UNIS 入侵防御系统产品

二层技术-以太网交换配置指导

北京紫光恒越网络科技有限公司
http://www.unis-hy.com

资料版本：5PW100-20160929

前 言

UNIS 入侵防御系统产品配置指导介绍了入侵防御系统产品各软件特性的原理及其配置方法，包含原理简介、配置任务描述和配置举例。《二层技术-以太网交换配置指导》主要介绍 MAC 地址表、以太网链路聚合、VLAN、VLAN 终结、LLDP 和二层转发相关的特性。

前言部分包含如下内容：

适用款型

入侵防御系统产品款型较多，形态丰富，本手册所描述的内容适用于如下产品款型：

表1 手册适用的产品款型

款型	形态
UNIS T5000-M06	分布式设备，可以运行在： • 独立运行模式 • IRF 模式
UNIS T5000-G20	集中式IRF设备
UNIS T1000-G20/G50/G60/G80	集中式IRF设备

读者对象

本手册主要适用于如下工程师：

- 网络规划人员
- 现场技术支持与维护人员
- 负责网络配置和维护的网络管理员

本书约定

1. 命令行格式约定

格 式	意 义
粗体	命令行关键字（命令中保持不变、必须照输的部分）采用**加粗**字体表示。
斜体	命令行参数（命令中必须由实际值进行替代的部分）采用*斜体*表示。

[ ]	表示用“[ ]”括起来的部分在命令配置时是可选的。
{ x | y | … }	表示从多个选项中仅选取一个。
[x | y | …]	表示从多个选项中选取一个或者不选。
{ x | y | … } *	表示从多个选项中至少选取一个。
[x | y | …] *	表示从多个选项中选取一个、多个或者不选。
&<1-n>	表示符号&前面的参数可以重复输入1～n次。
#	由“#”号开始的行表示为注释行。

2. 图形界面格式约定

格　式	意　义
< >	带尖括号“< >”表示按钮名，如“单击<确定>按钮”。
[ ]	带方括号“[ ]”表示窗口名、菜单名和数据表，如“弹出[新建用户]窗口”。
/	多级菜单用“/”隔开。如[文件/新建/文件夹]多级菜单表示[文件]菜单下的[新建]子菜单下的[文件夹]菜单项。

3. 各类标志

本书还采用各种醒目标志来表示在操作过程中应该特别注意的地方，这些标志的意义如下：

警告	该标志后的注释需给予格外关注，不当的操作可能会对人身造成伤害。
注意	提醒操作中应注意的事项，不当的操作可能会导致数据丢失或者设备损坏。
提示	为确保设备配置成功或者正常工作而需要特别关注的操作或信息。
说明	对操作内容的描述进行必要的补充和说明。
窍门	配置、操作、或使用设备的技巧、小窍门。

4. 图标约定

本书使用的图标及其含义如下：

	该图标及其相关描述文字代表一般网络设备，如路由器、交换机、防火墙等。
	该图标及其相关描述文字代表一般意义下的路由器，以及其他运行了路由协议的设备。
	该图标及其相关描述文字代表二、三层以太网交换机，以及运行了二层协议的设备。
	该图标及其相关描述文字代表无线控制器、无线控制器业务板和有线无线一体化交换机的无线控制引擎设备。

	该图标及其相关描述文字代表无线接入点设备。
	该图标及其相关描述文字代表无线终结单元。
	该图标及其相关描述文字代表无线终结者。
	该图标及其相关描述文字代表无线Mesh设备。
	该图标代表发散的无线射频信号。
	该图标代表点到点的无线射频信号。
	该图标及其相关描述文字代表防火墙、UTM、多业务安全网关、负载均衡等安全设备。
	该图标及其相关描述文字代表防火墙插卡、负载均衡插卡、NetStream插卡、SSL VPN插卡、IPS插卡、ACG插卡等安全插卡。

5. 端口编号示例约定

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

技术支持

用户支持邮箱：zgsm_service@thunis.com

技术支持热线电话：400-910-9998（手机、固话均可拨打）

网址：http://www.unis-hy.com

资料意见反馈

如果您在使用过程中发现产品资料的任何问题，可以通过以下方式反馈：

E-mail：zgsm_info@thunis.com

感谢您的反馈，让我们做得更好！

目 录

1 MAC地址表

1.1 MAC地址表简介

MAC（Media Access Control，媒体访问控制）地址表记录了 MAC 地址与接口的对应关系，以及接口所属的 VLAN 等信息。设备在转发报文时，根据报文的目的 MAC 地址查询 MAC 地址表，如果MAC地址表中包含与报文目的MAC地址对应的表项，则直接通过该表项中的出接口转发该报文；如果 MAC 地址表中没有包含报文目的 MAC 地址对应的表项时，设备将采取广播方式通过对应 VLAN 内除接收接口外的所有接口转发该报文。

1.1.1 MAC地址表项的生成方式

MAC 地址表项的生成方式有两种：自动生成、手工配置。

1. 自动生成MAC地址表项

一般情况下，MAC 地址表由设备通过源 MAC 地址学习自动生成。设备学习 MAC 地址的过程如下：

- 从某接口（假设为接口 A）收到一个数据帧，设备分析该数据帧的源 MAC 地址（假设为 MAC-SOURCE），并认为目的 MAC 地址为 MAC-SOURCE 的报文可以由接口 A 转发。
- 如果 MAC 地址表中已经包含 MAC-SOURCE，设备将对该表项进行更新。
- 如果 MAC 地址表中尚未包含 MAC-SOURCE，设备则将这个新 MAC 地址以及该 MAC 地址对应的接口 A 作为一个新的表项加入到 MAC 地址表中。

为适应网络拓扑的变化，MAC 地址表需要不断更新。MAC 地址表中自动生成的表项并非永远有效，每一条表项都有一个生存周期，到达生存周期仍得不到刷新的表项将被删除，这个生存周期被称作老化时间。如果在到达生存周期前某表项被刷新，则重新计算该表项的老化时间。

2. 手工配置MAC地址表项

设备通过源 MAC 地址学习自动生成 MAC 地址表时，无法区分合法用户和非法用户的报文，带来了安全隐患。如果非法用户将攻击报文的源 MAC 地址伪装成合法用户的 MAC 地址，并从设备的其他接口进入，设备就会学习到错误的 MAC 地址表项，于是将本应转发给合法用户的报文转发给非法用户。

为了提高安全性，网络管理员可手工在 MAC 地址表中加入特定 MAC 地址表项，将用户设备与接口绑定，从而防止非法用户骗取数据。

1.1.2 MAC地址表项的分类

MAC 地址表项分为以下几种：

- 静态 MAC 地址表项：由用户手工配置，用于目的是某个 MAC 地址的报文从对应接口转发出去，表项不老化。静态 MAC 地址表项优先级高于自动生成的 MAC 地址表项。
- 动态 MAC 地址表项：可以由用户手工配置，也可以由设备通过源 MAC 地址学习自动生成，用于目的是某个 MAC 地址的报文从对应接口转发出去，表项有老化时间。手工配置的动态 MAC 地址表项优先级等于自动生成的 MAC 地址表项。

- 黑洞 MAC 地址表项：由用户手工配置，用于丢弃源 MAC 地址或目的 MAC 地址为指定 MAC 地址的报文（例如，出于安全考虑，可以禁止某个用户发送和接收报文），表项不老化。黑洞 MAC 地址表项优先级高于自动生成的 MAC 地址表项。

静态 MAC 地址表项和黑洞 MAC 地址表项不会被动态 MAC 地址表项覆盖，而动态 MAC 地址表项可以被静态 MAC 地址表项和黑洞 MAC 地址表项覆盖。静态 MAC 地址表项和黑洞 MAC 地址表项不会彼此覆盖。

1.2 配置MAC地址表

以下配置均为可选配置，且配置过程无先后顺序，用户可以根据实际情况选择配置。

1.2.1 配置MAC地址表项

配置 MAC 地址表项时，需要注意：

- 在手工配置动态 MAC 地址表项时，如果 MAC 地址表中已经存在 MAC 地址相匹配的自动生成表项，但该表项的接口与配置不符，那么该手工配置失败。
- 如果不保存配置，设备重启后所有手工配置的 MAC 地址表项都会丢失；如果保存配置，设备重启后手工配置的静态 MAC 地址表项和黑洞 MAC 地址表项不会丢失，手工配置的动态 MAC 地址表项会丢失。

配置 MAC 地址表项后，当设备收到的报文的源 MAC 地址与配置表项中的 MAC 地址相同时，不同类型的 MAC 地址表项处理方式不同：

表1-1 不同类型 MAC 地址表项对源 MAC 地址匹配报文的处理方式

MAC 地址表项类型	报文源 MAC 地址与配置表项中的 MAC 地址相同
静态MAC地址表项	不检查报文入接口与表项中的接口是否相同，直接根据目的MAC地址转发该报文
黑洞MAC地址表项	丢弃该报文
动态MAC地址表项	• 如果报文入接口与该表项中的接口不同，则进行 MAC 地址学习，并覆盖该表项 • 如果报文入接口与该表项中的接口相同，则转发该报文，并更新该表项老化时间

1. 配置静态/动态MAC地址表项

(1) 全局配置静态/动态 MAC 地址表项

表1-2 全局配置静态/动态 MAC 地址表项

操作	命令	说明
进入系统视图	**system-view**	-
添加或者修改静态/动态MAC地址表项	**mac-address** { **dynamic** | **static** } *mac-address* **interface** *interface-type interface-number* **vlan** *vlan-id*	缺省情况下，未配置任何MAC地址表项 **interface**参数指定的接口必须属于*vlan-id*参数指定的VLAN，而且该VLAN必须事先创建，否则将配置失败

(2) 接口配置静态/动态 MAC 地址表项

表1-3 接口配置静态/动态 MAC 地址表项

操作		命令	说明
进入系统视图		**system-view**	-
进入接口视图	二层以太网接口视图	**interface** *interface-type interface-number*	-
	二层聚合接口视图	**interface bridge-aggregation** *interface-number*	
在当前接口下添加或者修改静态/动态MAC地址表项		**mac-address** { **dynamic** | **static** } *mac-address* **vlan** *vlan-id*	缺省情况下，接口下未配置任何MAC地址表项 当前接口必须属于*vlan-id*参数指定的VLAN，而且该VLAN必须事先创建，否则将配置失败

2. 配置黑洞MAC地址表项

表1-4 配置黑洞 MAC 地址表项

操作	命令	说明
进入系统视图	**system-view**	-
添加或者修改黑洞MAC地址表项	**mac-address blackhole** *mac-address* **vlan** *vlan-id*	缺省情况下，未配置任何MAC地址表项 *vlan-id*参数指定的VLAN必须事先创建，否则将配置失败

1.2.2 关闭MAC地址学习功能

缺省情况下，MAC 地址学习功能处于开启状态。有时为了保证设备的安全，需要关闭 MAC 地址学习功能。常见的危及设备安全的情况是：非法用户使用大量源 MAC 地址不同的报文攻击设备，导致设备 MAC 地址表资源耗尽，造成设备无法根据网络的变化更新 MAC 地址表。关闭 MAC 地址学习功能可以有效防止这种攻击。

关闭 MAC 地址学习功能后，对于已经存在的动态 MAC 地址表项待老化时间超时后将自然老化。

1. 关闭全局的MAC地址学习功能

关闭全局的 MAC 地址学习功能后，接口将不再学习新的 MAC 地址。

表1-5 关闭全局 MAC 地址学习功能

操作	命令	说明
进入系统视图	**system-view**	-
关闭全局的MAC地址学习功能	**undo mac-address mac-learning enable**	缺省情况下，全局的MAC地址学习功能处于开启状态

2. 关闭接口的MAC地址学习功能

在开启全局的 MAC 地址学习功能的前提下，用户可以关闭设备上单个接口的 MAC 地址学习功能。

表1-6 关闭接口的 MAC 地址学习功能

操作		命令	说明
进入系统视图		**system-view**	-
进入接口视图	二层以太网接口视图	**interface** *interface-type interface-number*	-
	二层聚合接口视图	**interface bridge-aggregation** *interface-number*	
关闭接口的MAC地址学习功能		**undo mac-address mac-learning enable**	缺省情况下，接口的MAC地址学习功能处于开启状态

1.2.3 配置动态MAC地址表项的老化时间

当网络拓扑改变后，如果动态 MAC 地址表项不及时更新，会导致用户流量不能正常转发。配置动态 MAC 地址表项的老化时间后，超过老化时间的动态 MAC 地址表项会被自动删除，设备将重新进行 MAC 地址学习，构建新的动态 MAC 地址表项。

用户配置的老化时间过长或者过短，都可能影响设备的运行性能：

- 如果用户配置的老化时间过长，设备可能会保存许多过时的 MAC 地址表项，从而耗尽 MAC 地址表资源，导致设备无法根据网络的变化更新 MAC 地址表。
- 如果用户配置的老化时间太短，设备可能会删除有效的MAC地址表项，导致设备广播大量的数据报文，增加网络的负担。

用户需要根据实际情况，配置合适的老化时间。如果网络比较稳定，可以将老化时间配置得长一些或者配置为不老化；否则，可以将老化时间配置得短一些。比如在一个比较稳定的网络，如果长时间没有流量，动态 MAC 地址表项会被全部删除，可能导致设备突然广播大量的数据报文，造成安全隐患，此时可将动态 MAC 地址表项的老化时间设得长一些或不老化，以减少广播，增加网络稳定性和安全性。

动态 MAC 地址表项的老化时间作用于全部接口上。

表1-7 配置动态 MAC 地址表项的老化时间

操作	命令	说明
进入系统视图	**system-view**	-
配置动态MAC地址表项的老化时间	**mac-address timer** { **aging** *seconds* | **no-aging** }	缺省情况下，动态MAC地址表项的老化时间为300秒

1.2.4 配置接口的MAC地址数学习上限

通过配置接口的 MAC 地址数学习上限，用户可以控制设备维护的 MAC 地址表的表项数量。如果 MAC 地址表过于庞大，可能导致设备的转发性能下降。当接口学习到的 MAC 地址数达到上限时，该接口将不再对 MAC 地址进行学习。

表1-8 配置接口的 MAC 地址数学习上限

操作		命令	说明
进入系统视图		**system-view**	-
进入接口视图	二层以太网接口视图	**interface** *interface-type interface-number*	-
	二层聚合接口视图	**interface bridge-aggregation** *interface-number*	
配置接口的MAC地址数学习上限		**mac-address max-mac-count** *count*	-

1.2.5 配置当达到接口MAC地址数学习上限时的报文转发规则

表1-9 配置允许转发源 MAC 地址不在 MAC 地址表里的报文

操作		命令	说明
进入系统视图		**system-view**	-
进入接口视图	二层以太网接口视图	**interface** *interface-type interface-number*	-
	二层聚合接口视图	**interface bridge-aggregation** *interface-number*	
配置当达到接口的MAC地址数学习上限时，允许转发源MAC地址不在MAC地址表里的报文		**mac-address max-mac-count enable-forwarding**	缺省情况下，当达到接口的MAC地址数学习上限时，允许转发源MAC地址不在MAC地址表里的报文

1.2.6 配置接口的MAC地址学习优先级

基于 MAC 地址转发报文的网络有时会因为下行接口的攻击行为或者环路，下行接口学习到网关等上层设备的 MAC 地址。为了避免这种情况，将接口的 MAC 地址学习功能分为两个优先级：高优先级和低优先级。对于高优先级的接口，可以学习任何 MAC 地址；对于低优先级的接口，在学习 MAC 地址时需要查看高优先级接口是否已经学到该 MAC 地址，如果已经学到，则不允许学习该 MAC 地址。比如，可以将上行接口的 MAC 地址学习优先级配置为高优先级，下行接口的 MAC 地址学习优先级配置为低优先级，那么，下行接口就不会学到网关等上层设备的 MAC 地址，避免了攻击。

表1-10 配置接口的 MAC 地址学习优先级

操作		命令	说明
进入系统视图		**system-view**	-
进入接口视图	二层以太网接口视图	**interface** *interface-type interface-number*	-
	二层聚合接口视图	**interface bridge-aggregation** *interface-number*	

操作	命令	说明
配置接口的MAC地址学习优先级	**mac-address mac-learning priority { high | low }**	缺省情况下，MAC地址学习优先级为低优先级

1.3 MAC地址表显示和维护

在完成上述配置后，在任意视图下执行 **display** 命令可以显示配置后 MAC 地址表的运行情况，通过查看显示信息验证配置的效果。

表1-11 MAC 地址表显示和维护

操作	命令
显示MAC地址表信息	**display mac-address** [*mac-address* [**vlan** *vlan-id*] | [[**dynamic** | **static**] [**interface** *interface-type interface-number*] | **blackhole**] [**vlan** *vlan-id*] [**count**]]
显示MAC地址表动态表项的老化时间	**display mac-address aging-time**
显示MAC地址学习功能的使能状态	**display mac-address mac-learning** [**interface** *interface-type interface-number*]

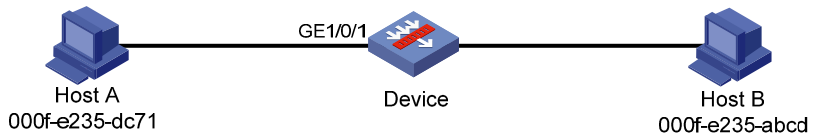
1.4 MAC地址表典型配置举例

1. 组网需求

- 现有一台用户主机，它的 MAC 地址为 000f-e235-dc71，属于 VLAN 1，连接 Device 的端口 GigabitEthernet1/0/1。为防止假冒身份的非法用户骗取数据，在设备的 MAC 地址表中为该用户主机添加一条静态表项。
- 另有一台用户主机，它的 MAC 地址为 000f-e235-abcd，属于 VLAN 1。由于该用户主机曾经接入网络进行非法操作，为了避免此种情况再次发生，在设备上添加一条黑洞 MAC 地址表项，使该用户主机接收不到报文。
- 配置设备的动态 MAC 地址表项老化时间为 500 秒。

2. 组网图

图1-1 MAC 地址表典型配置组网图



3. 配置步骤

增加一个静态 MAC 地址表项，目的地址为 000f-e235-dc71，出接口为 GigabitEthernet1/0/1，且该接口属于 VLAN 1。

```
<Device> system-view
```

```
[Device] mac-address static 000f-e235-dc71 interface gigabitethernet 1/0/1 vlan 1
```

增加一个黑洞 MAC 地址表项，地址为 000f-e235-abcd，属于 VLAN 1。

```
[Device] mac-address blackhole 000f-e235-abcd vlan 1
```

配置动态 MAC 地址表项的老化时间为 500 秒。

```
[Device] mac-address timer aging 500
```

4. 验证配置

查看端口 GigabitEthernet1/0/1 上的静态 MAC 地址表项信息。

```
[Device] display mac-address static interface gigabitethernet 1/0/1
MAC Address      VLAN ID    State           Port/NickName           Aging
000f-e235-dc71   1          Static          GE1/0/1                 N
```

查看黑洞 MAC 地址表信息。

```
[Device] display mac-address blackhole
MAC Address      VLAN ID    State           Port/NickName           Aging
000f-e235-abcd   1          Blackhole       N/A                     N
```

查看动态 MAC 地址表项的老化时间。

```
[Device] display mac-address aging-time
MAC address aging time: 500s.
```

目 录

1 以太网链路聚合

说明

- 仅 T5000-M06 支持引擎聚合接口相关配置。
- 自定义 Context 中不支持本特性。

1.1 以太网链路聚合简介

以太网链路聚合通过将多条以太网物理链路捆绑在一起形成一条以太网逻辑链路，实现增加链路带宽的目的，同时这些捆绑在一起的链路通过相互备份，可以有效地提高链路的可靠性。

如 图 1-1 所示，Device A与Device B之间通过三条以太网物理链路相连，将这三条链路捆绑在一起，就成为了一条逻辑链路Link aggregation 1。这条逻辑链路的带宽最大可等于三条以太网物理链路的带宽总和，增加了链路的带宽；同时，这三条以太网物理链路相互备份，当其中某条物理链路down，还可以通过其他两条物理链路转发报文。

图1-1 链路聚合示意图

1.1.2 基本概念

1. 聚合组、成员端口和聚合接口

链路捆绑是通过接口捆绑实现的，多个以太网接口捆绑在一起后形成一个聚合组，而这些被捆绑在一起的以太网接口就称为该聚合组的成员端口。每个聚合组唯一对应着一个逻辑接口，称为聚合接口。聚合组与聚合接口的编号是相同的，例如聚合组 1 对应于聚合接口 1。聚合组/聚合接口可以分为以下几种类型：

- 二层聚合组/二层聚合接口：二层聚合组的成员端口全部为二层以太网接口，其对应的聚合接口称为二层聚合接口。
- 三层聚合组/三层聚合接口：三层聚合组的成员端口全部为三层以太网接口，其对应的聚合接口称为三层聚合接口。在创建了三层聚合接口之后，还可继续创建该三层聚合接口的子接口，即三层聚合子接口。
- 引擎聚合组/引擎聚合接口：引擎聚合组的成员端口是系统自动添加，其对应的聚合接口称为引擎聚合接口。

聚合接口的速率和双工模式取决于对应聚合组内的选中端口（请参见“1.1.2 2. 成员端口的状态”）：聚合接口的速率等于所有选中端口的速率之和，聚合接口的双工模式则与选中端口的双工模式相同。

2. 成员端口的状态

聚合组内的成员端口具有以下三种状态：

- 选中（Selected）状态：此状态下的成员端口可以参与数据的转发，处于此状态的成员端口称为“选中端口”。
- 非选中（Unselected）状态：此状态下的成员端口不能参与数据的转发，处于此状态的成员端口称为“非选中端口”。
- 独立（Individual）状态：此状态下的成员端口可以作为普通物理口参与数据的转发。当聚合接口配置为聚合边缘接口，其成员端口未收到对端端口发送的 LACP（Link Aggregation Control Protocol，链路聚合控制协议）报文时，处于该状态。

3. 操作Key

操作 Key 是系统在进行链路聚合时用来表征成员端口聚合能力的一个数值，它是根据成员端口上的一些信息（包括该端口的速率、双工模式等）的组合自动计算生成的，这个信息组合中任何一项的变化都会引起操作 Key 的重新计算。在同一聚合组中，所有的选中端口都必须具有相同的操作 Key。

4. 配置分类

根据对成员端口状态的影响不同，成员端口上的配置可以分为以下两类：

(1) 属性类配置：包含的配置内容如 表 1-1 所示。在聚合组中，只有与对应聚合接口的属性类配置完全相同的成员端口才能够成为选中端口。

表1-1 属性类配置的内容

配置项	内容
VLAN配置	端口上允许通过的VLAN、端口缺省VLAN、端口的链路类型（即Trunk、Hybrid、Access类型）、有关VLAN配置的详细描述，请参见“二层技术-以太网交换配置指导”中的“VLAN”


说明

- 在聚合接口上所作的属性类配置，将在聚合接口以及相应的所有成员端口下生效。当聚合接口被删除后，这些配置仍将保留在这些成员端口上。
- 由于成员端口上属性类配置的改变可能导致其选中/非选中状态发生变化，进而对业务产生影响，因此当在成员端口上进行此类配置时，系统将给出提示信息，由用户来决定是否继续执行该配置。

(2) 协议类配置：是相对于属性类配置而言的，包含的配置内容有 MAC 地址学习、生成树等。在聚合组中，即使某成员端口与对应聚合接口的协议配置存在不同，也不会影响该成员端口成为选中端口。


