active-detect information
  Current status : normal
  Direct detect information:
    ge-1/1/0/2     up
  Excluded ports(configurable):
    ge-1/1/0/5
  Excluded ports(can not be configured):
    ge-1/1/0/1
```

设备 2 状态：

```
JX(config)#show multi-active-detect information
  Current status : conflict
  Direct detect information:
    ge-2/1/0/2     up
  Excluded ports(configurable):
    ge-2/1/0/5
  Excluded ports(can not be configured):
    ge-2/1/0/1
```

10.18 NQA

10.18.1 简介

网络质量分析 NQA（Network Quality Analysis）是一种实时的网络性能探测和统计技术，可以对响应时间、网络抖动、丢包率等网络信息进行统计。NQA 能够实时监视网络 QoS，在网络发生故障时进行有效的故障诊断和定位。

10.18.2 配置准备

场景

为了使网络服务质量可见，使用户能够自行检查网络服务质量是否达到要求。需要在网络中部署探针设备对网络服务质量进行监控。

当设备提供 NQA 时，就不用部署专门的探针设备，可以有效的节约成本。NQA 可以实现对网络运行状况的准确测试，输出统计信息。

NQA 监测网络上运行的多种协议的性能，使用户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。

前提

无

10.18.3 NQA 的缺省配置

设备上 NQA 功能的缺省配置如下。

功能	缺省值
NQA 功能	禁用

10.18.4 配置 ICMP-echo 测试

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**nqa test-instance** *admin-name operate-tag* **type icmp-echo**	配置测试实例,类型为 icmp-echo
3	JX(nqa-admin-test-icmp-echo)#**destination ip** *ip-address*	配置目的 ip 地址

4	JX(nqa-admin-test-icmp-echo)#**frequency** *freq-ms*	配置测试频率，单位：毫秒
5	JX(nqa-admin-test-icmp-echo)#**probe count** *count*	配置每次测试探测的次数
6	JX(nqa-admin-test-icmp-echo)#**probe timeout** *timeout-time-ms*	配置每次探测等待的时间，单位：毫秒
7	JX(config)#**nqa schedule** *admin-name operate-tag* **start-time now life-time forever**	配置检测调度策略为立即开始，一直循环 注：该配置与绑定 timerange 策略二选一即可
8	JX(config)#**nqa schedule** *admin-name operate-tag* **time-range** *timer-ange-list*	配置检测调度策略绑定 timerange 注：该策略与立即开始策略二选一即可

10.18.5 配置 UDP-echo 测试

请在服务端设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**nqa server enable**	配置 nqa 服务端使能
3	JX(config)#**nqa server udp-connect** *ip-address udp-port*	配置 udp 服务端

请在客户端设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**nqa test-instance** *admin-name operate-tag* **type udp-echo**	配置测试实例,类型为 udp-echo
3	JX(nqa-admin-test-udp-echo)#**destination ip** *ip-address*	配置目的 ip 地址
4	JX(nqa-admin-test-udp-echo)#**destination port** *udp-port*	配置目的 udp 端口号
5	JX(nqa-admin-test-udp-echo)#**frequency** *freq-ms*	配置测试频率，单位：毫秒
6	JX(nqa-admin-test-udp-echo)#**probe count** *count*	配置每次测试探测的次数
7	JX(nqa-admin-test-udp-echo)#**probe timeout** *timeout-time-ms*	配置每次探测等待的时间，单位：毫秒

8	JX(config)#**nqa schedule** *admin-name operate-tag* **start-time now life-time forever**	配置检测调度策略为立即开始，一直循环 注：该配置与绑定 timerange 策略二选一即可
9	JX(config)#**nqa schedule** *admin-name operate-tag* **time-range** *timer-ange-list*	配置检测调度策略绑定 timerange 注：该策略与立即开始策略二选一即可

10.18.6 配置 TCP 测试

请在服务端设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**nqa server enable**	配置 nqa 服务端使能
3	JX(config)#**nqa server tcp-connect** *ip-address tcp-port*	配置 tcp 服务端

请在客户端设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**nqa test-instance** *admin-name operate-tag* **type tcp**	配置测试实例,类型为 tcp
3	JX(nqa-admin-test-tcp)#**destination ip** *ip-address*	配置目的 ip 地址
4	JX(nqa-admin-test-tcp)#**destination port** *tcp-port*	配置目的 tcp 端口号
5	JX(nqa-admin-test-tcp)#**frequency** *freq-ms*	配置测试频率，单位：毫秒
6	JX(nqa-admin-test-tcp)#**probe count** *count*	配置每次测试探测的次数
7	JX(nqa-admin-test-tcp)#**probe timeout** *timeout-time-ms*	配置每次探测等待的时间，单位：毫秒
8	JX(config)#**nqa schedule** *admin-name operate-tag* **start-time now life-time forever**	配置检测调度策略为立即开始，一直循环 注：该配置与绑定 timerange 策略二选一即可
9	JX(config)#**nqa schedule** *admin-name operate-tag* **time-range** *timer-ange-list*	配置检测调度策略绑定 timerange 注：该策略与立即开始策略二选一即可

10.18.7 配置 DNS 测试

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**nqa test-instance** *admin-name operate-tag* **type dns**	配置测试实例,类型为 dns
3	JX(nqa-admin-test-dns)#**destination ip** *ip-address*	配置 DNS 服务器地址
4	JX(nqa-admin-test-dns)#**resolve-target** *domain-name*	配置需要解析的域名
5	JX(nqa-admin-test-tcp)#**frequency** *freq-ms*	配置测试频率，单位：毫秒
6	JX(nqa-admin-test-tcp)#**probe count** *count*	配置每次测试探测的次数
7	JX(nqa-admin-test-tcp)#**probe timeout** *timeout-time-ms*	配置每次探测等待的时间，单位：毫秒
8	JX(config)#**nqa schedule** *admin-name operate-tag* **start-time now life-time forever**	配置检测调度策略为立即开始，一直循环 注：该配置与绑定 timerange 策略二选一即可
9	JX(config)#**nqa schedule** *admin-name operate-tag* **time-range** *timer-ange-list*	配置检测调度策略绑定 timerange 注：该策略与立即开始策略二选一即可

10.18.8 配置 HTTP 测试

