配置手册

RG-S3760 系列

双协议栈多层交换机

RGOS 10.3(5b1)

版权声明

免责声明

本手册依据现有信息制作，其内容如有更改，恕不另行通知，请关注锐捷网络有限公司网站提供的最新信息。锐捷网络有限公司在编写本手册时已尽力保证其内容准确可靠，但对于本手册中的遗漏、不准确或错误，以及由此导致的损失和损害，锐捷网络有限公司不承担责任。

前 言

版本说明

本手册对应的软件版本为：RGOS®10.3(5b1)版。

读者对象

本书适合下列人员阅读

- 网络工程师
- 技术推广人员
- 网络管理员

本书约定

1. 通用格式约定

宋体：正文采用 5 号宋体。

注意、说明等提示内容前后增加线条与正文隔离。

终端信息显示格式：英文用 `Courier New` ，中文用宋体，文字大小 5 号，表示屏幕输出信息。信息中夹杂的用户从终端输入的信息，采用**加粗**字体表示。

2. 命令行格式约定

命令行字体采用用 Arial，具体相关格式意义如下：

粗体：命令行关键字（命令中保持不变必须照输的部分）采用加粗字体表示。

斜体：命令行参数（命令中必须由实际值进行替代的部分）采用斜体表示

[ ] ：表示用[ ] 括起来的部分，在命令配置时是可选的。

{ x | y | ... }：表示从两个或多个选项中选取一个。

[x | y | ...]：表示从两个或多个选项中选取一个或者不选。

//：由双斜杠开始的行表示为注释行。

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：

注意： 注意、警告、提醒操作中应注意的事项。

说明： 说明、提示、窍门、对操作内容的描述进行必要的补充

说明：

1) 本手册举例说明部分的端口类型同实际可能不符，实际操作中需要按照各产品所支持的端口类型进行配置

2) 本手册部分举例的显示信息中可能含有其它产品系列的内容（如产品型号、描述等），具体显示信息请以实际使用的设备信息为准。

3) 本手册中涉及的路由器及路由器产品图标，代表了一般意义下的路由器，以及运行了路由协议的三层交换机。

---

1 命令行界面配置

本章节说明使用命令行界面的方法，您可以通过使用命令行界面来管理网络设备。

本章节主要包括以下内容：

- 命令模式
- 获得帮助
- 简写命令
- 使用命令的 **no** 和 **default** 选项
- 理解 CLI 的提示信息
- 使用历史命令
- 使用编辑特性
- CLI 输出信息的过滤和查找
- 访问 CLI

1.1 命令模式

锐捷网络设备管理界面分成若干不同的模式，用户当前所处的命令模式决定了可以使用的命令。

在命令提示符下输入问号键（?）可以列出每个命令模式支持使用的命令。

当用户和网络设备管理界面建立一个新的会话连接时，用户首先处于用户模式（User EXEC 模式），可以使用用户模式的命令。在用户模式下，只可以使用少量命令，并且命令的功能也受到一些限制，例如像 show 命令等。用户模式的命令的操作结果不会被保存。

要使用所有的命令，首先必须进入特权模式（Privileged EXEC 模式）。通常，在进入特权模式时必须输入特权模式的口令。在特权模式下，用户可以使用所有的特权命令，并且能够由此进入全局配置模式。

使用配置模式（全局配置模式、接口配置模式等）的命令，会对当前运行的配置产生影响。如果用户保存了配置信息，这些命令将被保存下来，并在系统重新启动时再次执行。要进入各种配置模式，首先必须进入全局配置模式。从全局配置模式出发，可以进入接口配置模式等各种配置子模式。

下表列出了命令的模式、如何访问每个模式、模式的提示符、如何离开模式。这里假定网络设备的名字为缺省的“Ruijie”。

命令模式概要：

命令模式	访问方法	提示符	离开或访问下一模式	关于该模式
User EXEC（用户模式）	访问网络设备时首先进入该模式。	Ruijie>	输入 **exit** 命令离开该模式。 要进入特权模式，输入 **enable** 命令。	使用该模式来进行基本测试、显示系统信息
Privileged EXEC（特权模式）	在用户模式下，使用 **enable** 命令进入该模式。	Ruijie#	要返回到用户模式，输入 **disable** 命令。 要进入全局配置模式，输入 **configure** 命令。	使用该模式来验证设置命令的结果。该模式是具有口令保护的。
Global configuration（全局配置模式）	在特权模式下，使用 **configure** 命令进入该模式。	Ruijie(config)#	要返回到特权模式，输入 **exit** 命令或 **end** 命令，或者键入 Ctrl+C 组合键。 要进入接口配置模式，输入 **interface** 命令。在 **interface** 命令中必须指明要进入哪一个接口配置子模式。 要进入 VLAN 配置模式，输入 **vlan** vlan_id 命令。	使用该模式的命令来配置影响整个网络设备的全局参数。
Interface configuration（接口配置模式）	在全局配置模式下，使用 **interface** 命令进入该模式。	Ruijie(config-if)#	要返回到特权模式，输入 **end** 命令，或键入 Ctrl+C 组合键。要返回到全局配置模式，输入 **exit** 命令。在 **interface** 命令中必须指明要进入哪一个接口配置子模式。	使用该模式配置网络设备的各种接口。
Config-vlan（VLAN 配置模式）	在全局配置模式下，使用 **vlan** *vlan_id* 命令进入该模式。	Ruijie(config-vlan)#	要返回到特权模式，输入 **end** 命令，或键入 Ctrl+C 组合键。 要返回到全局配置模式，输入 **exit** 命令。	使用该模式配置 VLAN 参数。

1.2 获得帮助

用户可以在命令提示符下输入问号键（**?**）列出每个命令模式支持的命令。用户也可以列出相同开头的命令关键字或者每个命令的参数信息。见下表：

命令	说明
Help	在任何命令模式下获得帮助系统的摘要描述信息。
abbreviated-command-entry?	获得相同开头的命令关键字字符串。 例子： Ruijie**# di**? dir disable
abbreviated-command-entry<Tab>	使命令的关键字完整。 例子： Ruijie**# show conf<Tab>** Ruijie**# show configuration**
?	列出该命令的下一个关联的关键字。 例子： Ruijie**# show ?**
command keyword ?	列出该关键字关联的下一个变量。 例子： Ruijie(config)**# snmp-server community ?** WORD SNMP community string

1.3 简写命令

如果想简写命令，只需要输入命令关键字的一部分字符，只要这部分字符足够识别唯一的命令关键字即可。

例如 **show configuration** 命令可以写成：

```
Ruijie# show conf
```

1.4 使用命令的 no 和 default 选项

几乎所有命令都有 **no** 选项。通常，使用 **no** 选项来禁止某个特性或功能，或者执行与命令本身相反的操作。例如接口配置命令 **no shutdown** 执行关闭接口命令 **shutdown** 的相反操作，即打开接口。使用不带 **no** 选项的关键字打开被关闭的特性或者打开缺省是关闭的特性。

配置命令大多有 **default** 选项，命令的 **default** 选项将命令的设置恢复为缺省值。大多数命令的缺省值是禁止该功能，因此在许多情况下 **default** 选项的作用和 **no** 选项是相同的。然而部分命令的缺省值是允许该功能，在这种情况下，**default** 选项和 **no** 选项的作用是相反的。这时 **default** 选项打开该命令的功能，并将变量设置为缺省的允许状态。

1.5 理解 CLI 的提示信息

下表列出了用户在使用 CLI 管理网络设备时可能遇到的错误提示信息。

常见的 CLI 错误信息：

错误信息	含义	如何获取帮助
% Ambiguous command: "show c"	用户没有输入足够的字符，网络设备无法识别唯一的命令。	重新输入命令，紧接着发生歧义的单词输入一个问号。可能输入的关键字将被显示出来。
% Incomplete command	用户没有输入该命令的必需的关键字或者变量参数。	重新输入命令，输入空格再输入一个问号。可能输入的关键字或者变量参数将被显示出来。
% Invalid input detected at '^' marker	用户输入命令错误，符号（^）指明了产生错误的单词的位置。	在所在地命令模式提示符下输入一个问号，该模式允许的命令的关键字将被显示出来。

1.6 使用历史命令

系统提供了用户输入的命令的记录。该特性在重新输入长而且复杂的命令时将十分有用。

从历史命令记录重新调用输入过的命令，执行下表中的操作：

操作	结果
Ctrl-P 或上方向键	在历史命令表中浏览前一条命令。从最近的一条记录开始，重复使用该操作可以查询更早的记录。
Ctrl-N 或下方向键	在使用了 Ctrl-P 或上方向键操作之后，使用该操作在历史命令表中回到更近的一条命令。重复使用该操作可以查询更近的记录。

注意：

标准的终端支持方向键，例如 VT100 系列。

1.7 使用编辑特性

本节描述在进行命令行编辑时可能使用到的编辑功能。包括：

- 编辑快捷键
- 命令行滑动窗口

1.7.1 编辑快捷键

下表列出编辑快捷键：

功能	快捷键	说明
在编辑行内移动光标。	左方向键或 Ctrl-B	光标移到左边一个字符。
	右方向键或 Ctrl-F	光标移到右边一个字符。
	Ctrl-A	光标移到命令行的首部。
	Ctrl-E	光标移到命令行的尾部。
删除输入的字符。	Backspace 键	删除光标左边的一个字符。
	Delete 键	删除光标所在的字符。
输出时屏幕滚动一行或一页。	Return 键	在显示内容时用回车键将输出的内容向上滚动一行，显示下一行的内容，仅在输出内容未结束时使用。
	Space 键	在显示内容时用空格键将输出的内容向上滚动一页，显示下一页内容，仅在输出内容未结束时使用。

1.7.2 命令行滑动窗口

用户可以使用编辑功能中的滑动窗口特性，来编辑超过单行宽度的命令，使命令行的长度得以延伸。当编辑的光标接近右边框时，整个命令行会整体向左移动 20 个字符，但是仍然可以使光标回到前面的字符或者回到命令行的首部。

编辑命令行时光标移动操作如下表：

功能	快捷键
光标向左回退一个字符	左方向键或 Ctrl-B
光标回到行首	Ctrl-A
光标向右前进一个字符	右方向键或 Ctrl-F

光标移动到行尾	Ctrl-E

例如配置模式的命令 **mac-address-table static** 的输入可能超过一个屏幕的宽度。当光标第一次接近行尾时，整个命令行整体向左移动 20 个字符。命令行前部被隐藏的部分被符号（$）代替。每次接近右边界时都会向左移动 20 个字符长度。

```
mac-address-table  static  00d0.f800.0c0c  vlan  1
interface
$tatic  00d0.f800.0c0c  vlan  1  interface  fastEthernet
$tatic  00d0.f800.0c0c  vlan  1  interface  fastEthernet  0/1
```

可以使用 Ctrl-A 快捷键回到命令行的首部。这时命令行尾部被隐藏的部分将被符号（$）代替：

```
-address-table  static  00d0.f800.0c0c  vlan  1 interface  $
```

注意：

默认的终端行宽是 80 个字符。

使用命令行滑动窗口结合历史命令的功能，可以重复调用复杂的命令。具体的快捷键的使用方法查看编辑快捷键。

1.8 CLI 输出信息的过滤和查找

1.8.1 Show 命令的查找和过滤

要在 **show** 命令输出的信息中查找指定的内容，可以在使用以下命令：

命令	说明
Ruijie# **show** *any-command* **| begin** *regular-expression*	在 show 命令的输出内容中查找指定的内容，将第一个包含该内容的行以及该行以后的全部信息输出。

注意：

1) 支持在任意模式下执行 **Show** 命令

2) 查找的信息内容需要区分大小写，以下相同。

要在 **show** 命令的输出信息中过滤指定的内容，可以使用以下命令：

命令	说明
Ruijie# **show** *any-command* | **exclude** *regular-expression*	在 show 命令的输出内容中进行过滤，除了包含指定内容的行以外，输出其他的信息内容。
Ruijie# **show** *any-command* | **include** *regular-expression*	在 show 命令的输出内容中进行过滤，仅输出包含指定内容的行，其他信息将被过滤。

注意：

要在 **show** 命令的输出内容中进行查找和过滤，需要输入管道符号（竖线，"|"）。在管道字符之后，可以选择查找和过滤的规则和查找和过滤的内容（字符或字符串）。并且查找和过滤的内容需要区分大小写。

1.9 使用命令别名

系统提供命令别名功能，可以指定任意单词作为命令的别名，例如：将单词"mygateway"定义为"ip route 0.0.0.0 0.0.0.0 192.1.1.1"的别名，

则输入这个单词就相当于输入后面的整个字符串。

通过配置命令别名，可以用一个单词来代替一条命令。例如，创建一个别名来代表一条命令的前一部分，然后可以继续输入后面的部分。

别名所代表的命令所处的命令模式是当前系统中存在的命令模式，在全局配置模式下，输入 **alias ?**可以列出当前可以配置别名的全部命令模式：

```
Ruijie(config)#alias ?
aaa-gs              AAA server group mode
acl                 acl configure mode
bgp                 Configure bgp Protocol
config              globle configure mode
......
```

命令别名支持帮助信息，在别名前面会显示一个星号（*），并且会用以下格式显示：

**command-alias=original-command*

例如，在 EXEC 模式下，默认的命令别名"s"表示"show"关键字。则输入"s?"可以获取's'开头的关键字和别名的帮助信息：

```
Ruijie#s?
*s=show  show  start-chat  start-terminal-service
```

如果别名所代表的命令不止一个单词，则会使用引号将命令包括起来。例如，在 EXEC 模式下配置别名"sv"代替命令"show version"，则：

```
Ruijie#s?
```

```
*s=show  *sv=”show version” show  start-chat
start-terminal-service
```

别名必须从输入的命令行的第一个字符开始，前面不能有空格。如上面的例子，如果在命令之前输入了空格，就不能表示合法的别名：

```
Ruijie# s?
show  start-chat  start-terminal-service
```

命令别名也可以支持获取命令的参数的帮助信息，例如配置接口模式下的命令别名”ia”代表”ip address”，则在接口模式下：

```
Ruijie(config-if)#ia ?
  A.B.C.D  IP address
  dhcp     IP Address via DHCP

Ruijie(config-if)#ip address
```

这里列出了”**ip address**”命令后面的参数信息，并且将别名替换成实际的命令。

命令别名在使用时必须完整输入，否则不能被识别。

使用 **show aliases** 命令可以查看系统中的别名设置。

1.9.1 访问 CLI

在使用 CLI 之前，用户需要使用一个终端或 PC 和网络设备连接。启动网络设备，在网络设备硬件和软件初始化后就可以使用 CLI。在网络设备的首次使用时只能使用串口（Console）方式连接网络设备，称为带外（Outband）管理方式。在进行了相关配置后，可以通过 Telnet 虚拟终端方式连接和管理网络设备。通过这两者都可以访问命令行界面。

2 交换机基础管理配置

2.1 概述

本章描述对我司产品的一些管理操作：

- 通过命令的授权控制用户访问
- 登录认证控制
- 系统时间配置
- 定时重启
- 系统名称和命令提示符
- 标题配置
- 查看系统信息
- 控制台速率配置
- 在网络设备上使用 telnet
- 链接超时设置
- 批处理执行文件中的命令
- 服务开关设置

说明：
有关本节引用的 CLI 命令的详细使用信息及说明，请参照《配置交换机管理命令》。

2.2 通过命令的授权控制用户访问

2.2.1 概述

控制网络上的终端访问网络设备的一个简单办法，就是使用口令保护和划分特权级别。口令可以控制对网络设备的访问，特权级别可以在用户登录成功后，控制其可以使用的命令。

从安全角度来看，口令是保存在配置文件中的，在网络上传输这些文件时（比如使用 TFTP），我们希望保证口令的安全。因此口令在保存入参数文件之前将被加密处理，明文形式的口令变成密文形式的口令。命令 **enable secret** 使用了私有的加