说明

- 在聚合接口上所作的协议类配置，只在当前聚合接口下生效。
- 在成员端口上所作的协议类配置，只有当该成员端口退出聚合组后才能生效。

5. 聚合模式

链路聚合分为静态聚合和动态聚合两种模式，它们各自的优点如下所示：

- 静态聚合模式：一旦配置好后，端口的选中/非选中状态就不会受网络环境的影响，比较稳定。
- 动态聚合模式：能够根据对端和本端的信息调整端口的选中/非选中状态，比较灵活。

处于静态聚合模式下的聚合组称为静态聚合组，处于动态聚合模式下的聚合组称为动态聚合组。

1.1.3 静态聚合模式

静态聚合模式的工作机制如下所述。

1. 选择参考端口

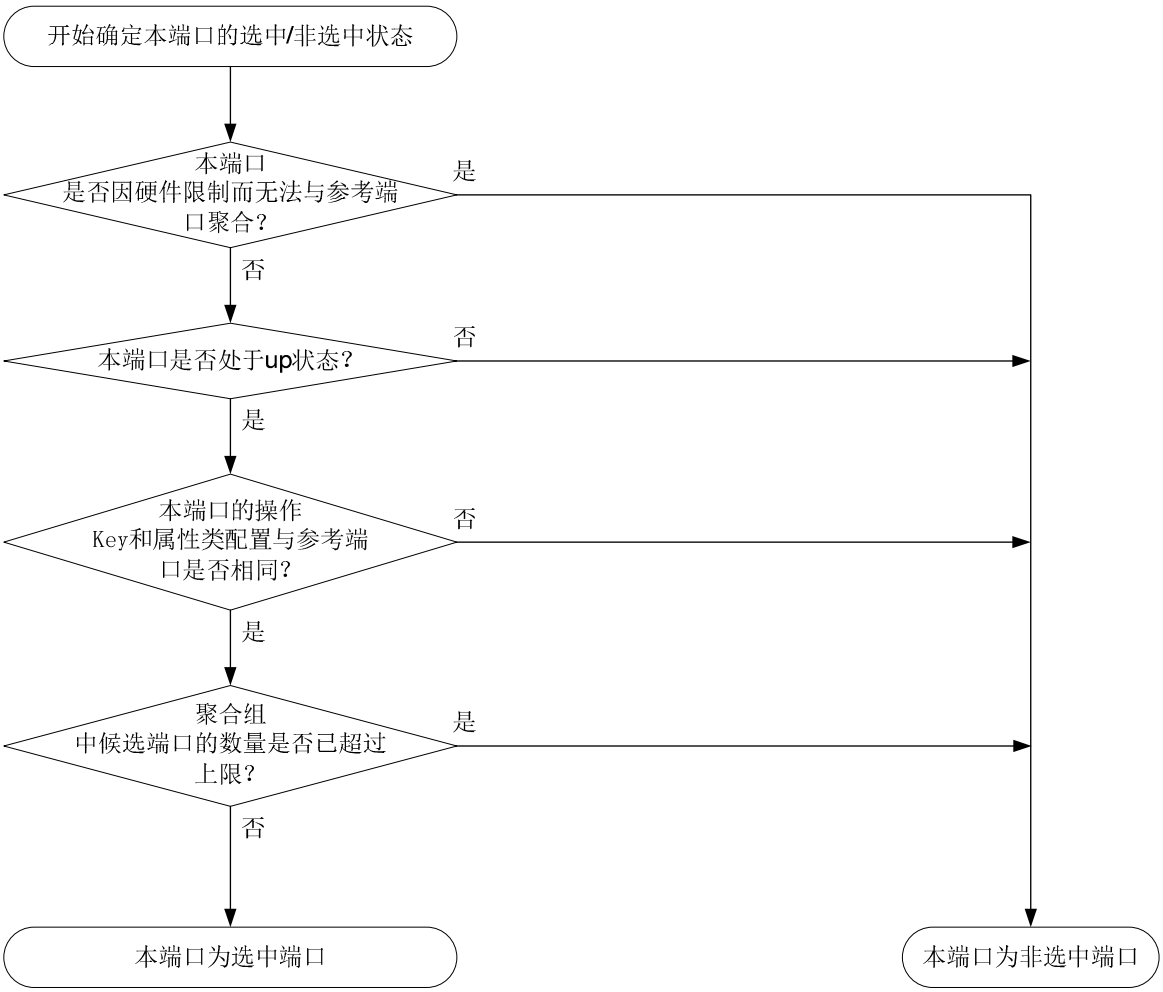
参考端口从本端的成员端口中选出，其操作 Key 和属性类配置将作为同一聚合组内的其他成员端口的参照，只有操作 Key 和属性类配置与参考端口一致的成员端口才能被选中。

对于聚合组内处于 up 状态的端口，按照端口的高端口优先级->全双工/高速率->全双工/低速率->半双工/高速率->半双工/低速率的优先次序，选择优先次序最高、且属性类配置与对应聚合接口相同的端口作为参考端口；如果多个端口优先次序相同，首先选择原来的选中端口作为参考端口；如果此时多个优先次序相同的端口都是原来的选中端口，则选择其中端口号最小的端口作为参考端口；如果多个端口优先次序相同，且都不是原来的选中端口，则选择其中端口号最小的端口作为参考端口。

2. 确定成员端口的状态

静态聚合组内成员端口状态的确定流程如 图 1-2 所示。

图1-2 静态聚合组内成员端口状态的确定流程



确定静态聚合组内成员端口状态时，需要注意：

- 当一个成员端口的操作 Key 或属性类配置改变时，其所在静态聚合组内各成员端口的选中/非选中状态可能会发生改变。
- 当静态聚合组内选中端口的数量已达到上限，对于后加入的成员端口和聚合组内选中端口的端口优先级：
  - 全部相同时，后加入的成员端口即使满足成为选中端口的所有条件，也不会立即成为选中端口。这样能够尽量维持当前选中端口上的流量不中断，但是由于设备重启时会重新计算选中端口，因此可能导致设备重启前后各成员端口的选中/非选中状态不一致。
  - 存在不同时，若后加入的成员端口的属性类配置与对应聚合接口相同，且端口优先级高于聚合组内选中端口的端口优先级，则端口优先级高的成员端口会立刻取代端口优先级低的选中端口成为新的选中端口。

1.1.4 动态聚合模式

动态聚合模式通过 LACP 协议实现，LACP 协议的内容及动态聚合模式的工作机制如下所述。

1. LACP协议

基于 IEEE802.3ad 标准的 LACP 协议是一种实现链路动态聚合的协议，运行该协议的设备之间通过互发 LACPDU 来交互链路聚合的相关信息。

动态聚合组内的成员端口可以收发 LACPDU（Link Aggregation Control Protocol Data Unit，链路聚合控制协议数据单元），本端通过向对端发送 LACPDU 通告本端的信息。当对端收到该 LACPDU 后，将其中的信息与所在端其他成员端口收到的信息进行比较，以选择能够处于选中状态的成员端口，使双方可以对各自接口的选中/非选中状态达成一致。

(1) LACP 协议的功能

LACP协议的功能分为基本功能和扩展功能两大类，如 表 1-2 所示。

表1-2 LACP 协议的功能分类

类别	说明
基本功能	利用LACPDU的基本字段可以实现LACP协议的基本功能。基本字段包含以下信息：系统LACP优先级、系统MAC地址、端口优先级、端口编号和操作Key
扩展功能	通过对LACPDU的字段进行扩展，可以实现对LACP协议的扩展。通过在扩展字段中定义一个新的TLV（Type/Length/Value，类型/长度/值）数据域，可以实现IRF（Intelligent Resilient Framework，智能弹性架构）中的LACP MAD（Multi-Active Detection，多Active检测）机制。有关IRF和LACP MAD机制的详细介绍，请参见“虚拟化配置指导”中的“IRF”。

(2) LACP 工作模式

LACP 工作模式分为 ACTIVE 和 PASSIVE 两种。

如果动态聚合组内成员端口的LACP工作模式为PASSIVE，且对端的LACP工作模式也为PASSIVE时，两端将不能发送 LACPDU。如果两端中任何一端的 LACP 工作模式为 ACTIVE 时，两端将可以发送 LACPDU。

(3) LACP 优先级

根据作用的不同，可以将LACP优先级分为系统LACP优先级和端口优先级两类，如 表 1-3 所示。

表1-3 LACP 优先级的分类

类别	说明	比较标准
系统LACP优先级	用于区分两端设备优先级的高低。当两端设备中的一端具有较高优先级时，另一端将根据优先级较高的一端来选择本端的选中端口，这样便使两端设备的选中端口达成了一致	优先级数值越小，优先级越高
端口优先级	用于区分各成员端口成为选中端口的优先程度	

(4) LACP 超时时间

LACP 超时时间是指成员端口等待接收 LACPDU 的超时时间，在 LACP 超时时间之后，如果本端成员端口仍未收到来自对端的 LACPDU，则认为对端成员端口已失效。

LACP 超时时间同时也决定了对端发送 LACPDU 的速率。LACP 超时有短超时（3 秒）和长超时（90 秒）两种。若 LACP 超时时间为短超时，则对端将快速发送 LACPDU（每 1 秒发送 1 个 LACPDU）；若 LACP 超时时间为长超时，则对端将慢速发送 LACPDU（每 30 秒发送 1 个 LACPDU）。

2. 动态聚合模式的工作机制：

(1) 选择参考端口

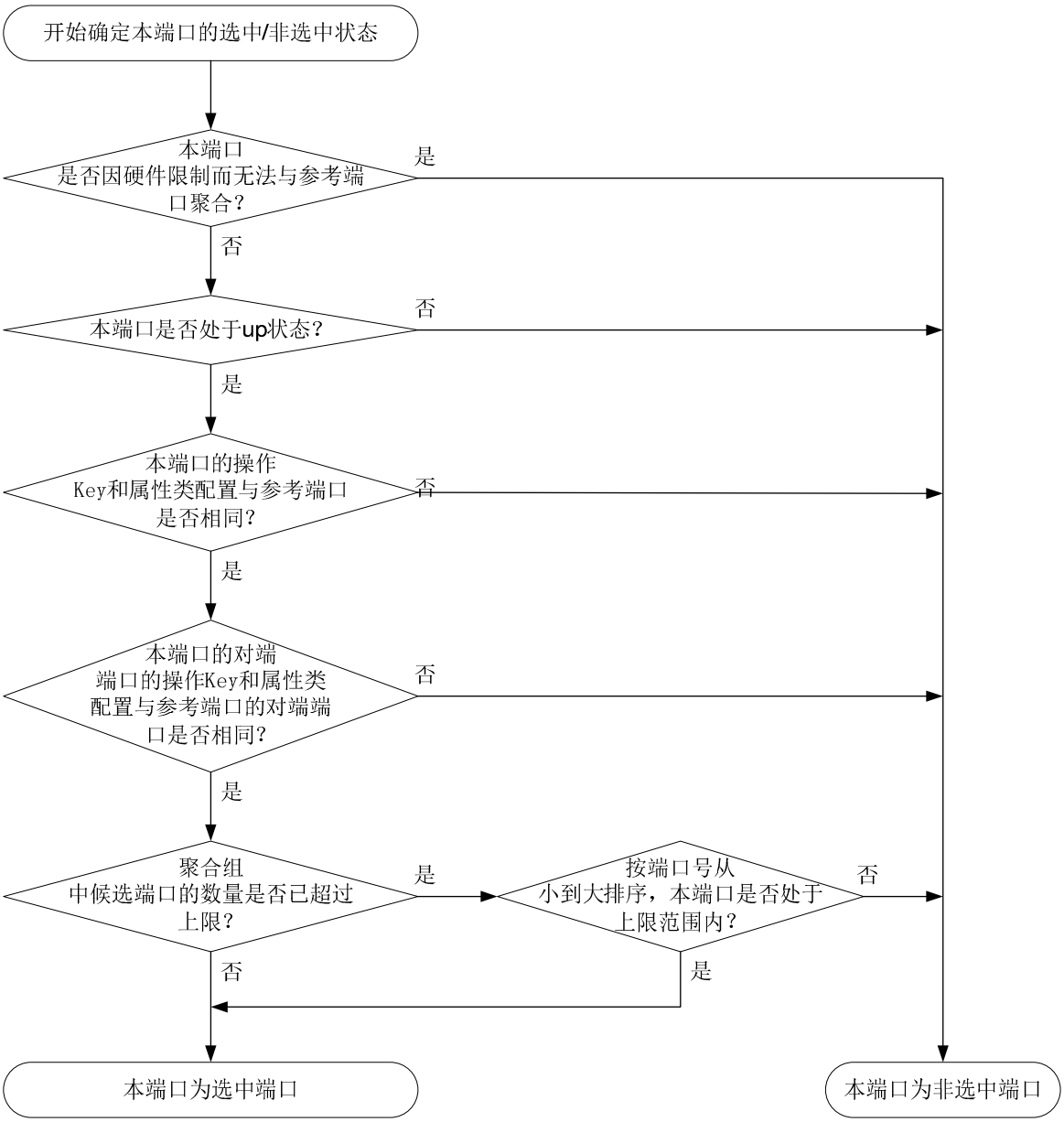
参考端口从聚合链路两端处于 up 状态的成员端口中选出，其操作 Key 和属性类配置将作为同一聚合组内的其他成员端口的参照，只有操作 Key 和属性类配置与参考端口一致的成员端口才能被选中。

- 首先，从聚合链路的两端选出设备 ID（由系统的 LACP 优先级和系统的 MAC 地址共同构成）较小的一端：先比较两端的系统 LACP 优先级，优先级数值越小其设备 ID 越小；如果优先级相同再比较其系统 MAC 地址，MAC 地址越小其设备 ID 越小。
- 其次，对于设备 ID 较小的一端，再比较其聚合组内各成员端口的端口 ID（由端口优先级和端口的编号共同构成）：先比较端口优先级，优先级数值越小其端口 ID 越小；如果优先级相同再比较其端口号，端口号越小其端口 ID 越小。端口 ID 最小、且属性类配置与对应聚合接口相同的端口作为参考端口。

(2) 确定成员端口的状态

在设备ID较小的一端，动态聚合组内成员端口状态的确定流程如 图 1-3 所示。

图1-3 动态聚合组内成员端口状态的确定流程



与此同时，设备 ID 较大的一端也会随着对端成员端口状态的变化，随时调整本端各成员端口的状态，以确保聚合链路两端成员端口状态的一致。

确定动态聚合组内成员端口状态时，需要注意：

- 仅全双工端口可成为选中端口。
- 当一个成员端口的操作 Key 或属性类配置改变时，其所在动态聚合组内各成员端口的选中/非选中状态可能会发生改变。
- 当本端端口的选中/非选中状态发生改变时，其对端端口的选中/非选中状态也将随之改变。
- 当动态聚合组内选中端口的数量已达到上限时，后加入的成员端口一旦满足成为选中端口的所有条件，就会立刻取代已不满足条件的端口成为选中端口。

1.1.5 聚合边缘接口

在网络设备与服务器等终端设备相连的场景中，当网络设备配置了动态聚合模式，而终端设备未配置动态聚合模式时，聚合链路不能成功建立，网络设备与该终端设备相连多条链路中只能有一条作为普通链路正常转发报文，因而链路间也不能形成备份，当该普通链路发生故障时，可能会造成报文丢失。

若要求在终端设备未配置动态聚合模式时，该终端设备与网络设备间的链路可以形成备份，可通过配置网络设备与终端设备相连的聚合接口为聚合边缘接口，使该聚合组内的所有成员端口都作为普通物理口转发报文，从而保证终端设备与网络设备间的多条链路可以相互备份，增加可靠性。当终端设备完成动态聚合模式配置时，其聚合成员端口正常发送 LACP 报文后，网络设备上符合选中条件的聚合成员端口会自动被选中，从而使聚合链路恢复正常工作。

1.1.6 聚合负载分担类型

通过采用不同的聚合负载分担类型，可以实现灵活地对聚合组内流量进行负载分担。聚合负载分担的类型可以归为以下几类：

- 逐流负载分担：按照报文的源/目的 MAC 地址、VLAN 标签、源/目的服务端口、入端口、源/目的 IP 地址、IP 协议类型或 MPLS 标签中的一种或某几种的组合区分流，使属于同一数据流的报文从同一条成员链路上通过。
- 逐包负载分担：不区分数据流，而是以报文为单位，将流量分担到不同的成员链路上进行传输。

1.2 以太网链路聚合配置任务简介

表1-4 以太网链路聚合配置任务简介

配置任务		说明	详细配置
配置聚合组	配置二层聚合组	三者必选其一	1.3.1
	配置三层聚合组		1.3.2
	配置引擎聚合组		1.3.3
聚合接口相关配置	配置聚合接口的描述信息	可选	1.4.1
	配置聚合接口的MAC地址	可选	1.4.2
	配置二层聚合接口的忽略VLAN	可选	1.4.3
	配置三层聚合接口MTU	可选	1.4.4
	限制聚合组内选中端口的数量	可选	1.4.5
	配置聚合接口的期望带宽	可选	1.4.6
	配置聚合接口为聚合边缘接口	可选	1.4.7
	关闭聚合接口	可选	1.4.8
	恢复聚合接口的缺省配置	可选	1.4.9
配置聚合负载分担	配置聚合负载分担类型	可选	1.5.1

配置任务		说明	详细配置
	配置聚合负载分担采用本地转发优先	可选	1.5.2
配置聚合流量重定向功能		可选	1.6

1.3 配置聚合组

配置聚合组时，需要注意：

- 配置了下列功能的端口将不能加入二层聚合组：冗余组节点（请参见“可靠性配置指导/冗余备份”中的“冗余组”）。
- 配置了下列功能的端口将不能加入三层聚合组：以太网冗余接口（请参见“可靠性配置指导/冗余备份”中的“以太网冗余接口”）、冗余组节点（请参见“可靠性配置指导/冗余备份”中的“冗余组”）。
- 用户删除聚合接口时，系统将自动删除对应的聚合组，且该聚合组内的所有成员端口将全部离开该聚合组。
- 聚合链路的两端应配置相同的聚合模式。
- 二层聚合组和三层聚合组都分为静态聚合和动态聚合两种模式。
- 对于静态聚合模式，用户需要保证在同一链路两端端口的选中/非选中状态的一致性，否则聚合功能无法正常使用。
- 对于动态聚合模式，聚合链路两端的设备会自动协商同一链路两端的端口在各自聚合组内的选中/非选中状态，用户只需保证本端聚合在一起的端口的对端也同样聚合在一起，聚合功能即可正常使用。

1.3.1 配置二层聚合组

1. 配置二层静态聚合组

表1-5 配置二层静态聚合组

操作	命令	说明
进入系统视图	**system-view**	-
创建二层聚合接口，并进入二层聚合接口视图	**interface bridge-aggregation** *interface-number*	创建二层聚合接口后，系统将自动生成同编号的二层聚合组，且该聚合组缺省工作在静态聚合模式下
退回系统视图	**quit**	-
进入二层以太网接口视图	**interface** *interface-type interface-number*	多次执行此步骤可将多个二层以太网接口加入聚合组
将二层以太网接口加入聚合组	**port link-aggregation group** *group-id*	

2. 配置二层动态聚合组

表1-6 配置二层动态聚合组

操作	命令	说明
进入系统视图	**system-view**	-
配置系统的LACP优先级	**lacp system-priority** *priority*	缺省情况下，系统的LACP优先级为32768 改变系统的LACP优先级，将会影响到动态聚合组成员端口的选中/非选中状态
创建二层聚合接口，并进入二层聚合接口视图	**interface bridge-aggregation** *interface-number*	创建二层聚合接口后，系统将自动生成同编号的二层聚合组，且该聚合组缺省工作在静态聚合模式下
配置聚合组工作在动态聚合模式下	**link-aggregation mode dynamic**	缺省情况下，聚合组工作在静态聚合模式下
退回系统视图	**quit**	-
进入二层以太网接口视图	**interface** *interface-type interface-number*	多次执行此步骤可将多个二层以太网接口加入聚合组
将二层以太网接口加入聚合组	**port link-aggregation group** *group-id*	
配置当前端口的LACP工作模式为PASSIVE	**lacp mode passive**	二者选其一 缺省情况下，端口的LACP工作模式为ACTIVE
配置当前端口的LACP工作模式为ACTIVE	**undo lacp mode**	
配置端口优先级	**link-aggregation port-priority** *priority*	缺省情况下，端口优先级为32768
配置端口的LACP超时时间为短超时（3秒），并使对端快速发送LACPDU	**lacp period short**	缺省情况下，端口的LACP超时时间为长超时（90秒），对端慢速发送LACPDU 请不要在ISSU升级前配置LACP超时时间为短超时，否则在ISSU升级期间会出现网络流量中断，导致流量转发不通。有关ISSU升级的详细介绍请参见“基础配置指导”中的“ISSU配置”

1.3.2 配置三层聚合组

1. 配置三层静态聚合组

表1-7 配置三层静态聚合组

操作	命令	说明
进入系统视图	**system-view**	-

操作	命令	说明
创建三层聚合接口，并进入三层聚合接口视图	**interface route-aggregation** *interface-number*	创建三层聚合接口后，系统将自动生成同编号的三层聚合组，且该聚合组缺省工作在静态聚合模式下
退回系统视图	**quit**	-
进入三层以太网接口视图	**interface** *interface-type interface-number*	多次执行此步骤可将多个三层以太网接口加入聚合组
将三层以太网接口加入聚合组	**port link-aggregation group** *group-id*	

2. 配置三层动态聚合组

表1-8 配置三层动态聚合组

操作	命令	说明
进入系统视图	**system-view**	-
配置系统的LACP优先级	**lacp system-priority** *priority*	缺省情况下，系统的LACP优先级为32768 改变系统的LACP优先级，将会影响到动态聚合组成员的选中/非选中状态
创建三层聚合接口，并进入三层聚合接口视图	**interface route-aggregation** *interface-number*	创建三层聚合接口后，系统将自动生成同编号的三层聚合组，且该聚合组缺省工作在静态聚合模式下
配置聚合组工作在动态聚合模式下	**link-aggregation mode dynamic**	缺省情况下，聚合组工作在静态聚合模式下
退回系统视图	**quit**	-
进入三层以太网接口视图	**interface** *interface-type interface-number*	多次执行此步骤可将多个三层以太网接口加入聚合组
将三层以太网接口加入聚合组	**port link-aggregation group** *group-id*	
配置当前端口的LACP工作模式为PASSIVE	**lacp mode passive**	二者选其一 缺省情况下，端口的LACP工作模式为ACTIVE
配置当前端口的LACP工作模式为ACTIVE	**undo lacp mode**	
配置端口优先级	**link-aggregation port-priority** *priority*	缺省情况下，端口优先级为32768
配置端口的LACP超时时间为短超时（3秒），并使对端快速发送LACPDU	**lacp period short**	缺省情况下，端口的LACP超时时间为长超时（90秒），对端慢速发送LACPDU 请不要在ISSU升级前配置LACP超时时间为短超时，否则在ISSU升级期间会出现网络流量中断，导致流量转发不通。有关ISSU升级的详细介绍请参见“基础配置指导”中的“ISSU配置”