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**nqa test-instance** *admin-name operate-tag* **type http**	配置测试实例,类型为 http
3	JX(nqa-admin-test-http)#**destination ip** *ip-address*	配置 HTTP 服务器地址
4	JX(nqa-admin-test-http)#**url** *url-path*	配置探测的 URL
5	JX(nqa-admin-test-http)#**frequency** *freq-ms*	配置测试频率，单位：毫秒

6	JX(nqa-admin-test-http)#**probe count** *count*	配置每次测试探测的次数
7	JX(nqa-admin-test-http)#**probe timeout** *timeout-time-ms*	配置每次探测等待的时间，单位：毫秒
8	JX(config)#**nqa schedule** *admin-name operate-tag* **start-time now life-time forever**	配置检测调度策略为立即开始，一直循环 注：该配置与绑定 timerange 策略二选一即可
9	JX(config)#**nqa schedule** *admin-name operate-tag* **time-range** *timer-ange-list*	配置检测调度策略绑定 timerange 注：该策略与立即开始策略二选一即可

10.18.9 配置 FTP 测试

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**nqa test-instance** *admin-name operate-tag* **type ftp**	配置测试实例,类型为 ftp
3	JX(nqa-admin-test-ftp)#**destination ip** *ip-address*	配置 FTP 服务器地址
4	JX(nqa-admin-test-ftp)#**username** *ftp-username*	配置 FTP 用户名
5	JX(nqa-admin-test-ftp)#**password** *ftp-password*	配置 FTP 用户密码
6	JX(nqa-admin-test-ftp)#**filename** *ftp-filename*	配置用于探测的文件名
7	JX(nqa-admin-test-ftp)#**operation （get | put）**	（可选）配置 ftp 操作，默认：get
5	JX(nqa-admin-test-ftp)#**frequency** *freq-ms*	配置测试频率，单位：毫秒
6	JX(nqa-admin-test-ftp)#**probe count** *count*	配置每次测试探测的次数
7	JX(nqa-admin-test-ftp)#**probe timeout** *timeout-time-ms*	配置每次探测等待的时间，单位：毫秒
8	JX(config)#**nqa schedule** *admin-name operate-tag* **start-time now life-time forever**	配置检测调度策略为立即开始，一直循环 注：该配置与绑定 timerange 策略二选一即可
9	JX(config)#**nqa schedule** *admin-name operate-tag* **time-range** *timer-ange-list*	配置检测调度策略绑定 timerange 注：该策略与立即开始策略二选一即可

10.18.10 配置 SNMP 测试

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**nqa test-instance** *admin-name operate-tag* **type snmp**	配置测试实例,类型为 snmp
3	JX(nqa-admin-test-snmp)#**destination ip** *ip-address*	配置 SNMP 服务器地址
4	JX(nqa-admin-test-snmp)#**frequency** *freq-ms*	配置测试频率，单位：毫秒
5	JX(nqa-admin-test-snmp)#**probe count** *count*	配置每次测试探测的次数
6	JX(nqa-admin-test-snmp)#**probe timeout** *timeout-time-ms*	配置每次探测等待的时间，单位：毫秒
7	JX(config)#**nqa schedule** *admin-name operate-tag* **start-time now life-time forever**	配置检测调度策略为立即开始，一直循环 注：该配置与绑定 timerange 策略二选一即可
8	JX(config)#**nqa schedule** *admin-name operate-tag* **time-range** *timer-ange-list*	配置检测调度策略绑定 timerange 注：该策略与立即开始策略二选一即可

10.18.11 配置测试历史记录功能

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局模式。
2	JX(config)#**nqa test-instance** *admin-name operate-tag* { **type** { **icmp-echo** | **udp-echo** | **tcp** | **dns** | **http** | **ftp** | **snmp** } }	进入任意 nqa 测试实例
3	JX(nqa-admin-test-icmp-echo)#**history-record enable**	使能历史记录功能
	JX(nqa-admin-test-icmp-echo)#**history-record keep-time** *time-minutes*	（可选）配置历史记录保存时间，单位：分钟

10.18.12 配置测试统计功能

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局模式。
2	JX(config)#**nqa test-instance** *admin-name operate-tag* { **type** { **icmp-echo** | **udp-echo** | **tcp** | **dns** | **http** | **ftp** | **snmp** } }	进入任意 nqa 测试实例
3	JX(nqa-admin-test-icmp-echo)#**statistics enable**	使能测试统计功能
4	JX(nqa-admin-test-icmp-echo)#**statistics interval** *interval-minutes*	（可选）配置统计周期

10.18.13 配置测试告警功能

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局模式。
2	JX(config)#**nqa test-instance** *admin-name operate-tag* { **type** { **icmp-echo** | **udp-echo** | **tcp** | **dns** | **http** | **ftp** | **snmp** } }	进入任意 nqa 测试实例
3	SwitchA(nqa-admin-test-icmp-echo)#**reaction trap probe-failure** *fail-count*	每次探测连续发生 n 次失败时，发送 trap
4	SwitchA(nqa-admin-test-icmp-echo)#**reaction trap test-failure** *fail-count*	每次探测累积发生 n 次失败时，发送 trap
5	SwitchA(nqa-admin-test-icmp-echo)#**reaction trap test-complete**	每次探测完成，发送 trap

10.18.14 检查配置

配置完成后，请在设备上执行以下命令检查配置结果。

序号	检查项	说明
1	JX#**show nqa config**	查看 nqa 配置信息。

10.18.15 维护

用户可以通过以下命令维护 NQA 特性。

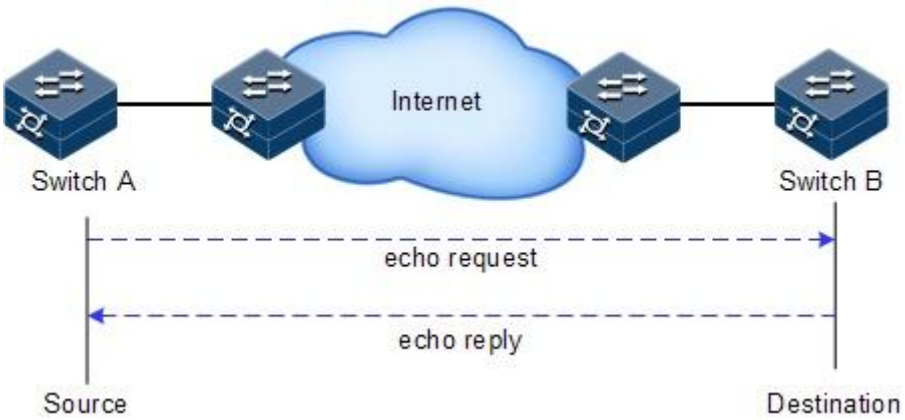
命令	描述
JX#**show nqa agent**	查看 nqa 客户端信息
JX#**show nqa server**	查看 nqa 服务端信息
JX#**show nqa result** *admin-name operate-tag*	查看 nqa 客户端测试结果
JX#**show nqa history** *admin-name operate-tag*	查看 nqa 客户端测试结果历史
JX#**show nqa statistics** *admin-name operate-tag*	查看 nqa 客户端测试结果统计

10.18.16 配置 ICMP-echo 测试功能示例

组网需求

如图 10-9 所示，在 Switch A 使能 NQA icmp-echo 探测，探测 SwitchA 与 SwitchB 之间的 IP 层连通性。

图 10-21 配置 ICMP-echo 测试功能组网示意图



配置步骤

步骤 1 配置 IP 层互通。

配置 Switch A。