密算法。

2.2.2 缺省的口令和特权级别配置

缺省没有设置任何级别的口令，缺省的级别是 15 级。

2.2.3 设置和改变各级别的口令

我司产品提供下面的命令用于设置和改变各级别的口令。

命令	目的
Ruijie(config)# **enable password** [**level** *level*] {*password* | *encryption-type encrypted-password*}	设置静态口令。目前只能设置 15 级用户的口令，并且只能在未设置安全口令的情况下有效。 如果设置非 15 级的口令，系统将会给出一个提示，并自动转为安全口令。 如果设置的 15 级静态口令和 15 级安全口令完全相同，系统将会给出一个警告信息。
Ruijie(config)# **enable secret** [**level** *level*] {*encryption-type encrypted-password*}	设置安全口令，功能与静态口令相同，但使用了更好的口令加密算法。为了安全起见，建议您使用安全口令。
Ruijie# **enable** [*level*] 和 Ruijie# **disable** [*level*]	切换用户级别，从权限较低的级别切换到权限较高的级别需要输入相应级别的口令。

在设置口令中，如果您使用带 **level** 关键字时，则为指定特权级别定义口令。设置了特定级别的口令后，给定的口令只适用于那些需要访问该级别的用户。

2.2.4 配置多个特权级别

在缺省情况下，系统只有两个受口令保护的授权级别：普通用户级别（1 级）和特权用户级别（15 级）。但是用户可以为每个模式的命令划分 16 个授权级别。通过给不同的级别设置口令，就可以通过不同的授权级别使用不同的命令集合。

在特权用户级别口令没有设置的情况下，进入特权级别亦不需要口令校验。为了安全起见，我们提醒您最好为特权用户级别设置口令。

2.2.4.1 命令授权配置

如果想让更多的授权级别使用某一条命令，则可以将该命令的使用权授予较低的用户级别；而如果想让命令的使用范围小一些，则可以将该命令的使用权授予较高的用户级别。

你可以使用如下命令对命令进行授权：

命令	目的
Ruijie**# configure terminal**	进入全局配置模式
Ruijie(config)**# privilege mode** [**all**] {**level** *level* | **reset**} *command-string*	设置命令的级别划分。 mode －要授权的命令所属的 CLI 命令模式，例如：**config** 表示全局配置模式；**exec** 表示特权命令模式，**interface** 表示接口配置模式等等。 all －将指定命令的所有子命令的权限，变为相同的权限级别。 **level** *level* －授权级别，范围从 0 到 15。**level** 1 是普通用户级别，**level** 15 是特权用户级别，在各用户级别间切换可以使用 **enable/disable** 命令。 *command-string* －要授权的命令。

要恢复一条已知的命令授权，可以在全局配置模式下使用 **no privilege mode** [**all**] **level** *level command* 命令。

2.2.4.2 命令授权配置实例

下面是将 **reload** 命令及其子命令授予级别 1 并且设置级别 1 为有效级别（通过设置口令为"**test**"）的配置过程：

```
Ruijie# configure terminal
Ruijie(config)# privilege exec all level 1 reload
Ruijie(config)# enable secret level 1 0 test
Ruijie(config)# end
```

进入 1 级，可以看见命令和子命令：

```
Ruijie# disable 1
Ruijie> reload ?
at                  reload at a specific time/date
```

```
cancel                cancel pending reload scheme
in                    reload after a time interval
<cr>
```

下面是将 reload 命令及其子命令的权限恢复为默认值的配置过程：

```
Ruijie# configure terminal
Ruijie(config)# privilege exec all reset reload
Ruijie(config)# end
```

进入 1 级，命令权限已经被收回：

```
Ruijie# disable 1
Ruijie> reload ?
% Unrecognized command.
```

2.2.5 配置线路（line）口令保护

我司产品支持对远程登录（如 TELNET）进行口令验证，要配置 **line** 口令保护，请在 **line** 配置模式下执行以下命令：

命令	目的
Ruijie(config-line)# **password** *password*	指定 **line** 线路口令
Ruijie(config-line)# **login**	启用 **line** 线路口令保护

说明：

如果没有配置登录认证，即使配置了 **line** 口令，登录时，**line** 层口令认证会被忽略。登录认证在下一节介绍。

2.2.6 支持会话锁定

我司产品支持通过 **lock** 命令将会话终端临时锁住，以防止访问。要使用锁住会话终端的功能，需要在 **line** 配置模式下打开支持终端锁定的功能，并在相应终端的 EXEC 模式下，通过使用 **lock** 命令锁住终端：

命令	目的
Ruijie(config-line)# **lockable**	启用锁住 **line** 终端的功能
Ruijie# **lock**	锁住当前 **line** 终端

2.3 登录认证控制

2.3.1 概述

前面一节我们描述了如何通过配置本地保存的口令来控制对网络设备的访问。除了线路口令保护和本地认证外，如果启用了 AAA 模式，则在用户登录网络设备进行管理时，在登录时我们还可以通过一些服务器来根据用户名和密码进行用户的管理权限的认证，目前我们还支持利用 RADIUS 服务器根据用户登录时的用户名和密码控制用户对网络设备的管理权限。

利用 RADIUS 服务器对用户登录时的用户名和密码进行控制，这样网络设备不再用本地保存的密码信息进行认证，而是将加密后的用户信息发送到 RADIUS 服务器上进行验证，服务器统一配置用户的用户名、用户密码、共享密码和访问策略等信息，便于管理和控制用户访问，提高用户信息的安全性。

2.3.2 配置本地用户

我司产品支持基于本地数据库的身份认证系统，主要用于 AAA 模式下，通过方法列表中的本地认证；以及非 AAA 模式下，线路登录管理中的本地登录认证。

要建立用户名身份认证，请在全局配置模式下，根据具体需求执行以下命令：

命令	作用
Ruijie(config)# **username** *name* [**password** *password* | **password** *encryption-type encrypted password*]	使用加密口令建立用户名身份认证
Ruijie(config)# **username** *name* [**privilege** *level*]	为用户设置权限级别（可选）

2.3.3 配置线路登录认证

要建立线路登录身份认证，请在线路配置模式下，根据具体需求执行以下命令：

命令	作用
Ruijie(config-line)# **login local**	非 AAA 模式下，设置线路登录进行本地认证
Ruijie(config-line)# **login authentication** {**default** | *list-name*}	AAA 模式下，设置线路登录进行 AAA 认证。认证时使用 AAA 方法列表中的认证方法，包括 Radius 认证、本地认证、无认证等。

说明：
设置 AAA 模式、Radius 服务器配置以及方法列表的配置，请参见 AAA 配置相关章节。

2.4 系统时间配置

2.4.1 概述

每台网络设备中均有自己的系统时钟，该时钟提供具体日期(年、月、日)和时间(时、分、秒)以及星期等信息。对于一台网络设备，当第一次使用时你需要首先手工配置网络设备系统时钟为当前的日期和时间。当然，根据需要，你也可以随时修正系统时钟。网络设备的系统时钟主要用于系统日志等需要记录事件发生时间的地方。

2.4.2 设置系统时间

你可以通过手工的方式来设置网络设备上的时间。当你设置了网络设备的时钟后，网络设备的时钟将以你设置的时间为准一直运行下去，即使网络设备下电，网络设备的时钟仍然继续运行。所以网络设备的时钟设置一次后，原则上不需要再进行设置，除非你需要修正网络设备上的时间。

但是对于没有提供硬件时钟的网络设备，手工设置网络设备上的时间实际上只是设置软件时钟，它仅对本次运行有效，当网络设备下电后，手工设置的时间将失效。

命令	作用
Ruijie# **clock set** *hh:mm:ss month day year*	设置系统的日期和时钟。

例如把系统时间改成 2003-6-20，10:10:12

```
Ruijie# clock set 10:10:12 6 20 2003        //设置系统时间和日期
Ruijie# show clock                          //确认修改系统时间生效
10:10:15 UTC Fri, Jun 20, 2003
```

2.4.3 查看系统时间

你可以在特权模式下使用 **show clock** 命令来显示系统时间信息，显示的格式如下：

```
Ruijie# sh clock                            //显示当前系统时间
clock: 2003-5-20 11:11:34
```

2.4.4 硬件时钟更新

一些平台使用硬件时钟（calendar）来补充软件时钟，硬件时钟是不间断持续运转的，因为硬件时钟是走电池的，即使设备关闭或重启状态下也在运转。
如果硬件时钟和软件时钟不同步，软件时钟是比较精确的，采用该命令将软件时钟的日期和时间复制给硬件时钟。

用软件时钟来更新硬件时钟，在特权模式下执行 **clock update-calendar** 这个命令，软件时钟就会覆盖硬件时钟的值。

命令	作用
Ruijie# **clock update-calendar**	用软件时钟来更新硬件时钟

使用如下命令可以将当前软件时钟的时间和日期复制到硬件时钟：

```
Ruijie# clock update-calendar
```

2.5 定时重启

2.5.1 概述

本节描述如何使用 **reload** [modifiers]命令制定重启计划（scheme），以实现系统定时重启。定时重启功能，它在某些场合下(比如出于测试目的或其它需要)可以为用户提供操作上的便利。modifers 是 **reload** 提供的一组命令选项，可以使得该命令的使用更加灵活。可选的 modifiers 有 **in**、**at**、**cancel**。具体使用说明如下：

1. **reload in** *mmm* | *hhh:mm* [*string*]

指定系统在经过一定时间间隔后重启。这里的时间间隔由 *mmm* 或 *hhh:mm* 决定，以分钟为单位，用户可以任选一种格式输入。参数 *string* 是一个帮助提示，用户可以在这里为这个计划起一个助记名，以便能直观地反映该重启的用途，比如如果出于测试目的，需要系统 10 分钟后重启，我们可以键入 **reload in** *10 test*。

2. **reload at** *hh:mm month day year* [*string*]

指定系统在将来的某个时间点重启。输入的时间值必须是将来的某个时间点，限制时间跨度不能超过 200 天。string 的用法同上。比如当前系统时间是 2005-01-10 14:31，我们想要系统在明天上班时重启，我们可以键入 **reload at** *08:30 11 1 2005 newday*。或者假如当前系统时间是 2005-12-10 14:31，我们想要系统在 2006-01-01 12:00 重启，我们可以键入 **reload at** *12:00 1 1 2006 newyear*。

3. **reload cancel**

该命令是删除用户已经指定的重启计划。比如前面我们指定了系统在明天 8 点 30 重启，键入 **reload cancel** 后，该设定将被删除。

说明：

如果用户要使用 **at** 选项，则要求当前系统必须支持时钟功能。建议使用之前先配置好系统的时钟，以便更切合您的用途。如果用户之前已经设置了重启计划，则后面再设置的计划将覆盖前面的设置。如果用户已经设置了重启计划，假如在该计划生效前用户重启了系统，则该计划将丢失。

重启计划中的时间与当前时间的跨度不能超过 200 天并且要大于当前系统时间。同时用户在设置了重启计划之后最好不要再修改系统时钟,否则有可能会导致设置失效，比如将系统时间调到重启时间之后。

2.5.2 指定系统在某个时间重启

在特权模式下，通过如下命令，可以指定系统在将来的某个时间重启：

命令	作用
Ruijie# **reload at** *hh:mm month day year* [*reload-reason*]	指定系统在 year 年 month 月 day 日 hh 时 mm 分 reload。reload 的原因是 reload-reason (如果有输入的话)。

下面是一个指定系统在 2005 年 1 月 11 日中午 12:00 重启的例子(假定系统当前时钟是 2005 年 1 月 11 日 8:30)：

```
Ruijie# reload at 12:00 1 11 2005 midday//设置重启系统时间和日期
Ruijie# show reload                          //确认修改重启时间生效
Reload scheduled in 16581 seconds.
At 2005-01-11 12:00
Reload reason: midday
```

2.5.3 指定系统一段时间后重启

在特权模式下，使用如下命令指定系统一段时间后重启：

命令	作用
Ruijie# **reload in** *mmm* [*reload-reason*]	指定系统 *mmm* 分钟后 **reload**，**reload** 的原因是 *reload-reason*(如果有输入的话)
Ruijie# **reload in** *hhh:mm* [*reload-reason*]	指定系统 *hhh* 小时 *mm* 分钟后 **reload**， **reload** 的原因是 *reload-reason*(如果有输入的话)

下面是一个指定系统 125 分钟后 **reload** 的例子(假定当前系统时间是 2005-01-10 12:00)：

```
Ruijie# reload in 125 test                    //设置重启系统时间
或者是：
Ruijie# reload in 2:5 test                    //设置重启系统时间
Ruijie# show reload                           //确认修改重启时间生效
System will reload in 7485 seconds.
```

2.5.4 直接重启

不带重启计划参数的 **reload** 命令表示立即重启设备，用户可以在特权模式下直接键入 **reload** 命令来重启系统。

2.5.5 删除已设置的重启策略

在特权模式下，使用如下命令删除已设置的重启计划：

命令	作用
Ruijie# **reload cancel**	删除已设置的重启计划

如果之前没有设置重启计划，则用户会看到错误操作的提示信息。

2.6 系统名称和命令提示符

2.6.1 概述

为了管理的方便，你可以为一台网络设备配置系统名称(System Name)来标识它。同时如果你还没有为 CLI 配置命令提示符，则系统名称（如果系统名称超过 32 个字符，则截取其前 32 个字符）将作为默认的命令提示符，提示符将随着系统名称的变化而变化。默认情况下，以具体的设备名称作为系统名称，例如”S2924G”、”R2692”等。

2.6.2 配置系统名称

我司产品提供全局配置模式下的命令来配置系统的名称：

命令	作用
Ruijie(Config)# **hostname** *name*	设置系统名称，名称必须由可打印字符组成，长度不能超过 255 个字节。

你可以在全局配置模式下使用 **no hostname** 来将系统名称恢复位缺省值。下面的例子将网络设备的名称改成 RGOS：

```
Ruijie# configure terminal                    //进入全局配置模式
Ruijie(config)# hostname RGOS                 //设置网络设备名称为 RGOS
RGOS(config)#                                 //名称已经修改
```

2.6.3 配置命令提示符

如果你没有配置命令提示符，则系统名称（如果系统名称超过 32 个字符，则截取其前 32 个字符）将作为缺省提示符，提示符将随着系统名称的变化而变化。你可以在全局配置模式下使用 **prompt** 命令配置命令提示符，命令的提示符只对 EXEC 模式有效。

命令	作用
Ruijie# **prompt** *string*	设置命令提示符，名称必须由可打印字符组成，如果长度超过 32 个字符，则截取其前 32 个字符。

你可以在全局配置模式下使用 **no prompt** 来将命令提示符恢复为缺省值。

2.7 标题配置

2.7.1 概述

当用户登录网络设备时，你可能需要告诉用户一些必要的信息。你可以通过设置标题来达到这个目的。你可以创建两种类型的标题（banner）：每日通知和登录标题。每日通知针对所有连接到网络设备的用户，当用户登录网络设备时，通知消息将首先显示在终端上。利用每日通知，你可以发送一些较为紧迫的消息（比如系统即将关闭等）给网络用户。登录标题显示在每日通知之后，它的主要作用是提供一些常规的登录提示信息。缺省情况下，每日通知和登录标题均未设置。