1.3.3 配置引擎聚合组

设备各款型对于本节所描述的特性支持情况有所不同，详细差异信息如下：

型号	特性	描述
T5000-M06	引擎聚合组	支持
T5000-G20		不支持
T1000-G20/G50/G60/G80		不支持

在安全设备中，可以将多个引擎板上的引擎口捆绑形成一个引擎聚合组，以增加链路带宽的目的。

在创建引擎聚合组之前，需要先创建引擎组，之后，系统会自动创建一个和引擎组编号一致的引擎聚合组。需要注意的是，由于缺省引擎聚合组编号是 1，所以系统将自动分配编号 2。

引擎聚合组创建后，还需要将引擎板加入引擎组，之后，系统会自动将引擎板上的引擎口加入刚才创建好的引擎聚合组。有关引擎组的详细描述，请参见“虚拟化技术配置指导”中的“Context”。

引擎聚合口对应的引擎聚合组只能工作在静态聚合模式下。引擎聚合口的创建和删除，在创建和删除引擎组时自动完成，由于默认引擎聚合接口 1 所对应的缺省引擎组不能被删除，所以默认引擎组合接口 1 不能被删除。

表1-9 配置引擎聚合组

操作	命令	说明
进入系统视图	**system-view**	-
创建引擎组并进入引擎组视图	**blade-controller-team** *blade-controller-team-name* [**id** *blade-controller-team-id*]	缺省情况下，设备有一个引擎组，名称为Default，编号为1 引擎组创建后，系统自动创建一个编号和引擎组编号相同的引擎聚合组 有关**blade-controller-team**命令的详细介绍，请参见“虚拟化技术命令参考”中的“Context”
将引擎板加入引擎组，进而将引擎口加入引擎聚合组（分布式设备—独立运行模式）	**location blade-controller slot** *slot-number* **cpu** *cpu-number*	缺省情况下，引擎板插入时，会自动加入缺省的引擎组 引擎板加入引擎组后，系统自动将引擎板对应的引擎口加入之前系统自动创建的引擎聚合组
将引擎板加入引擎组，进而将引擎口加入引擎聚合组（分布式设备—IRF模式）	**location blade-controller chassis** *chassis-number* **slot** *slot-number* **cpu** *cpu-number*	缺省情况下，引擎板插入时，会自动加入缺省的引擎组 引擎板加入引擎组后，系统自动将引擎板对应的引擎口加入之前系统自动创建的引擎聚合组

1.4 聚合接口相关配置

本节对能够在聚合接口上进行的部分配置进行介绍。除本节所介绍的配置外，能够在二层/三层以太网接口上进行的配置大多数也能在二层/三层聚合接口上进行，具体配置请参见相关的配置指导。

1.4.1 配置聚合接口的描述信息

通过在接口上配置描述信息，可以方便网络管理员根据这些信息来区分各接口的作用。

表1-10 配置聚合接口的描述信息

操作		命令	说明
进入系统视图		**system-view**	-
进入聚合接口视图	进入二层聚合接口视图	**interface bridge-aggregation** *interface-number*	-
	进入三层聚合接口/子接口视图	**interface route-aggregation** { *interface-number* | *interface-number.subnumber* }	
配置当前接口的描述信息		**description** *text*	缺省情况下，接口的描述信息为“*接口名* Interface”

1.4.2 配置聚合接口的MAC地址

同一设备上所有聚合接口的缺省 MAC 地址都相同，不同设备上聚合接口的缺省 MAC 地址不同。通常情况下，不需要修改聚合接口的 MAC 地址。

表1-11 配置聚合接口的 MAC 地址

操作	命令	说明
进入系统视图	**system-view**	-
进入三层聚合接口/子接口视图	**interface route-aggregation** { *interface-number* | *interface-number.subnumber* }	-
配置聚合接口的MAC地址	**mac-address** *mac-address*	缺省情况下，同一设备上所有聚合接口的MAC地址都相同，不同设备上聚合接口的MAC地址不同

1.4.3 配置二层聚合接口的忽略VLAN

未配置二层聚合接口的忽略 VLAN 时，只有当其成员端口上关于 VLAN 允许通过的配置（包括是否允许 VLAN 通过，以及通过的方式）与该二层聚合接口的配置完全相同时，该成员端口才有可能成为选中端口；配置了二层聚合接口的忽略 VLAN 后，即使其成员端口上关于这些 VLAN 允许通过的配置与该二层聚合接口上的配置不一致，也不影响该成员端口成为选中端口。

表1-12 配置二层聚合接口的忽略 VLAN

操作	命令	说明
进入系统视图	**system-view**	-
进入二层聚合接口视图	**interface bridge-aggregation** *interface-number*	-
配置二层聚合接口的忽略VLAN	**link-aggregation ignore vlan** *vlan-id-list*	缺省情况下，二层聚合接口未配置忽略VLAN

1.4.4 配置三层聚合接口MTU

MTU（Maximum Transmission Unit，最大传输单元）参数会影响 IP 报文的分片与重组，可以通过下面的配置来改变 MTU 值。

表1-13 配置三层聚合接口 MTU

操作	命令	说明
进入系统视图	**system-view**	-
进入三层聚合接口/子接口视图	**interface route-aggregation** { *interface-number* | *interface-number.subnumber* }	-
配置三层聚合接口/子接口的MTU值	**mtu** *size*	缺省情况下，三层聚合接口/子接口的MTU值为1500字节

1.4.5 限制聚合组内选中端口的数量

提示

本端和对端配置的聚合组中的最小/最大选中端口数必须一致。

聚合链路的带宽取决于聚合组内选中端口的数量，用户通过配置聚合组中的最小选中端口数，可以避免由于选中端口太少而造成聚合链路上的流量拥塞。当聚合组内选中端口的数量达不到配置值时，对应的聚合接口将不会 up。具体实现如下：

- 如果聚合组内能够被选中的成员端口数小于配置值，这些成员端口都将变为非选中状态，对应聚合接口的链路状态也将变为 down。
- 当聚合组内能够被选中的成员端口数增加至不小于配置值时，这些成员端口都将变为选中状态，对应聚合接口的链路状态也将变为 up。

当配置了聚合组中的最大选中端口数之后，最大选中端口数将同时受配置值和设备硬件能力的限制，即取二者的较小值作为限制值。用户借此可实现两端口间的冗余备份：在一个聚合组中只添加两个成员端口，并配置该聚合组中的最大选中端口数为 1，这样这两个成员端口在同一时刻就只能有一个成为选中端口，而另一个将作为备份端口。

表1-14 限制聚合组内选中端口的数量

操作		命令	说明
进入系统视图		**system-view**	-
进入聚合接口视图	进入二层聚合接口视图	**interface bridge-aggregation** *interface-number*	-
	进入三层聚合接口	**interface route-aggregation** *interface-number*	
配置聚合组中的最小选中端口数		**link-aggregation selected-port minimum** *min-number*	缺省情况下，聚合组中的最小选中端口数不受限制
配置聚合组中的最大选中端口数		**link-aggregation selected-port maximum** *max-number*	缺省情况下，聚合组中的最大选中端口数仅受设备硬件能力的限制

1.4.6 配置聚合接口的期望带宽

表1-15 配置聚合接口的期望带宽

操作		命令	说明
进入系统视图		**system-view**	-
进入聚合接口视图	进入二层聚合接口视图	**interface bridge-aggregation** *interface-number*	-
	进入三层聚合接口/子接口视图	**interface route-aggregation** { *interface-number* | *interface-number.subnumber* }	
配置当前接口的期望带宽		**bandwidth** *bandwidth-value*	缺省情况下，接口的期望带宽＝接口的波特率÷1000（kbps）

1.4.7 配置聚合接口为聚合边缘接口

配置聚合接口为聚合边缘接口时，需要注意，该配置仅在聚合接口对应的聚合组为动态聚合组时生效。

表1-16 配置聚合接口为聚合边缘接口

操作		命令	说明
进入系统视图		**system-view**	-
进入聚合接口视图	进入二层聚合接口视图	**interface bridge-aggregation** *interface-number*	-
	进入三层聚合接口视图	**interface route-aggregation** *interface-number*	
配置聚合接口为聚合边缘接口		**lacp edge-port**	缺省情况下，聚合接口不为聚合边缘接口

1.4.8 关闭聚合接口

对聚合接口的开启/关闭操作，将会影响聚合接口对应的聚合组内成员端口的选中/非选中状态和链路状态：

- 关闭聚合接口时，将使对应聚合组内所有处于选中状态的成员端口都变为非选中端口，且所有成员端口的链路状态都将变为 down。
- 开启聚合接口时，系统将重新计算对应聚合组内成员端口的选中/非选中状态。

表1-17 关闭聚合接口

操作		命令	说明
进入系统视图		**system-view**	-
进入聚合接口视图	进入二层聚合接口视图	**interface bridge-aggregation** *interface-number*	-
	进入三层聚合接口/子接口视图	**interface route-aggregation** { *interface-number* | *interface-number.subnumber* }	
关闭当前接口		**shutdown**	未关闭当前接口

1.4.9 恢复聚合接口的缺省配置

通过执行本操作可以将聚合接口下的所有配置都恢复为缺省配置。

表1-18 恢复聚合接口的缺省配置

操作		命令	说明
进入系统视图		**system-view**	-
进入聚合接口视图	进入二层聚合接口视图	**interface bridge-aggregation** *interface-number*	-
	进入三层聚合接口/子接口视图	**interface route-aggregation** { *interface-number* | *interface-number.subnumber* }	
恢复当前聚合接口的缺省配置		**default**	-

1.5 配置聚合负载分担

1.5.1 配置聚合负载分担类型

聚合负载分担类型支持全局配置或在聚合组内配置两种方式：全局的配置对所有聚合组都有效，而聚合组内的配置只对当前聚合组有效。对于一个聚合组来说，优先采用该聚合组内的配置，只有该聚合组内未进行配置时，才采用全局的配置。

1. 全局配置聚合负载分担类型

表1-19 全局配置聚合负载分担类型

操作	命令	说明
进入系统视图	**system-view**	-
配置全局采用的聚合负载分担类型	**link-aggregation global load-sharing mode** { **destination-ip** | **destination-mac** | **destination-port** | **ingress-port** | **source-ip** | **source-mac** | **source-port** } *	各参数支持情况请参见相关命令手册

2. 在聚合组内配置聚合负载分担类型

表1-20 在聚合组内配置聚合负载分担类型

操作		命令	说明
进入系统视图		**system-view**	-
进入聚合接口视图	进入二层聚合接口视图	**interface bridge-aggregation** *interface-number*	-
	进入三层聚合接口视图	**interface route-aggregation** *interface-number*	
	进入引擎聚合接口视图	**interface blade-aggregation** *interface-number*	
配置聚合组内采用的聚合负载分担类型		**link-aggregation load-sharing mode** { { **destination-ip** | **destination-mac** | **destination-port** | **ip-protocol** | **mpls-label1** | **source-ip** | **source-mac** | **source-port** } * | **per-packet** }	各参数支持情况请参见相关命令手册

1.5.2 配置聚合负载分担采用本地转发优先

配置聚合负载分担采用本地转发优先机制可以降低数据流量对IRF物理端口之间链路的冲击，IRF中成员设备间聚合负载分担处理流程如 图 1-4 所示。有关IRF的详细介绍，请参见“虚拟化配置指导”中的“IRF”。

图1-4 IRF 中成员设备间聚合负载分担处理流程

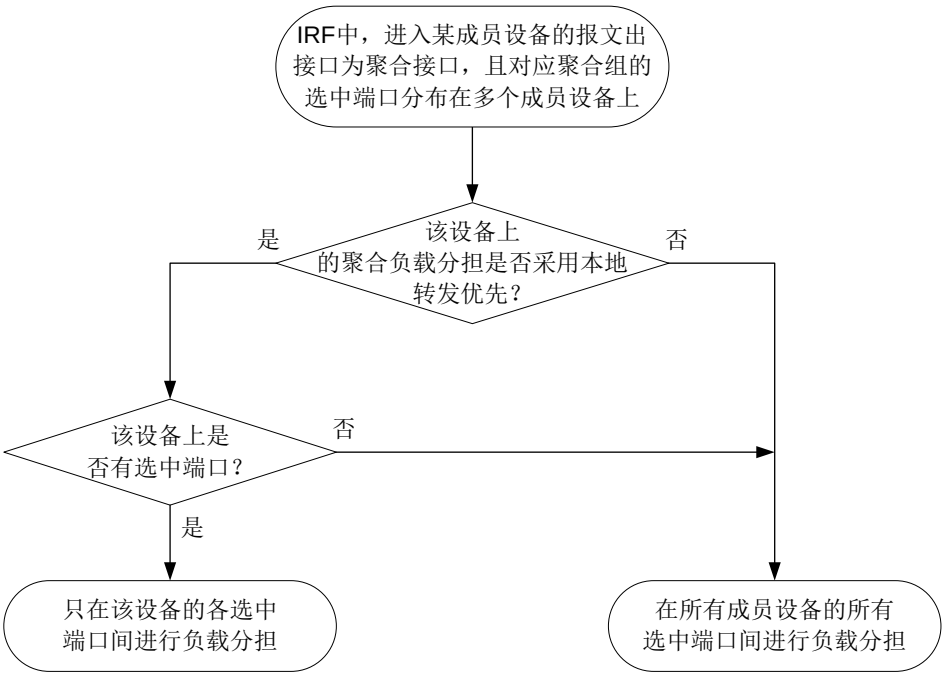


表1-21 配置聚合负载分担采用本地转发优先

操作	命令	说明
进入系统视图	**system-view**	-
配置聚合负载分担采用本地转发优先	**link-aggregation load-sharing mode local-first**	缺省情况下，聚合负载分担采用本地转发优先

1.6 配置聚合流量重定向功能

在开启了聚合流量重定向功能后，当重启设备上某块有聚合组选中端口的单板时，系统可以将该单板上的流量重定向到其他单板上，从而实现聚合链路上流量的不中断。（分布式设备－独立运行模式）

在开启了聚合流量重定向功能后，当重启 IRF 中某台有聚合组选中端口的成员设备时，系统可以将该设备上的流量重定向到其他成员设备上，从而实现聚合链路上流量的不中断。（集中式 IRF 设备）

在开启了聚合流量重定向功能后，当重启 IRF 中某台有聚合组选中端口的成员设备或成员设备上某块有聚合组选中端口的单板时，系统可以将该设备或单板上的流量重定向到其他成员设备或单板上，从而实现聚合链路上流量的不中断。（分布式设备－IRF 模式）

配置聚合流量重定向功能时，需要注意：

- 必须在聚合链路两端都开启聚合流量重定向功能才能实现聚合链路上流量的不中断。
- 如果同时开启聚合流量重定向功能和生成树功能，在重启单板/设备时会出现少量的丢包，因此不建议同时开启上述两个功能。
- 当聚合接口配置为聚合边缘接口后，聚合流量重定向功能将不能正常使用。

- 只有动态聚合组支持聚合流量重定向功能。

表1-22 配置聚合流量重定向功能

操作	命令	说明
进入系统视图	**system-view**	-
开启聚合流量重定向功能	**link-aggregation lacp traffic-redirect-notification enable**	缺省情况下，聚合流量重定向功能处于关闭状态

1.7 以太网链路聚合显示与维护

在完成上述配置后，在任意视图下执行 **display** 命令可以显示配置后以太网链路聚合的运行情况，通过查看显示信息验证配置的效果。

在用户视图下执行 **reset** 命令可以清除端口的 LACP 和聚合接口上的统计信息。

表1-23 以太网链路聚合显示与维护

操作	命令
显示聚合接口的相关信息	**display interface** [{ **blade-aggregation** | **bridge-aggregation** | **route-aggregation** } [*interface-number*]] [**brief** [**description** | **down**]]
显示本端系统的设备ID	**display lacp system-id**
显示全局或聚合组内采用的聚合负载分担类型	**display link-aggregation load-sharing mode** [**interface** [{ **blade-aggregation** | **bridge-aggregation** | **route-aggregation** } *interface-number*]]
显示成员端口上链路聚合的详细信息	**display link-aggregation member-port** [*interface-list*]
显示所有聚合组的摘要信息	**display link-aggregation summary**
显示已有聚合接口所对应聚合组的详细信息	**display link-aggregation verbose** [{ **blade-aggregation** | **bridge-aggregation** | **route-aggregation** } [*interface-number*]]
清除成员端口上的LACP统计信息	**reset lacp statistics** [**interface** *interface-list*]
清除聚合接口上的统计信息	**reset counters interface** [{ **blade-aggregation** | **bridge-aggregation** | **route-aggregation** } [*interface-number*]]

1.8 以太网链路聚合典型配置举例

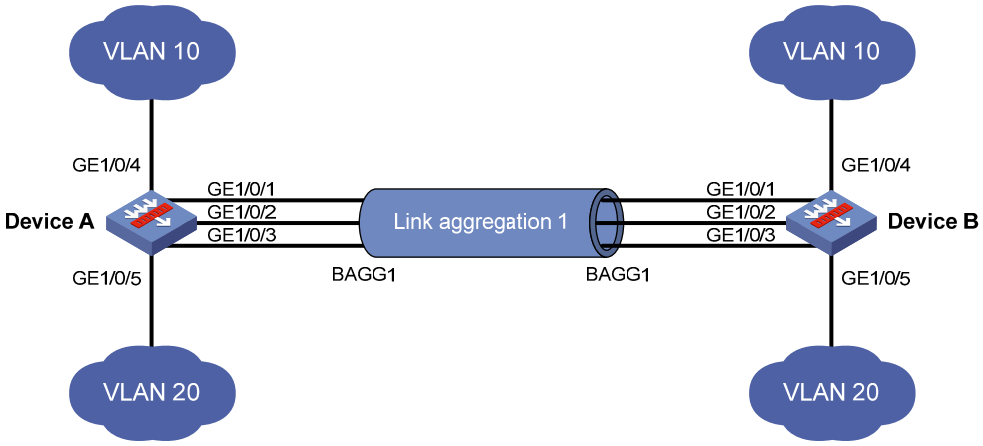
1.8.1 二层静态聚合配置举例

1. 组网需求

- Device A 与 Device B 通过各自的二层以太网接口 GigabitEthernet1/0/1～GigabitEthernet1/0/3 相互连接。
- 在 Device A 和 Device B 上分别配置二层静态链路聚合组，并实现设备间 VLAN 10 和 VLAN 20 分别互通。

2. 组网图

图1-5 二层静态聚合配置组网图



3. 配置步骤

(1) 配置 Device A

创建 VLAN 10，并将端口 GigabitEthernet1/0/4 加入到该 VLAN 中。

```
<DeviceA> system-view
[DeviceA] vlan 10
[DeviceA-vlan10] port gigabitethernet 1/0/4
[DeviceA-vlan10] quit
```

创建 VLAN 20，并将端口 GigabitEthernet1/0/5 加入到该 VLAN 中。

```
[DeviceA] vlan 20
[DeviceA-vlan20] port gigabitethernet 1/0/5
[DeviceA-vlan20] quit
```

创建二层聚合接口 1。

```
[DeviceA] interface bridge-aggregation 1
[DeviceA-Bridge-Aggregation1] quit
```

分别将端口 GigabitEthernet1/0/1 至 GigabitEthernet1/0/3 加入到聚合组 1 中。

```
[DeviceA] interface gigabitethernet 1/0/1
[DeviceA-GigabitEthernet1/0/1] port link-aggregation group 1
[DeviceA-GigabitEthernet1/0/1] quit
[DeviceA] interface gigabitethernet 1/0/2
[DeviceA-GigabitEthernet1/0/2] port link-aggregation group 1
[DeviceA-GigabitEthernet1/0/2] quit
[DeviceA] interface gigabitethernet 1/0/3
[DeviceA-GigabitEthernet1/0/3] port link-aggregation group 1
[DeviceA-GigabitEthernet1/0/3] quit
```

配置二层聚合接口 1 为 Trunk 端口，并允许 VLAN 10 和 20 的报文通过。

```
[DeviceA] interface bridge-aggregation 1
[DeviceA-Bridge-Aggregation1] port link-type trunk
[DeviceA-Bridge-Aggregation1] port trunk permit vlan 10 20
[DeviceA-Bridge-Aggregation1] quit
```

(2) 配置 Device B

Device B 的配置与 Device A 相似，配置过程略。

4. 验证配置

查看 Device A 上所有聚合组的详细信息。

```
[DeviceA] display link-aggregation verbose
Loadsharing Type: Shar -- Loadsharing, NonS -- Non-Loadsharing
Port Status: S -- Selected, U -- Unselected, I -- Individual
Flags:  A -- LACP_Activity, B -- LACP_Timeout, C -- Aggregation,
        D -- Synchronization, E -- Collecting, F -- Distributing,
        G -- Defaulted, H -- Expired

Aggregate Interface: Bridge-Aggregation1
Aggregation Mode: Static
Loadsharing Type: NonS
  Port             Status  Priority Oper-Key
--------------------------------------------------------------------------------
  GE1/0/1          S       32768    1
  GE1/0/2          S       32768    1
  GE1/0/3          S       32768    1
```

以上信息表明，聚合组 1 为非负载分担类型的二层静态聚合组，包含有三个选中端口。

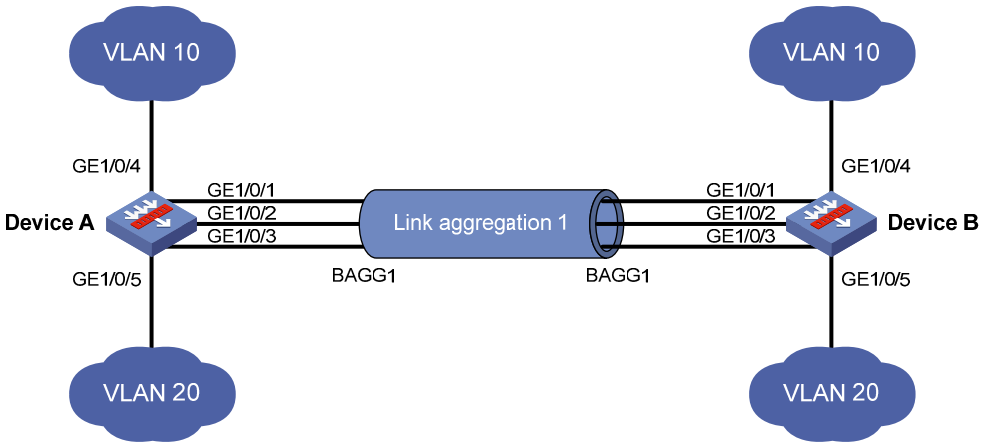
1.8.2 二层动态聚合配置举例

1. 组网需求

- Device A 与 Device B 通过各自的二层以太网接口 GigabitEthernet1/0/1～GigabitEthernet1/0/3 相互连接。
- 在 Device A 和 Device B 上分别配置二层动态链路聚合组，并实现设备间 VLAN 10 和 VLAN 20 分别互通。

2. 组网图

图1-6 二层动态聚合配置组网图



3. 配置步骤

(1) 配置 Device A

创建 VLAN 10，并将端口 GigabitEthernet1/0/4 加入到该 VLAN 中。

```
<DeviceA> system-view
[DeviceA] vlan 10
[DeviceA-vlan10] port gigabitethernet 1/0/4
[DeviceA-vlan10] quit
```

创建 VLAN 20，并将端口 GigabitEthernet1/0/5 加入到该 VLAN 中。

```
[DeviceA] vlan 20
[DeviceA-vlan20] port gigabitethernet 1/0/5
[DeviceA-vlan20] quit
```