```
JX#hostname SwitchA
SwitchA#config
```

```
SwitchA(config)#vlan 10
SwitchA(config-vlan-10)#quit
SwitchA(config)#interface ge 1/0/1
SwitchA(config-ge-1/0/1)#port hybrid vlan 10 tagged
SwitchA(config-ge-1/0/1)#quit
SwitchA(config)#interface vlan 10
SwitchA(config-vlanif-10)#ip address 10.1.1.1/24
```

配置 Switch B。

```
JX#hostname SwitchB
SwitchB#config
SwitchB(config)#vlan 10
SwitchB(config-vlan-10)#quit
SwitchB(config)#interface ge 1/0/1
SwitchB(config-ge-1/0/1)#port hybrid vlan 10 tagged
SwitchB(config-ge-1/0/1)#quit
SwitchB(config)#interface vlan 10
SwitchB(config-vlanif-10)#ip address 10.1.1.2/24
```

步骤 2 配置 NQA 的 icmp-echo 探测实例。

配置 Switch A。

```
SwitchA(config)#nqa test-instance admin test type icmp-echo
SwitchA(nqa-admin-test-icmp-echo)#destination ip 10.1.1.2
SwitchA(nqa-admin-test-icmp-echo)#frequency 10000
SwitchA(nqa-admin-test-icmp-echo)#probe count 3
SwitchA(nqa-admin-test-icmp-echo)#history-record enable
SwitchA(nqa-admin-test-icmp-echo)#reaction trap probe-failure 3
SwitchA(config)#nqa schedule admin test start-time now life-time forever
```

检查结果

通过 **show nqa config** 命令查看本地配置是否正确。

```
SwitchA(config)#show nqa config
!
nqa test-instance admin test type icmp-echo
 frequency 10000
 history-record enable
 statistics enable
 reaction trap probe-failure 3
 probe count 3
 destination ip 10.1.1.2
nqa schedule admin test start-time now life-time forever
……
```

通过 **show nqa history admin test** 命令查看测试结果历史。

```
SwitchA(config)#show nqa history admin test
 Index    Response          Status            Time
--------------------------------------------------------------------------------
 9        11                successed         2023-11-10 17:05:26
```

```
8       7               successed       2023-11-10 17:05:26
7       10              successed       2023-11-10 17:05:26
6       12              successed       2023-11-10 17:05:16
5       7               successed       2023-11-10 17:05:16
4       9               successed       2023-11-10 17:05:16
3       11              successed       2023-11-10 17:05:05
2       8               successed       2023-11-10 17:05:05
1       9               successed       2023-11-10 17:05:05
--------------------------------------------------------------------------
```

10.19 PATCH 补丁功能

10.19.1 简介

patch 功能允许在不更换系统软件的情况下，通过补丁方式在线修复系统漏洞，解决软件问题。系统最多支持 32 个补丁。

10.19.2 配置准备

场景

系统发现的问题，可以通过补丁方式进行修复，无需重启设备。

前提

已准备好补丁文件。

10.19.3 加载补丁

设备上加载一个补丁的配置如下。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**patch 1 load** *FILENAME*	加载一个 patch 补丁

10.19.4 激活补丁

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。

2	JX(config)#**patch 1 active**	激活 patch 1 补丁

10.19.5 去激活补丁

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**patch 1 deactive**	去激活 patch 1 补丁，使其失效

10.19.6 删除补丁

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**patch 1 delete**	删除 patch 1 补丁，使其失效

10.19.7 维护

用户可以通过以下命令维护 PATCH 特性。

命令	描述
JX#**show patch information**	查看设备的补丁情况

10.19.8 PATCH 示例

组网需求

通过打补丁的方式修复系统功能。

设备和 PC 通过 SNMP 接口互联，PC 上开启 TFTP 软件，以便从设备下载补丁文件。

配置步骤

步骤 1 通过 tftp 协议下载补丁文件到设备

```
JX(config)#tftp get 22.1.1.1 libmspaaa_patch.pat localfile
libmspaaa_patch.pat
```

步骤 2 加载补丁。

```
JX(config)#patch 1 load libmspaaa_patch.pat
```

步骤 3 激活补丁

```
JX(config)#patch 1 active
```

检查结果

通过 **show** patch information 命令查看 patch 状态：