2.7.2 配置每日通知

你可以创建包含一行或多行信息的通知信息，当用户登录网络设备时，这些信息将会被显示。您可以通过以下全局配置模式的命令来设置每日通知信息：

命令	作用

Ruijie(config)# **banner motd** *c message c*	设置每日通知(message of the day)的文本。c 表示分界符，这个分界符可以是任何字符(比如'&'等字符)。输入分界符后，然后按回车键，现在你可以开始输入文本，你需要在键入分界符并按回车键来结束文本的输入，需要注意的是，如果键入结束的分界符后仍然输入字符，则这些字符将被系统丢弃。需要注意的是，通知信息的文本中不应该出现作为分界符的字母，文本的长度不能超过 255 个字节。

你可以在全局配置模式下使用 **no banner motd** 来删除已配置的每日通知信息。下面的例子说明了如何配置一个每日通知，我们使用(#)作为分界符，每日通知的文本信息为"Notice: system will shutdown on July 6th."，配置实例如下：

```
Ruijie(config)# banner motd #                          //开始分界符
Enter TEXT message.  End with the character '#'.
Notice: system will shutdown on July 6th.#        //结束分界符
Ruijie(config)#
```

2.7.3 配置登录标题

您可以通过以下全局配置模式的命令来设置登录标题信息：

命令	作用
Ruijie(config)# **banner login** *c message c*	设置登录标题的文本。*c* 表示分界符，这个分界符可以是任何字符(比如'&'等字符)。输入分界符后，然后按回车键，现在你可以开始输入文本，你需要在键入分界符并按回车键来结束文本的输入，需要注意的是，如果键入结束的分界符后仍然输入字符，则这些字符将被系统丢弃。需要注意的是，登录标题的文本中不应该出现作为分界符的字母，文本的长度不能超过 255 个字节。

你可以在全局配置模式下使用 **no banner login** 来删除登录标题。

下面的例子说明了如何配置一个登录标题，我们使用(#)作为分界符，登录标题的文本为"Access for authorized users only. Please enter your password."，配置实例如下：

```
Ruijie(config)# banner login #                         //开始分界符
Enter TEXT message.  End with the character '#'.
Access for authorized users only. Please enter your password.
#                                                      //结束分界符
Ruijie(config)#
```

2.7.4 显示标题

标题的信息将在你登录网络设备时显示，下面是一个标题显示的例子：

```
C:\>telnet 192.168.65.236
 Notice:  system will shutdown on July 6th.
 Access for authorized users only. Please enter your password.
 User Access Verification
 Password:
```

其中“Notice: system will shutdown on July 6th.” 为每日通知，“Access for authorized users only. Please enter your password.”为登录标题。

2.8 查看系统信息

2.8.1 概述

你可以通过命令行中的显示命令查看一些系统的信息，主要包括系统的版本信息，系统中的设备信息等。

2.8.2 查看系统、版本信息

系统信息主要包括系统描述，系统上电时间，系统的硬件版本，系统的软件版本，系统的 Ctrl 层软件版本，系统的 Boot 层软件版本。你可以通过这些信息来了解这个网络设备系统的概况。你可以在特权模式下使用下表所列的命令来显示这些系统信息：

命令	作用
Ruijie# **show version**	显示系统、版本信息

2.8.3 显示硬件实体信息

硬件信息主要包括物理设备信息及设备上的插槽和模块信息。设备本身信息包括：设备的描述，设备拥有的插槽的数量；插槽信息：插槽在设备上的编号，插槽上的模块的描述（如果插槽没有插模块，则描述为空），插槽所插模块包括的物理端口数，插槽最多可能包含的端口的最大个数（所插模块包括的端口数）你可以在特权模式下使用下表所列的命令来显示设备和插槽的信息：

命令	作用
Ruijie# **show version devices**	显示网络设备当前的设备信息

Ruijie# **show version slots**	显示网络设备当前的插槽和模块信息

2.9 控制台速率配置

2.9.1 概述

网络设备有一个控制台接口（Console），通过这个控制台接口，可以对网络设备进行管理。当网络设备第一次使用的时候，必须采用通过控制台口方式对其进行配置。您可以根据需要改变网络设备串口的速率。需要注意的是，用来管理网络设备的终端的速率设置必须和网络设备的控制台的速率一致。

2.9.2 设置控制台速率

在线路配置模式下，您可以使用以下命令来设置控制台的速率：

命令	作用
Ruijie(config-line)# **speed** *speed*	设置控制台的传输速率，单位是bps。对于串行接口，你只能将传输速率设置为 9600、19200、38400、57600、115200 中的一个，缺省的速率是 9600。

下面的例子表示如何将串口速率设置为 57600 bps：

```
Ruijie# configure terminal                    //进入全局配置模式
Ruijie(config)# line console 0                //进入控制台线路配置模式
Ruijie(config-line)# speed 57600              //设置控制台速率为 57600
Ruijie(config-line)# end                      //回到特权模式
Ruijie# show line console 0                   //查看控制台配置
CON    Type    speed   Overruns
* 0    CON     57600   0
Line 0, Location: "", Type: "vt100"
Length: 25 lines, Width: 80 columns
Special Chars: Escape  Disconnect  Activation
            ^^x    none       ^M
Timeouts:     Idle EXEC    Idle Session
            never       never
History is enabled, history size is 10.
Total input: 22 bytes
Total output:  115 bytes
```

```
Data overflow:  0 bytes
stop rx interrupt:  0 times
Modem: READY
```

2.10 在网络设备上使用 telnet

2.10.1 概述

Telnet 在 TCP/IP 协议族中属于应用层协议，它给出了通过网络提供远程登录和虚拟终端通讯功能的规范。Telnet Client 服务为已登录到本网络设备上的本地用户或远程用户提供使用本网络设备的 Telnet Client 程序访问网上其他远程系统资源的服务。如下图所示用户在微机上通过终端仿真程序或 Telnet 程序建立与网络设备 A 的连接后，可通过输入 telnet 命令再登录设备 B，并对其进行配置管理。

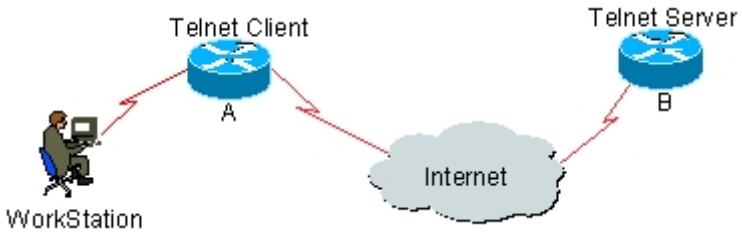


图 1

2.10.2 使用 Telnet Client

您可以通过网络设备上的 telnet 命令登录到远程设备上去：

命令	作用
Ruijie# **telnet** *host-ip-address*	通过 **telnet** 登录到远程设备，可以是主机名或 IP 地址。

下面的例子是如何建立 **Telnet** 会话并管理远程网络设备，远程网络设备的 ip 地址是 192.168.65.119：

```
Ruijie# telnet 192.168.65.119      //建立到远程设备的 telnet 会话
Trying 192.168.65.119 ... Open
User Access Verification                  //进入远程设备的登录界面
Password:
```

2.11 连接超时设置

2.11.1 概述

可以通过配置设备的连接超时时间，控制该设备已经建立的连接（包括已接受连接，以及该设备到远程终端的会话），当空闲时间超过设置值，没有任何输入输出信息时，中断此连接。

2.11.2 连接超时

当前已接受的连接，在指定时间内，没有任何输入信息，服务器端将中断此连接。

我司产品提供 LINE 配置模式下的命令来配置连接超时时间：

命令	作用
Ruijie(config-line)# **exec-timeout** *20*	配置 LINE 上，已接受连接的超时时间，当超过配置时间，没有任何输入时，将中断此连接。

可以在 LINE 配置模式下使用 **no exec-timeout** 命令，取消 LINE 下连接的超时设置。

```
Ruijie# configure terminal                    //进入全局配置模式
Ruijie# line vty 0                            //进入 LINE 配置模式
Ruijie(config-line)# exec-timeout 20          //设置超时时间为 20min
```

2.11.3 会话超时

当前 LINE 上已经建立的会话，在指定时间内，没有任何输入信息，将中断当前连接到远程终端的会话。并且恢复终端为空闲状态。

我司产品提供 LINE 配置模式下的命令来配置到远程终端的会话超时时间：

命令	作用
Ruijie(config-line)# **session-timeout** *20*	配置 LINE 上，连接到远程终端的会话超时时间，在指定时间内，没有任何输入时，将中断此会话。

可以在 LINE 配置模式下使用 **no exec-timeout** 命令，取消 LINE 下到远程终端的会话超时时间设置。

```
Ruijie# configure terminal                    //进入全局配置模式
```

```
Ruijie(config)# line vty 0                          //进入 LINE 配置模式
Ruijie(config-line)# session-timeout 20      //设置超时时间为 20min
```

2.12 批处理执行文件中的命令

在系统管理中，有时候需要输入较多的配置命令来实现对某个功能的管理，完全通过 CLI 界面输入需要较长的时间，也很容易造成错误和遗漏。如果将这些功能的配置命令按配置步骤全部放在一个批处理文件中，在需要配置时，执行这个批处理文件，就可以将相关的配置全部配置完毕。

命令	作用
Ruijie# **execute** {[**flash:**] *filename*}	执行一个批处理文件。

例如：批处理文件 line_rcms_script.text 用于打开所有异步口上的反向 **Telnet** 功能，文件内容如下：

```
configure terminal
line tty 1 16
transport input all
no exec
end
```

执行的结果：

```
Ruijie# execute flash:line_rcms_script.text
executing script file line_rcms_script.text ......
executing done
Ruijie# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Ruijie(config)# line vty 1 16
Ruijie(config-line)# transport input all
Ruijie(config-line)# no exec
Ruijie(config-line)# end
```

说明：

批处理文件的文件名和文件中的内容可以自行指定，一般是在用户的 PC 上编辑完毕通过 TFTP 方式传输到设备的 Flash 中。批处理的内容完全是模仿用户的输入，因此，必须按照 CLI 命令的配置顺序来编辑批处理文件的内容。另外，对于一些交互式命令，则需要在批处理文件中预先写入相应的应答信息，保证命令能够正常执行。

2.13 服务开关设置

在系统运行过程中，可以动态地调整系统所提供的服务，打开与关闭指定的服务（SSH Server/Telnet Server/Web Server）。

命令	作用
Ruijie(Config)# **enable service ssh-sesrver**	打开 SSH Server
Ruijie(Config)# **enable service telnet-server**	打开 Telnet Server
Ruijie(Config)# **enable service web-server**	打开 Http Server

可以在配置模式下，使用 **no enable service** 命令，关闭对应的服务。

```
Ruijie# configure terminal                    //进入全局配置模式
Ruijie(config)# enable service ssh-server     //打开 SSH Server
```

3 LINE 模式配置

3.1 概述

本章描述对 LINE 的一些操作：

- 进入 LINE 模式
- 增加/减少 LINE VTY 数目
- 配置 LINE 下可以通讯的协议

3.2 LINE 模式配置

3.2.1 进入 LINE 模式

通过进入到指定的 LINE 模式，可以在 LINE 模式下，对具体的 LINE 进行配置。要进入到指定的 LINE 模式，执行以下命令：

命令	作用
Ruijie(config)# **line** [**console** | **vty**] *first-line* [*last-line*]	进入指定的 LINE 模式

3.2.2 增加/减少 LINE VTY 数目

默认情况下，line vty 的数目为 5。可以通过命令增加或者减少 line vty 的数目。VTY 最大数目可以增加到 36。

命令	作用
Ruijie(config)# **line vty** *line-number*	将 LINE VTY 数目增加到某个值
Ruijie(config)# **no line vty** *line-number*	将 LINE VTY 数目减少到某个值

3.2.3 配置 Line 下的可通讯协议

如果需要限制 LINE 线路下可以通讯的协议类型，可以通过此命令进行设置。缺省情况下，VTY 类型可以允许所有协议进行通讯；而其它类型的 TTY，不允许任何协议进行通讯。

命令	说明
configure terminal	进入配置模式
Line vty *line number*	进入 Line 配置模式
transport input {**all** | **ssh** | **telnet** | **none**}	配置对应 Line 下可以通讯的协议
no transport input	配置 LINE 下不允许任何协议通讯
default transport input	恢复 LINE 下的通讯协议为默认配置

3.2.4 配置 Line 下的访问控制列表

如果需要配置 LINE 线路下的访问控制，可以通过此命令进行设置。缺省情况下，Line 下没有配置任何访问控制列表。接收所有连接，并允许所有外出的连接。

命令	说明
configure terminal	进入配置模式
Line vty *line number*	进入 Line 配置模式
access-class *access-list-number* {**in | out**}	配置对应 Line 下的访问控制列表
no access-class *access-list-number* {**in |out**}	取消 Line 下配置的访问控制列表

4 系统升级维护配置

4.1 概述

系统升级维护指的是在命令行界面下进行主程序或者 CTRL 程序的升级或者文件的上传和下载，通常有两种手段：一种是使用 TFTP 协议通过网口进行升级，另一种是使用 Xmodem 协议通过串口进行升级。

4.2 升级维护方法

我们将从以下几个小节描述如何升级维护设备的文件

- 通过 TFTP 协议传输文件
- 通过 XMODEM 协议传输文件

4.2.1 通过 TFTP 协议传输文件

一种是从主机端下载文件到设备，另一种是从设备上传文件到主机端。

在 CLI 命令模式下，按如下步骤设置来完成文件的下载：

下载前，首先在本地主机端打开 TFTP Server 软件；然后选定要下载的文件所在的目录；而后登录到设备，在特权模式下使用以下命令下载文件，如果没有指明 Location 则需要单独输入 TFTP Server 的 IP 地址。

命令	作用
Ruijie# **copy tftp:** *//location/filename* **flash:** *filename* [**vrf** vrfname]	下载主机端的 URL 指定的文件 *filename* 到设备。

在 CLI 命令模式下，按如下步骤设置来完成文件的上传：

上传前，首先在本地主机端打开 TFTP Server 软件；然后在主机端选定要保存上传文件的目录，而后在特权模式下使用以下命令上传文件。

命令	作用
Ruijie# **copy flash:** *filename* **tftp:** *//loca tion/filename* [**vrf** vrfname]	从设备端上传文件 *filename* 到主机端的 URL 指定的目录下，可重设文件名。

注意：

如果源文件的文件名中有空格，则需要给tftp链接加上引号，如下：

copy tftp:”//localtion/filename” **flash:**filename [**vrf** vrfname]

同样的，如果目标的文件名中有空格，也需要将文件名加上引号，如下：
copy tftp://localtion/filename **flash:**”filename” [**vrf** vrfname]

4.2.2 通过 XMODEM 协议传输文件

一种是从主机端下载文件到设备，另一种是从设备上传文件到主机端。

在 CLI 命令模式下，按如下步骤设置来完成文件的下载：

下载前，首先通过 Windows 超级终端登录到设备的带外管理界面；然后在特权模式下使用以下命令下载文件；而后在本地主机的 Windows 超级终端中，选择“传送”菜单中的“发送文件”功能，如图 1 所示：