创建二层聚合接口 1，并配置该接口为动态聚合模式。

```
[DeviceA] interface bridge-aggregation 1
[DeviceA-Bridge-Aggregation1] link-aggregation mode dynamic
[DeviceA-Bridge-Aggregation1] quit
```

分别将端口 GigabitEthernet1/0/1 至 GigabitEthernet1/0/3 加入到聚合组 1 中。

```
[DeviceA] interface gigabitethernet 1/0/1
[DeviceA-GigabitEthernet1/0/1] port link-aggregation group 1
[DeviceA-GigabitEthernet1/0/1] quit
[DeviceA] interface gigabitethernet 1/0/2
[DeviceA-GigabitEthernet1/0/2] port link-aggregation group 1
[DeviceA-GigabitEthernet1/0/2] quit
[DeviceA] interface gigabitethernet 1/0/3
[DeviceA-GigabitEthernet1/0/3] port link-aggregation group 1
[DeviceA-GigabitEthernet1/0/3] quit
```

配置二层聚合接口 1 为 Trunk 端口，并允许 VLAN 10 和 20 的报文通过。

```
[DeviceA] interface bridge-aggregation 1
[DeviceA-Bridge-Aggregation1] port link-type trunk
[DeviceA-Bridge-Aggregation1] port trunk permit vlan 10 20
[DeviceA-Bridge-Aggregation1] quit
```

(2) 配置 Device B

Device B 的配置与 Device A 相似，配置过程略。

4. 验证配置

查看 Device A 上所有聚合组的详细信息。

```
[DeviceA] display link-aggregation verbose
Loadsharing Type: Shar -- Loadsharing, NonS -- Non-Loadsharing
Port Status: S -- Selected, U -- Unselected, I -- Individual
Flags:  A -- LACP_Activity, B -- LACP_Timeout, C -- Aggregation,
        D -- Synchronization, E -- Collecting, F -- Distributing,
        G -- Defaulted, H -- Expired

Aggregate Interface: Bridge-Aggregation1
Aggregation Mode: Dynamic
Loadsharing Type: NonS
System ID: 0x8000, 000f-e267-6c6a
```

```
Local:
  Port            Status   Priority Oper-Key  Flag
--------------------------------------------------------------------------------
  GE1/0/1         S        32768    1         {ACDEF}
  GE1/0/2         S        32768    1         {ACDEF}
  GE1/0/3         S        32768    1         {ACDEF}
Remote:
  Actor           Partner  Priority Oper-Key  SystemID               Flag
--------------------------------------------------------------------------------
  GE1/0/1         1        32768    1         0x8000, 000f-e267-57ad {ACDEF}
  GE1/0/2         2        32768    1         0x8000, 000f-e267-57ad {ACDEF}
  GE1/0/3         3        32768    1         0x8000, 000f-e267-57ad {ACDEF}
```

以上信息表明，聚合组 1 为非负载分担类型的二层动态聚合组，包含有三个选中端口。

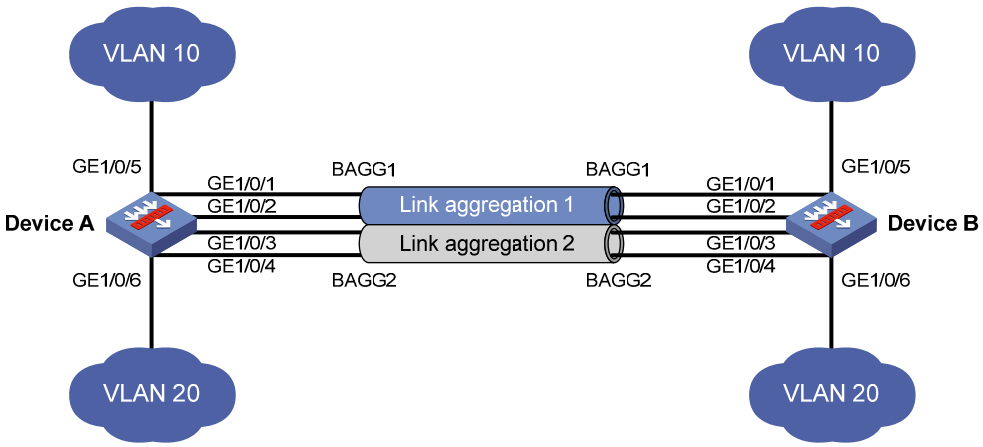
1.8.3 二层聚合负载分担配置举例

1. 组网需求

- Device A 与 Device B 通过各自的二层以太网接口 GigabitEthernet1/0/1～GigabitEthernet1/0/4 相互连接。
- 在 Device A 和 Device B 上分别配置两个二层静态链路聚合组，并使两端的 VLAN 10 通过二层聚合接口 1 互通、VLAN 20 通过二层聚合接口 2 互通。
- 通过在聚合组 1 上按照源 MAC 地址进行聚合负载分担、在聚合组 2 上按照目的 MAC 地址进行聚合负载分担的方式，来实现数据流量在各成员端口间的负载分担。

2. 组网图

图1-7 二层聚合负载分担配置组网图



3. 配置步骤

(1) 配置 Device A

创建 VLAN 10，并将端口 GigabitEthernet1/0/5 加入到该 VLAN 中。

```
<DeviceA> system-view
[DeviceA] vlan 10
[DeviceA-vlan10] port gigabitethernet 1/0/5
```

```
[DeviceA-vlan10] quit
```
创建 VLAN 20，并将端口 GigabitEthernet1/0/6 加入到该 VLAN 中。
```
[DeviceA] vlan 20
[DeviceA-vlan20] port gigabitethernet 1/0/6
[DeviceA-vlan20] quit
```
创建二层聚合接口 1，并配置该接口对应的聚合组内按照源 MAC 地址进行聚合负载分担。
```
[DeviceA] interface bridge-aggregation 1
[DeviceA-Bridge-Aggregation1] link-aggregation load-sharing mode source-mac
[DeviceA-Bridge-Aggregation1] quit
```
分别将端口 GigabitEthernet1/0/1 和 GigabitEthernet1/0/2 加入到聚合组 1 中。
```
[DeviceA] interface gigabitethernet 1/0/1
[DeviceA-GigabitEthernet1/0/1] port link-aggregation group 1
[DeviceA-GigabitEthernet1/0/1] quit
[DeviceA] interface gigabitethernet 1/0/2
[DeviceA-GigabitEthernet1/0/2] port link-aggregation group 1
[DeviceA-GigabitEthernet1/0/2] quit
```
配置二层聚合接口 1 为 Trunk 端口，并允许 VLAN 10 的报文通过。
```
[DeviceA] interface bridge-aggregation 1
[DeviceA-Bridge-Aggregation1] port link-type trunk
[DeviceA-Bridge-Aggregation1] port trunk permit vlan 10
[DeviceA-Bridge-Aggregation1] quit
```
创建二层聚合接口 2，并配置该接口对应的聚合组内按照目的 MAC 地址进行聚合负载分担。
```
[DeviceA] interface bridge-aggregation 2
[DeviceA-Bridge-Aggregation2] link-aggregation load-sharing mode destination-mac
[DeviceA-Bridge-Aggregation2] quit
```
分别将端口 GigabitEthernet1/0/3 和 GigabitEthernet1/0/4 加入到聚合组 2 中。
```
[DeviceA] interface gigabitethernet 1/0/3
[DeviceA-GigabitEthernet1/0/3] port link-aggregation group 2
[DeviceA-GigabitEthernet1/0/3] quit
[DeviceA] interface gigabitethernet 1/0/4
[DeviceA-GigabitEthernet1/0/4] port link-aggregation group 2
[DeviceA-GigabitEthernet1/0/4] quit
```
配置二层聚合接口 2 为 Trunk 端口，并允许 VLAN 20 的报文通过。
```
[DeviceA] interface bridge-aggregation 2
[DeviceA-Bridge-Aggregation2] port link-type trunk
[DeviceA-Bridge-Aggregation2] port trunk permit vlan 20
[DeviceA-Bridge-Aggregation2] quit
```
(2)　配置 Device B

Device B 的配置与 Device A 相似，配置过程略。

4. 验证配置

查看 Device A 上所有聚合组的详细信息。
```
[DeviceA] display link-aggregation verbose
Loadsharing Type: Shar -- Loadsharing, NonS -- Non-Loadsharing
Port Status: S -- Selected, U -- Unselected, I -- Individual
Flags:  A -- LACP_Activity, B -- LACP_Timeout, C -- Aggregation,
```

```
        D -- Synchronization, E -- Collecting, F -- Distributing,
        G -- Defaulted, H -- Expired

Aggregate Interface: Bridge-Aggregation1
Aggregation Mode: Static
Loadsharing Type: Shar
  Port             Status  Priority Oper-Key
--------------------------------------------------------------------------------
  GE1/0/1          S       32768    1
  GE1/0/2          S       32768    1

Aggregate Interface: Bridge-Aggregation2
Aggregation Mode: Static
Loadsharing Type: Shar
  Port             Status  Priority Oper-Key
--------------------------------------------------------------------------------
  GE1/0/3          S       32768    2
  GE1/0/4          S       32768    2
```

以上信息表明，聚合组 1 和聚合组 2 都是负载分担类型的二层静态聚合组，各包含有两个选中端口。

查看 Device A 上所有聚合接口所对应聚合组内采用的聚合负载分担类型。

```
[DeviceA] display link-aggregation load-sharing mode interface
Bridge-Aggregation1 Load-Sharing Mode:
source-mac address

Bridge-Aggregation2 Load-Sharing Mode:
destination-mac address
```

以上信息表明，二层聚合组 1 按照报文的源 MAC 地址进行聚合负载分担，二层聚合组 2 按照报文的目的 MAC 地址进行聚合负载分担。

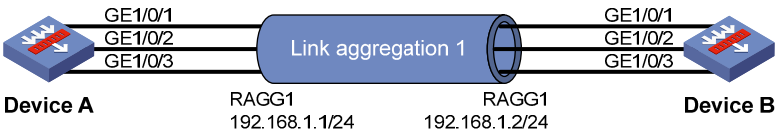
1.8.4 三层静态聚合配置举例

1. 组网需求

- Device A 与 Device B 通过各自的三层以太网接口 GigabitEthernet1/0/1～GigabitEthernet1/0/3 相互连接。
- 在 Device A 和 Device B 上分别配置三层静态链路聚合组，并为对应的三层聚合接口配置 IP 地址和子网掩码。

2. 组网图

图1-8 三层静态聚合配置组网图



3. 配置步骤

(1) 配置 Device A

创建三层聚合接口 1，并为该接口配置 IP 地址和子网掩码。

```
<DeviceA> system-view
[DeviceA] interface route-aggregation 1
[DeviceA-Route-Aggregation1] ip address 192.168.1.1 24
[DeviceA-Route-Aggregation1] quit
```

分别将接口 GigabitEthernet1/0/1 至 GigabitEthernet1/0/3 加入到聚合组 1 中。

```
[DeviceA] interface gigabitethernet 1/0/1
[DeviceA-GigabitEthernet1/0/1] port link-aggregation group 1
[DeviceA-GigabitEthernet1/0/1] quit
[DeviceA] interface gigabitethernet 1/0/2
[DeviceA-GigabitEthernet1/0/2] port link-aggregation group 1
[DeviceA-GigabitEthernet1/0/2] quit
[DeviceA] interface gigabitethernet 1/0/3
[DeviceA-GigabitEthernet1/0/3] port link-aggregation group 1
[DeviceA-GigabitEthernet1/0/3] quit
```

(2) 配置 Device B

Device B 的配置与 Device A 相似，配置过程略。

4. 验证配置

查看 Device A 上所有聚合组的详细信息。

```
[DeviceA] display link-aggregation verbose
Loadsharing Type: Shar -- Loadsharing, NonS -- Non-Loadsharing
Port Status: S -- Selected, U -- Unselected, I -- Individual
Flags:  A -- LACP_Activity, B -- LACP_Timeout, C -- Aggregation,
        D -- Synchronization, E -- Collecting, F -- Distributing,
        G -- Defaulted, H -- Expired

Aggregate Interface: Route-Aggregation1
Aggregation Mode: Static
Loadsharing Type: NonS
  Port             Status  Priority Oper-Key
--------------------------------------------------------------------------------
  GE1/0/1          S       32768    1
  GE1/0/2          S       32768    1
  GE1/0/3          S       32768    1
```

以上信息表明，聚合组 1 为非负载分担类型的三层静态聚合组，包含有三个选中端口。

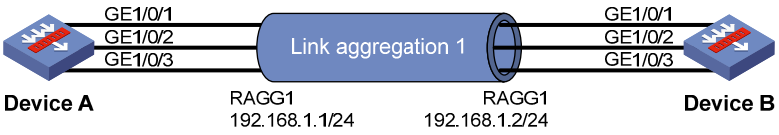
1.8.5 三层动态聚合配置举例

1. 组网需求

- Device A 与 Device B 通过各自的三层以太网接口 GigabitEthernet1/0/1～GigabitEthernet1/0/3 相互连接。
- 在 Device A 和 Device B 上分别配置三层动态链路聚合组，并为对应的三层聚合接口配置 IP 地址和子网掩码。

2. 组网图

图1-9 三层动态聚合配置组网图



3. 配置步骤

(1) 配置 Device A

创建三层聚合接口 1，配置该接口为动态聚合模式，并为其配置 IP 地址和子网掩码。

```
<DeviceA> system-view
[DeviceA] interface route-aggregation 1
[DeviceA-Route-Aggregation1] link-aggregation mode dynamic
[DeviceA-Route-Aggregation1] ip address 192.168.1.1 24
[DeviceA-Route-Aggregation1] quit
```

分别将接口 GigabitEthernet1/0/1 至 GigabitEthernet1/0/3 加入到聚合组 1 中。

```
[DeviceA] interface gigabitethernet 1/0/1
[DeviceA-GigabitEthernet1/0/1] port link-aggregation group 1
[DeviceA-GigabitEthernet1/0/1] quit
[DeviceA] interface gigabitethernet 1/0/2
[DeviceA-GigabitEthernet1/0/2] port link-aggregation group 1
[DeviceA-GigabitEthernet1/0/2] quit
[DeviceA] interface gigabitethernet 1/0/3
[DeviceA-GigabitEthernet1/0/3] port link-aggregation group 1
[DeviceA-GigabitEthernet1/0/3] quit
```

(2) 配置 Device B

Device B 的配置与 Device A 相似，配置过程略。

4. 验证配置

查看 Device A 上所有聚合组的详细信息。

```
[DeviceA] display link-aggregation verbose
Loadsharing Type: Shar -- Loadsharing, NonS -- Non-Loadsharing
Port Status: S -- Selected, U -- Unselected, I -- Individual
Flags:  A -- LACP_Activity, B -- LACP_Timeout, C -- Aggregation,
        D -- Synchronization, E -- Collecting, F -- Distributing,
        G -- Defaulted, H -- Expired

Aggregate Interface: Route-Aggregation1
Aggregation Mode: Dynamic
Loadsharing Type: NonS
System ID: 0x8000, 000f-e267-6c6a
Local:
  Port             Status  Priority Oper-Key  Flag
--------------------------------------------------------------------------------
  GE1/0/1          S       32768    1         {ACDEF}
```

```
  GE1/0/2           S       32768    1         {ACDEF}
  GE1/0/3           S       32768    1         {ACDEF}
Remote:
  Actor             Partner Priority Oper-Key  SystemID               Flag
--------------------------------------------------------------------------------
  GE1/0/1           1       32768    1         0x8000, 000f-e267-57ad {ACDEF}
  GE1/0/2           2       32768    1         0x8000, 000f-e267-57ad {ACDEF}
  GE1/0/3           3       32768    1         0x8000, 000f-e267-57ad {ACDEF}
```

以上信息表明，聚合组 1 为非负载分担类型的三层动态聚合组，包含有三个选中端口。

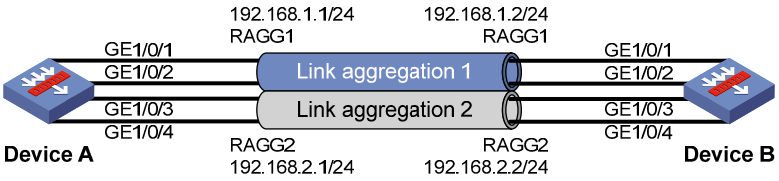
1.8.6 三层聚合负载分担配置举例

1. 组网需求

- Device A 与 Device B 通过各自的三层以太网接口 GigabitEthernet1/0/1～GigabitEthernet1/0/4 相互连接。
- 在 Device A 和 Device B 上分别配置两个三层静态链路聚合组，并为对应的三层聚合接口都配置 IP 地址和子网掩码。
- 通过在聚合组 1 上按照源 IP 地址进行聚合负载分担、在聚合组 2 上按照目的 IP 地址进行聚合负载分担的方式，来实现数据流量在各成员端口间的负载分担。

2. 组网图

图1-10 三层聚合负载分担配置组网图



3. 配置步骤

(1) 配置 Device A

创建三层聚合接口 1，配置该接口对应的聚合组内按照源 IP 地址进行聚合负载分担，并为其配置 IP 地址和子网掩码。

```
<DeviceA> system-view
[DeviceA] interface route-aggregation 1
[DeviceA-Route-Aggregation1] link-aggregation load-sharing mode source-ip
[DeviceA-Route-Aggregation1] ip address 192.168.1.1 24
[DeviceA-Route-Aggregation1] quit
```

分别将接口 GigabitEthernet1/0/1 和 GigabitEthernet1/0/2 加入到聚合组 1 中。

```
[DeviceA] interface gigabitethernet 1/0/1
[DeviceA-GigabitEthernet1/0/1] port link-aggregation group 1
[DeviceA-GigabitEthernet1/0/1] quit
[DeviceA] interface gigabitethernet 1/0/2
[DeviceA-GigabitEthernet1/0/2] port link-aggregation group 1
[DeviceA-GigabitEthernet1/0/2] quit
```

创建三层聚合接口 2，配置该接口对应的聚合组内按照目的 IP 地址进行聚合负载分担，并为其配置 IP 地址和子网掩码。

```
[DeviceA] interface route-aggregation 2
[DeviceA-Route-Aggregation2] link-aggregation load-sharing mode destination-ip
[DeviceA-Route-Aggregation2] ip address 192.168.2.1 24
[DeviceA-Route-Aggregation2] quit
```

分别将接口 GigabitEthernet1/0/3 和 GigabitEthernet1/0/4 加入到聚合组 2 中。

```
[DeviceA] interface gigabitethernet 1/0/3
[DeviceA-GigabitEthernet1/0/3] port link-aggregation group 2
[DeviceA-GigabitEthernet1/0/3] quit
[DeviceA] interface gigabitethernet 1/0/4
[DeviceA-GigabitEthernet1/0/4] port link-aggregation group 2
[DeviceA-GigabitEthernet1/0/4] quit
```

(2) 配置 Device B

Device B 的配置与 Device A 相似，配置过程略。

4. 验证配置

查看 Device A 上所有聚合组的详细信息。

```
[DeviceA] display link-aggregation verbose
Loadsharing Type: Shar -- Loadsharing, NonS -- Non-Loadsharing
Port Status: S -- Selected, U -- Unselected, I -- Individual
Flags:  A -- LACP_Activity, B -- LACP_Timeout, C -- Aggregation,
        D -- Synchronization, E -- Collecting, F -- Distributing,
        G -- Defaulted, H -- Expired

Aggregate Interface: Route-Aggregation1
Aggregation Mode: Static
Loadsharing Type: Shar
  Port             Status  Priority Oper-Key
--------------------------------------------------------------------------------
  GE1/0/1          S       32768    1
  GE1/0/2          S       32768    1

Aggregate Interface: Route-Aggregation2
Aggregation Mode: Static
Loadsharing Type: Shar
  Port             Status  Priority Oper-Key
--------------------------------------------------------------------------------
  GE1/0/3          S       32768    2
  GE1/0/4          S       32768    2
```

以上信息表明，聚合组 1 和聚合组 2 都是负载分担类型的三层静态聚合组，各包含有两个选中端口。

查看 Device A 上所有聚合接口所对应聚合组内采用的聚合负载分担类型。

```
[DeviceA] display link-aggregation load-sharing mode interface
Route-Aggregation1 Load-Sharing Mode:
source-ip address

Route-Aggregation2 Load-Sharing Mode:
```

```
destination-ip address
```
以上信息表明，三层聚合组 1 按照报文的源 IP 地址进行聚合负载分担，三层聚合组 2 按照报文的目的 IP 地址进行聚合负载分担。

目 录

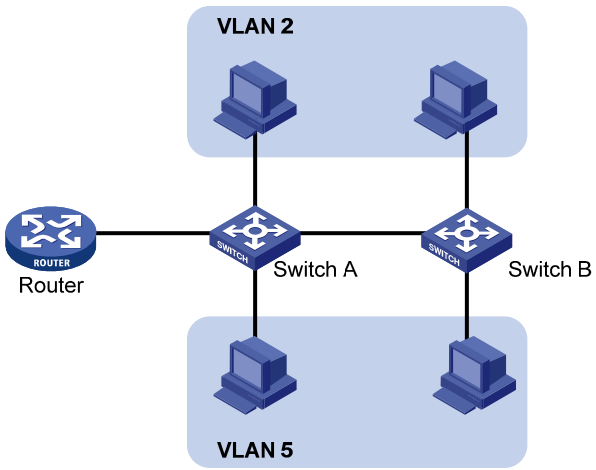
1 VLAN

1.1 VLAN简介

1.1.1 VLAN概述

以太网是一种基于CSMA/CD（Carrier Sense Multiple Access/Collision Detect，带冲突检测的载波侦听多路访问）技术的共享通讯介质。采用以太网技术构建的局域网，既是一个冲突域，又是一个广播域。当网络中主机数目较多时会导致冲突严重、广播泛滥、性能显著下降，甚至网络不可用等问题。通过在以太网中部署网桥或二层交换机，可以解决冲突严重的问题，但仍然不能隔离广播报文。在这种情况下出现了VLAN（Virtual Local Area Network，虚拟局域网）技术，这种技术可以把一个物理LAN划分成多个逻辑的LAN——VLAN。处于同一VLAN的主机能直接互通，而处于不同VLAN的主机则不能直接互通。这样，广播报文被限制在同一个VLAN内，即每个VLAN是一个广播域。如 图 1-1 所示，VLAN 2 内的主机可以互通，但与VLAN 5 内的主机不能互通。

图1-1 VLAN 示意图



VLAN 的划分不受物理位置的限制：物理位置不在同一范围的主机可以属于同一个 VLAN；一个 VLAN 包含的主机可以连接在同一个交换机上，也可以跨越交换机，甚至可以跨越路由器。

VLAN 根据划分方式不同可以分为不同类型。基于端口划分 VLAN 是其中最简单、最有效的 VLAN 划分方式。它按照设备端口来定义 VLAN 成员，将指定端口加入到指定 VLAN 中之后，端口就可以转发该 VLAN 的报文。本章将介绍基于端口的 VLAN。