```
JX(config)#show patch information
 Patch max number     : 32
 Patch current number  : 1
 Seq  ProcName   State  Type  Valid  EffetiveTime       PatchName
 ------------------------------------------------------------------------
 1    mspaaa   ACTIVE  REP   YES    2024-03-20 17:08:52  libmspaaa_patch.pat
```

10.20 定时备份配置功能

10.20.1 简介

定时备份配置功能目前是基于自动上传配置绑定 timerange 实现。

10.20.2 配置准备

场景

定时备份设备的当前配置到指定服务器。

前提

设备和 pc 可以 ping 通，且 pc 端 tftp 或者 ftp 服务器已打开。

10.20.3 配置 timerange

设备上配置一个 timerange list，并配置相应的规则配置如下。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**time-range list** *id*	创建一个 timerange 列表，进入节点

3	JX(config-timerange-*)#**time-range** *timerange-id* **absolute from** *hour:minute:second year/month/monthday* **to** *hour:minute:second year/month/monthday* JX(config-timerange-*)#**time-range** *timerange-id* **everyhour** *minute:second* **to** *minute:second* JX(config-timerange-*)#**time-range** *timerange-id* **everyday** *hour:minute:second* **to** *hour:minute:second* JX(config-timerange-*)#**time-range** *timerange-id* **everyweek** *hour:minute:second* { **mon** | **tue** | **wed** | **thu** | **fri** | **sat** | **sun** } **to** { **mon** | **tue** | **wed** | **thu** | **fri** | **sat** | **sun** } JX(config-timerange-*)#**time-range** *timerange-id* **everymonth** *hour:minute:second monthday* **to** *hour:minute:second monthday* JX(config-timerange-*)#**time-range** *timerange-id* **everyweekend** *hour:minute:second* **to** *hour:minute:second* JX(config-timerange-*)#**time-range** *timerange-id* **everyyear** *hour:minute:second month/monthday* **to** *hour:minute:second month/monthday*	创建时间段，可应用于自动上传。 相对时间段（周期时间段）：是指由 everyxxx 描述的以一周为间隔的周期性时间，同时可以依靠绝对时间段指明时间范围有效日期。 绝对时间段：是指由 absolute 指定的特定的时间范围。可以通过绝对时间段指定的时间范围来限制相对时间段（周期时间段）在什么时间范围生效。 如果该时间段配置了多个生效时间，生效原则为：周期时间段之间取“或”，周期时间段和绝对时间段之间取“与”

10.20.4 配置自动上传

请在设备上进行以下配置。

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**auto-upload start**	启动自动上传功能
3	JX(config)#**auto-upload tftp server** {*ipv4-address*|*ipv6-address*} *remotefile* {**config**|**running-config**|**log**} [**interval** *interval-value*| **port** *port-number*|**time-range** *list-number*]*	配置自动上传 tftp 服务器属性并指定上传的文件，绑定的 timerange 列表
4	JX(config)#**auto-upload ftp server** {*ipv4-address*|*ipv6-address*} *username password remotefile* {**config**|**running-config**|**log**} [**interval** *interval-value*| **port** *port-number*|**time-range** *list-number*]*	配置自动上传 ftp 服务器属性并指定上传的文件，绑定的 timerange 列表

10.20.5 检查配置

配置完成后，请在设备上执行以下命令检查配置结果。

命令	描述
JX#show auto-upload server	查看配置的自动上传表项
JX#show auto-upload config	查看自动上传的配置

10.20.6 定时备份配置示例

组网需求

通过 tftp 方式自动备份设备当前配置到 PC。

设备和 PC 通过 SNMP 接口互联，PC 上开启 TFTP 软件。

配置步骤

步骤 1 配置 timerange list 1 并配置规则 1（以每小时定时备份举例）

```
JX(config)#time-range list 1。
JX(config-timerange-1)#time-range 1 everyhour 3:10 to 13:10
```

步骤 2 配置自动上传属性,启动自动上传功能，创建表项（tftp 服务器地址 192.168.62.1,本地文件为运行配置文件，上传到服务器的文件为 config.txt），并绑定 timerange list 1

```
JX(config)#auto-upload start
JX(config)#auto-upload tftp server 192.168.62.1 config.txt running-config
time-range 1
```

步骤 3 配置完成后，每小时 3 分 10 秒设备会定时上传运行配置文件到 PC 端 tftp 服务器一次

检查结果

通过 **show auto-upload server** 命令查看所配置的自动上传属性：