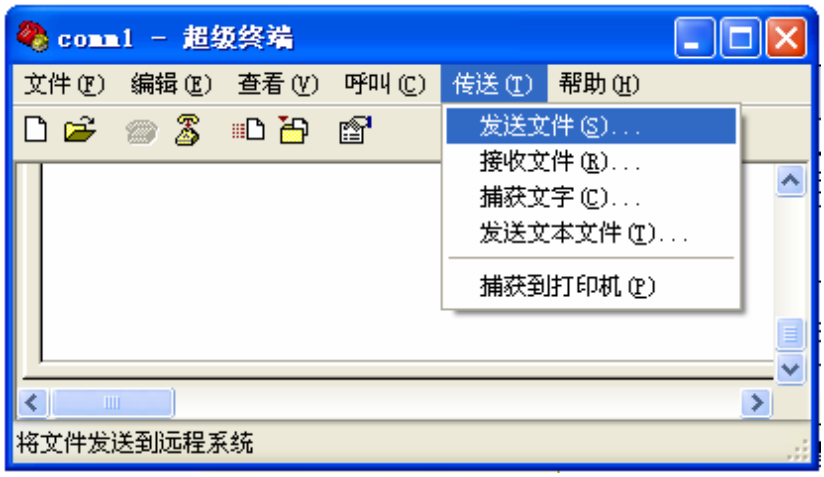


图 1

在弹出的对话框的文件名选择本地主机要下载的文件，协议选择“Xmodem”，点击“发送”，则 Windows 超级终端显示发送的进度以及数据包。如图 2 所示：

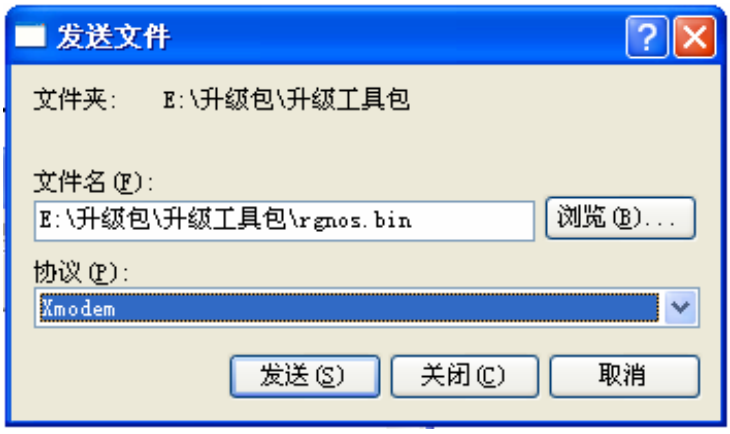


图 2

命令	作用

Ruijie# **copy xmodem flash**:*filename*	从主机端下载文件到设备端并命名为 *filename*。

在 CLI 命令模式下，按如下步骤设置来完成文件的上传：

上传前，首先通过 Windows 超级终端登录到设备的带外管理界面；然后在特权模式下使用以下命令上传文件；最后在本地主机的 Windows 超级终端中，选择“传送”菜单中的“接收文件”功能。如图 3 所示：

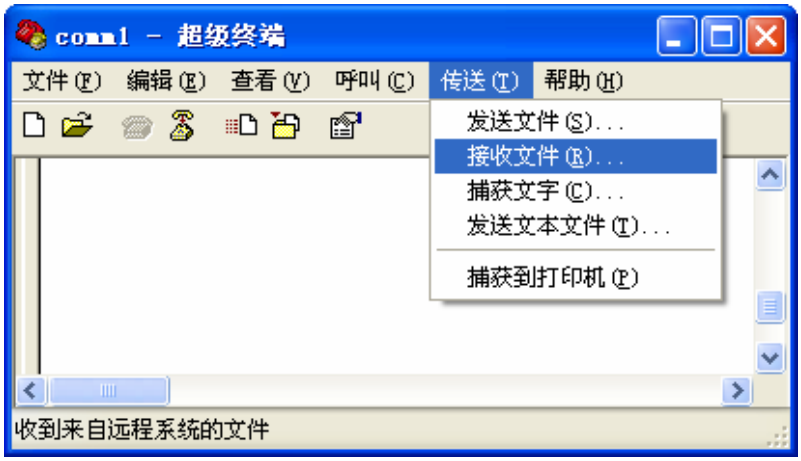


图 3

在弹出的对话框选择上传文件的存储位置，接收协议选择“Xmodem”，点击“接收”，超级终端会进一步提示用户本地存储文件的名称，点击“确认”后开始接收文件。如图 4 所示：

图 4

命令	作用
Ruijie# **copy flash**：*filename* **xmodem**	从设备端上传文件 *filename* 到主机端

注意：

如果文件名中有空格，需要将文件名加上引号，如下：

copy xmodem flash:”filename”**或者copy flash:**”filename” **xmodem**

4.2.3 升级系统

不论是盒式设备还是机箱设备，使用都可以使用上述的 tftp 或者 xmodem 将升级文件传输到设备上。传输成功后，重新启动设备，升级文件就会自动完成当前系统的检测和升级，这个过程不需要人工干预和介入。

升级文件在盒式设备和机箱设备上的升级动作有所不同：

1）盒式设备的升级只完成自己单板系统的升级操作，升级完毕，系统自动复位，机器再次启动正常运行。

2）机箱设备包含有管理板，线卡以及多业务卡，要通过一个升级文件完成整套系统的升级操作。首先待管理板端升级完毕，系统复位。机器再次启动时，版本自动同步功能会启动起来，完成线卡和多业务卡的系统升级。

自动升级功能：是运行于主管理板端的一个功能，会对系统中存在的从管理板，线卡和多业务卡进行版本一致性检查，当发现他们的版本和主管理板中对应的单板版本不一致时，就将他们的单板升级文件传送过去，完成他们的升级，以使整个系统的版本保持一致。

注意：

任何时候升级机箱设备的主管理板，都会同时升级从管理板，从而保持版本的一致性。升级线卡的操作会使整个系统中插入的线卡同时得到升级。在升级结束提示前不可断电，否则可能导致升级程序丢失。

在机箱设备升级未完成前，可通过**show version**检查所有线卡和管理板的软件版本是否与升级的目标版本一致的方式检查升级是否完成，不能进行主备切换（例如：**redundancy force-switchover**），否则会导致升级失败而退回原始版本。

- 通过升级文件来升级机箱设备：

1） 确认要载入的升级文件的文件名为 **rgos.bin**

2） 使用上述 **copy** 命令，将此文件下载到设备上

3） 如果设备上存在从管理板，需要等待主从管理板的主程序升级成功，成功时将有如下提示：

```
Upgrade Slave CM MAIN successful!!
Upgrade CM MAIN successful!!
```

1） 执行一次整机复位操作

2） 系统再次启动后，升级文件会开始运行，将会看到类似如下提示：

```
Installing is in process ......
Do not restart your machine before finish !!!!!!
```

```
......
```

3） 在升级操作完成后，将会看到类似如下提示：

```
Installing process finished ......
Restart machine operation is permited now !!!!!!
```

4） 升级文件运行完毕后系统会自动复位，出现如下提示：

```
System restarting, for reason 'Upgrade product !'.
```

5） 系统再次启动后，管理板的整个系统会完成升级，然后加载管理板的单板升级文件运行，这还会有步骤 5 和 6 的提示信息，但不会有步骤 7 的提示信息，取而代之的是：

```
System load main program from install package ......
```

直接从升级文件中加载管理板的主程序运行

6） 主程序正常运行后，自动升级功能就会启动，如果机箱中有从管理板或者其他模块会看到如下提示：

```
A new card is found in slot [1].
System is doing version synchronization checking ......
Current software version in slot [1] is synchronous.
System needn't to do version synchronization for this
card ......
```

或者如下提示：

```
System is doing version synchronization checking ......
Card in slot [3] need to do version synchronization ......
```

其他打印信息

```
Version synchronization begain ......
Keep power on, don't draw out the card and don't restart your
machine before finished !!!!!!
```

其他打印信息

```
Transmission is OK, now, card in slot [3] need restart ...
Software installation of card in slot [3] is in process ......
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!
Software installation of card in slot [3] has finished
successfully ......
The version synchronization of card in slot [3] get finished
successfully.
```

上述两种情况一种表示，线卡的版本已经是同步的不需要再次自动升级，另一种表

示线卡的版本，还需要自动升级，然后做升级操作。

系统会依次对从板和每一个模块完成上述操作。

根据提示等待系统完成所有模块的版本一致检查和升级操作，系统便可正常工作了。

注意：

在升级或者自动升级过程中，会有不允许重启的提示，一旦出现类似提示，请务必不要断电或者复位系统，也不要随便插拔其他模块。

说明：

对于热插入的模块系统会有同样的自动升级检查操作。

- 通过升级文件来升级盒式设备：

盒式设备的升级只需要完成上述步骤的 1－7，然后系统自行复位后就会正常运行了。

5 网络通信检测工具

5.1 Ping 连通性测试

为了测试网络的连通性，很多的网络设备都支持 Echo 协议，该协议包括发送一个特殊的数据包给指定的网络地址，然后等待该地址应答回来的数据包，通过 Echo 协议，可以评估网络的连通性、延时和网络的可靠性，利用 RGOS 提供的 Ping 工具，可以有效的帮助用户诊断、定位网络中的连通性问题。

Ping 命令运行在普通用户模式和特权用户模式下，在普通用户模式下，只能运行基本的 Ping 功能，而在特权用户模式下，还可以运行 Ping 的扩展功能。

命令	作用
Ruijie# **ping** [*ip*] [*address* [**length** *length*] [**ntimes** *times*] [**data** *data*][**source** *source*] [**timeout** *seconds*]]	**Ping**：网络连通性测试工具

普通的 Ping 功能，可以在普通用户模式和特权用户模式下执行，缺省将 5 个长度为 100Byte 的数据包发送到指定的 IP 地址，在指定的时间（缺省为 2 秒）内，如果有应答，则显示‘!’符号；如果没有应答，则显示‘.’符号。最后输出一个统计信息。以下为普通 ping 的实例：

```
Ruijie# ping 192.168.5.1
Sending 5, 100-byte ICMP Echoes to 192.168.5.1, timeout is 2
seconds:
< press Ctrl+C to break >
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max =
1/2/10 ms
```

扩展的 Ping 功能，只能在特权用户模式下执行。在扩展 Ping 中，可以指定发送数据包的个数、长度、超时的时间等等。和普通的 Ping 功能一样，最后也输出一个统计信息， 以下为一个扩展 Ping 的实例：

```
Ruijie# ping 192.168.5.197 length 1500 ntimes 100 data ffff
source 192.168.4.190 timeout 3
Sending 100, 1000-byte ICMP Echoes to 192.168.5.197, timeout
is 3 seconds:
  < press Ctrl+C to break >
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
Success rate is 100 percent (100/100), round-trip min/avg/max
= 2/2/3 ms
```

5.2 Traceroute 连通性测试

执行 **Traceroute** 命令，可以显示数据包从源地址到目的地址，所经过的所有网关。**Traceroute** 命令主要用于检查网络的连通性，并在网络故障发生时，准确地定位故障发生的位置。

网络传输的规则是，一个数据包每经过一个网关，数据包中的 TTL 域的数据执行减 1 操作。当 TTL 域的数据为 0 时，该网关便丢弃这个数据包，并送回一个地址不可达的错误数据包给源地址。根据这个规则，Traceroute 命令的执行过程是：首先给目的地址发送一个 TTL 为 1 的数据包，第一个网关便送回一个 ICMP 错误消息，以指明此数据包不能被发送，因为 TTL 超时，之后将数据包的 TTL 域加 1 后重新发送，同样第二个网关返回 TTL 超时错误，这个过程一直继续下去，直到到达目的地址，记录每一个回送 ICMP TTL 超时信息的源地址，便记录下了数据从源地址到达目的地址，IP 数据包所经历的整个完整的路径。

Traceroute 命令可以在普通用户模式和特权用户模式下执行，具体的命令格式如下：

命令	功能
Ruijie# **traceroute** [*protocol*] [*destination* [**probe** *probe*] [**ttl** *minimum maximum*] [**source** *source*] [**timeout** *seconds*]]	跟踪数据包发送网络路径

以下为应用 Traceroute 的两个例子，一个为网络连接畅通，一个为网络连接存在某些网关不通的情况。

1．网络畅通的 Traceroute 例子：

```
Ruijie# traceroute 61.154.22.36
  < press Ctrl+C to break >
Tracing the route to 61.154.22.36
1     192.168.12.1       0 msec  0 msec  0 msec
2     192.168.9.2        4 msec  4 msec 4 msec
3     192.168.9.1        8 msec  8 msec  4 msec
4     192.168.0.10       4 msec  28 msec  12 msec
5     202.101.143.130    4 msec  16 msec  8 msec
6     202.101.143.154   12 msec  8 msec  24 msec
7     61.154.22.36       12 msec  8 msec  22 msec
```

从上面的结果可以清楚地看到，从源地址要访问 IP 地址为 61.154.22.36 的主机，网络数据包都经过了哪些网关（1－6），同时给出了到达该网关所花费的时间，这对于网络分析，是非常有用的。

2．网络中某些网关不通的 Traceroute 例子：

```
Ruijie# traceroute 202.108.37.42
< press Ctrl+C to break >
```

```
Tracing the route to 202.108.37.42
1     192.168.12.1      0 msec  0 msec  0 msec
2     192.168.9.2       0 msec  4 msec 4 msec
3     192.168.110.1    16 msec  12 msec  16 msec
4     *  *  *
5     61.154.8.129      12 msec  28 msec  12 msec
6     61.154.8.17        8 msec  12 msec  16 msec
7     61.154.8.250      12 msec  12 msec  12 msec
8     218.85.157.222    12 msec  12 msec  12 msec
9     218.85.157.130    16 msec  16 msec  16 msec
10    218.85.157.77     16 msec  48 msec  16 msec
11    202.97.40.65      76 msec  24 msec  24 msec
12    202.97.37.65      32 msec  24 msec  24 msec
13    202.97.38.162     52 msec  52 msec  224 msec
14    202.96.12.38      84 msec  52 msec  52 msec
15    202.106.192.226   88 msec  52 msec  52 msec
16    202.106.192.174   52 msec  52 msec  88 msec
17    210.74.176.158   100 msec  52 msec  84 msec
18    202.108.37.42      48 msec  48 msec  52 msec
```

从上面的结果可以清楚地看到，从源地址要访问 IP 地址为 202.108.37.42 的主机，网络数据包都经过了哪些网关（1－17），并且网关 4 出现了故障。

6 接口配置

6.1 接口类型概述

本章主要对锐捷设备的接口类型进行划分，并对每种接口类型进行详细定义。锐捷设备的接口类型可分为以下两大类：

- 二层接口(L2 interface)
- 三层接口(L3 interface) (三层设备支持)

6.1.1 二层接口(L2 interface)

本节主要描述二层接口的类型及相关的定义，可分为以下几种类型

- Switch Port
- L2 Aggregate Port

6.1.1.1 Switch Port

Switch Port由设备上的单个物理端口构成，只有二层交换功能。该端口可以是一个 Access Port或一个 Trunk Port，您可以通过Switch Port接口配置命令，把一个端口配置为一个 Access Port或者 Trunk Port。Switch Port被用于管理物理接口和与之相关的第二层协议，并且不处理路由和桥接。

6.1.1.1.1 Access Port

每个 Access Port 只能属于一个 VLAN,它只传输属于这个 VLAN 的帧。一般用于连接计算机。

缺省 VLAN

每个 Access Port 只属于一个 VLAN，所以它的缺省 VLAN 就是它所在的 VLAN，可以不用设置。

帧的接收与发送

Access Port 发送出的数据帧是不带 TAG 的，且它只能接收以下三种格式的帧：

- Untagged 帧
- VID 为 Access Port 所属 VLAN 的 Tagged 帧
- VID 为 0 的 Tagged 帧

Untagged 帧

Access Port 接收不带 TAG 标志的帧，并为无 TAG 帧添加缺省 VLAN 的 TAG。发送前，去掉添加的 TAG，再发送。

Tagged 帧

Access 端口接收到的数据帧带有 TAG 时，将按照以下条件进行处理：

- 当 TAG 的 VID（VLAN ID）与缺省 VLAN ID 相同时，接收该数据帧，并在发送时去掉 TAG 标志后发送。
- 当 TAG 的 VID（VLAN ID）为 0 时，接收该数据帧。在 TAG 中，VID＝0 用于识别帧优先级。
- 当 TAG 的 VID（VLAN ID）与缺省 VLAN ID 不同且不为 0 时，丢弃该帧。