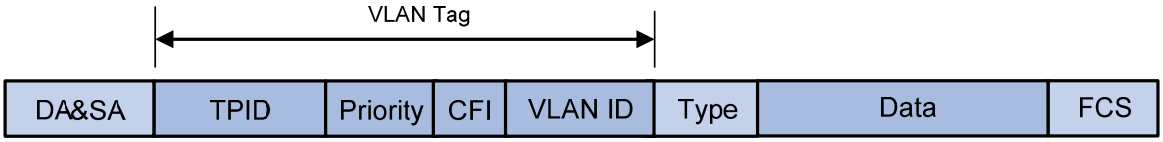
VLAN 的优点如下：

- 限制广播域。广播域被限制在一个 VLAN 内，节省了带宽，提高了网络处理能力。
- 增强局域网的安全性。VLAN 间的二层报文是相互隔离的，即一个 VLAN 内的主机不能和其他 VLAN 内的主机直接通信，如果不同 VLAN 要进行通信，则需通过路由器或三层交换机等三层设备。
- 灵活构建虚拟工作组。通过 VLAN 可以将不同的主机划分到不同的工作组，同一工作组的主机可以位于不同的物理位置，网络构建和维护更方便灵活。

1.1.2 VLAN报文封装

要使网络设备能够分辨不同 VLAN 的报文，需要在报文中添加标识 VLAN 的字段。IEEE 802.1Q 协议规定，在以太网报文的目的 MAC 地址和源 MAC 地址字段之后、协议类型字段之前加入 4 个字节的 VLAN Tag，用以标识 VLAN 的相关信息。

图1-2 VLAN Tag 的组成字段



如 图 1-2 所示，VLAN Tag包含四个字段，分别是TPID（Tag Protocol Identifier，标签协议标识符）、Priority、CFI（Canonical Format Indicator，标准格式指示位）和VLAN ID。

- TPID：协议规定 TPID 取值为 0x8100 时表示报文带有 VLAN Tag，但各设备厂商可以自定义该字段的值。当邻居设备将 TPID 值配置为非 0x8100 时，为了能够识别这样的报文，实现互通，必须在本设备上修改 TPID 值，确保和邻居设备的 TPID 值配置一致。如果报文的 TPID 值为配置值或0x8100，则该报文被认为带有VLAN Tag。配置TPID值的相关命令请参见“二层技术-以太网交换命令参考”中的“VLAN 终结”和“QinQ”。
- Priority：用来表示报文的 802.1p 优先级，长度为 3 比特，相关内容请参见“ACL 和 QoS 配置指导/QoS”中的“附录”。
- CFI：用来表示 MAC 地址在不同的传输介质中是否以标准格式进行封装，长度为 1 比特。取值为 0 表示 MAC 地址以标准格式进行封装，为 1 表示以非标准格式封装。在以太网中，CFI 取值为 0。
- VLAN ID：用来表示该报文所属 VLAN 的编号，长度为 12 比特。由于 0 和 4095 为协议保留取值，所以 VLAN ID 的取值范围为 1～4094。

网络设备根据报文是否携带VLAN Tag以及携带的VLAN Tag信息，来对报文进行处理，利用VLAN ID来识别报文所属的VLAN。详细的处理方式请参见“1.4.1 基于端口的VLAN简介”。

说明

- 以太网支持 Ethernet II、802.3/802.2 LLC、802.3/802.2 SNAP 和 802.3 raw 封装格式，本文以 Ethernet II 型封装为例。802.3/802.2 LLC、802.3/802.2 SNAP 和 802.3 raw 封装格式添加 VLAN Tag 字段的方式请参见相关协议规范。
- 对于携带有多层 VLAN Tag 的报文，设备会根据其最外层 VLAN Tag 进行处理，而内层 VLAN Tag 会被视为报文的普通数据部分。

1.1.3 协议规范

与 VLAN 相关的协议规范有：

- IEEE 802.1Q：IEEE Standard for Local and Metropolitan Area Networks-Virtual Bridged Local Area Networks

1.2 配置VLAN基本属性

表1-1 配置 VLAN 基本属性

配置	命令	说明
进入系统视图	**system-view**	-
（可选）创建一个VLAN并进入VLAN视图，或批量创建VLAN	**vlan** { *vlan-id1* [**to** *vlan-id2*] | **all** }	缺省情况下，系统只有一个缺省VLAN（VLAN 1）
进入VLAN视图	**vlan** *vlan-id*	批量创建VLAN时，为必选；否则，无需执行本命令
指定当前VLAN的名称	**name** *text*	缺省情况下，VLAN的名称为“VLAN *vlan-id*”，其中*vlan-id*为该VLAN的四位数编号，如果该VLAN的编号不足四位，则会在编号前增加0，补齐四位。例如，VLAN 100的名称为“VLAN 0100”
配置当前VLAN的描述信息	**description** *text*	缺省情况下，VLAN的描述信息为“VLAN *vlan-id*”，其中*vlan-id*为该VLAN的四位数编号，如果该VLAN的编号不足四位，则会在编号前增加0，补齐四位。例如，VLAN 100的描述信息为“VLAN 0100”

说明

- VLAN 1 为系统缺省 VLAN，用户不能手工创建和删除。
- 动态学习到的 VLAN，以及被其他应用锁定不让删除的 VLAN，都不能使用 **undo vlan** 命令直接删除。只有将相关配置删除之后，才能删除相应的 VLAN。

1.3 配置VLAN接口基本属性

不同 VLAN 间的主机不能直接通信，通过在设备上创建并配置 VLAN 接口，可以实现 VLAN 间的三层互通。

VLAN 接口是一种三层的虚拟接口，它不作为物理实体存在于设备上。每个 VLAN 对应一个 VLAN 接口，在为 VLAN 接口配置了 IP 地址后，该 IP 地址即可作为本 VLAN 内网络设备的网关地址，此时该 VLAN 接口能对需要跨网段的报文进行三层转发。

配置 VLAN 接口基本属性时，需要注意，在创建 VLAN 接口之前，对应的 VLAN 必须已经存在，否则将不能创建指定的 VLAN 接口。

表1-2 配置 VLAN 接口基本属性

配置	命令	说明
进入系统视图	**system-view**	-

配置	命令	说明
创建VLAN接口并进入VLAN接口视图	**interface vlan-interface** *interface-number*	如果该VLAN接口已经存在，则直接进入该VLAN接口视图 缺省情况下，不存在VLAN接口
配置VLAN接口的IP地址	**ip address** *ip-address* { *mask* | *mask-length* } [**sub**]	缺省情况下，未配置VLAN接口的IP地址
配置当前VLAN接口的描述信息	**description** *text*	缺省情况下，VLAN接口的描述信息为该VLAN接口的接口名，如“Vlan-interface1 Interface”
配置VLAN接口的MTU值	**mtu** *size*	缺省情况下，VLAN接口的MTU值为1500
（可选）配置VLAN接口的期望带宽	**bandwidth** *bandwidth-value*	缺省情况下，接口的期望带宽＝接口的波特率÷1000（kbps）
（可选）恢复当前VLAN接口的缺省配置	**default**	-
（可选）取消手工关闭VLAN接口	**undo shutdown**	缺省情况下，未手工关闭VLAN接口

1.4 配置基于端口的VLAN

1.4.1 基于端口的VLAN简介

基于端口划分 VLAN 是最简单、最有效的 VLAN 划分方法。它按照设备端口来定义 VLAN 成员，将指定端口加入到指定 VLAN 中之后，该端口就可以转发该 VLAN 的报文。

用户可以配置端口的链路类型及缺省 VLAN，其中，链路类型决定了端口能否加入多个 VLAN。

1. 端口的链路类型

端口的链路类型分为三种，不同链路类型的端口在转发报文时对 VLAN Tag 的处理方式不同：

- Access：端口只能发送一个 VLAN 的报文，发出去的报文不带 VLAN Tag。一般用于和不能识别 VLAN Tag 的用户终端设备相连，或者不需要区分不同 VLAN 成员时使用。
- Trunk：端口能发送多个 VLAN 的报文，发出去的端口缺省 VLAN 的报文不带 VLAN Tag，其他 VLAN 的报文都必须带 VLAN Tag。通常用于网络传输设备之间的互连。
- Hybrid：端口能发送多个 VLAN 的报文，端口发出去的报文可根据需要配置某些 VLAN 的报文带 VLAN Tag，某些 VLAN 的报文不带 VLAN Tag。
- 端口缺省 VLAN

除了可以配置端口允许通过的 VLAN 外，还可以配置端口的缺省 VLAN，即端口 VLAN ID（Port VLAN ID，PVID）。当端口收到 Untagged 报文时，会认为该报文所属的 VLAN 为缺省 VLAN。

- Access 端口的缺省 VLAN 就是它所在的 VLAN。
- Trunk 端口和 Hybrid 端口可以允许多个 VLAN 通过，能够配置端口缺省 VLAN。
- 当执行 **undo vlan** 命令删除的 VLAN 是某个端口的缺省 VLAN 时，对 Access 端口，端口的缺省 VLAN 会恢复到 VLAN 1；对 Trunk 或 Hybrid 端口，端口的缺省 VLAN 配置不会改变，即它们可以使用已经不存在的 VLAN 作为端口缺省 VLAN。

说明

- 建议本端设备端口的缺省 VLAN ID 和相连的对端设备端口的缺省 VLAN ID 保持一致。
- 建议保证端口的缺省 VLAN 为端口允许通过的 VLAN。如果端口不允许某 VLAN 通过，但是端口的缺省 VLAN 为该 VLAN，则端口会丢弃收到的该 VLAN 的报文或者不带 VLAN Tag 的报文。

2. 端口对报文的处理方式

在配置了端口链路类型和端口缺省VLAN后，端口对报文的接收和发送的处理有几种不同情况，具体情况请参看 表 1-3。

表1-3 不同链路类型端口收发报文的差异

端口类型	对接收报文的处理		对发送报文的处理
	当接收到的报文不带 Tag 时	当接收到的报文带有 Tag 时	
Access端口	为报文添加端口缺省VLAN的Tag	• 当报文的 VLAN ID 与端口的缺省 VLAN ID 相同时，接收该报文 • 当报文的 VLAN ID 与端口的缺省 VLAN ID 不同时，丢弃该报文	去掉Tag，发送该报文
Trunk端口	• 当端口的缺省 VLAN ID 在端口允许通过的 VLAN ID 列表里时，接收该报文，给报文添加端口缺省 VLAN 的 Tag • 当端口的缺省 VLAN ID 不在端口允许通过的 VLAN ID 列表里时，丢弃该报文	• 当报文的 VLAN ID 在端口允许通过的 VLAN ID 列表里时，接收该报文 • 当报文的 VLAN ID 不在端口允许通过的 VLAN ID 列表里时，丢弃该报文	• 当报文的 VLAN ID 与端口的缺省 VLAN ID 相同，且是该端口允许通过的 VLAN ID 时：去掉 Tag，发送该报文 • 当报文的 VLAN ID 与端口的缺省 VLAN ID 不同，且是该端口允许通过的 VLAN ID 时：保持原有 Tag，发送该报文
Hybrid端口			当报文的VLAN ID是端口允许通过的VLAN ID时，发送该报文，并可以通过**port hybrid vlan**命令配置端口在发送该VLAN的报文时是否携带Tag

1.4.2 配置基于Access端口的VLAN

配置基于 Access 端口的 VLAN 有两种方法：一种是在 VLAN 视图下进行配置，另一种是在接口视图下进行配置。

表1-4 配置基于 Access 端口的 VLAN（在 VLAN 视图下）

配置	命令	说明
进入系统视图	**system-view**	-

配置	命令	说明
进入VLAN视图	**vlan** *vlan-id*	-
向当前VLAN中添加一个或一组Access端口	**port** *interface-list*	缺省情况下，系统将所有端口都加入到VLAN 1

表1-5 配置基于 Access 端口的 VLAN（在接口视图下）

操作		命令	说明
进入系统视图		**system-view**	-
进入相应视图	二层以太网接口视图	**interface** *interface-type interface-number*	-
	二层聚合接口视图	**interface bridge-aggregation** *interface-number*	
配置端口的链路类型为Access类型		**port link-type access**	缺省情况下，端口的链路类型为Access
将当前Access端口加入到指定VLAN		**port access vlan** *vlan-id*	缺省情况下，所有Access端口都属于VLAN 1 在将Access端口加入到指定VLAN之前，该VLAN必须已经存在

1.4.3 配置基于Trunk端口的VLAN

Trunk 端口可以允许多个 VLAN 通过，只能在接口视图下进行配置。

配置基于 Trunk 端口的 VLAN 时，需要注意：

- Trunk 端口不能直接切换为 Hybrid 端口，只能先将 Trunk 端口配置为 Access 端口，再配置为 Hybrid 端口。
- 配置端口缺省 VLAN 后，必须使用 **port trunk permit vlan** 命令配置允许端口缺省 VLAN 的报文通过，接口才能转发端口缺省 VLAN 的报文。

表1-6 配置基于 Trunk 端口的 VLAN

操作		命令	说明
进入系统视图		**system-view**	-
进入相应视图	二层以太网接口视图	**interface** *interface-type interface-number*	-
	二层聚合接口视图	**interface bridge-aggregation** *interface-number*	
配置端口的链路类型为Trunk类型		**port link-type trunk**	缺省情况下，端口的链路类型为Access类型
允许指定的VLAN通过当前Trunk端口		**port trunk permit vlan** { *vlan-id-list* | **all** }	缺省情况下，Trunk端口只允许VLAN 1的报文通过
（可选）配置Trunk端口的缺省VLAN		**port trunk pvid vlan** *vlan-id*	缺省情况下，Trunk端口的缺省VLAN为VLAN 1

1.4.4 配置基于Hybrid端口的VLAN

Hybrid 端口可以允许多个 VLAN 通过，只能在接口视图下进行配置。

配置基于 Hybrid 端口的 VLAN 时，需要注意：

- Hybrid 端口不能直接切换为 Trunk 端口，只能先将 Hybrid 端口配置为 Access 端口，再配置为 Trunk 端口。
- 在配置允许指定的 VLAN 通过 Hybrid 端口之前，允许通过的 VLAN 必须已经存在。
- 配置端口缺省 VLAN 后，必须使用 **port hybrid vlan** 命令配置允许端口缺省 VLAN 的报文通过，出接口才能转发端口缺省 VLAN 的报文。

表1-7 配置基于 Hybrid 端口的 VLAN

操作		命令	说明
进入系统视图		**system-view**	-
进入相应视图	二层以太网接口视图	**interface** *interface-type interface-number*	-
	二层聚合接口视图	**interface bridge-aggregation** *interface-number*	
配置端口的链路类型为Hybrid类型		**port link-type hybrid**	缺省情况下，端口的链路类型为Access类型
允许指定的VLAN通过当前Hybrid端口		**port hybrid vlan** *vlan-id-list* { **tagged** | **untagged** }	缺省情况下，Hybrid端口只允许该端口在链路类型为Access时的所属VLAN的报文以Untagged方式通过
（可选）配置Hybrid端口的缺省VLAN		**port hybrid pvid vlan** *vlan-id*	缺省情况下，Hybrid端口的缺省VLAN为该端口在链路类型为Access时的所属VLAN

1.5 配置VLAN组

VLAN 组是一组 VLAN 的集合。VLAN 组内可以添加多个 VLAN 列表，一个 VLAN 列表表示一组 VLAN ID 连续的 VLAN。

表1-8 配置 VLAN 组

操作	命令	说明
进入系统视图	**system-view**	-
创建一个VLAN组，并进入VLAN组视图	**vlan-group** *group-name*	缺省情况下，不存在VLAN组
在当前VLAN组内添加VLAN成员	**vlan-list** *vlan-id-list*	缺省情况下，当前VLAN组中不存在VLAN列表 可以多次在当前VLAN组内添加VLAN成员

1.6 VLAN显示和维护

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

在用户视图下执行 **reset** 命令可以清除 VLAN 接口统计信息。

表1-9 VLAN 显示和维护

操作	命令
显示VLAN接口相关信息	**display interface vlan-interface** [*interface-number*] [**brief** [**description** | **down**]]
显示VLAN相关信息	**display vlan** [*vlan-id1* [**to** *vlan-id2*] | **all** | **dynamic** | **reserved** | **static**]
显示设备上所有已创建VLAN的概要信息	**display vlan brief**
显示创建的VLAN组及其VLAN成员列表	**display vlan-group** [*group-name*]
显示设备上当前存在的Hybrid或Trunk端口	**display port** { **hybrid** | **trunk** }
清除VLAN接口的统计信息	**reset counters interface vlan-interface** [*interface-number*]

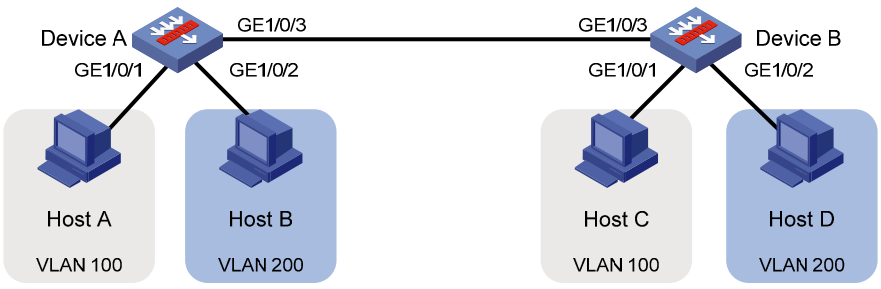
1.7 基于端口的VLAN典型配置举例

1. 组网需求

- Host A 和 Host C 属于部门 A，但是通过不同的设备接入公司网络；Host B 和 Host D 属于部门 B，也通过不同的设备接入公司网络。
- 为了通信的安全性，也为了避免广播报文泛滥，公司网络中使用 VLAN 技术来隔离部门间的二层流量。其中部门 A 使用 VLAN 100，部门 B 使用 VLAN 200。

2. 组网图

图1-3 基于端口的 VLAN 组网图



3. 配置步骤

(1) 配置 Device A

创建 VLAN 100，并将 GigabitEthernet1/0/1 加入 VLAN 100。

```
<DeviceA> system-view
```

```
[DeviceA] vlan 100
[DeviceA-vlan100] port gigabitethernet 1/0/1
[DeviceA-vlan100] quit
```

创建 VLAN 200，并将 GigabitEthernet1/0/2 加入 VLAN 200。

```
[DeviceA] vlan 200
[DeviceA-vlan200] port gigabitethernet 1/0/2
[DeviceA-vlan200] quit
```

为了使 Device A 上 VLAN 100 和 VLAN 200 的报文能发送给 Device B，将 GigabitEthernet1/0/3 的链路类型配置为 Trunk，并允许 VLAN 100 和 VLAN 200 的报文通过。

```
[DeviceA] interface gigabitethernet 1/0/3
[DeviceA-GigabitEthernet1/0/3] port link-type trunk
[DeviceA-GigabitEthernet1/0/3] port trunk permit vlan 100 200
```

(2) Device B 上的配置与 Device A 上的配置相同，不再赘述。

(3) 将 Host A 和 Host C 配置在一个网段，比如 192.168.100.0/24；将 Host B 和 Host D 配置在一个网段，比如 192.168.200.0/24。

4. 验证配置

(1) Host A 和 Host C 能够互相 ping 通，但是均不能 ping 通 Host B 和 Host D。Host B 和 Host D 能够互相 ping 通，但是均不能 ping 通 Host A 和 Host C。

(2) 通过查看显示信息验证配置是否成功。

查看 Device A 上 VLAN 100 和 VLAN 200 的配置信息，验证以上配置是否生效。

```
[DeviceA-GigabitEthernet1/0/3] display vlan 100
 VLAN ID: 100
 VLAN type: Static
 Route interface: Not configured
 Description: VLAN 0100
 Name: VLAN 0100
 Tagged ports:
    GigabitEthernet1/0/3
 Untagged ports:
    GigabitEthernet1/0/1
[DeviceA-GigabitEthernet1/0/3] display vlan 200
 VLAN ID: 200
 VLAN type: Static
 Route interface: Not configured
 Description: VLAN 0200
 Name: VLAN 0200
 Tagged ports:
    GigabitEthernet1/0/3
 Untagged ports:
    GigabitEthernet1/0/2
```

目 录

1 VLAN终结

1.1 VLAN终结简介

VLAN 终结是指对接收到的报文，按照报文携带的 VLAN Tag 信息匹配对应的接口后，去除报文 VLAN Tag，再将报文进行三层转发或交由其他业务处理。转发出去的报文是否带有 VLAN Tag 由出接口决定，对从配置了 VLAN 终结的接口发送的报文，按照该接口上的终结配置，将相应的 VLAN Tag 添加到报文中后发送该报文。

1.1.1 VLAN终结分类

根据对所终结的报文的不同处理方式，VLAN 终结分为以下三种：

- Dot1q 终结：用来终结带有一层及以上 VLAN Tag 的报文（要求最外层 VLAN ID 必须匹配配置值），从配置了 Dot1q 终结的接口发送的报文，都添加一层 VLAN Tag。
- Untagged 终结：用来终结收到的不带 VLAN Tag 的报文，从配置了 Untagged 终结的接口发送的报文，都不添加 VLAN Tag。
- Default 终结：用来终结同一主接口上其他子接口上无法处理的报文，从配置了 Default 终结的接口发送的报文，都不添加 VLAN Tag。

说明

不支持将配置模糊 VLAN 终结的子接口共享分配给 Context，因为目前 Context 下下发引流规则时，无法获取 VLAN ID 范围

为便于描述，本特性部分内容对带有两层及以上 VLAN Tag 的报文，将其最外两层 VLAN Tag 按从外层到内层的方向，分别用第一层 VLAN Tag、第二层 VLAN Tag 表示，对 VLAN ID 的描述类似。

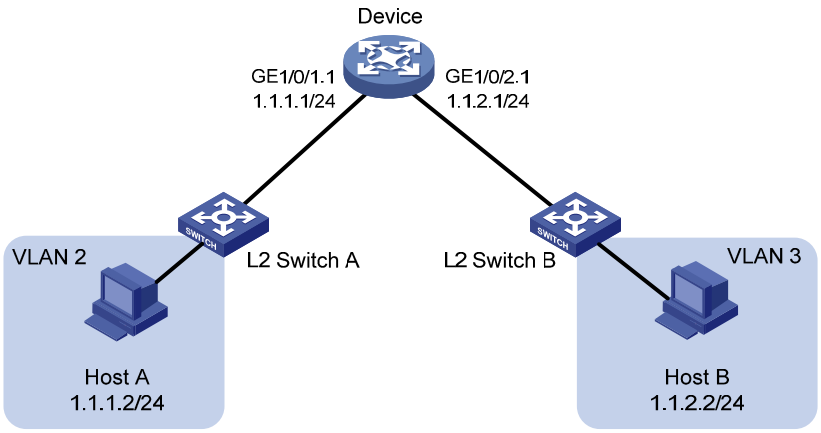
1.1.2 VLAN终结应用场景

1. 指定VLAN间的互通

划分 VLAN 后，不同 VLAN 间的主机不能直接通信，使用三层路由技术可以实现所有 VLAN 间报文的互通。此时如果要对互通的 VLAN 范围做限制，即要求只有指定的部分 VLAN 间可以互通，可以借助 VLAN 终结功能来实现。目前可以通过三层以太网子接口/三层聚合子接口实现指定 VLAN 间的互通。

如下图所示，Host A 属于 VLAN 2，Host B 属于 VLAN 3，将 Host A 的网关地址指定为 1.1.1.1/24，Host B 的网关地址指定为 1.1.2.1/24，就可以通过在三层以太网子接口 GigabitEthernet1/0/1.1 和 GigabitEthernet1/0/2.1 上配置 VLAN 终结来实现 Host A 和 Host B 之间的互通了。