```
JX(config)#show auto-upload server
 Type  Server               Port    Interval(min) Localfile         Vpn
TimerangeId
------------------------------------------------------------------------
---------------------------
 tftp  192.168.62.1          69     10          running-config     n/a
1
------------------------------------------------------------------------
---------------------------
JX(config)#
```

11 L3VPN 业务

本章介绍 L3VPN 业务特性的基本原理和配置过程，并提供相关的配置案例。

- L3VPN 基础配置

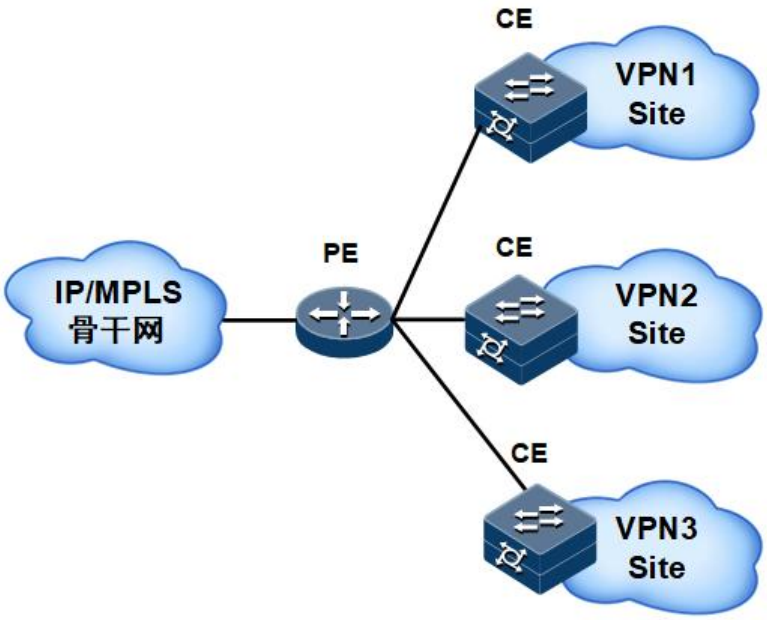
11.1 L3VPN 基础配置

11.1.1 简介

本章提到的 L3VPN 功能实现的是 MCE 功能，MCE 是 Multi-VPN-Instance CE（多实例 CE）的简称，具有 MCE 功能的设备可以在 VPN 组网应用中承担多个 VPN 实例的 CE 功能，减少用户网络设备的投入。

VPN 以隧道的方式解决了在公网中传送私网数据的问题，但传统的 VPN 架构要求每个 VPN 实例单独使用一个 CE 与 PE 相连，如图 11-1 所示。

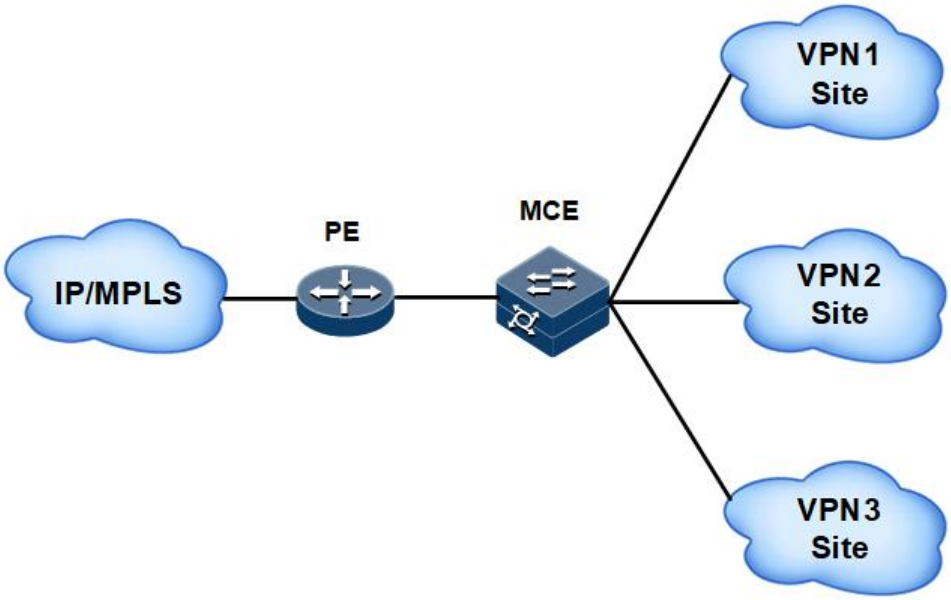
图 11-1 部署 MCE 前的组网



随着用户业务的不断细化和安全需求的提高，很多情况下一个私有网络内的用户需要划分成多个 VPN，不同 VPN 用户间的业务需要完全隔离。此时，为每个 VPN 单独配置一台 CE 将增加用户的设备开支和维护成本；而多个 VPN 共用一台 CE，使用同一个路由转发表，又无法保证数据的安全性。

使用 MCE 技术，可以有效解决多 VPN 网络带来的数据安全与网络成本之间的矛盾。如图 11-2 所示。

图 11-2 部署 MCE 后的组网



MCE 将 PE 的部分功能扩展到 CE 设备，通过将不同的接口与 VPN 绑定，并为每个 VPN 创建和维护独立的路由转发表（Multi-VRF）。这样不但能够隔离私网内不同 VPN 的报文转发路径，而且通过与 PE 间的配合，也能够将每个 VPN 的路由正确发布至对端 PE，保证 VPN 报文在公网内的传输。

11.1.2 配置准备

场景

在 MCE 上配置 VPN 实例（每个业务配置一个 VPN 实例）。

前提

VPN 实例名，VLAN 接口

11.1.3 创建 VPN 实例

步骤	配置	说明
1	`JX#config`	进入全局配置模式。
2	`JX(config)#ip vpn-instance NAME`	创建 VPN 实例。
3	`JX(config-vpn-instance-*)#ipv4-family`	使能 VPN 实例的 IPv4 地址族。
4	`JX(config-vpn-instance-*-af-ipv4)#route-distinguisher 9:9`	配置 RD。
5	`JX(config-vpn-instance-*-af-ipv4)#vpn-target 9:9 both`	配置当前 VPN 实例与 VPN target 进行关联。

11.1.4 配置 VLAN 接口绑定 VPN 实例

步骤	配置	说明
1	JX#**config**	进入全局配置模式。
2	JX(config)#**interface vlan** *vlan-id*	进入 VLAN 接口配置模式。
3	JX(config-vlanif-*)#**ip vpn-instance** *NAME*	配置 VLAN 接口上绑定 VPN 实例

11.1.5 检查配置

配置完成后，请在设备上执行以下命令检查配置结果。

序号	检查项	说明
1	JX#**show ip vpn-instance**	显示所有的 VPN 实例信息。

11.1.6 配置 VLAN 接口绑定 VPN 示例

组网需求

配置交换机设备 VLAN 接口绑定 VPN

配置步骤

步骤 1 创建 VLAN，并将接口加入 VLAN。

```
JX#config
JX(config)#vlan 10
JX(config)#interface ge 1/0/1
JX(config-ge-1/0/1)#port hybrid pvid 10
JX(config-ge-1/0/1)#port hybrid vlan 10 untagged
JX(config-ge-1/0/1)#quit
```

步骤 2 在交换机设备上创建三层接口。

```
JX(config)#interface vlan 10
```

步骤 3 创建 VPN 实例。

```
JX(config)#ip vpn-instance 123
JX(config-vpn-instance-123)#ipv4-family
JX(config-vpn-instance-123-af-ipv4)#route-distinguisher 9:9
JX(config-vpn-instance-123-af-ipv4)#vpn-target 9:9 both
JX(config-vpn-instance-123-af-ipv4)#quit
JX(config-vpn-instance-123)#quit
```

步骤 4 VLAN 绑定 VPN 实例。