6.1.1.1.2 Trunk Port

每个 Trunk port 可以属于多个 VLAN，能够接收和发送属于多个 VLAN 的帧，一般用于设备之间的连接，也可以用于连接用户的计算机。

缺省 VLAN

因为 Trunk Port 可以属于多个 VLAN，所以需要设置一个 Native vlan 作为缺省 VLAN。缺省情况下 Trunk port 将传输所有 VLAN 的帧，为了减轻设备的负载，减少对带宽的浪费，可通过设置 VLAN 许可列表来限制 Trunk port 传输哪些 VLAN 的帧。

注意:

建议将本端设备 Trunk 端口的 native vlan 和相连的对端设备 Trunk 端口的 native vlan 配置为一致，否则端口可能无法正确转发报文。

帧的接收与发送

Trunk port 可接收 Untagged 帧和端口允许 VLAN 范围内的 tagged 帧。Trunk Port 发送的非 Native vlan 的帧都是带 TAG 的，而发送的 Native vlan 的帧都不带 TAG。

Untagged 帧

若 Trunk port 接收到的帧不带 IEEE802.1Q TAG,那么帧将在这个接口的 Native VLAN 中传输。

Tagged 帧

若 Trunk port 接收到帧是带 TAG 的，将按照以下条件进行处理：

- 当 Trunk Port 接收到的帧所带 TAG 的 VID 等于该 Trunk port 的 Native vlan 时，允许接收该数据帧；在发送该帧时，将去掉 TAG 后再发送。

- 当 Trunk Port 接收到的帧所带 TAG 的 VID 不等于该 Trunk port 的 Native vlan，但 VID 是该端口允许通过的 VLAN ID 时，接收该数据帧；发送时，将保持原有 TAG。
- 当 Trunk Port 接收到的帧所带 TAG 的 VID 不等于该 Trunk port 的 Native vlan，且 VID 是该端口不允许通过的 VLAN ID 时，丢弃该报文。

说明：

Untagged 报文就是普通的 Ethernet 报文，普通 PC 机的网卡是可以识别这样的报文进行通讯；TAG 报文结构的变化是在源 MAC 地址和目的 MAC 地址之后，加上了 4bytes 的 VLAN 信息，也就是 VLAN TAG 头。

6.1.1.1.3 Hybrid 端口

Hybrid 类型的端口可以属于多个 VLAN，可以接收和发送多个 VLAN 的报文，可以用于设备之间连接，也可以用于连接用户的计算机。Hybrid 端口和 Trunk 端口的不同之处在于 Hybrid 端口可以允许多个 VLAN 的报文发送时不打标签，而 Trunk 端口只允许缺省 VLAN 的报文发送时不打标签，需要注意的是：Hybrid 端口加入的 VLAN 必须已经存在。

6.1.1.2 L2 Aggregate Port

Aggregate port 是由多个物理成员端口聚合而成的。我们可以把多个物理链接捆绑在一起形成一个简单的逻辑链接，这个逻辑链接我们称之为一个 Aggregate Port（以下简称 AP ）。

对于二层交换来说 AP 就好像一个高带宽的 Switch port，它可以把多个端口的带宽叠加起来使用，扩展了链路带宽。此外，通过 L2 Aggregate port 发送的帧还将在 L2 Aggregate port 的成员端口上进行流量平衡，如果 AP 中的一条成员链路失效，L2 Aggregate port 会自动将这个链路上的流量转移到其他有效的成员链路上，提高了连接的可靠性。

注意：

L2 Aggregate Port 的成员端口类型可以为 Access port 或 Trunk Port，但同一个 AP 的成员端口必须为同一类型，要么全是 Access Port，要么全是 Trunk port。

6.1.2 三层接口(L3 interface)

本节主要描述三层接口的类型及相关的定义，可分为以下几种类型

- SVI (Switch virtual interface)

- Routed Port
- L3 Aggregate Port

6.1.2.1 SVI(Switch virtual interface)

SVI 是交换虚拟接口，用来实现三层交换的逻辑接口。SVI 可以做为本机的管理接口，通过该管理接口管理员可管理设备。您也可以创建 SVI 为一个网关接口，就相当于是对应各个 VLAN 的虚拟的子接口，可用于三层设备中跨 VLAN 之间的路由。创建一个 SVI 很简单，您可通过 **interface vlan** 接口配置命令来创建 SVI，然后给 SVI 分配 IP 地址来建立 VLAN 之间的路由。

如图所示，VLAN20 的主机可直接互相通讯，无需通过三层设备的路由，若 VLAN20 内的主机 A 想和 VLAN30 内的主机 B 通讯必须通过 VLAN20 对应的 SVI1 和 VLAN30 对应的 SVI2 才能实现。

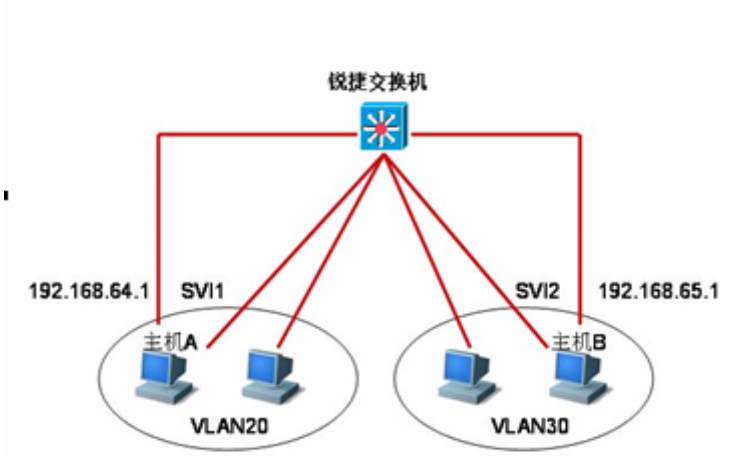


图 1

6.1.2.2 Routed Port

一个 Routed Port 是一个物理端口，就如同三层设备上的一个端口，能用一个三层路由协议配置。在三层设备上，可以把单个物理端口设置为 Routed port，作为三层交换的网关接口。一个 Routed Port 与一个特定的 Vlan 没有关系，而是作为一个访问端口。Routed port 不具备二层交换的功能。您可通过 **no switchport** 命令将一个二层接口 Switch port 转变为 Routed port,然后给 Routed port 分配 IP 地址来建立路由。注意的是，当使用 **no switchport** 接口配置命令时，该端口关闭并重启，将删除该端口的所有二层特性。

注意:

当一个端口是 L2 Aggregate Port 的成员端口或者是未认证成功的 DOT1X 认证口时，是不能用 **switchport/ no switchport** 命令进行层次切换的。

6.1.2.3 L3 Aggregate Port

L3 Aggregate port 同 L2 Aggregate Port 一样，也是由多个物理成员端口汇聚构成的一个逻辑上的聚合端口组。汇聚的端口必须为同类型的三层接口。对于三层交换来说，AP 作为三层交换的网关接口，它相当于把同一聚合组内的多条物理链路视为一条逻辑链路，是链路带宽扩展的一个重要途径。此外，通过 L3 Aggregate port 发送的帧同样能在 L3 Aggregate port 的成员端口上进行流量平衡，当 AP 中的一条成员链路失效后，L3 Aggregate port 会自动将这个链路上的流量转移到其他有效的成员链路上，提高了连接的可靠性。

L3 Aggregate port 不具备二层交换的功能。您可通过 **no switchport** 将一个无成员二层接口 L2 Aggregate port 转变为 L3 Aggregate Port，接着将多个 Routed Port 加入此 L3 Aggregate port，然后给 L3 Aggregate Port 分配 IP 地址来建立路由。

6.2 配置接口

本节描述接口的缺省配置，配置指南，配置步骤，配置实例

6.2.1 接口编号规则

对于 Switch Port，其编号由两个部分组成：插槽号，端口在插槽上的编号。例如端口所在的插槽编号为 2，端口在插槽上的编号为 3，则端口对应的接口编号为 2/3。插槽的编号是从 0－插槽的个数。插槽的编号规则是：面对设备的面板，插槽按照从前至后，从左至右，从上至下的顺序一次排列，对应的插槽号从 1 开始依次增加。插槽上的端口编号是从 1－插槽上的端口数，编号顺序是从左到右。对于可以选择介质类型的设备，端口包括两种介质（光口和电口），无论使用那种介质，都使用相同的端口编号。您也可以通过命令行中的 **show** 命令来查看插槽以及插槽上的端口信息。

对于 Aggregate Port，其编号的范围为 1－设备支持的 Aggregate Port 个数。

对于 SVI，其编号就是这个 SVI 对应的 VLAN 的 VID。

注意：

设备上的静态插槽的编号固定为 0，而动态插槽(可插拔模块或线卡)的编号从 1 开始。

6.2.2 接口配置命令的使用

您可在全局配置模式下使用 interface 命令进入接口配置模式。

命令	作用
Ruijie(config)# **interface** *接口 ID*	在全局配置模式下输入 interface 命令，进入接口配置模式。用户也可以在全局配置模式下使用 interface range 或 interface range macro 命令配置一定范围的接口。但是定义在一个范围内的接口必须是相同类型和具有相同特性的

下例给出了进入 Gigabitethernet 2/1 接口的示例：

```
Ruijie(config)# interface gigabitethernet 2/1
Ruijie(config-if)#
```

在接口配置模式下您可配置接口的相关属性。

6.2.3 使用 interface range 命令

6.2.3.1 配置一定范围的接口

用户可以使用全局配置模式下的 **interface range** 命令同时配置多个接口。当进入 **interface range** 配置模式时，此时设置的属性适用于所选范围内的所有接口。

命令	作用
Ruijie(config)# **interface range** {*port-range* | **macro** *macro_name*}	输入一定范围的接口。 **interface range** 命令可以指定若干范围段。 macro 参数可以使用范围段的宏定义，参见配置和使用端口范围的宏定义。 每个范围段可以使用逗号（,）隔开。 同一条命令中的所有范围段中的接口必须属于相同类型。

当使用 **interface range** 命令时，请注意 range 参数的格式：

有效的接口范围格式：

vlan *vlan-ID* - *vlan-ID*, VLAN ID 范围 1～4094；

Fastethernet *slot*/{*第一个 port*} - {*最后一个 port*}；

Gigabitethernet *slot*/{*第一个 port*} - {*最后一个 port*}；

TenGigabitethernet *slot*/{*第一个 port*} - {*最后一个 port*}；

Aggregate Port Aggregate *port 号 - Aggregate port 号*, 范围 1～MAX；

在一个 **interface range** 中的接口必须是相同类型的，即或者全是 fastethernet，gigabitethernet 或者全是 Aggregate Port，或者全是 SVI。

下面的例子是在全局配置模式下使用 **interface range** 命令：

```
Ruijie# configure terminal
Ruijie(config)# interface range fastethernet 0/1 - 10
Ruijie(config-if-range)# no shutdown
Ruijie(config-if-range)#
```

下面的例子是如何使用分隔符号（,）隔开多个 range：

```
Ruijie# configure terminal
Ruijie(config)# interface range fastethernet 0/1-5, 1/7-8
Ruijie(config-if-range)# no shutdown
Ruijie(config-if-range)#
```

6.2.3.2 配置和使用端口范围的宏定义

用户可以自行定义一些宏来取代端口范围的输入。但在用户使用 **interface range** 命令中的 **macro** 关键字之前，必须先在全局配置模式下使用 **define interface-range** 命令定义这些宏。

命令	作用
Ruijie(config)# **define interface-range** *macro_name interface-range*	定义接口范围的宏定义。 macro_name－宏定义的名字，不超过 32 个字符。 宏定义的内部可以包括多个范围段。 同一宏定义中的所有范围段中的接口必须属于相同类型。
Ruijie(config)# **interface range macro** *macro_name*	宏定义的字符串将被保存在内存中，使用 **interface range** 命令时，可以使用宏定义的名字来取代需要输入的表示接口范围的字符串。

在全局配置模式下使用 **no define interface-range macro_name** 命令来删除设置的宏定义。

当使用 **define interface-range** 命令来定义接口范围时，注意：

有效的接口范围格式：

- **vlan** *vlan-ID - vlan-ID*, VLAN ID 范围 1～4094；
- **fastethernet** *slot*/{*第一个 port*} - {*最后一个 port*}；

- **gigabitethernet** *slot*/{*第一个 port*} - {*最后一个 port*};
- **Aggregate Port** *Aggregate port 号 - Aggregate port 号*, 范围 1～MAX;

在一个 **interface range** 中的接口必须是相同类型的，即或者全是 switch port，或者全是 Aggregate Port，或者全是 SVI。

下面的例子是如何使用 **define interface-range** 命令来定义 fastethernet0/1-4 的宏定义：

```
Ruijie# configure terminal
Ruijie(config)# define interface-range resource
fastethernet 0/1-4
Ruijie(config)# end
```

下面的例子显示如何定义多个接口范围段的宏定义：

```
Ruijie# configure terminal
Ruijie(config)# define interface-range ports1to2N5to7
fastethernet 0/1-2, 1/5-7
Ruijie(config)# end
```

下面的例子显示使用宏定义 ports1to2N5to7 来配置指定范围的接口：

```
Ruijie# configure terminal
Ruijie(config)# interface range macro ports1to2N5to7
Ruijie(config-if-range)#
```

下面的例子显示如何删除宏定义 ports1to2N5to7：

```
Ruijie# configure terminal
Ruijie(config)# no define interface-range ports1to2N5to7
Ruijie# end
```

6.2.4 选择接口介质类型

有些接口，可以有多种介质类型供用户选择。您可以选择其中一种介质使用。一旦您选定介质类型，接口的连接状态、速度、双工、流控等属性都是指该介质类型的属性，如果您改变介质类型，新选介质类型的这些属性将使用默认值，您可以根据需要重新设定这些属性。

此配置命令只对物理端口有效。Aggregate Port 和 SVI 接口不支持介质类型设置。

此配置命令只对支持介质选择的端口有效。

配置为 Aggregate Port 成员口的端口，其介质类型必须一致，否则无法加入到 AP 中。Aggregate Port 成员口的端口类型不能改变。

命令	作用
Ruijie(config-if)# **medium-type** { **fiber** | **copper** }	设置端口的介质类型

下面的例子显示了如何设置接口 Gigabitethernet 1/1 的介质类型：

```
Ruijie# config terminal
Enter configuration commands, one per line. End with CNTL/Z.
Ruijie(config)# interface gigabitethernet 1/1
Ruijie(config-if)# medium-type fiber
Ruijie(config-if)# end
```

6.2.5 配置接口的描述和管理状态

为了有助于您记住一个接口的功能，您可以为一个接口起一个专门的名字来标识这个接口，也就是接口的描述(Description)。您可以根据要表达的含义来设置接口的具体名称，比如，您想将 Gigabitethernet 1/1 分配给用户 A 专门使用，您就可以将这个接口的描述设置为“Port for User A”。

命令	作用
Ruijie(config-if)# **description** *string*	设置接口的描述，最多 32 个字符。

下面的例子显示了如何设置接口 Gigabitethernet 1/1 的描述：

```
Ruijie# config terminal
Enter configuration commands, one per line. End with CNTL/Z.
Ruijie(config)# interface gigabitethernet 1/1
Ruijie(config-if)# description PortForUser A
Ruijie(config-if)# end
```

在某些情况下，您可能需要禁用某个接口。您可以通过设置接口的管理状态来直接关闭一个接口。如果关闭一个接口，则这个接口上将不会接收和发送任何帧，这个接口将丧失这个接口对应的所有功能。您也可以通过设置管理状态来重新打开一个已经关闭的接口。接口的管理状态有两种：Up 和 Down，当端口被关闭时，端口的管理状态为 down，否则为 up。