图1-1 VLAN 终结用于不同 VLAN 之间互通



1.2 VLAN终结配置任务简介

表1-1 VLAN 终结配置任务简介

配置任务		说明	详细配置
配置Dot1q终结	配置明确的Dot1q终结	请根据设备的支持情况选择一种方式	1.3.1
	配置模糊的Dot1q终结		1.3.2
配置Untagged终结			1.4
配置Default终结			1.5
配置VLAN终结支持广播/组播		可选	1.6
配置VLAN Tag的TPID值		可选	1.7

1.3 配置Dot1q终结

根据每个子接口所能终结的 VLAN 报文中最外层 VLAN ID 范围的不同，Dot1q 终结分为：

- 明确的 Dot1q 终结：只允许接收最外层 VLAN ID 为指定值的 VLAN 报文，其他 VLAN 报文则不允许通过该子接口。收到报文后，将报文最外层 VLAN Tag 剥离。发送报文时，给报文添加一层 VLAN Tag，VLAN ID 为指定值。
- 模糊的 Dot1q 终结：只允许接收最外层 VLAN ID 在指定范围内的 VLAN 报文，不属于该范围的 VLAN 报文则不允许通过该子接口。收到报文后，将报文最外层 VLAN Tag 剥离。发送报文时，会给报文添加一层 VLAN Tag，对于 DHCP Relay 转发的 DHCP Server 端报文，通过查找 DHCP 会话表项获取相应的 VLAN ID。

1.3.1 配置明确的Dot1q终结

表1-2 配置明确的 Dot1q 终结

操作		命令	说明
进入系统视图		**system-view**	-
进入接口视图	三层以太网子接口视图	**interface** *interface-type interface-number.subnumber*	-
	三层聚合子接口视图	**interface route-aggregation** *interface-number.subnumber*	
开启当前接口的Dot1q终结功能，并指定当前子接口能够终结的VLAN报文最外层VLAN ID		**vlan-type dot1q vid** *vlan-id* [**loose**]	缺省情况下，未开启接口的Dot1q终结功能

1.3.2 配置模糊的Dot1q终结

表1-3 配置模糊的 Dot1q 终结

操作		命令	说明
进入系统视图		**system-view**	-
进入接口视图	三层以太网子接口视图	**interface** *interface-type interface-number.subnumber*	-
	三层聚合子接口视图	**interface route-aggregation** *interface-number.subnumber*	
开启当前接口的Dot1q终结功能，并指定当前子接口能够终结的VLAN报文的最外层VLAN ID范围		**vlan-type dot1q vid** *vlan-id-list* [**loose**]	缺省情况下，未开启接口的Dot1q终结功能

1.4 配置Untagged终结

表1-4 配置 Untagged 终结

操作		命令	说明
进入系统视图		**system-view**	-
进入接口视图	三层以太网子接口视图	**interface** *interface-type interface-number.subnumber*	-
	三层聚合子接口视图	**interface route-aggregation** *interface-number.subnumber*	
开启当前接口的Untagged终结功能，使当前接口可以处理不带VLAN Tag的报文		**vlan-type dot1q untagged**	缺省情况下，未开启接口的Untagged终结功能

1.5 配置Default终结

表1-5 配置 Default 终结

操作		命令	说明
进入系统视图		system-view	-
进入接口视图	三层以太网子接口视图	**interface** *interface-type interface-number.subnumber*	-
	三层聚合子接口视图	**interface route-aggregation** *interface-number.subnumber*	
使能当前接口的Default终结功能，使当前接口可以处理其他子接口都无法处理的报文		**vlan-type dot1q default**	缺省情况下，未开启接口的Default终结功能

1.6 配置VLAN终结支持广播/组播

当接口下配置了模糊的 Dot1q 终结功能后，不允许发送广播、组播报文。只有配置了 VLAN 终结支持广播/组播功能，这些接口才能发送广播/组播报文。

表1-6 配置 VLAN 终结支持广播/组播

操作		命令	说明
进入系统视图		**system-view**	-
进入接口视图	三层以太网子接口视图	**interface** *interface-type interface-number.subnumber*	-
	三层聚合子接口视图	**interface route-aggregation** *interface-number.subnumber*	
配置允许当前接口发送广播和组播报文		**vlan-termination broadcast enable**	缺省情况下，当前接口配置了模糊的Dot1q终结功能后，不允许发送广播、组播报文

1.7 配置VLAN Tag的TPID值

如果要在三层以太网子接口/三层聚合子接口上使用 VLAN 终结功能，可以通过以下配置指定接口接收和发送报文的最外层 VLAN Tag 的 TPID 值。在配置 TPID 值后，当接收报文时，只有报文最外层 VLAN Tag 的 TPID 值为 0x8100 或者指定值的报文才会作为 VLAN 报文来处理；发送报文时，会给报文最外层 VLAN Tag 的 TPID 值填入指定值，如果报文带有两层及以上 VLAN Tag，则给报文其他层 VLAN Tag 的 TPID 值都填入 0x8100。

表1-7 配置 VLAN Tag 的 TPID 值

操作	命令	说明
进入系统视图	**system-view**	-

操作		命令	说明
进入接口视图	三层以太网接口视图	**interface** *interface-type interface-number*	在三层以太网接口、三层聚合接口视图下配置，会对相应接口的所有子接口生效
	三层聚合接口视图	**interface route-aggregation** *interface-number*	
配置当前接口接收和发送的报文最外层VLAN Tag的TPID值		**dot1q ethernet-type** *hex-value*	缺省情况下，当前接口接收和发送的报文最外层VLAN Tag的TPID值均为0x8100

1.8 VLAN终结配置举例

1.8.1 明确的Dot1q终结配置举例

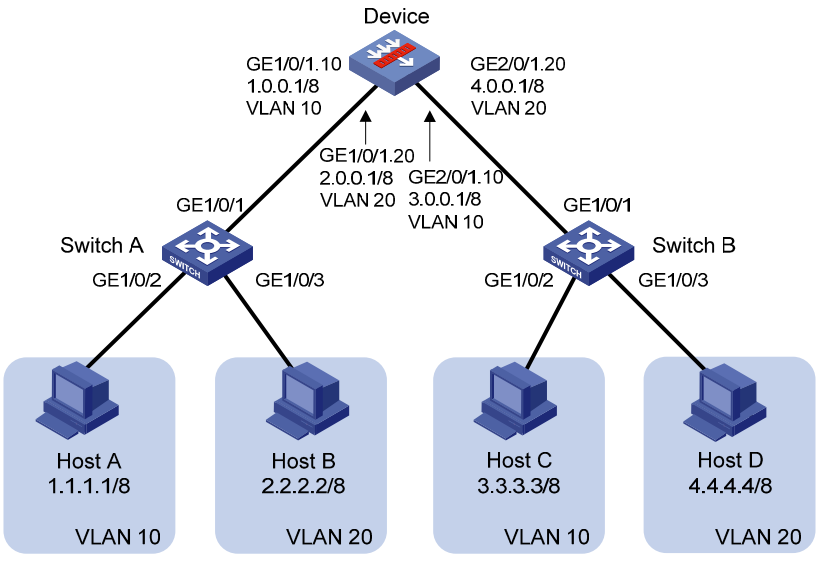
1. 组网需求

如下图所示，Host A、Host B 和 Switch A 相连，Host C、Host D 和 Switch B 相连。Host A 和 Host C 属于 VLAN 10，Host B 和 Host D 属于 VLAN 20。Device 子接口 GigabitEthernet1/0/1.10、GigabitEthernet1/0/1.20、GigabitEthernet2/0/1.10、GigabitEthernet2/0/1.20 的 IP 地址分别为 1.0.0.1/8、2.0.0.1/8、3.0.0.1/8 和 4.0.0.1/8。要求实现：

- Host A 和 Host B 之间、Host C 和 Host D 之间能够互相通信，即同一交换机、不同 VLAN 之间能够互相通信；
- Host A 和 Host C 之间、Host B 和 Host D 之间能够互相通信，即不同交换机、同一 VLAN 之间能够互相通信；
- Host A 和 Host D 之间、Host B 和 Host C 之间能够互相通信，即不同交换机、不同 VLAN 之间能够互相通信。

2. 组网图

图1-2 明确的 Dot1q 终结配置组网图



3. 配置步骤

(1) Host A、Host B、Host C、Host D 的配置

将 Host A 的 IP 地址指定为 1.1.1.1/8，网关地址指定为 1.0.0.1/8。

将 Host B 的 IP 地址指定为 2.2.2.2/8，网关地址指定为 2.0.0.1/8。

将 Host C 的 IP 地址指定为 3.3.3.3/8，网关地址指定为 3.0.0.1/8。

将 Host D 的 IP 地址指定为 4.4.4.4/8，网关地址指定为 4.0.0.1/8。

(2) L2 Switch A、L2 Switch B 的配置

下面以 L2 Switch A 的配置步骤为例，L2 Switch B 的配置与 L2 Switch A 的配置相同，不再赘述。

向 VLAN 10 中加入端口 GigabitEthernet1/0/2。

```
<L2_SwitchA> system-view
[L2_SwitchA] vlan 10
[L2_SwitchA-vlan10] port gigabitethernet 1/0/2
[L2_SwitchA-vlan10] quit
```

向 VLAN 20 中加入端口 GigabitEthernet1/0/3。

```
[L2_SwitchA] vlan 20
[L2_SwitchA-vlan20] port gigabitethernet 1/0/3
[L2_SwitchA-vlan20] quit
```

配置端口 GigabitEthernet1/0/1 的链路类型为 Trunk 类型，并允许 VLAN 10 和 VLAN 20 通过。

```
[L2_SwitchA] interface gigabitethernet 1/0/1
[L2_SwitchA-GigabitEthernet1/0/1] port link-type trunk
[L2_SwitchA-GigabitEthernet1/0/1] port trunk permit vlan 10 20
```

(3) Device 的配置

创建以太网子接口 GigabitEthernet1/0/1.10、GigabitEthernet1/0/1.20、GigabitEthernet2/0/1.10 和 GigabitEthernet2/0/1.20，分别为其配置 IP 地址，配置 GigabitEthernet1/0/1.10 和 GigabitEthernet2/0/1.10 用来终结 VLAN 10 的报文，配置 GigabitEthernet1/0/1.20 和 GigabitEthernet2/0/1.20 用来终结 VLAN 20 的报文。

```
<Device> system-view
[Device] interface gigabitethernet 1/0/1.10
[Device-GigabitEthernet1/0/1.10] ip address 1.0.0.1 255.0.0.0
[Device-GigabitEthernet1/0/1.10] vlan-type dot1q vid 10
[Device-GigabitEthernet1/0/1.10] quit
[Device] interface gigabitethernet 1/0/1.20
[Device-GigabitEthernet1/0/1.20] ip address 2.0.0.1 255.0.0.0
[Device-GigabitEthernet1/0/1.20] vlan-type dot1q vid 20
[Device-GigabitEthernet1/0/1.20] quit
[Device] interface gigabitethernet 2/0/1.10
[Device-GigabitEthernet2/0/1.10] ip address 3.0.0.1 255.0.0.0
[Device-GigabitEthernet2/0/1.10] vlan-type dot1q vid 10
[Device-GigabitEthernet2/0/1.10] quit
[Device] interface gigabitethernet 2/0/1.20
[Device-GigabitEthernet2/0/1.20] ip address 4.0.0.1 255.0.0.0
[Device-GigabitEthernet2/0/1.20] vlan-type dot1q vid 20
[Device-GigabitEthernet2/0/1.20] quit
```

4. 验证配置

配置完成后，Host A、Host B、Host C、Host D 可以相互 ping 通。

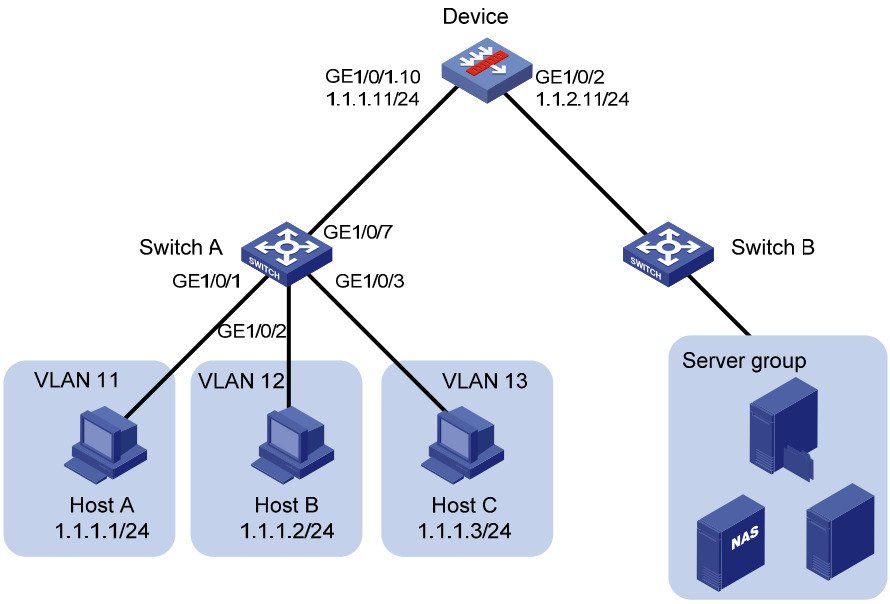
1.8.2 模糊的Dot1q终结配置举例

1. 组网需求

如下图所示，Host A、Host B、Host C 和 Switch A 相连，Server group 和 Switch B 相连。Host A、Host B、Host C 分别属于 VLAN 11、VLAN 12、VLAN 13；这些 Host 需要与 Server group 互相通信。

2. 组网图

图1-3 模糊的 Dot1q 终结配置组网图



3. 配置步骤

(1) Host A、Host B、Host C 的配置

将 Host A 的 IP 地址指定为 1.1.1.1/24，Host B 的 IP 地址为 1.1.1.2/24，Host C 的 IP 地址为 1.1.1.3/24，网关地址均指定为 1.1.1.11/24。

(2) L2 Switch A 的配置

向 VLAN 11 中加入端口 GigabitEthernet1/0/1。

```
<L2_SwitchA> system-view
[L2_SwitchA] vlan 11
[L2_SwitchA-vlan11] port gigabitethernet 1/0/1
[L2_SwitchA-vlan11] quit
```

向 VLAN 12 中加入端口 GigabitEthernet1/0/2。

```
[L2_SwitchA] vlan 12
[L2_SwitchA-vlan12] port gigabitethernet 1/0/2
[L2_SwitchA-vlan12] quit
```

向 VLAN 13 中加入端口 GigabitEthernet1/0/3。

```
[L2_SwitchA] vlan 13
[L2_SwitchA-vlan13] port gigabitethernet 1/0/3
[L2_SwitchA-vlan13] quit
```

配置端口 GigabitEthernet1/0/7 的链路类型为 Trunk 类型，并允许 VLAN 11～13 通过。

```
[L2_SwitchA] interface gigabitethernet 1/0/7
[L2_SwitchA-GigabitEthernet1/0/7] port link-type trunk
[L2_SwitchA-GigabitEthernet1/0/7] port trunk permit vlan 11 to 13
```

(3) Device 的配置

创建以太网子接口 GigabitEthernet1/0/1.10，为其配置 IP 地址，开启 Dot1q 终结功能，指定终结最外层 VLAN ID 在范围 11～13 内的报文，并允许该子接口发送广播、组播报文。

```
<Device> system-view
[Device] interface gigabitethernet 1/0/1.10
[Device-GigabitEthernet1/0/1.10] ip address 1.1.1.11 255.255.255.0
[Device-GigabitEthernet1/0/1.10] vlan-type dot1q vid 11 to 13
[Device-GigabitEthernet1/0/1.10] vlan-termination broadcast enable
[Device-GigabitEthernet1/0/1.10] quit
```

配置以太网接口 GigabitEthernet1/0/2 的 IP 地址。

```
[Device] interface gigabitethernet 1/0/2
[Device-GigabitEthernet1/0/2] ip address 1.1.2.11 255.255.255.0
```

(4) L2 Switch B 的配置

L2 Switch B 采用出厂配置即可。

(5) Server group 的配置

将 Server group 里所有设备的 IP 地址配置在 1.1.2.0/24 网段，网关地址指定为 1.1.2.11/24 即可。

4. 验证配置

配置完成后， Host A、Host B、Host C 可以与 Server group 相互 ping 通。

目 录

1 LLDP

说明

对于自定义 context，不支持配置本特性。

1.1 LLDP简介

1.1.1 LLDP产生背景

目前，网络设备的种类日益繁多且各自的配置错综复杂，为了使不同厂商的设备能够在网络中相互发现并交互各自的系统及配置信息，需要有一个标准的信息交流平台。

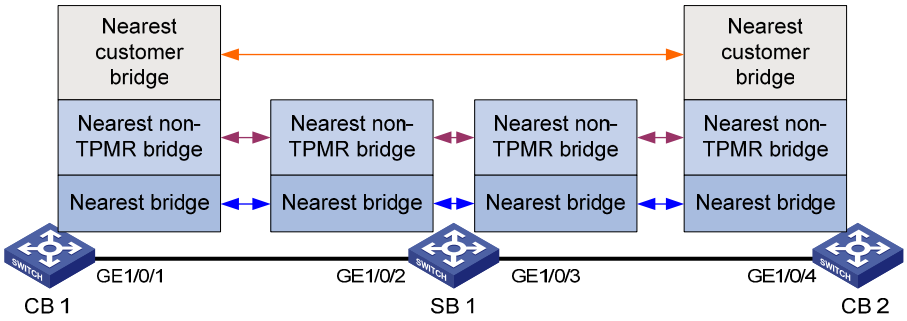
LLDP（Link Layer Discovery Protocol，链路层发现协议）就是在这样的背景下产生的，它提供了一种标准的链路层发现方式，可以将本端设备的信息（包括主要能力、管理地址、设备标识、接口标识等）组织成不同的 TLV（Type/Length/Value，类型/长度/值），并封装在 LLDPDU（Link Layer Discovery Protocol Data Unit，链路层发现协议数据单元）中发布给与自己直连的邻居，邻居收到这些信息后将其以标准 MIB（Management Information Base，管理信息库）的形式保存起来，以供网络管理系统查询及判断链路的通信状况。有关 MIB 的详细介绍，请参见“网络管理和监控配置指导”中的“SNMP”。

1.1.2 LLDP基本概念

1. LLDP代理

LLDP代理是LLDP协议运行实体的一个抽象映射。一个接口下，可以运行多个LLDP代理。目前LLDP定义的代理类型包括：Nearest Bridge（最近桥代理）、Nearest non-TPMR Bridge（最近非TPMR桥代理）和Nearest Customer Bridge（最近客户桥代理）。其中TPMR（Two-Port MAC Relay，双端口MAC中继），是一种只有两个可供外部访问桥端口的桥，支持MAC桥的功能子集。TPMR对于所有基于帧的介质无关协议都是透明的，但如下协议除外：以TPMR为目的地的协议、以保留MAC地址为目的地址但TPMR定义为不予转发的协议。LLDP在相邻的代理之间进行协议报文交互，并基于代理创建及维护邻居信息。如 图 1-1 所示，是LLDP不同类型的代理邻居关系示意图。其中，CB（Customer Bridge，客户桥）和SB（Service Bridge，服务桥）表示LLDP的两种桥模式。

图1-1 LLDP 邻居关系示意图

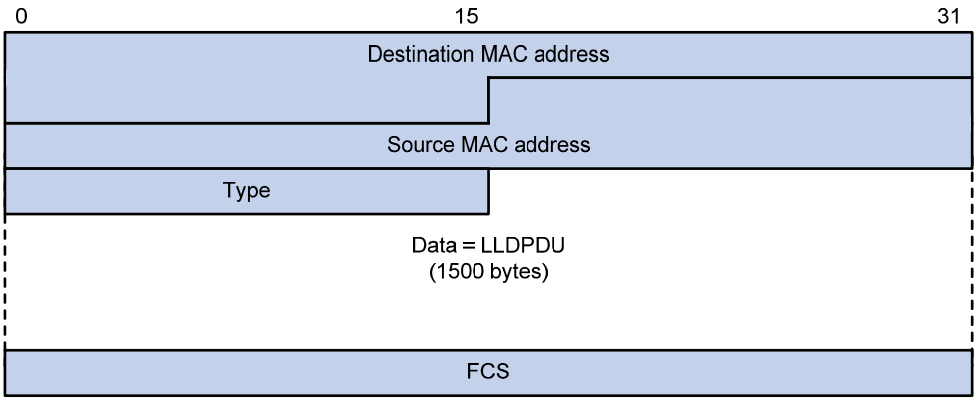


2. LLDP报文

封装有 LLDPDU 的报文称为 LLDP 报文，其封装格式有两种：Ethernet II 和 SNAP（Subnetwork Access Protocol，子网访问协议）。

(1) Ethernet II 格式封装的 LLDP 报文

图1-2 Ethernet II 格式封装的 LLDP 报文

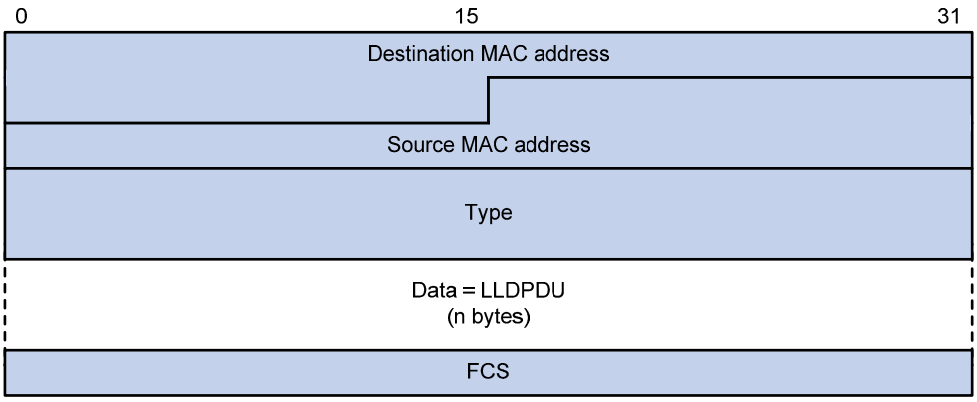


如 图 1-2 所示，是以Ethernet II格式封装的LLDP报文，其中各字段的含义如下：

- Destination MAC address：目的 MAC 地址。
- Source MAC address：源 MAC 地址，为端口 MAC 地址。
- Type：报文类型，为 0x88CC。
- Data：数据内容，为 LLDPDU。
- FCS：帧检验序列，用来对报文进行校验。

(2) SNAP 格式封装的 LLDP 报文

图1-3 SNAP 格式封装的 LLDP 报文



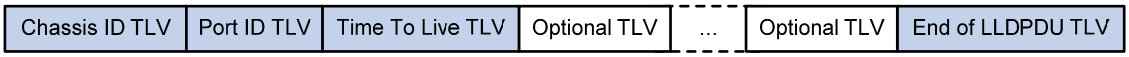
如 图 1-3 所示，是以SNAP格式封装的LLDP报文，其中各字段的含义如下：

- Destination MAC address：目的 MAC 地址，与 Ethernet II 格式封装的 LLDP 报文目的 MAC 地址相同。
- Source MAC address：源 MAC 地址，为端口 MAC 地址。
- Type：报文类型，为 0xAAAA-0300-0000-88CC。
- Data：数据内容，为 LLDPDU。
- FCS：帧检验序列，用来对报文进行校验。