```
JX{ config-vlanif-10 }#ip vpn-instance 123
```

检查结果

通过 **show ip vpn-instance verbose** 命令查看 vpn 信息。

```
JX(config)#show ip vpn-instance verbose
------------------------------------------------------------
VPN-Instance Name       : 123
VPN-Instance ID        : 2
 Creation Time         : 1970-01-01 08:00:02
 Description           : --
 Ipv4 Family
  Route Distinguisher   : 9:9
  Multicast            : disable
  Apply Mode           : Route
  Shared Group         : --
  Sourece Interface     : --
  Binding              : --
  Vpn Target Import     : 9:9
  Vpn Target Export     : 9:9

------------------------------------------------------------
```

12 附录

本章介绍了本文档中涉及的缩略语和术语。

- 术语
- 缩略语

12.1 术语

B

半双工	Half-duplex	半双工指在同一时间只能在同一个方向进行的双向通信。一方在接受信息，而另一方在发送信息的通信，即为半双工。
保护地线	Protection Ground Wire	连接设备和保护地的线缆，通常为黄绿相间的同轴线缆。
标签	Label	线缆、机箱以及警告标识。

D

单模光纤	Single Mode Fiber	单模光纤指在同一条光纤中只能传输单个模式光信号的光纤。
第一英里以太网	EFM（Ethernet in the First Mile）	遵循 IEEE 802.3ah 协议，是一种链路级以太网 OAM 技术，针对两台直连设备之间的链路，提供链路连通性检测功能、链路故障监控功能、远端故障通知功能等。EFM 主要用于用户接入的网络边缘的以太网链路。
电气和电子工程师协会	IEEE（Institute of Electrical and Electronics Engineers）	一个国际性的电子技术与信息科学工程师的协会，是世界上最大的专业技术组织之一（成员人数）。

动态 ARP 检测	DAI（Dynamic ARP Inspection）	一种能够验证网络中 ARP 地址解析协议数据报的安全特性。通过 DAI，网络管理员能够拦截、记录和丢弃具有无效 MAC 地址/IP 地址绑定的 ARP 数据包，以防止网络中常见的 ARP 欺骗攻击。
动态主机配置协议	DHCP（Dynamic Host Configuration Protocol）	在网络中动态分配 IP 地址的技术。它可以自动地为网络中的所有客户端分配 IP 地址，从而减轻管理员的工作量，实现 IP 地址集中管理。
多模光纤	Multi-mode Fiber	多模光纤指在同一条光纤中能够传输多个模式光的光纤。

F

访问控制列表	ACL（Access Control List）	访问控制列表是由 permit | deny 语句组成的一系列有顺序的规则，设备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。
服务质量	QoS（Quality of Service）	一种网络安全机制，是用来解决网络延迟和阻塞问题的一种技术。当网络过载或拥塞时，QoS 能确保重要业务不被延迟或丢弃，同时保证网络高效运行。

G

故障转移	Failover	提供了一种端口联动方案，可以扩展链路备份的范围。该功能通过监控上行链路并对下行链路进行同步设置，使上层设备的故障迅速传达给下层，从而触发主备切换，避免因上行链路故障无法被下层设备感知而出现的流量丢失。
光纤配线架	ODF（Optical Distribution Frame）	光缆和光通信设备之间的配线连接设备。它是光传输系统中一个重要的配套设施，主要用于光缆终端的光纤熔接、光连接器安装、光路的调接、多余尾纤的存储及光缆的保护等。

H

互联网编号分配委员会	IANA（Internet Assigned Numbers Authority）	主要职能是分配和维护在互联网技术标准（或协议）中的唯一编码和数值，如 IP 地址，组播地址等。

J

激光器自动关断	ALS（Automatic Laser Shutdown）	当光接口的光纤被拔出或光发射器的输出功率过大时，光接口自动关断激光器，防止出现维护或运行风险。

简单网络管理协议	SNMP（Simple Network Management Protocol）	由 IETF（Internet Engineering Task Force，互联网工程任务组）为了解决 Internet 中网络设备的管理问题而提出的一套网络管理协议。SNMP 可以使一个网管系统远程管理所有支持 SNMP 的网络设备，包括监视网络状态、修改网络设备配置、接收网络事件告警等。它是目前 TCP/IP 网络中应用最广泛的网络管理协议。
简单网络时间协议	SNTP（Simple Network Time Protocol）	SNTP 主要用于同步网络中的设备时间。

K

开放最短路径优先	OSPF（Open Shortest Path First）	一种内部网关动态路由协议，用于在单一自治系统内决策路由。
快速生成树协议	RSTP（Rapid Spanning Tree Protocol）	RSTP 是为了弥补 STP（Spanning Tree Protocol，生成树协议）收敛速度慢的不足而开发的。RSTP 在 STP 的基础上进行了改进，实现了网络拓扑快速收敛。

L

链路汇聚控制协议	LACP（Link Aggregation Control Protocol）	一种实现链路动态汇聚的协议。LACP 协议通过 LACPDU（Link Aggregation Control Protocol Data Unit，链路汇聚控制协议数据单元）与对端交互信息。
链路聚合	Link Aggregation	通过将多个物理以太网端口聚合在一起形成一个逻辑上的聚合组，并把同一聚合组内的多条物理链路视为一条逻辑链路。链路聚合可以实现流量在聚合组各成员端口之间负载分担，在有效的提高了设备之间链路可靠性的同时，还在不进行硬件升级的条件下增大了带宽。

M

密码认证协议	PAP（Password Authentication Protocol）	PPP 认证协议中的密码认证协议，是一种 2 次握手协议，在网络上采用明文方式传输用户名和密码。

Q

全双工	Full-duplex	通信链路上双方可以同时发送和接收数据。
QinQ	Stacked VLAN 或 Double VLAN	是 802.1Q 的扩展，IEEE 在 802.1ad 标准中定义。在运营商的骨干网络（公网）中，报文携带两层的 VLAN Tag：公网 VLAN Tag 和私网 VLAN Tag。公网中私网 VLAN Tag 被当作报文数据部分进行传输。可以分为基本 QinQ 和灵活 QinQ 两种类型。

S

生成树协议	STP（Spanning Tree Protocol）	STP 可以在局域网中消除网络环路，并实现数据链路备份功能。在逻辑上阻断环路，防止广播风暴的产生。当未阻断的链路出现故障时，阻断的链路会被重新激活，充当备份线路的功能。

T

挑战握手认证协议	CHAP（Challenge-Handshake Authentication Protocol）	PPP 认证协议中的挑战握手认证协议，是一种 3 次握手验证协议，只在网络上传输用户名，而不传输用户密码。

V