命令	作用
Ruijie(config-if)# **shutdown**	关闭一个接口

下面的例子描述如何关闭接口 Gigabitethernet 1/2：

```
Ruijie# configure terminal
Ruijie(config)# interface gigabitethernet 1/2
Ruijie(config-if)# shutdown
Ruijie(config-if)# end
```

6.2.6 配置接口的速度，双工，流控

本节描述如何配置接口的速率、双工和流控模式。

以下配置命令只对 Switch Port,Routed Port 有效。

命令	作用
Ruijie(config-if)# **speed {10 | 100 | 1000 | auto }**	设置接口的速率参数，或者设置为 auto。 注意： 1000 只对千兆口有效，设备的光口速率强制为 1000M。
Ruijie(config-if)# **duplex {auto | full | half }**	设置接口的双工模式。
Ruijie(config-if)# **flowcontrol {auto | on | off }**	设置接口的流控模式。 注意：当 speed,duplex,flowcontrol 都设为非 auto 模式时，该接口关闭自协商过程

在接口配置模式下使用 **no speed** ， **no duplex** 和 **no flowcontrol** 命令，将接口的速率、双工和流控配置恢复为缺省值（自协商）。下面的例子显示如何将 Gigabitethernet 1/1 的速率设为 1000M，双工模式设为全双工，流控关闭：

```
Ruijie# configure terminal
Ruijie(config)# interface gigabitethernet 1/1
Ruijie(config-if)# speed 1000
Ruijie(config-if)# duplex full
Ruijie(config-if)# flowcontrol off
Ruijie(config-if)# end
```

注意:

S3760 系列的光口的协商能力配置，需要与对端的配置一致，否则可能会出现端口不能 link up 问题。

6.2.7 配置接口的 MTU

当端口进行大吞吐量数据交换时，可能会遇到大于以太网标准帧长度的帧，这种帧被称为 jumbo 帧。用户可以通过设置端口的 MTU 来控制该端口允许收发的最大帧长。

MTU 是指帧中有效数据段的长度，不包括以太网封装的开销。

端口的 MTU 检查只在输入时进行。输出时不会检查 MTU。端口收到的帧，如果长度超过设置的 MTU，将被丢弃。

MTU 允许设置的范围为 64~9216 字节，粒度为 4 字节，缺省为 1500 字节。

此配置命令只对物理端口有效。SVI 接口暂时不支持 MTU 设置。

命令	作用
Ruijie(config-if)# **mtu** *num*	设置端口的 MTU，Num：<64-9216>

下面的例子显示了如何设置接口 Gigabitethernet 1/1 的 MTU：

```
Ruijie# config terminal
Enter configuration commands, one per line. End with CNTL/Z.
Ruijie(config)# interface gigabitethernet 1/1
Ruijie(config-if)# mtu 64
Ruijie(config-if)# end
```

说明：

S3760 系列产品的端口工作在 10/100M 模式下生效的 MTU 配置为：1490 Bytes、1500Bytes、1530Bytes 与 2026Bytes，端口工作在 1000M 模式下生效的 MTU 为：1490 Bytes、1500Bytes、1530Bytes、2026Bytes、9000Bytes、9170Bytes 与 10218Bytes。

6.2.8 配置二层接口

下表显示了二层接口的缺省配置，有关 VLAN 及端口的配置请参照“配置 VLAN”和“配置基于端口的流量控制”。

二层接口的缺省配置如下表：

属性	缺省设置
工作模式	二层交换模式
Switch port 模式	access port
允许的 VLAN 范围	VLAN 1～4094
缺省 VLAN（对于 access port 而言）	VLAN 1
Native VLAN（对于 trunk port 而言）	VLAN 1
介质类型	copper
接口管理状态	Up
接口描述	空
速度	自协商
双工模式	自协商

流控	自协商
Aggregate port	无
风暴控制	关闭
保护端口	关闭
端口安全	关闭

6.2.8.1 配置 Switch Port

6.2.8.1.1 配置 access/trunk port

本节主要讲述配置 Switchport 的操作模式(access/trunk port)及每种模式下的相关配置。

您可在接口配置模式下通过 **switchport** 或其他命令来配置 Switch Port 的相关属性：

命令	作用
Ruijie(config-if)**# switchport mode {access | trunk }**	配置接口的操作模式。

下例显示如何配置 gigabitethernet 1/2 的操作模式为 access port。

```
Ruijie# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Ruijie(config)# interface gigabitethernet 1/2
Ruijie(config-if)# switchport mode access
Ruijie(config-if)# end
```

命令	作用
Ruijie(config-if)**# switchport access vlan** *vlan-id*	配置 access port 所属的 VLAN。

下例显示如何配置 access port gigabitethernet 2/1 所属 vlan 为 100。

```
Ruijie# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Ruijie(config)# interface gigabitethernet 2/1
Ruijie(config-if)# switchport access vlan 100
Ruijie(config-if)# end
```

配置 trunk port 的 native VLAN

命令	作用

Ruijie(config-if)**# switchport trunk native vlan** *vlan-id*	配置 trunk port 的 NATIVE VLAN。

下例显示如何配置 Trunk Port Gigabitethernet 2/1 的 Native vlan 为 10。

```
Ruijie# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Ruijie(config)# interface gigabitethernet 2/1
Ruijie(config-if)# switchport trunk native vlan 10
Ruijie(config-if)# end
```

配置接口的端口安全，有关端口安全更详细的信息请参照“基于端口的流量控制”：

命令	作用
Ruijie(config-if)**# switchport port-security**	配置接口的端口安全。

下例显示如何打开 Gigabitethernet 2/1 的端口安全。。

```
Ruijie# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Ruijie(config)# interface gigabitethernet 2/1
Ruijie(config-if)# switchport port-security
Ruijie(config-if)# end
```

配置接口的速度，双工，流控请参照“配置接口的速度，双工，流控”。

下例显示如何配置 Gigabitethernet 2/1 为 access port，所属 VLAN 为 100，速度，双工，流控为自协商模式，端口安全打开：

```
Ruijie# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Ruijie(config)# interface gigabitethernet 2/1
Ruijie(config-if)# switchport access vlan 100
Ruijie(config-if)# speed auto
Ruijie(config-if)# duplex auto
Ruijie(config-if)# flowcontrol auto
Ruijie(config-if)# switchport port-security
Ruijie(config-if)# end
```

6.2.8.1.2 配置 Hybrid 端口

您可以通过以下步骤配置 Hybrid 端口：

命令	说明
configure terminal	进入配置模式

interface <*interface*>	进入接口配置模式 百兆,千兆，万兆
switchport mode hybrid	配置为端口为 hybrid 口
no switchport mode	删除端口模式
switchport hybrid native vlan *id*	设置hybrid口的默认VLAN
switchport hybrid allowed vlan [[**add**] [**tagged**	**untaged**]] |**remove**] **vlist**	设置端口的输出规则

```
ruijie# configure terminal
ruijie(config)# interface g 0/1
ruijie(config-if)# switchport mode hybrid
ruijie(config-if)# switchport hybrid native vlan 3
ruijie(config-if)# switchport hybrid allowed vlan untagged
20-30
ruijie(config-if)# end
ruijie# show running interface g 0/1
```

6.2.8.2 配置 L2 Aggregate Port

本节主要讲述如何创建 L2 Aggregate Port 及和 L2 Aggregate Port 相关的一些配置。

您可以在接口配置模式下使用 **aggregateport** 来创建 L2 Aggregate Port，具体的配置过程请参照“配置 Aggregate Port”。

6.2.8.3 清除接口的统计值并复位该接口

在特权模式下您可通过 **clear** 命令清除接口的统计值并复位该接口。该命令只对 Switch Port,L2 Aggregrate port 的成员端口,Routed port,L3 Aggregate port 的成员端口有效，以下为 clear 命令：

命令	作用
Ruijie**# clear counters** [*interface-id*]	清除接口统计值。
Ruijie**# clear interrface** *interface-id*	接口硬件复位

接口的统计值可以通过特权模式命令 **show interfaces** 查看，在特权模式下使用 **clear counters** 命令，可以将接口的统计值清零。如果不指定接口，则将所有的 L2 接口计数器清零。

下面的例子显示如何清除 Gigabitethernet 1/1 的计数器：

```
Ruijie# clear counters gigabitethernet 1/1
```

6.2.9 配置三层接口

配置三层接口：

命令	作用
Ruijie(config-if)**# no switchport**	将该接口 Shut Down 并且重新转换成三层模式。该命令只适用于 Switch Port 和 L2 Aggregrate port。
Ruijie(config-if)**# ip address** *ip_address subnet_mask*{[**secondary**|**tertiary**|**quartus**][**broadcast**]}	配置 IP 地址和子网掩码。

删除一个三层接口的 IP 地址可以使用接口配置模式的 **no ip address** 命令。

一个 L2 Aggregate Port 的成员口，不能进行 **no switchport** 操作。

下面的例子是描述如何将一个二层接口配置成 Routed Port，并且给该接口分配 IP 地址：

```
Ruijie# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Ruijie(config)# interface gigabitethernet 2/1
Ruijie(config-if)# no switchport
Ruijie(config-if)# ip address 192.20.135.21 255.255.255.0
Ruijie(config-if)# no shutdown
Ruijie(config-if)# end
```

6.2.9.1 配置 SVI

本节主要描述如何创建 SVI 及和 SVI 的一些相关配置。

您可在通过 **interface vlan** *vlan-id* 创建一个 SVI 或修改一个已经存在的 SVI。

SVI 的配置：

命令	作用
Ruijie(config)**# interface vlan** *vlan-id*	进入 SVI 接口配置模式。

然后可对 SVI 的相关属性进行配置，详细的信息请参考“配置 IP 单址路由”。

下面的例子显示如何进入接口配置模式，并且给 SVI 100 分配 IP 地址：

```
Ruijie# configure terminal
```

```
Enter configuration commands, one per line. End with CNTL/Z.
Ruijie(config)# interface vlan 100
Ruijie(config-if)# ip address 192.168.1.1 255.255.255.0
Ruijie(config-if)# end
```

6.2.9.2 配置 Routed port

本节描述如何创建 Routed Port 及和 Routed Port 的一些相关配置。

您可在接口模式下，进入某个接口后，使用 **no switchport** 来创建 Routed port。

创建一个 Routed port 并给该 Routed port 分配 IP 地址：

命令	作用
Ruijie(config-if)# **no switchport**	将该接口 Shut Down 并转换成三层模式
Ruijie(config-if)# **ip address** *ip_address subnet_mask*	配置 IP 地址和子网掩码。

注意：

当一个接口是 L2 Aggregate Port 的成员口时，是不能用 **switchport/ no switchport** 命令进行层次切换的。

下面的例子显示如何将一个二层接口配置成 Routed Port，并且给该接口分配 IP 地址：

```
Ruijie# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Ruijie(config)# interface fastethernet 0/6
Ruijie(config-if)# no switchport
Ruijie(config-if)# ip address 192.168.1.1 255.255.255.0
Ruijie(config-if)# no shutdown
Ruijie(config-if)# end
```

6.2.9.3 配置 L3 Aggregate Port

本节描述如何创建 L3 Aggregate Port 及和 L3 Aggregate Port 的一些相关配置。

您可在接口模式下使用 **no switchport** 将某个 L2 Aggregate Port 转化为 L3 Aggregate Port:

命令	作用

Ruijie(config-if)**# no switchport**	将该接口 Shut Down 并转换成三层模式。
Ruijie(config-if)**# ip address** *ip_address subnet_mask*	配置 IP 地址和子网掩码。

下面的例子显示如何创建 L3 Aggregate Port，并且给该接口分配 IP 地址：

```
Ruijie# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Ruijie(config)# interface aggregateport 2
Ruijie(config-if)# no switchport
Ruijie(config-if)# ip address 192.168.1.1 255.255.255.0
Ruijie(config-if)# no shutdown
Ruijie(config-if)# end
```

6.3 显示接口配置和状态

本节描述接口的显示内容，显示实例。您可在特权模式下通过 **show** 命令可来查看接口状态。在特权模式下您可使用以下命令显示接口状态：

命令	作用
Ruijie**# show interfaces** [*interface-id*]	显示指定接口的全部状态和配置信息。
Ruijie**# show interfaces** *interface-id* **status**	显示接口的状态。
Ruijie**# show interfaces** [*interface-id*] **switchport**	显示可交换接口（非路由接口）的 administrative 和 operational 状态信息。
Ruijie**# show interfaces** [*interface-id*] **description**	显示指定接口的描述配置和接口状态。
Ruijie**# show interfaces** [*interface-id*] **counters**	显示指定端口的统计值信息， 其中速率显示可能有 0.5%内的误差。

以下例子为显示接口 Gigabitethernet 1/1 的接口状态：

```
Ruijie# show interfaces gigabitethernet 1/1
GigabitEthernet   : Gi 1/1
Description    : user A
AdminStatus   : up
OperStatus    : down
Hardware     : 1000BASE-TX
Mtu         : 1500
PhysAddress   :
```

```
LastChange    : 0:0h:0m:0s
AdminDuplex   : Auto
OperDuplex    : Unknown
AdminSpeed    : 1000M
OperSpeed     : Unknown
FlowControlAdminStatus : Enabled
FlowControlOperStatus : Disabled
Priority    : 1
```

以下例子为显示接口 SVI 5 的接口状态和配置信息：

```
Ruijie# show interfaces vlan 5
VLAN   : V5
Description          : SVI 5
AdminStatus         : up
OperStatus          : down
Primary Internet address     :  192.168.65.230/24
Broadcast address         :  192.168.65.255
PhysAddress         : 00d0.f800.0001
LastChange          : 0:0h:0m:5s
```

以下例子为显示接口 Aggregate Port 3 的接口状态：

```
Ruijie# show interfaces aggregateport 3：
Interface   : AggreatePort 3
Description :
AdminStatus : up
OperStatus  : down
Hardware    : -
Mtu        : 1500
LastChange  : 0d:0h:0m:0s
AdminDuplex : Auto
OperDuplex  : Unknown
AdminSpeed  : Auto
OperSpeed   : Unknown
FlowControlAdminStatus : Autonego
FlowControlOperStatus  : Disabled
Priority    : 0
```

以下例子显示接口 GigabitEthernet 1/1 的接口配置信息：

```
Ruijie# show interfaces gigabitEthernet 1/1 switchport
Interface  Switchport Mode     Access    Native    Protected
VLAN lists
---------- ---------- --------- --------- --------- ---------
gigabitethernet 1/1       Enabled   Access       1          1
Enabled   All
```

以下例子为显示接口 Gigabitethernet 2/1 的接口描述：

```
Ruijie# show interfaces gigabitethernet 1/2 description
Interface          Status     Administrative    Description
------------  ---------      -----------    ---------------
gigabitethernet 2/1     down       down        Gi 2/1
```

以下例子为显示端口统计值

```
Ruijie# show interfaces gigabitethernet 1/2 counters
Interface : gigabitethernet 1/2
5 minute input rate  : 9144 bits/sec, 9 packets/sec
5 minute output rate : 1280 bits/sec, 1 packets/sec
InOctets           : 17310045
InUcastPkts         : 37488
InMulticastPkts     : 28139
InBroadcastPkts     : 32472
OutOctets          : 1282535
OutUcastPkts        : 17284
OutMulticastPkts    : 249
OutBroadcastPkts    : 336
Undersize packets   : 0
Oversize packets    : 0
collisions         : 0
Fragments          : 0
Jabbers            : 0
CRC alignment errors : 0
AlignmentErrors     : 0
FCSErrors          : 0
dropped packet events (due to lack of resources): 0
packets received of length (in octets):
64:46264, 65-127: 47427, 128-255: 3478,
256-511: 658, 512-1023: 18016, 1024-1518: 125
```

6.4 LinkTrap 策略配置

在设备中可以基于接口配置是否发送该接口的 LinkTrap，当功能打开时，如果接口发生 Link 状态变化，SNMP 将发出 LinkTrap,反之则不发。缺省情况下，该功能打开。

6.4.1 配置命令

命令	作用

Ruijie(config-if)# [**no**] **snmp trap link-status**	打开或者关闭发送该接口 link trap 的功能.