3. LLDPDU

LLDPDU 就是封装在 LLDP 报文数据部分的数据单元。在组成 LLDPDU 之前，设备先将本地信息封装成 TLV 格式，再由若干个 TLV 组合成一个 LLDPDU 封装在 LLDP 报文的数据部分进行传送。

图1-4 LLDPDU 的封装格式



如 图 1-4 所示，蓝色的Chassis ID TLV、Port ID TLV、Time To Live TLV和End of LLDPDU TLV这四种TLV是每个LLDPDU都必须携带的，其余的TLV则为可选携带。每个LLDPDU最多可携带 32 种 TLV。

4. TLV

TLV 是组成 LLDPDU 的单元，每个 TLV 都代表一个信息。LLDP 可以封装的 TLV 包括基本 TLV、802.1 组织定义 TLV、802.3 组织定义 TLV 和 LLDP-MED（Link Layer Discovery Protocol Media Endpoint Discovery，链路层发现协议媒体终端发现） TLV。

基本 TLV 是网络设备管理基础的一组 TLV，802.1 组织定义 TLV、802.3 组织定义 TLV 和 LLDP-MED TLV 则是由标准组织或其他机构定义的 TLV，用于增强对网络设备的管理，可根据实际需要选择是否在 LLDPDU 中发送。

(1) 基本 TLV

在基本TLV中，有几种TLV对于实现LLDP功能来说是必选的，即必须在LLDPDU中发布，如 表 1-1 所示。

表1-1 基本 TLV

TLV 名称	说明	是否必须发布
Chassis ID	发送设备的桥MAC地址	是
Port ID	标识LLDPDU发送端的端口。如果LLDPDU中携带有LLDP-MED TLV，其内容为端口的MAC地址；否则，其内容为端口的名称	是
Time To Live	本设备信息在邻居设备上的存活时间	是
End of LLDPDU	LLDPDU的结束标识，是LLDPDU的最后一个TLV	是
Port Description	端口的描述	否
System Name	设备的名称	否
System Description	系统的描述	否
System Capabilities	系统的主要功能以及已开启的功能项	否
Management Address	管理地址，以及该地址所对应的接口号和OID（Object Identifier，对象标识符）	否

(2) 802.1 组织定义 TLV

IEEE 802.1 组织定义TLV的内容如 表 1-2 所示。

表1-2 IEEE 802.1 组织定义的 TLV

TLV 名称	说明
Port VLAN ID(PVID)	端口VLAN ID
Port and protocol VLAN ID(PPVID)	端口协议VLAN ID
VLAN Name	端口所属VLAN的名称
Protocol Identity	端口所支持的协议类型
DCBX	数据中心桥能力交换协议（Data Center Bridging Exchange Protocol）
Link Aggregation	端口是否支持链路聚合以及是否已开启链路聚合
Management VID	管理VLAN
VID Usage Digest	包含VLAN ID使用摘要的数据
ETS Configuration	增强传输选择（Enhanced Transmission Selection）配置
ETS Recommendation	增强传输选择推荐
PFC	基于优先级的流量控制（Priority-based Flow Control）
APP	应用协议（Application Protocol）
QCN	量化拥塞通知（Quantized Congestion Notification）

说明

- 目前，UNIS 设备不支持发送 Protocol Identity TLV 和 VID Usage Digest TLV，但可以接收这两种类型的 TLV。
- 三层以太网接口仅支持 Link Aggregation TLV。

(3) 802.3 组织定义 TLV

IEEE 802.3 组织定义TLV的内容如 表 1-3 所示。

表1-3 IEEE 802.3 组织定义的 TLV

TLV 名称	说明
MAC/PHY Configuration/Status	端口支持的速率和双工状态、是否支持端口速率自动协商、是否已开启自动协商功能以及当前的速率和双工状态
Power Via MDI	端口的供电能力，包括PoE（Power over Ethernet，以太网供电）的类型（包括PSE（Power Sourcing Equipment，供电设备）和PD（Powered Device，受电设备）两种）、PoE端口的远程供电模式、是否支持PSE供电、是否已开启PSE供电、供电方式是否可控、供电类型、功率来源、功率优先级、PD请求功率值、PSE分配功率值
Maximum Frame Size	端口支持的最大帧长度，取端口配置的MTU（Maximum Transmission Unit，最大传输单元）
Power Stateful Control	端口的电源状态控制，包括PSE/PD所采用的电源类型、供/受电的优先级以及供/受电的功率
Energy-Efficient Ethernet	节能以太网

说明

Power Stateful Control TLV 是在 IEEE P802.3at D1.0 版本中被定义的，之后的版本不再支持该 TLV。UNIS 设备只有在收到 Power Stateful Control TLV 后才会发送该类型的 TLV。

(4) LLDP-MED TLV

LLDP-MED TLV为VoIP（Voice over IP，在IP网络上传送语音）提供了许多高级的应用，包括基本配置、网络策略配置、地址信息以及目录管理等，满足了语音设备的不同生产厂商在投资收效、易部署、易管理等方面的要求，并解决了在以太网中部署语音设备的问题，为语音设备的生产者、销售者以及使用者提供了便利。LLDP-MED TLV的内容如 表 1-4 所示。

表1-4 LLDP-MED TLV

TLV 名称	说明
LLDP-MED Capabilities	网络设备所支持的LLDP-MED TLV类型
Network Policy	网络设备或终端设备上端口的VLAN类型、VLAN ID以及二三层与具体应用类型相关的优先级等
Extended Power-via-MDI	网络设备或终端设备的扩展供电能力，对Power Via MDI TLV进行了扩展

TLV 名称	说明
Hardware Revision	终端设备的硬件版本
Firmware Revision	终端设备的固件版本
Software Revision	终端设备的软件版本
Serial Number	终端设备的序列号
Manufacturer Name	终端设备的制造厂商名称
Model Name	终端设备的模块名称
Asset ID	终端设备的资产标识符，以便目录管理和资产跟踪
Location Identification	网络设备的位置标识信息，以供终端设备在基于位置的应用中使用

说明

如果禁止发布 802.3 的组织定义的 MAC/PHY Configuration/Status TLV，则 LLDP-MED TLV 将不会被发布，不论其是否被允许发布；如果禁止发布 LLDP-MED Capabilities TLV，则其他 LLDP-MED TLV 将不会被发布，不论其是否被允许发布。

5. 管理地址

管理地址是供网络管理系统标识网络设备并进行管理的地址。管理地址可以明确地标识一台设备，从而有利于网络拓扑的绘制，便于网络管理。管理地址被封装在 LLDP 报文的 Management Address TLV 中向外发布。

1.1.3 LLDP工作机制

1. LLDP的工作模式

在指定类型的 LLDP 代理下，LLDP 有以下四种工作模式：

- TxRx：既发送也接收 LLDP 报文。
- Tx：只发送不接收 LLDP 报文。
- Rx：只接收不发送 LLDP 报文。
- Disable：既不发送也不接收 LLDP 报文。

当端口的 LLDP 工作模式发生变化时，端口将对协议状态机进行初始化操作。为了避免端口工作模式频繁改变而导致端口不断执行初始化操作，可配置端口初始化延迟时间，当端口工作模式改变时延迟一段时间再执行初始化操作。

2. LLDP报文的发送机制

在指定类型 LLDP 代理下，当端口工作在 TxRx 或 Tx 模式时，设备会周期性地向邻居设备发送 LLDP 报文。如果设备的本地配置发生变化则立即发送 LLDP 报文，以将本地信息的变化情况尽快通知给邻居设备。但为了防止本地信息的频繁变化而引起 LLDP 报文的大量发送，使用令牌桶机制对 LLDP 报文发送作限速处理。有关令牌桶的详细介绍，请参见“ACL 和 QoS 配置指导”中的“流量监管、流量整形和接口限速”。

当设备的工作模式由 Disable/Rx 切换为 TxRx/Tx，或者发现了新的邻居设备（即收到一个新的 LLDP 报文且本地尚未保存发送该报文设备的信息）时，该设备将自动启用快速发送机制，即将 LLDP 报文的发送周期设置为快速发送周期，并连续发送指定数量的 LLDP 报文后再恢复为正常的发送周期。

3. LLDP报文的接收机制

当端口工作在 TxRx 或 Rx 模式时，设备会对收到的 LLDP 报文及其携带的 TLV 进行有效性检查，通过检查后再将邻居信息保存到本地，并根据 Time To Live TLV 中 TTL（Time to Live，生存时间）的值来设置邻居信息在本地设备上的老化时间，若该值为零，则立刻老化该邻居信息。

1.1.4 协议规范

与 LLDP 相关的协议规范有：

- IEEE 802.1AB-2005：Station and Media Access Control Connectivity Discovery
- IEEE 802.1AB 2009：Station and Media Access Control Connectivity Discovery
- ANSI/TIA-1057：Link Layer Discovery Protocol for Media Endpoint Devices
- DCB Capability Exchange Protocol Specification Rev 1.0
- DCB Capability Exchange Protocol Base Specification Rev 1.01
- IEEE Std 802.1Qaz-2011：Media Access Control (MAC) Bridges and Virtual Bridged Local Area Networks-Amendment 18: Enhanced Transmission Selection for Bandwidth Sharing Between Traffic Classes

1.2 LLDP配置任务简介

表1-5 LLDP 配置任务简介

配置任务		说明	详细配置
配置LLDP基本功能	开启LLDP功能	必选	1.3.1
	配置LLDP桥模式	可选	1.3.2
	配置LLDP工作模式	可选	1.3.3
	配置接口初始化延迟时间	可选	1.3.4
	配置轮询功能	可选	1.3.5
	配置允许发布的TLV类型	可选	1.3.6
	配置管理地址及其封装格式	可选	1.3.7
	调整LLDP相关参数	可选	1.3.8
	配置LLDP报文的封装格式	可选	1.3.9
	关闭LLDP的PVID不一致检查功能	可选	1.3.10
配置LLDP Trap和LLDP-MED Trap功能		可选	1.4
配置LLDP报文的源MAC地址为指定VLAN关联子接口的MAC地址		可选	1.5
配置设备支持通过LLDP生成对端管理地址的ARP或ND表项		可选	1.6

1.3 配置LLDP基本功能

1.3.1 开启LLDP功能

只有当全局和接口上都开启了 LLDP 功能后，该功能才会生效。

表1-6 开启 LLDP 功能

操作	命令	说明
进入系统视图	**system-view**	-
全局开启LLDP功能	**lldp global enable**	本命令缺省未开启
进入二/三层以太网接口视图、二/三层聚合接口视图	**interface** *interface-type interface-number*	
在接口上开启LLDP功能	**lldp enable**	缺省情况下，LLDP功能在接口上处于开启状态

1.3.2 配置LLDP桥模式

LLDP 桥模式有客户桥模式和服务桥模式两种：

- 工作于客户桥模式时，设备可支持最近桥代理、最近非 TPMR 桥代理和最近客户桥代理，即设备对报文目的 MAC 地址为上述代理的 MAC 地址的 LLDP 报文进行处理，对报文目的 MAC 地址为其他 MAC 地址的 LLDP 报文进行 VLAN 内透传。
- 工作于服务桥模式时，设备可支持最近桥代理和最近非 TPMR 桥代理，即设备对报文目的 MAC 地址为上述代理的 MAC 地址的 LLDP 报文进行处理，对报文目的 MAC 地址为其他 MAC 地址的 LLDP 报文进行 VLAN 内透传。

表1-7 配置 LLDP 桥模式

操作	命令	说明
进入系统视图	**system-view**	-
配置LLDP桥模式	**lldp mode service-bridge**	缺省情况下，LLDP桥模式为客户桥模式

1.3.3 配置LLDP工作模式

LLDP 工作模式分为以下四种：

- TxRx：既发送也接收 LLDP 报文。
- Tx：只发送不接收 LLDP 报文。
- Rx：只接收不发送 LLDP 报文。
- Disable：既不发送也不接收 LLDP 报文。

表1-8 配置 LLDP 工作模式

操作	命令	说明
进入系统视图	**system-view**	-
进入二/三层以太网接口视图、二/三层聚合接口视图、	**interface** *interface-type interface-number*	
配置LLDP的工作模式	在二/三层以太网接口视图或管理以太网接口视图下： **lldp [agent { nearest-customer | nearest-nontpmr }] admin-status { disable | rx | tx | txrx }** 在二/三层聚合接口视图下： **lldp agent { nearest-customer | nearest-nontpmr } admin-status { disable | rx | tx | txrx }**	缺省情况下，最近桥代理类型的LLDP工作模式为TxRx，最近客户桥代理和最近非TPMR桥代理类型的LLDP工作模式为Disable 以太网接口视图下，未指定**agent**参数时，表示配置最近桥代理的工作模式 聚合接口视图下，只支持配置最近桥客户桥代理和最近非TPMR代理的工作模式

1.3.4 配置接口初始化延迟时间

当接口上 LLDP 的工作模式发生变化时，接口将对协议状态机进行初始化操作，通过配置接口初始化的延迟时间，可以避免由于工作模式频繁改变而导致接口不断地进行初始化。

表1-9 配置接口初始化延迟时间

操作	命令	说明
进入系统视图	**system-view**	-
配置接口初始化的延迟时间	**lldp timer reinit-delay** *delay*	缺省情况下，接口初始化的延迟时间为2秒

1.3.5 配置轮询功能

在开启了轮询功能后，LLDP 将以轮询间隔周期性地查询本设备的相关配置是否发生改变，如果发生改变将触发 LLDP 报文的发送，以将本设备的配置变化迅速通知给其他设备。

表1-10 配置轮询功能

操作	命令	说明
进入系统视图	**system-view**	-
进入二/三层以太网接口视图、二/三层聚合接口视图	**interface** *interface-type interface-number*	

操作	命令	说明
开启轮询功能并配置轮询间隔	在二/三层以太网接口视图或管理以太网接口视图下： **lldp** [**agent** { **nearest-customer** | **nearest-nontpmr** }] **check-change-interval** *interval* 在二/三层聚合接口视图下： **lldp agent** { **nearest-customer** | **nearest-nontpmr** } **check-change-interval** *interval*	缺省情况下，轮询功能处于关闭状态

1.3.6 配置允许发布的TLV类型

表1-11 配置允许发布的 TLV 类型

操作	命令	说明
进入系统视图	**system-view**	-
进入二/三层以太网接口视图、二/三层聚合接口视图	**interface** *interface-type interface-number*	
配置接口上允许发布的TLV类型（二层以太网接口视图）	**lldp tlv-enable** { **basic-tlv** { **all** | **port-description** | **system-capability** | **system-description** | **system-name** | **management-address-tlv** [**ipv6**] [*ip-address*] } | **dot1-tlv** { **all** | **congestion-notification** | **port-vlan-id** | **link-aggregation** | **protocol-vlan-id** [*vlan-id*] | **vlan-name** [*vlan-id*] | **management-vid** [*mvlan-id*] } | **dot3-tlv** { **all** | **mac-physic** | **max-frame-size** | **power** } | **med-tlv** { **all** | **capability** | **inventory** | **network-policy** [*vlan-id*] | **power-over-ethernet** | **location-id** { **civic-address** *device-type country-code* { *ca-type ca-value* }&<1-10> | **elin-address** *tel-number* } } } **lldp agent nearest-nontpmr tlv-enable** { **basic-tlv** { **all** | **port-description** | **system-capability** | **system-description** | **system-name** | **management-address-tlv** [**ipv6**] [*ip-address*] } | **dot1-tlv** { **all** | **congestion-notification** | **evb** | **port-vlan-id** | **link-aggregation** } } **lldp agent nearest-customer tlv-enable** { **basic-tlv** { **all** | **port-description** | **system-capability** | **system-description** | **system-name** | **management-address-tlv** [**ipv6**] [*ip-address*] } | **dot1-tlv** { **all** | **congestion-notification** | **port-vlan-id** | **link-aggregation** } }	缺省情况下： • 最近桥代理允许发布除 Location-id TLV、Port And Protocol VLAN ID TLV、VLAN Name TLV、Management VLAN ID TLV 之外所有类型的 TLV • 最近客户桥代理允许发布基本 TLV 和 IEEE 802.1 组织定义 TLV evb参数的支持情况请参见LLDP命令。

操作	命令	说明
配置接口上允许发布的TLV类型（三层以太网接口视图）	**lldp tlv-enable** { **basic-tlv** { **all** | **port-description** | **system-capability** | **system-description** | **system-name** | **management-address-tlv** [**ipv6**] [*ip-address* | **interface loopback** *interface-number*] } | **dot1-tlv** { **all** | **link-aggregation** } | **dot3-tlv** { **all** | **mac-physic** | **max-frame-size** | **power** } | **med-tlv** { **all** | **capability** | **inventory** | **power-over-ethernet** | **location-id** { **civic-address** *device-type country-code* { *ca-type ca-value* }&<1-10> | **elin-address** *tel-number* } } } **lldp agent** { **nearest-nontpmr** | **nearest-customer** } **tlv-enable** { **basic-tlv** { **all** | **port-description** | **system-capability** | **system-description** | **system-name** | **management-address-tlv** [**ipv6**] [*ip-address*] } | **dot1-tlv** { **all** | **link-aggregation** } }	缺省情况下： • 最近桥代理允许发布除Network Policy TLV之外所有类型的TLV，其中IEEE 802.1组织定义的TLV只支持Link Aggregation TLV • 最近非TPMR桥代理不发布任何TLV • 最近客户桥代理允许发布基本TLV和IEEE 802.1组织定义TLV，其中IEEE 802.1组织定义的TLV只支持Link Aggregation TLV
配置接口上允许发布的TLV类型（管理以太网接口视图）	**lldp tlv-enable** { **basic-tlv** { **all** | **port-description** | **system-capability** | **system-description** | **system-name** | **management-address-tlv** [**ipv6**] [*ip-address*] } | **dot1-tlv** { **all** | **link-aggregation** } | **dot3-tlv** { **all** | **mac-physic** | **max-frame-size** | **power** } | **med-tlv** { **all** | **capability** | **inventory** | **power-over-ethernet** | **location-id** { **civic-address** *device-type country-code* { *ca-type ca-value* }&<1-10> | **elin-address** *tel-number* } } } **lldp agent** { **nearest-nontpmr** | **nearest-customer** } **tlv-enable** { **basic-tlv** { **all** | **port-description** | **system-capability** | **system-description** | **system-name** | **management-address-tlv** [**ipv6**] [*ip-address*] } | **dot1-tlv** { **all** | **link-aggregation** } }	缺省情况下： • 最近桥代理允许发布除Network Policy TLV之外所有类型的TLV，其中IEEE 802.1组织定义的TLV只支持Link Aggregation TLV • 最近非TPMR桥代理不发布任何TLV • 最近客户桥代理允许发布基本TLV和IEEE 802.1组织定义TLV，其中IEEE 802.1组织定义的TLV只支持Link Aggregation TLV
配置接口上允许发布的TLV类型（二层聚合接口视图）	**lldp agent nearest-nontpmr tlv-enable** { **basic-tlv** { **all** | **management-address-tlv** [**ipv6**] [*ip-address*] | **port-description** | **system-capability** | **system-description** | **system-name** } | **dot1-tlv** { **all** | **evb** | **port-vlan-id** } } **lldp agent nearest-customer tlv-enable** { **basic-tlv** { **all** | **management-address-tlv** [**ipv6**] [*ip-address*] | **port-description** | **system-capability** | **system-description** | **system-name** } | **dot1-tlv** { **all** | **port-vlan-id** } } **lldp tlv-enable dot1-tlv** { **protocol-vlan-id** [*vlan-id*] | **vlan-name** [*vlan-id*] | **management-vid** [*mvlan-id*] }	不存在最近桥代理 缺省情况下： • 不存在最近桥代理 • 最近非TPMR桥代理只允许发布EVB TLV • 最近客户桥代理允许发布基本TLV和IEEE 802.1组织定义TLV，其中IEEE 802.1组织定义的TLV只支持Port And Protocol VLAN ID TLV、VLAN Name TLV及Management VLAN ID TLV

操作	命令	说明
配置接口上允许发布的TLV类型（三层聚合接口视图）	**lldp agent** { **nearest-customer** | **nearest-nontpmr** } **tlv-enable basic-tlv** { **all** | **management-address-tlv** [**ipv6**] [*ip-address*] | **port-description** | **system-capability** | **system-description** | **system-name** }	不存在最近桥代理 缺省情况下： • 不存在最近桥代理 • 最近非 TPMR 桥代理不发布任何 TLV • 最近客户桥代理只允许发布基本 TLV

1.3.7 配置管理地址及其封装格式

管理地址被封装在 Management Address TLV 中向外发布，封装格式可以是数字或字符串。如果邻居将管理地址以字符串格式封装在 TLV 中，用户可在本地设备上也将封装格式改为字符串，以保证与邻居设备的正常通信。

表1-12 配置管理地址及其封装格式

操作	命令	说明
进入系统视图	**system-view**	-
进入二/三层以太网接口视图、二/三层聚合接口视图	**interface** *interface-type interface-number*	
允许在LLDP报文中发布管理地址并配置所发布的管理地址	在二层以太网接口视图/管理以太网接口视图下： **lldp** [**agent** { **nearest-customer** | **nearest-nontpmr** }] **tlv-enable basic-tlv management-address-tlv** [**ipv6**] [*ip-address*] 在三层以太网接口视图下： **lldp** [**agent** { **nearest-customer** | **nearest-nontpmr** }] **tlv-enable basic-tlv management-address-tlv** [**ipv6**] [*ip-address*] | **interface loopback** *interface-number*] 在二/三层聚合接口视图下： **lldp agent** { **nearest-customer** | **nearest-nontpmr** } **tlv-enable basic-tlv management-address-tlv** [**ipv6**] [*ip-address*]	缺省情况下，最近桥代理和最近客户桥代理类型的LLDP允许在LLDP报文中发布管理地址，最近非TPMR桥代理类型LLDP不允许在LLDP报文中发布管理地址
配置管理地址在TLV中的封装格式为字符串格式	在二/三层以太网接口视图或管理以太网接口视图下： **lldp** [**agent** { **nearest-customer** | **nearest-nontpmr** }] **management-address-format string** 在二/三层聚合接口视图下： **lldp agent** { **nearest-customer** | **nearest-nontpmr** } **management-address-format string**	缺省情况下，管理地址在TLV中的封装格式为数字格式

1.3.8 调整LLDP相关参数

LLDP 报文所携 Time To Live TLV 中 TTL 的值用来设置邻居信息在本地设备上的老化时间，由于 TTL＝Min（65535，（TTL 乘数×LLDP 报文的发送间隔＋1）），即取 65535 与（TTL 乘数×LLDP 报文的发送间隔＋1）中的最小值，因此通过调整 TTL 乘数可以控制本设备信息在邻居设备上的老化时间。

表1-13 调整 LLDP 相关参数

操作	命令	说明
进入系统视图	**system-view**	-
配置TTL乘数	**lldp hold-multiplier** *value*	缺省情况下，TTL乘数为4
配置LLDP报文的发送间隔	**lldp timer tx-interval** *interval*	缺省情况下，LLDP报文的发送间隔为30秒
配置LLDP报文发包限速的令牌桶大小	**lldp max-credit** *credit-value*	缺省情况下，发包限速令牌桶大小为5
配置快速发送LLDP报文的个数	**lldp fast-count** *count*	缺省情况下，快速发送LLDP报文的个数为4个
配置快速发送LLDP报文的间隔	**lldp timer fast-interval** *interval*	缺省情况下，快速发送LLDP报文的发送间隔为1秒