VLAN 映射	VLAN Mapping	主要功能是将以太网业务报文中的私网 VLAN Tag 替换为运营商的 VLAN Tag，使其按照运营商的 VLAN 转发规则进行传输。在报文从运营商网络转发到对端用户私网时，再按照同样的规则将 VLAN Tag 恢复为原有的用户私网 VLAN Tag，使报文正确到达目的地。

W

网络时间协议	NTP（Network Time Protocol）	由 RFC1305 定义的时间同步协议，用来在分布式时间服务器和客户端之间进行时间同步。使用 NTP 的目的是快速对网络内所有具有时钟的设备进行时钟同步，从而使设备能够提供基于统一时间的不同应用。同时 NTP 还能保证很高的精度（误差在 10ms 左右）。

X

虚拟局域网	VLAN（Virtual Local Area Network）	是为解决以太网的广播问题和安全性而提出的一种协议。是一种通过将局域网内的设备逻辑地而不是物理地划分成一个个网段，从而实现多个互不影响的虚拟工作组的二层隔离技术。

Y

以太网承载 PPP 协议	PPPoE（Point-to-point Protocol over Ethernet）	通过 PPPoE 协议，远端接入设备能够实现对每个接入用户的控制和计费。

以太网环网保护倒换	ERPS（Ethernet Ring Protection Switching）	基于 ITU-T G.8032 标准的 APS（Automatic Protection Switching，自动保护倒换）协议，是一种专门应用于以太网环的链路层协议，正常情况下，它在以太网环中能够防止数据环路引起的广播风暴，当以太网环上链路或设备故障时，能迅速切换到备份链路，保证业务快速恢复。
因特网工程任务组	IETF（Internet Engineering Task Force）	成立于 1985 年底，是全球互联网最具权威的技术标准化组织，主要任务是负责互联网相关技术规范的研发和制定。
远程用户拨号认证系统	RADIUS（Remote Authentication Dial In User Service）	网络中对用户进行认证和计费的协议。

Z

自动保护倒换	APS（Automatic Protection Switched）	自动保护倒换技术是通过对传输线路变化的实时监视、告警信息的自动分析，能够及时发现故障及隐患，在出现严重故障时，快速将工作通道自动切换到备用通道，在极短的时间内恢复通信，完成对故障的快速反应和恢复机制。
自协商	Auto-Negotiation	自协商指接口根据协商结果自动选择接口速率和双工模式。

12.2 缩略语

A

AAA	Authentication，Authorization and Accounting	认证、授权和计费
ABR	Area Border Router	区域边界路由器
AC	Alternating Current	交流电
ACL	Access Control List	访问控制列表
ANSI	American National Standards Institute	美国国家标准协会
APS	Automatic Protection Switching	自动保护倒换
ARP	Address Resolution Protocol	地址解析协议
AS	Autonomous System	自治系统
ASCII	American Standard Code for Information Interchange	美国信息交换标准码

ASE	Autonomous System External	外部自治系统
ATM	Asynchronous Transfer Mode	异步传输模式
AWG	American Wire Gauge	美国线缆标准

B

BC	Boundary Clock	边界时钟
BDR	Backup Designated Router	备份指定路由器
BITS	Building Integrated Timing Supply System	通信楼综合定时供给系统
BOOTP	Bootstrap Protocol	自举协议
BPDU	Bridge Protocol Data Unit	网桥协议数据单元
BTS	Base Transceiver Station	基站收发信台

C

CAR	Committed Access Rate	承诺访问速率
CAS	Channel Associated Signaling	随路信令
CBS	Committed Burst Size	承诺突发数据量
CE	Customer Edge	用户网边缘
CHAP	Challenge Handshake Authentication Protocol	挑战握手认证协议
CIDR	Classless Inter-Domain Routing	无类域间路由
CIR	Committed Information Rate	承诺信息速率
CIST	Common Internal Spanning Tree	公共内部生成树
CLI	Command Line Interface	命令行接口
CoS	Class of Service	服务等级
CPU	Central Processing Unit	中央处理器
CRC	Cyclic Redundancy Check	循环冗余校验
CSMA/CD	Carrier Sense Multiple Access/Collision Detection	载波侦听多路访问
CST	Common Spanning Tree	公共生成树

D

DAI	Dynamic ARP Inspection	动态 ARP 检测

DBA	Dynamic Bandwidth Allocation	动态带宽分配
DC	Direct Current	直流电
DHCP	Dynamic Host Configuration Protocol	动态主机配置协议
DiffServ	Differentiated Service	区分服务
DNS	Domain Name System	域名系统
DS	Differentiated Services	差分服务
DSL	Digital Subscriber Line	数字用户线

E

EAP	Extensible Authentication Protocol	PPP 扩展认证协议
EAPoL	EAP over LAN	基于局域网的 EAP
EFM	Ethernet in the First Mile	第一英里以太网
EMC	Electro Magnetic Compatibility	电磁兼容
EMI	Electro Magnetic Interference	电磁干扰
EMS	Electro Magnetic Susceptibility	电磁敏感性
ERPS	Ethernet Ring Protection Switching	以太网环网保护倒换
ESD	Electro Static Discharge	静电释放
EVC	Ethernet Virtual Connection	以太网虚连接

F

FCS	Frame Check Sequence	帧校验序列
FE	Fast Ethernet	快速以太网
FIFO	First Input First Output	先入先出
FTP	File Transfer Protocol	文件传输协议

G

GARP	Generic Attribute Registration Protocol	通用属性注册协议
GE	Gigabit Ethernet	千兆以太网
GMRP	GARP Multicast Registration Protocol	基于 GARP 的组播注册协议
GPS	Global Positioning System	全球定位系统
GVRP	Generic VLAN Registration Protocol	通用 VLAN 注册协议

H

HDLC	High-level Data Link Control	高级数据链路控制
HTTP	Hyper Text Transfer Protocol	超文本传输协议

I

IANA	Internet Assigned Numbers Authority	互联网编号分配委员会
ICMP	Internet Control Message Protocol	Internet 控制报文协议
IE	Internet Explorer	IE 浏览器