6.4.2 配置举例

下面配置将配置接口为不发送 Link trap:

```
Ruijie(config)# interface gigabitEthernet 1/1
Ruijie(config-if)# no snmp trap link-status
```

7 Aggregate Port 配置

本章描述如何在锐捷设备上配置 Aggregate Port。

7.1 概述

7.1.1 理解 Aggregate Port

我们可以把多个物理链接捆绑在一起形成一个逻辑链接，这个逻辑链接我们称之为 Aggregate Port（以下简称 AP）。锐捷设备所提供的 AP 功能符合 IEEE802.3ad 标准，它可以用于扩展链路带宽，提供更高的连接可靠性。

AP 功能支持流量平衡，可以把流量均匀地分配给各成员链路。AP 功能还实现了链路备份，当 AP 中的一条成员链路断开时，系统会将该成员链路的流量自动地分配到 AP 中的其它有效成员链路上去。AP 中一条成员链路收到的广播或者多播报文，将不会被转发到其它成员链路上。

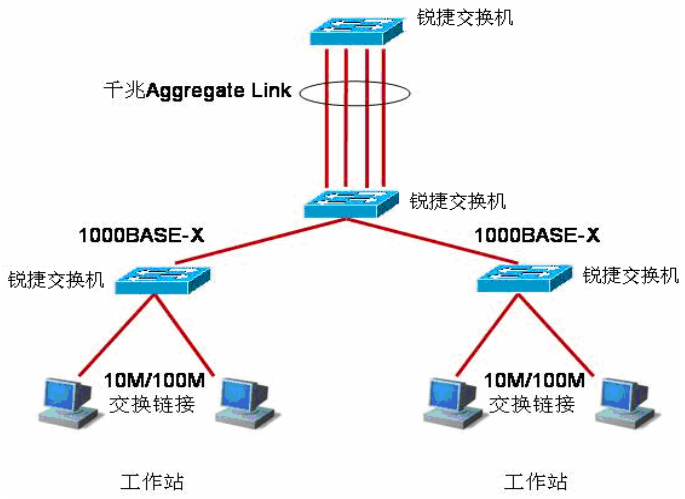


图 1 典型的 AP 配置

说明：

交换机 S3760 产品系列的每个 AP 口最多包含的成员口数量都为 8 个，该系列产品支持最大 AP 口数量为 31。

7.1.2 理解流量平衡

AP 可以根据报文的源 MAC 地址、目的 MAC 地址，源 MAC 地址+目的 MAC 地址、源 IP 地址，目的 IP 地址以及源 IP 地址+目的 IP 地址等特征值把流量平均地分配到 AP 的成员链路中。您可以用 **aggregateport load-balance** 设定流量分配方式。

源 MAC 地址流量平衡是根据报文的源 MAC 地址把报文分配到 AP 的各个成员链路中。不同源 MAC 的报文，一，根据源 MAC 地址在各成员链路间平衡分配，相同源 MAC 的报文，固定从同一个成员链路转发。

目的 MAC 地址流量平衡是根据报文的目的 MAC 地址把报文分配到 AP 的各个成员链路中。相同目的 MAC 的报文，固定从同一个成员链路转发，不同目的 MAC 的报文，根据目的 MAC 地址在各成员链路间平衡分配。

源 MAC+目的 MAC 地址流量平衡是根据报文的源 MAC 和目的 MAC 地址把报文分配到 AP 的各个成员链路中。具有不同的源 MAC+目的 MAC 地址的报文根据源 MAC+目的 MAC 地址在各成员链路间平衡分配，，而具有相同的源 MAC+目的 MAC 地址的报文则固定分配给同一个成员链路。

源 IP 地址或目的 IP 地址流量平衡是根据报文源 IP 或目的 IP 进行流量分配。不同源 IP 或目的 IP 的报文根据源 IP 或目的 IP 在各成员链路间平衡分配，相同源 IP 或目的 IP 的报文则固定通过相同的成员链路转发。该流量平衡方式用于三层报文，如果在此流量平衡模式下收到二层报文，则自动根据设备的默认方式进行流量平衡。

源 IP 地址+目的 IP 地址流量平衡是根据报文源 IP 和目的 IP 进行流量分配。该流量平衡方式用于三层报文，如果在此流量平衡模式下收到二层报文，则自动根据设备的默认方式进行流量平衡。具有不同的源 IP+目的 IP 地址的报文根据源 IP+目的 IP 地址在各成员链路间平衡分配，，具有相同的源 IP+目的 IP 地址的报文则固定分配给相同的成员链路。

以上所有平衡模式都适用于二层 AP 和三层 AP，即，源 IP 地址流量平衡、目的 IP 地址流量平衡、源 IP 地址+目的 IP 地址流量平衡模式也适用于二层 AP。

交换机 S3760 产品系列支持流量平衡模式如下表所示。“√”表示支持，“×”表示不支持。

表 1 S3760 系列交换机支持的流量平衡模式

流量平衡模式	S3760
基于源 MAC	×
基于目的 MAC	×
基于源 MAC+目的 MAC	√
基于目的 IP	×
基于源 IP	×
基于源 IP+目的 IP	√

我们应根据不同的网络环境设置合适的流量分配方式，以便能把流量较均匀地分配到各个链路上，充分利用网络的带宽。

在下图中，一个交换机通过 AP 与路由器进行通讯，所有内网中的设备（如图中的上面 4 台 PC 机）以路由器为网关，所有外网（如图中的下面 2 台 PC 机）经路由器发出的报文的源 MAC 都是网关的 MAC 地址，为了让路由器与其他主机之间的通讯流量能由其他链路来分担，应设置为根据目的 MAC 地址进行流量平衡；而在交换机处，则需要设置为根据源 MAC 地址进行流量平衡。

说明：

流量平衡模式设置为源 IP、目的 IP、源 IP＋目的 IP 时对 2 层报文采用设备的默认模式进行流量平衡。默认模式可在设备未配置 **aggregateport load-balance** 参数时通过 **show aggregateport load-balance** 命令获得。

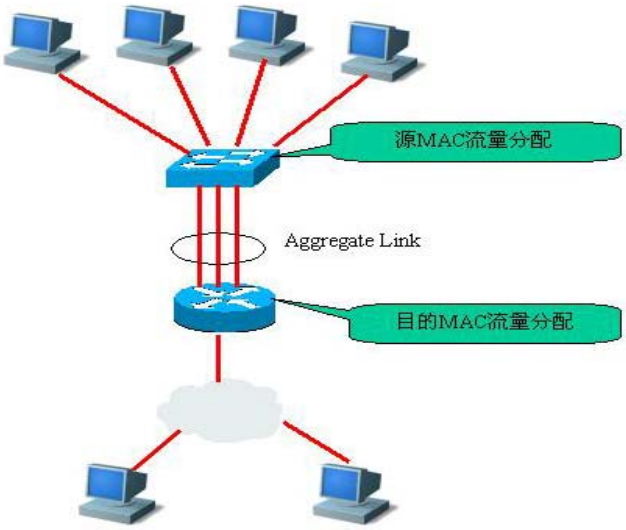


图 2　AP 流量平衡示意图

7.2　配置 Aggregate Port

7.2.1　缺省的 Aggregate Port 配置

AP 的缺省配置如下表所示：

属性	缺省值
二层 AP 接口	无
三层 AP 接口	无

流量平衡	根据输入报文的源 MAC 地址进行流量分配。

7.2.2 Aggregate Port 配置指导

- AP 成员端口的端口速率必须一致。
- 二层端口只能加入二层 AP，三层端口只能加入三层 AP；包含成员口的 AP 口不允许改变二层/三层属性。
- AP 不能设置端口安全功能。
- 一个端口加入 AP，端口的属性将被 AP 的属性所取代。
- 一个端口从 AP 中删除，则端口的属性将恢复为其加入 AP 前的属性。

注意：

当一个端口加入 AP 后，不能在该端口上进行任何配置，直到该端口退出 AP。

7.2.3 配置 Aggregate Port

在接口配置模式下，请按如下步骤将端口加入 AP：

命令	作用
Ruijie(config-if-range)# **port-group** *port-group-number*	将该接口加入一个 AP(如果这个 AP 不存在，则同时创建这个 AP)。

在接口配置模式下使用 **no port-group** 命令将一个物理端口退出 AP。

下面的例子是将二层的以太网接口 0/1 配置成二层 AP 5 成员：

```
Ruijie# configure terminal
Ruijie(config)# interface range gigabitEthernet 0/1
Ruijie(config-if-range)# port-group 5
Ruijie(config-if-range)# end
```

您可以在全局配置模式下使用命令 Ruijie(config)# **interface aggregateport** *n* (n 为 AP 号)来直接创建一个 AP(如果 AP n 不存在)。

注意：

将普通端口加入某个 AP 口后，当该端口再次从 ap 口退出时，普通端口上的原先相关的配置可能会恢复为缺省的配置。不同功能对 ap 口的成员的原有配置的处理方式有所不同，因此建议在端口从 ap 口退出后，应查看并确认端口的配置。

7.2.4 配置三层 Aggregate Port

缺省情况下，一个 Aggregate Port 是一个二层的 AP，如果您要配置一个三层 AP，您需要进行如下操作。

下面的例子是如何配置一个三层 AP 接口（AP 3），并且给它配置 IP 地址（192.168.1.1）：

```
Ruijie# configure terminal
Ruijie(config)# interface aggretegateport 3
Ruijie(config-if)# no switchport
Ruijie(config-if)# ip address 192.168.1.1 255.255.255.0
Ruijie(config-if)# end
```

注意：
通过 port-group 命令将三层接口加入一个 AP 时，如果该 AP 不存在，则不能加入成功，所以，您必须先创建该三层 AP 接口，再将各三层接口加入。

创建了一个三层的 AP 接口以后，您就可以往该 AP 口添加成员口了，如下例，将 fastEthernet 0/1-3 加入三层 AP 2：

```
Ruijie# configure terminal
Ruijie(config)# interface range fastEthernet 0/1-3
Ruijie(config-if)# no switchport
Ruijie(config-if)# port-group 2
```

7.2.5 配置 Aggregate Port 的流量平衡

在配置模式下，请按如下步骤配置 AP 的流量平衡算法：

命令	作用
Ruijie(config)# **aggregateport load-balance {src-dst-mac | src-dst-ip }**	设置 AP 的流量平衡，选择使用的算法： **src-dst-ip**：根据源 IP 与目的 IP 进行流量分配。 **src-dst-mac**：根据源 MAC 与目的 MAC 进行流量分配。

要将 AP 的流量平衡设置恢复到缺省值，可以在全局配置模式下使用：
no aggregateport load-balance 命令。

7.3 显示 Aggregate Port

在特权模式下，请按如下步骤显示 AP 设置。

命令	作用
Ruijie**# show aggregateport** [*port-number*]{**load-balance | summary**}	显示 AP 设置。

```
Ruijie# show aggregateport load-balance
Load-balance : Source MAC address
Ruijie#show aggregateport 1 summary
AggregatePort MaxPorts SwitchPort Mode   Ports
------------- -------- ---------- ------
Ag1           8        Enabled    ACCESS
```

8 链路聚合控制协议(LACP)配置

8.1 概述

IEEE 802.3ad 标准的 LACP(Link Aggregation Control Protocol, 链路聚合控制协议)是一个关于动态链路聚合的协议，它通过协议报文 LACPDU(Link Aggregation Control Protocol Data Unit, 链路聚合控制协议数据单元)和相连的设备交互信息。

当端口启用 LACP 协议后，端口通过发送 LACPDU 来通告自己的系统优先级，系统 MAC，端口的优先级，端口号和操作 key 等。相连设备收到该报文后，根据所存储的其他端口的信息，选择端口进行相应的聚合操作，从而可以使双方在端口退出或者加入聚合组上达到一致。

8.2 动态链路聚合的模式

LACP 的端口有两种模式：主动(Active)模式和被动模式(Passive)。

处于主动模式的端口会主动发起 LACP 报文协商。而处于被动模式的端口则只会对收到的 LACP 报文做应答。

处于主动模式的端口能够和处于被动模式或者主动模式的端口进行聚合，而处于被动模式的端口只能和处于主动模式的端口进行聚合。

8.3 LACP 端口的状态

聚合组内的成员有可能处于 3 种状态：

1. 当端口的链路状态处于 Down 时，端口不可能转发任何数据报文，显示为”down”状态。
2. 端口链路处于 Up 状态，并经过 LACP 协商后，端口被置于聚合状态(端口被作为一个聚合组的一个成员参与聚合组的数据报文转发)，显示为“bndl”状态。
3. 当端口链路处于 UP 状态，但是由于对端没有启用 LACP，或者因为端口属性和主端口不一致等一些因素导致经过报文协商端口被置于挂起状态（处于挂起状态的端口不参与聚合组的数据报文转发），显示为“sups”状态。

说明:

- 只有全双工的端口才能进行聚合
- 端口端口的速率、流控、介质类型以及端口二 三层属性必须一致才能聚合
- 端口聚合后修改端口的上述属性将导致同个聚合组内的其他端口也无法聚合

8.4 动态链路聚合的优先级关系

8.4.1 LACP 的系统 ID

每台设备仅能配置一个 LACP 聚合系统。每个 LACP 聚合系统都有唯一的系统优先级。系统 ID 由 LACP 的系统优先级和设备 MAC 地址组成。系统优先级越小，系统 ID 的优先级越高；在系统优先级相同的情况下，比较设备的 MAC 地址，设备 MAC 地址越小，系统 ID 的优先级越高。系统 ID 优先级较高的系统决定端口状态，低优先级系统的端口状态随高优先级系统的端口状态变化而变化。

8.4.2 LACP 的端口 ID

每个端口有独立的 LACP 端口优先级，这是一个可配置的数值。端口 ID 由 LACP 的端口优先级和端口号组成。端口优先级数值越小，端口 ID 的优先级越高；在端口优先级相同的情况下，端口号越小，端口 ID 的优先级越高。

8.4.3 LACP 的主端口

当有动态成员处于 up 状态时，LACP 会根据端口的速率，双工速率等关系，选择一个聚合组内端口 ID 优先级最高的端口作为主端口。只有和主端口属性相同的端口才能处于聚合状态，参与聚合组的数据转发。当端口的属性变化时，LACP 会在不解聚合的情况下，重新选择主端口；但是当新的主端口不处于聚合状态时，LACP 会把同一个聚合组内的成员解聚合，重新聚合。

8.4.4 LACP 的协商过程

在收到对端的 LACP 报文后，选取系统 ID 优先级比较高的系统。在系统 ID 优先级较高的一端，按照端口 ID 优先级从高到低的顺序，设置聚合组内端口的处于聚合状态（当聚合组内的端口数量超过设备的聚合组最大端口数量限制时，把超过聚合能力部分的端口置于挂起状态）。对端收到更新后的 LACP 报文后，也会把相应的端口设置成聚合状态。