1.3.9 配置LLDP报文的封装格式

LLDP 报文的封装格式有 Ethernet II 和 SNAP 两种：

- 当采用 Ethernet II 封装格式时，开启了 LLDP 功能的接口所发送的 LLDP 报文将以 Ethernet II 格式封装。
- 当采用 SNAP 封装格式时，开启了 LLDP 功能的接口所发送的 LLDP 报文将以 SNAP 格式封装。

需要注意的是，LLDP 早期版本要求只有配置为相同的封装格式才能处理该格式的 LLDP 报文，因此为了确保与运行 LLDP 早期版本的设备成功通信，必须配置为与之相同的封装格式。

表1-14 配置 LLDP 报文的封装格式

操作	命令	说明
进入系统视图	**system-view**	-
进入二/三层以太网接口视图、二/三层聚合接口视图	**interface** *interface-type interface-number*	

操作	命令	说明
配置LLDP报文的封装格式为SNAP格式	在二/三层以太网接口视图或管理以太网接口视图下： **lldp** [**agent** { **nearest-customer** | **nearest-nontpmr** }] **encapsulation snap** 在二/三层聚合接口视图下： **lldp agent** { **nearest-customer** | **nearest-nontpmr** } **encapsulation snap**	缺省情况下，LLDP报文的封装格式为Ethernet II格式

1.3.10 关闭LLDP的PVID不一致检查功能

一般组网情况下，要求链路两端的 PVID 保持一致。设备会对收到的 LLDP 报文中的 PVID TLV 进行检查，如果发现报文中的 PVID 与本端 PVID 不一致，则认为网络中可能存在错误配置，LLDP 会打印日志信息，提示用户。

但在一些特殊情况下，可以允许链路两端的 PVID 配置不一致。例如为了简化接入设备的配置，各接入设备的上行口采用相同的 PVID，而对端汇聚设备的各接口采用不同的 PVID，从而使各接入设备的流量进入不同 VLAN。此时，可以关闭 LLDP 的 PVID 不一致性检查功能。

表1-15 关闭 LLDP 的 PVID 不一致检查功能

操作	命令	说明
进入系统视图	**system-view**	-
关闭LLDP的PVID不一致检查功能	**lldp ignore-pvid-inconsistency**	缺省情况下， LLDP的PVID不一致检查功能处于关闭状态

1.4 配置LLDP Trap和LLDP-MED Trap功能

开启 LLDP Trap 或 LLDP-MED Trap 功能后，设备可以通过向网管系统发送 Trap 信息以通告如发现新的 LLDP 邻居或 LLDP-MED 邻居、与原来邻居的通信链路发生故障等重要事件。

LLDP Trap 和 LLDP-MED Trap 信息的发送间隔是指设备向网管系统发送 Trap 信息的最小时间间隔，通过调整该时间间隔，可以避免由于邻居信息频繁变化而导致 Trap 信息的频繁发送。

表1-16 配置 LLDP Trap 和 LLDP-MED Trap 功能

操作	命令	说明
进入系统视图	**system-view**	-
进入二层以太网接口视图/二层聚合接口视图/三层以太网接口视图/三层聚合接口视图	**interface** *interface-type interface-number*	

操作	命令	说明
开启LLDP Trap功能	在二/三层以太网接口视图下： **lldp** [**agent** { **nearest-customer** | **nearest-nontpmr** }] **notification remote-change enable** 在二/三层聚合接口视图下： **lldp agent** { **nearest-customer** | **nearest-nontpmr** } **notification remote-change enable**	缺省情况下，LLDP Trap功能处于关闭状态
开启LLDP-MED Trap功能	在二/三层以太网接口视图或管理以太网接口视图下： **lldp notification med-topology-change enable**	缺省情况下，LLDP-MED Trap功能处于关闭状态
退回系统视图	**quit**	-
（可选）配置LLDP Trap和LLDP-MED Trap信息的发送间隔	**lldp timer notification-interval** *interval*	缺省情况下，LLDP Trap和LLDP-MED Trap信息的发送间隔均为30秒

1.5 配置LLDP报文的源MAC地址为指定VLAN关联子接口的MAC地址

配置本特性后，LLDP 报文的源 MAC 地址为指定 VLAN 在 Dot1q 终结中关联的三层以太网子接口的 MAC 地址。有关 Dot1q 终结的详细介绍，请参见“二层技术-以太网交换配置指导”中的“VLAN 终结”。

表1-17 配置 LLDP 报文的源 MAC 地址为指定 VLAN 关联子接口的 MAC 地址

操作	命令	说明
进入系统视图	**system-view**	-
进入三层以太网接口视图	**interface** *interface-type interface-number*	-
配置LLDP报文源MAC地址为指定VLAN关联三层以太网子接口的MAC地址	**lldp source-mac vlan** *vlan-id*	缺省情况下，LLDP报文源MAC地址为当前接口的MAC地址 本命令中的*vlan-id*为Dot1q终结中三层以太网子接口关联的VLAN ID

1.6 配置设备支持通过LLDP生成对端管理地址的ARP或ND表项

配置本特性后，当接口收到携带 IPv4 格式 Management Address TLV 的 LLDP 报文后，会生成该报文携带的管理地址与报文源 MAC 地址组成的 ARP 表项；当接口收到携带 IPv6 格式 Management Address TLV 的 LLDP 报文后，会生成该报文携带的管理地址与报文源 MAC 地址组成的 ND 表项。

表1-18 配置设备支持通过 LLDP 生成对端管理地址的 ARP 或 ND 表项

操作	命令	说明
进入系统视图	**system-view**	-
进入三层以太网接口视图	**interface** *interface-type interface-number*	-
配置接口收到携带Management Address TLV的LLDP报文后生成ARP表项或ND表项	**lldp management-address** { **arp-learning** | **nd-learning** } [**vlan** *vlan-id*]	缺省情况下，接口收到携带Management Address TLV的LLDP报文后生成ARP表项和ND表项 本命令中的*vlan-id*为Dot1q终结中三层以太网子接口关联的VLAN ID ARP表项和ND表项的生成互不影响，可同时配置

1.7 LLDP显示和维护

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

表1-19 LLDP 显示和维护

操作	命令
显示LLDP本地信息	**display lldp local-information** [**global** | **interface** *interface-type interface-number*]
显示由邻居设备发来的LLDP信息	**display lldp neighbor-information** [[[**interface** *interface-type interface-number*] [**agent** { **nearest-bridge** | **nearest-customer** | **nearest-nontpmr** }] [**verbose**]] | **list** [**system-name** *system-name*]]
显示LLDP的统计信息	**display lldp statistics** [**global** | [**interface** *interface-type interface-number*] [**agent** { **nearest-bridge** | **nearest-customer** | **nearest-nontpmr** }]]
显示LLDP的状态信息	**display lldp status** [**interface** *interface-type interface-number*] [**agent** { **nearest-bridge** | **nearest-customer** | **nearest-nontpmr** }]
显示接口上可发送的可选TLV信息	**display lldp tlv-config** [**interface** *interface-type interface-number*] [**agent** { **nearest-bridge** | **nearest-customer** | **nearest-nontpmr** }]

1.8 LLDP典型配置举例

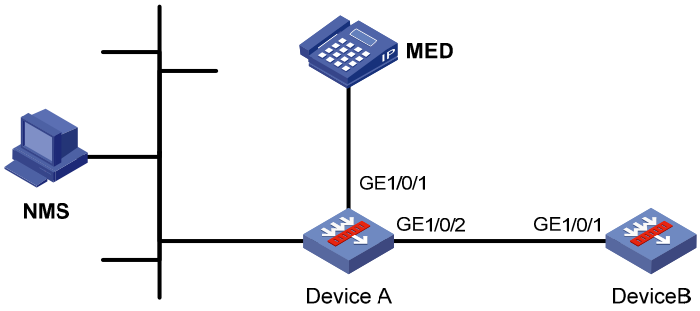
1.8.1 LLDP基本功能配置举例

1. 组网需求

- NMS（Network Management System，网络管理系统）与 Device A 相连，Device A 通过接口 GigabitEthernet1/0/1 和 GigabitEthernet1/0/2 分别与 MED 设备和 Device B 相连。
- 通过在 Device A 和 Device B 上配置 LLDP 功能，使 NMS 可以对 Device A 与 MED 设备之间、以及 Device A 与 Device B 之间链路的通信情况进行判断。

2. 组网图

图1-5 LLDP 基本功能配置组网图



3. 配置步骤

(1) 配置 Device A

全局开启 LLDP 功能。

```
<DeviceA> system-view
[DeviceA] lldp global enable
```

在接口 GigabitEthernet1/0/1 和 GigabitEthernet1/0/2 上分别开启 LLDP 功能（此步骤可省略，LLDP 功能在接口上缺省开启），并配置 LLDP 工作模式为 Rx。

```
[DeviceA] interface gigabitethernet 1/0/1
[DeviceA-GigabitEthernet1/0/1] lldp enable
[DeviceA-GigabitEthernet1/0/1] lldp admin-status rx
[DeviceA-GigabitEthernet1/0/1] quit
[DeviceA] interface gigabitethernet 1/0/2
[DeviceA-GigabitEthernet1/0/2] lldp enable
[DeviceA-GigabitEthernet1/0/2] lldp admin-status rx
[DeviceA-GigabitEthernet1/0/2] quit
```

(2) 配置 Device B

全局开启 LLDP 功能。

```
<DeviceB> system-view
[DeviceB] lldp global enable
```

在接口 GigabitEthernet1/0/1 上开启 LLDP 功能（此步骤可省略，LLDP 功能在接口上缺省开启），并配置 LLDP 工作模式为 Tx。

```
[DeviceB] interface gigabitethernet 1/0/1
[DeviceB-GigabitEthernet1/0/1] lldp enable
[DeviceB-GigabitEthernet1/0/1] lldp admin-status tx
[DeviceB-GigabitEthernet1/0/1] quit
```

4. 验证配置

显示 Device A 上全局和所有接口的 LLDP 状态信息。

```
[DeviceA] display lldp status
Global status of LLDP: Enable
Bridge mode of LLDP: customer-bridge
The current number of LLDP neighbors: 2
The current number of CDP neighbors: 0
```

```
LLDP neighbor information last changed time: 0 days, 0 hours, 4 minutes, 40 seconds
Transmit interval              : 30s
Fast transmit interval         : 1s
Transmit credit max            : 5
Hold multiplier                : 4
Reinit delay                   : 2s
Trap interval                  : 30s
Fast start times               : 4

LLDP status information of port 1 [GigabitEthernet1/0/1]:
LLDP agent nearest-bridge:
Port status of LLDP            : Enable
Admin status                   : Rx_Only
Trap flag                      : No
MED trap flag                  : No
Polling interval               : 0s
Number of LLDP neighbors       : 1
Number of MED neighbors        : 1
Number of CDP neighbors        : 0
Number of sent optional TLV    : 21
Number of received unknown TLV : 0

LLDP agent nearest-nontpmr:
Port status of LLDP            : Enable
Admin status                   : Disable
Trap flag                      : No
MED trap flag                  : No
Polling interval               : 0s
Number of LLDP neighbors       : 0
Number of MED neighbors        : 0
Number of CDP neighbors        : 0
Number of sent optional TLV    : 1
Number of received unknown TLV : 0

LLDP agent nearest-customer:
Port status of LLDP            : Enable
Admin status                   : Disable
Trap flag                      : No
MED trap flag                  : No
Polling interval               : 0s
Number of LLDP neighbors       : 0
Number of MED neighbors        : 0
Number of CDP neighbors        : 0
Number of sent optional TLV    : 16
Number of received unknown TLV : 0

LLDP status information of port 2 [GigabitEthernet1/0/2]:
LLDP agent nearest-bridge:
```

```
Port status of LLDP            : Enable
Admin status                   : Rx_Only
Trap flag                      : No
MED trap flag                  : No
Polling interval               : 0s
Number of LLDP neighbors       : 1
Number of MED neighbors        : 0
Number of CDP neighbors        : 0
Number of sent optional TLV    : 21
Number of received unknown TLV : 3

LLDP agent nearest-nontpmr:
Port status of LLDP            : Enable
Admin status                   : Disable
Trap flag                      : No
MED trap flag                  : No
Polling interval               : 0s
Number of LLDP neighbors       : 0
Number of MED neighbors        : 0
Number of CDP neighbors        : 0
Number of sent optional TLV    : 1
Number of received unknown TLV : 0

LLDP agent nearest-customer:
Port status of LLDP            : Enable
Admin status                   : Disable
Trap flag                      : No
MED trap flag                  : No
Polling interval               : 0s
Number of LLDP neighbors       : 0
Number of MED neighbors        : 0
Number of CDP neighbors        : 0
Number of sent optional TLV    : 16
Number of received unknown TLV : 0
```

由此可见，Device A 的接口 GigabitEthernet1/0/1 上连接了一个 MED 邻居设备，GigabitEthernet1/0/2 上则连接了一个非 MED 邻居设备，且这两个接口的 LLDP 工作模式都为 Rx，即只接收而不发送 LLDP 报文。

将 Device A 和 Device B 间的链路断掉后，再显示 Device A 上所有接口的 LLDP 状态信息。

```
[DeviceA] display lldp status
Global status of LLDP: Enable
The current number of LLDP neighbors: 1
The current number of CDP neighbors: 0
LLDP neighbor information last changed time: 0 days, 0 hours, 5 minutes, 20 seconds
Transmit interval              : 30s
Fast transmit interval         : 1s
Transmit credit max            : 5
Hold multiplier                : 4
Reinit delay                   : 2s
```

```
Trap interval                  : 30s
Fast start times               : 4

LLDP status information of port 1 [GigabitEthernet1/0/1]:
LLDP agent nearest-bridge:
Port status of LLDP            : Enable
Admin status                   : Rx_Only
Trap flag                      : No
MED trap flag                  : No
Polling interval               : 0s
Number of LLDP neighbors       : 1
Number of MED neighbors        : 1
Number of CDP neighbors        : 0
Number of sent optional TLV    : 0
Number of received unknown TLV : 5

LLDP agent nearest-nontpmr:
Port status of LLDP            : Enable
Admin status                   : Disabl
Trap flag                      : No
MED trap flag                  : No
Polling interval               : 0s
Number of LLDP neighbors       : 0
Number of MED neighbors        : 0
Number of CDP neighbors        : 0
Number of sent optional TLV    : 1
Number of received unknown TLV : 0

LLDP status information of port 2 [GigabitEthernet1/0/2]:
LLDP agent nearest-bridge:
Port status of LLDP            : Enable
Admin status                   : Rx_Only
Trap flag                      : No
MED trap flag                  : No
Polling interval               : 0s
Number of LLDP neighbors       : 0
Number of MED neighbors        : 0
Number of CDP neighbors        : 0
Number of sent optional TLV    : 0
Number of received unknown TLV : 0

LLDP agent nearest-nontpmr:
Port status of LLDP            : Enable
Admin status                   : Disable
Trap flag                      : No
MED trap flag                  : No
Polling interval               : 0s
Number of LLDP neighbors       : 0
```

```
Number of MED neighbors        : 0
Number of CDP neighbors        : 0
Number of sent optional TLV    : 1
Number of received unknown TLV : 0

LLDP agent nearest-customer:
Port status of LLDP            : Enable
Admin status                   : Disable
Trap flag                      : No
MED trap flag                  : No
Polling interval               : 0s
Number of LLDP neighbors       : 0
Number of MED neighbors        : 0
Number of CDP neighbors        : 0
Number of sent optional TLV    : 16
Number of received unknown TLV : 0
```

由此可见，Device A 的接口 GigabitEthernet1/0/2 上已经没有任何邻居设备了。

目 录

1 二层转发

说明

对于本节命令中的 CPU 参数，仅 T5000-M06 产品支持。

1.1 配置普通二层转发

1.1.1 普通二层转发的工作机制

如果设备接收到的报文的目的 MAC 地址匹配三层接口的 MAC 地址，则通过设备的三层接口进行三层转发；否则通过设备的二层接口进行二层转发。

二层转发根据报文的目的 MAC 地址查找 MAC 地址表，得到报文的出接口，然后将报文发送出去。

普通二层转发是设备默认启用的特性，不需要配置。

1.1.2 普通二层转发显示和维护

在任意视图下执行 **display** 命令可以显示二层转发过程中的统计信息，查看转发的结果。

在用户视图下执行 **reset** 命令可以清除二层转发的统计信息。

表1-1 普通二层转发显示和维护

操作	命令
显示二层转发统计信息	**display mac-forwarding statistics** [**interface** *interface-type interface-number*]
清除二层转发统计信息	**reset mac-forwarding statistics**

1.2 配置快速二层转发

1.2.1 快速二层转发的工作机制

快速二层转发采用高速缓存来处理报文，采用了基于数据流的技术，可以大大提高转发效率。

快速二层转发用源 IP 地址、源端口号、目的 IP 地址、目的端口号、协议号、输入接口、输出接口和 VLAN 来标识一条数据流。在二层转发过程中，会根据设备规则，对需要进行三层业务处理的报文，获取其 IP 地址等信息，生成 IP 快速转发表。当一条数据流的第一个报文转发后，会在高速缓存中生成相应的转发信息，该数据流后续报文的转发就可以通过直接查找快速转发表进行转发。这样便大大缩减了报文的排队流程，减少报文的转发时间，提高报文的转发速率。

快速二层转发是设备默认启用的特性，不需要配置。

1.2.2 快速二层转发显示和维护

在任意视图下执行 **display** 命令可以显示快速二层转发表信息。

表1-2 快速二层转发显示和维护

操作	命令
显示IP快速转发表信息（分布式设备－独立运行模式/集中式IRF设备）	**display mac-forwarding cache ip** [*ip-address*] [**slot** *slot-number* [**cpu** *cpu-number*]]
显示IP快速转发表信息（分布式设备－IRF模式）	**display mac-forwarding cache ip** [*ip-address*] [**chassis** *chassis-number* **slot** *slot-number* [**cpu** *cpu-number*]]
显示分片报文快速转发表信息（分布式设备－独立运行模式/集中式IRF设备）	**display mac-forwarding cache ip fragment** [*ip-address*] [**slot** *slot-number* [**cpu** *cpu-number*]]
显示分片报文快速转发表信息（分布式设备－IRF模式）	**display mac-forwarding cache ip fragment** [*ip-address*] [**chassis** *chassis-number* **slot** *slot-number* [**cpu** *cpu-number*]]
显示IPv6快速转发表信息（分布式设备－独立运行模式/集中式IRF设备）	**display mac-forwarding cache ipv6** [*ipv6-address*] [**slot** *slot-number* [**cpu** *cpu-number*]]
显示IPv6快速转发表信息（分布式设备－IRF模式）	**display mac-forwarding cache ipv6** [*ipv6-address*] [**chassis** *chassis-number* **slot** *slot-number* [**cpu** *cpu-number*]]

1.3 配置直通转发

设备各款型对于本节所描述的特性支持情况有所不同，详细差异信息如下：

型号	特性	描述
T5000-M06	配置直通转发	不支持
T5000-G20		支持
T1000-G20/G50/G60/G80		支持

直通转发指的是设备接收到报文的前 64 个字节后立即转发。该功能可以节省报文在设备中消耗的时间，提高转发性能。

表1-3 配置设备直通转发

操作	命令	说明
进入系统视图	**system-view**	-
配置设备直通转发	**cut-through enable**	缺省情况下，设备直通转发功能处于关闭状态

说明

报文在 CRC（Cyclic Redundancy Code，循环冗余校验码）接收前已经转发，因此设备也将转发 CRC 校验错误的报文。

1.4 配置Bridge转发

1.4.1 INLINE转发的工作机制

根据报文的转发特征，INLINE 转发有下列几种转发模式：

- 反射模式：用户通过配置将某接口收到的报文处理完以后，还从该接口发送出去。
- 透传模式：用户通过配置直接指定从某接口入的报文从特定接口出。
- 黑洞模式：用户通过配置将某接口收到的报文处理完以后丢弃。

Inline 转发是在数据链路层对流量进行安全监控的一种技术。目前这种技术主要应用在安全产品上，如应用在 VFW（Virtual Firewall，虚拟防火墙）上。

1.4.2 配置INLINE转发

1. 配置反射/透传/黑洞模式INLINE转发

表1-4 配置反射/透传/黑洞模式 Bridge 转发

操作	命令	说明
进入系统视图	**system-view**	-
创建反射模式Bridge转发实例，并进入Bridge视图	**bridge** *bridge-index* **reflect**	三者选其一 缺省情况下，不存在Bridge转发实例
创建透传模式Bridge转发实例，并进入Bridge视图	**bridge** *bridge-index* **forward**	
创建黑洞模式Bridge转发实例，并进入Bridge视图	**bridge** *bridge-index* **blackhole**	
向Bridge转发实例中添加接口	**add interface** *interface-type interface-number*	缺省情况下，Bridge转发实例中未添加任何接口 每个反射/黑洞模式Bridge转发实例只能添加一个接口；每个透传模式Bridge转发实例只能添加两个接口，且这两个接口的类型必须保持一致

1.5 配置快速Bridge转发

1.5.1 快速Bridge转发的工作机制

快速 Bridge 转发采用高速缓存来处理报文，采用了基于数据流的技术，可以大大提高转发效率。

快速 Bridge 转发用源 IP 地址、源端口号、目的 IP 地址、目的端口号、协议号、输入接口、输出接口和 VLAN 来标识一条数据流。在二层转发过程中，会根据设备规则，对需要进行三层业务处理的报文，获取其 IP 地址等信息，生成 IP 快速转发表。当一条数据流的第一个报文转发后，会在高速缓存中生成相应的转发信息，该数据流后续报文的转发就可以通过直接查找快速转发表进行转发。这样便大大缩减了报文的排队流程，减少报文的转发时间，提高报文的转发速率。

快速 Bridge 转发是设备默认启用的特性，不需要配置。

1.5.2 快速Bridge转发显示和维护

在任意视图下执行 **display** 命令可以显示快速 Bridge 转发表信息。

表1-5 快速 Bridge 转发显示和维护

操作	命令
显示Bridge转发创建的IP快速转发表信息（分布式设备－独立运行模式/集中式IRF设备）	**display bridge cache ip inline** [*ip-address*] [**slot** *slot-number* [**cpu** *cpu-number*]]
显示Bridge转发创建的IP快速转发表信息（分布式设备－IRF模式）	**display bridge cache ip inline** [*ip-address*] [**chassis** *chassis-number* **slot** *slot-number* [**cpu** *cpu-number*]]
显示Bridge转发创建的分片报文快速转发表信息（分布式设备－独立运行模式/集中式IRF设备）	**display bridge cache ip fragment inline** [*ip-address*] [**slot** *slot-number* [**cpu** *cpu-number*]]
显示Bridge转发创建的分片报文快速转发表信息（分布式设备－IRF模式）	**display bridge cache ip fragment inline** [*ip-address*] [**chassis** *chassis-number* **slot** *slot-number* [**cpu** *cpu-number*]]
显示Bridge转发创建的IPv6快速转发表信息（分布式设备－独立运行模式/集中式IRF设备）	**display bridge cache ipv6 inline** [*ipv6-address*] [**slot** *slot-number* [**cpu** *cpu-number*]]
显示Bridge转发创建的IPv6快速转发表信息（分布式设备－IRF模式）	**display bridge cache ipv6 inline** [*ipv6-address*] [**chassis** *chassis-number* **slot** *slot-number* [**cpu** *cpu-number*]]