IEC	International Electro technical Commission	国际电子技术委员会
IEEE	Institute of Electrical and Electronics Engineers	电气和电子工程师协会
IETF	Internet Engineering Task Force	因特网工程任务组
IGMP	Internet Group Management Protocol	因特网组管理协议
IP	Internet Protocol	网络互联协议
IS-IS	Intermediate System to Intermediate System Routing Protocol	中间系统到中间系统的路由选择协议
ISP	Internet Service Provider	Internet 服务提供商
ITU-T	International Telecommunications Union - Telecommunication Standardization Sector	国际电信联盟远程通信标准化组织

L

LACP	Link Aggregation Control Protocol	链路汇聚控制协议
LACPDU	Link Aggregation Control Protocol Data Unit	链路聚合控制协议数据单元
LAN	Local Area Network	局域网
LCAS	Link Capacity Adjustment Scheme	链路容量调整机制
LLDP	Link Layer Discovery Protocol	链路层发现协议
LLDPDU	Link Layer Discovery Protocol Data Unit	链路层发现协议数据单元

M

MAC	Medium Access Control	媒体访问控制
MDI	Medium Dependent Interface	介质相关接口

MDI-X	Medium Dependent Interface cross-over	介质相关交叉接口
MIB	Management Information Base	管理信息库
MSTI	Multiple Spanning Tree Instance	多生成树实例
MSTP	Multiple Spanning Tree Protocol	多生成树协议
MTBF	Mean Time Between Failure	平均无故障工作时间
MTU	Maximum Transmission Unit	最大传输单元
MVR	Multicast VLAN Registration	组播 VLAN 注册

N

NMS	Network Management System	网络管理系统
NNM	Network Node Management	网络节点管理
NTP	Network Time Protocol	网络时间协议
NView NNM	NView Network Node Management	NView 网络节点管理系统

O

OAM	Operation，Administration and Management	操作、管理和维护
OC	Ordinary Clock	普通时钟
ODF	Optical Distribution Frame	光纤配线架
OID	Object Identifiers	对象标识符
Option 82	DHCP Relay Agent Information Option	DHCP 中继代理信息选项
OSPF	Open Shortest Path First	开放最短路径优先

P

P2MP	Point to Multipoint	点到多点
P2P	Point-to-Point	点到点
PADI	PPPoE Active Discovery Initiation	PPPoE 活动发现发起报文
PADO	PPPoE Active Discovery Offer	PPPoE 活动发现提供报文
PADS	PPPoE Active Discovery Session-confirmation	PPPoE 活动发现会话确认报文
PAP	Password Authentication Protocol	密码认证协议
PDU	Protocol Data Unit	协议数据单元

PE	Provider Edge	运营商边缘
PIM-DM	Protocol Independent Multicast-Dense Mode	密集模式独立组播协议
PIM-SM	Protocol Independent Multicast-Sparse Mode	稀疏模式独立组播协议
Ping	Packet Internet Grope	因特网包探索器
PPP	Point to Point Protocol	点对点协议
PPPoE	PPP over Ethernet	以太网承载 PPP 协议
PTP	Precision Time Protocol	精确时钟同步协议

Q

QoS	Quality of Service	服务质量

R

RADIUS	Remote Authentication Dial In User Service	远程用户拨号认证系统
RCMP	JX Cluster Management Protocol	甲信集群管理协议
RED	Random Early Detection	随机早期检测
RH	Relative Humidity	相对湿度
RIP	Routing Information Protocol	路由信息协议
RMON	Remote Network Monitoring	远端网络监控
ROS	JX Operating System	甲信操作系统
RPL	Ring Protection Link	环保护链路
RRPS	JX Ring Protection Switching	甲信环保护倒换
RSTP	Rapid Spanning Tree Protocol	快速生成树协议
RSVP	Resource Reservation Protocol	资源预留协议

S

SCADA	Supervisory Control And Data Acquisition	数据采集与监视控制系统
SF	Signal Fail	信号失效
SFP	Small Form-factor Pluggable	小封装可插拔
SFTP	Secure File Transfer Protocol	安全文件传输协议
SLA	Service Level Agreement	服务等级规约

SNMP	Simple Network Management Protocol	简单网络管理协议
SNTP	Simple Network Time Protocol	简单网络时间协议
SP	Strict-Priority	严格优先级调度
SPF	Shortest Path First	最短路径优先
SSHv2	Secure Shell v2	安全外壳协议版本 2
STP	Spanning Tree Protocol	生成树协议

T

TACACS+	Terminal Access Controller Access Control System	终端访问控制器访问控制系统
TC	Transparent Clock	透传时钟
TCP	Transmission Control Protocol	传输控制协议
TFTP	Trivial File Transfer Protocol	普通文件传输协议
TLV	Type Length Value	类型、长度和取值
ToS	Type of Service	服务类型
TPID	Tag Protocol Identifier	标签协议标识
TTL	Time To Live	生存时间

U

UDP	User Datagram Protocol	用户数据包协议
UNI	User Network Interface	用户侧接口
USM	User-Based Security Model	基于用户的安全模型

V

VLAN	Virtual Local Area Network	虚拟局域网
VRRP	Virtual Router Redundancy Protocol	虚拟路由冗余协议

W

WAN	Wide Area Network	广域网
WRR	Weight Round Robin	加权循环调度