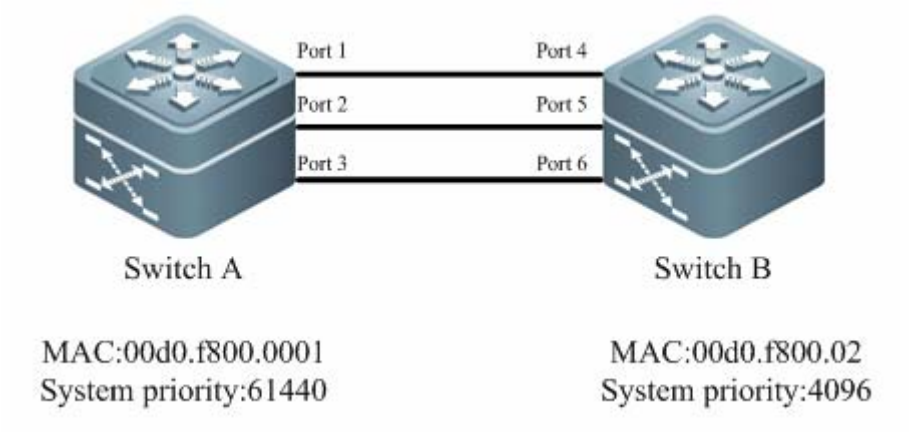


图 1 LACP 协商

例如：如上图 1 所示，交换机 A 和交换机 B 通过 3 个端口连接在一起。设置交换机 A 的系统优先级为 61440，设置交换机 B 的系统优先级为 4096。在交换机 A，B 的 3 个直连端口上打开 LACP 链路聚合，设置 3 个端口的聚合模式为主动模式，设置 3 个端口的端口优先级为默认优先级 32768。

在收到对端的 LACP 报文后，交换机 B 发现自己的系统 ID 优先级比较高(交换机 B 的系统优先级比交换机 A 高)，于是按照端口 ID 优先级的顺序(端口优先级相同的情况下，按照端口号从小到大的顺序)设置端口 4，5，6 处于聚合状态。交换机 A 收到交换机 B 更新后的 LACP 报文后，发现对端的系统 ID 优先级比较高，并且把端口设置成聚合状态了，也把端口 1，2，3 设置成聚合状态了。

8.5 动态链路聚合的要求

动态链路聚合是 LACP 协议自动地添加和删除聚合组内的端口，两个端口被自动地聚合在一起有一定的要求。

1. 只有相同的操作 key 才能被聚合在一起。
2. 只有和主端口具有相同的速率和双工等基本属性的端口才能被动态聚合在一起。
3. 端口链路处于 UP 状态，相连的端口启用 LACP，并且端口或者相连端口必须处于主动模式(Active)。

8.6 配置动态链路聚合(LACP)

你可以配置 LACP 的系统优先级，端口的优先级以及聚合组的管理 key。一台交换机所有的动态链路组只能有一个 LACP 系统优先级，修改这个值会影响到交换机上的所有聚合组。

在配置模式下,按如下步骤配置动态链路聚合：

命令	作用

Ruijie# **configure**	进入配置模式
Ruijie(config)# **lacp system-priority** *system-priority*	(可选)配置 LACP 系统的优先级，可选范围为 0-65535，默认优先级为 32768。
Ruijie(config)# **interface** *interface-id*	进入接口模式。
Ruijie(config-if)# **lacp port-priority** *port-priority*	(可选)配置端口的优先级,可选范围为 0-65535, 默认优先级为 32768。
Ruijie(config-if)# **port-group** *key* **mode active | passive**	把端口加入聚合组并指定端口的动态聚合模式，如果聚合组不存在，则会创建一个聚合组。 key 为聚合组的管理 key，key 取值范围根据不同产品支持的聚合组数量不同而变。 active|passive 端口在动态聚合组中的模式。
Ruijie(config-if)# **end**	退回特权模式。

8.6.1查看端口的动态链路聚合状态

在特权模式下使用如下命令来查看动态链路聚合的状态：

命令	作用
Ruijie# **show lacp summary**	查看 LACP 系统的动态链路聚合状态.

8.7 LACP 的 配置用例

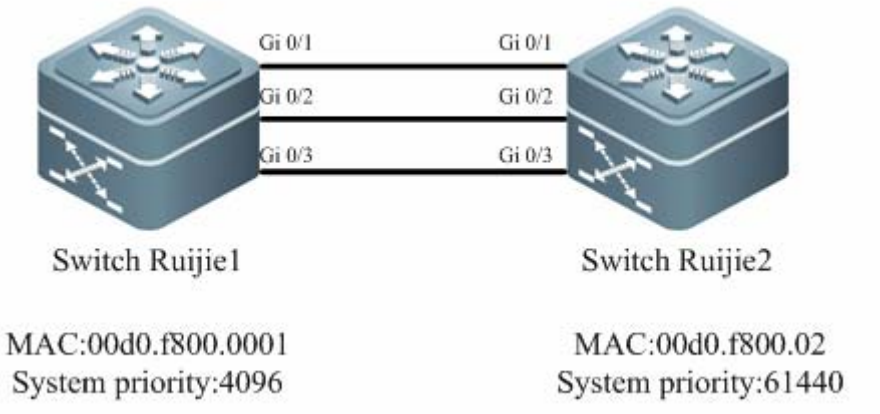


图 2 LACP 链路聚合

例如，如上图 2 所示拓扑，在交换机 Ruijie1 上设置 LACP 系统优先级为 4096,在端口 Gi 0/1、Gi 0/2、 Gi 0/3 上启用动态链路聚合协议，并设置端口的 LACP 端口优先级为 4096。

```
Ruijie1# configure terminal
Ruijie1(config)# lacp system-priority 4096
Ruijie1(config)# interface range GigabitEthernet 0/1-3
Ruijie1(config-if-range)# lacp port-priority 4096
Ruijie1(config-if-range)# port-group 3 mode active
Ruijie1(config-if-range)# end
```

在 Ruijie2 上设置 LACP 系统优先级为 61440，在端口 **Gi 0/1**、**Gi 0/2**、**Gi 0/3** 启用动态链路聚合协议，并设置端口的 LACP 端口优先级为 61440。

```
Ruijie2# configure terminal
Ruijie1(config)# lacp system-priority 61440
Ruijie2(config)# interface range GigabitEthernet 0/1-3
Ruijie2(config-if-range)# lacp port-priority 61440
Ruijie2(config-if-range)# port-group 3 mode active
Ruijie2(config-if-range)#end
```

配置完相关配置后，如果 LACP 协商成功，则会打印相应的 log：

```
*Feb 25 17:11:31: %LACP-5-BUNDLE: Interface Gi0/1 joined
AggregatePort 3.
*Feb 25 17:11:32: %LACP-5-BUNDLE: Interface Gi0/2 joined
AggregatePort 3.
*Feb 25 17:11:32: %LACP-5-BUNDLE: Interface Gi0/3 joined
AggregatePort 3.
*Feb 25 17:11:32: %LINEPROTO-5-UPDOWN: Line protocol on Interface
AggregatePort 3, changed state to up
```

表示端口 Gi 0/1、Gi 0/2、Gi 0/3 已经被成功地加入聚合组 3 了，这时在交换机 Ruijie1 上查看聚合组内成员口的状态。

```
Ruijie(config)#show LACP summary
Flags:  S - Device is sending Slow LACPDUs
        F - Device is sending fast LACPDUs.
A - Device is in active mode.       P - Device is in passive mode.

Aggregate port 3:

Local information:
                            LACP port      Oper    Port     Port
Port     Flags     State    Priority       Key     Number   State
-------------------------------------------------------------------
Gi0/1    SA        bndl       4096          0x3     0x1      0x3d
```

```
Gi0/2     SA       bndl       4096          0x3     0x2      0x3d
Gi0/3     SA       bndl       4096          0x3     0x3      0x3d


Partner information:
                  LACP port                 Oper    Port     Port
Port      Flags   Priority     Dev ID       Key     Number   State
--------------------------------------------------------------------
Gi0/1     SA       61440    00d0.f800.0002   0x3     0x1      0x3d
Gi0/2     SA       61440    00d0.f800.0002   0x3     0x2      0x3d
Gi0/3     SA       61440    00d0.f800.0002   0x3     0x3      0x3d
```

其中，

“Local information”部分，显示的是本系统维护的端口的 LACP 信息。

“Port” 显示的是本系统内端口的 ID。

“Flags” 显示的是端口的一些状态标志：‘S’标志着 LACP 状态已经稳定了，处于定时发送 LACPPDU 的状态；’A’标志着端口是出于主动模式。

“State” 显示的是端口状态：“bndl”标志着端口已经处于聚合状态了；其它状态说明参见前述“LACP 端口的状态”一节。

“LACP Port Priority” 显示的是端口的 LACP 优先级信息。

“Oper Key” 显示的是端口的操作 Key。

“Port Number” 显示的是端口的端口号。

“Port State” 显示的是端口的 LACP 协议状态。

“Partner infomation”部分，显示的是相连端口的 LACP 信息。

“Dev ID”部分显示的是端口相邻端口的系统 MAC 信息。

9 VLAN 配置

本章描述如何配置 IEEE802.1q VLAN

9.1 概述

VLAN 是虚拟局域网（Virtual Local Area Network）的简称，它是在一个物理网络上划分出来的逻辑网络。这个网络对应于 ISO 模型的第二层网络。VLAN 的划分不受网络端口的实际物理位置的限制。VLAN 有着和普通物理网络同样的属性，除了没有物理位置的限制，它和普通局域网一样。第二层的单播、广播和多播帧在一个 VLAN 内转发、扩散，而不会直接进入其他的 VLAN 之中。所以，如果一个端口所连接的主机想要和其它不在同一个 VLAN 的主机通讯，则必须通过一个三层设备，见下图。

可以把一个端口定义为一个 VLAN 的成员，所有连接到这个特定端口的终端都是虚拟网络的一部分，并且整个网络可以支持多个 VLAN。当在 VLAN 中增加、删除和修改用户的时候，不必从物理上调整网络配置。

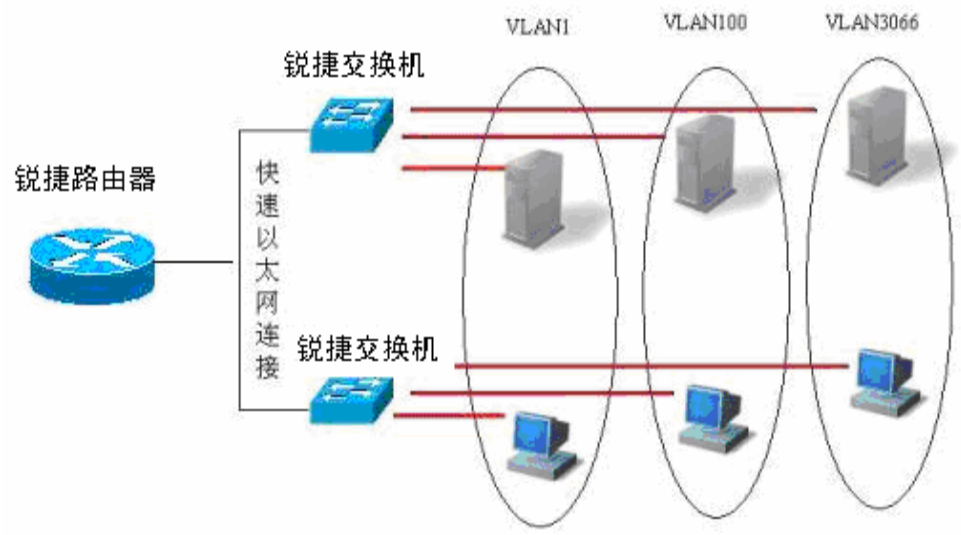


图 1

和一个物理网络一样，VLAN 通常和一个 IP 子网联系在一起。一个典型的例子是，所有在同一个 IP 子网中的主机属于同一个 VLAN，VLAN 之间的通讯必须通过三层设备（三层设备）。锐捷的三层设备可以通过 SVI 接口（Switch Virtual Interfaces）来进行 VLAN 之间的 IP 路由。关于 SVI 的配置，请见接口管理配置及 IP 单播路由配置。

9.1.1 支持的 VLAN

产品支持的 VLAN 遵循 IEEE802.1Q 标准，最多支持 4094 个 VLAN(VLAN ID 1-4094)其中 VLAN 1 是不可删除的默认 VLAN。

注意:

S3760 系列设备可支持配置 4094 个 Vlan

9.1.2 VLAN 成员类型

可以通过配置一个端口的 VLAN 成员类型，来确定这个端口能通过怎样的帧，以及这个端口可以属于多少个 VLAN。关于 VLAN 成员类型的详细说明，请看下表：

VLAN 成员类型	VLAN 端口特征
Access	一个 Access 端口，只能属于一个 VLAN，并且是通过手工设置指定 VLAN 的。
Trunk（802.1Q）	一个 Trunk 口，在缺省情况下是属于本设备所有 VLAN 的，它能够转发所有 VLAN 的帧。也可以通过设置许可 VLAN 列表(Allowed-VLANs)来加以限制。

9.2 配置 VLAN

一个 VLAN 是以 VLAN ID 来标识的。在设备中，您可以添加、删除、修改 VLAN 2-4094，而 VLAN 1 是由设备自动创建，并且不可被删除。

可以使用接口配置模式来配置一个端口的 VLAN 成员类型或加入、移出一个 VLAN。

9.2.1 VLAN 配置信息的保存

当您在特权命令模式下输入 **copy running-config startup-config** 命令后，VLAN 的配置信息便被保存进配置文件。要查看 VLAN 配置信息，可以使用 **show vlan** 命令。

9.2.2 缺省的 VLAN 配置

参数	缺省值	范围
VLAN ID	1	1－4094
VLAN Name	VLAN xxxx，xxxx 是 VLAN ID 数	无范围
VLAN State	Active	Active，Inactive

9.2.3 创建、修改一个 VLAN

在特权模式下，您可以创建或者修改一个 VLAN：

命令	作用
Ruijie(config)# **vlan** *vlan-id*	输入一个 VLAN ID。如果输入的是一个新的 VLAN ID，则设备会创建一个 VLAN，如果输入的是已经存在的 VLAN ID，则修改相应的 VLAN。
Ruijie(config)# **name** *vlan-name*	（可选）为 VLAN 取一个名字。如果没有进行这一步，则设备会自动为它起一个名字 VLAN xxxx，其中 xxxx 是用 0 开头的四位 VLAN ID 号。比如，VLAN 0004 就是 VLAN 4 的缺省名字。

如果您想把 VLAN 的名字改回缺省名字，只需输入 **no name** 命令即可。

下面是一个创建 VLAN 888，将它命名为 Test888，并且保存进配置文件的例子：

```
Ruijie# configure terminal
Ruijie(config)# vlan 888
Ruijie(config-vlan)# name test888
Ruijie(config-vlan)# end
```

9.2.4 删除一个 VLAN

您不能删除缺省 VLAN（VLAN 1）。

在特权模式下删除一个 VLAN：

命令	作用
Ruijie(config)# **no vlan** *vlan-id*	输入一个 VLAN ID，删除它。

9.2.5 向 VLAN 分配 Access 口

如果您把一个接口分配给一个不存在的 VLAN，那么这个 VLAN 将自动被创建。

在特权模式下，将一个端口分配给一个 VLAN：

命令	作用
Ruijie(config-if)# **switchport mode access**	定义该接口的 VLAN 成员类型（二层 ACCESS 口）
Ruijie(config-if)# **switchport access vlan** *vlan-id*	将这个口分配给一个 VLAN

下面这个例子把 fastEthernet 0/10 作为 Access 口加入了 VLAN20：

```
Ruijie# configure terminal
Ruijie(config)# interface fastethernet 1/10
Ruijie(config-if)# switchport mode access
Ruijie(config-if)# switchport access vlan 20
Ruijie(config-if)# end
```

下面这个例子显示了如何检查配置是否正确：

```
Ruijie(config)# show interfaces fastEthernet 0/1 switchport
Switchport is enabled
Mode is access port
Acsess vlan is 1,Native vlan is 1
Protected is disabled
Vlan lists is ALL
```

9.3 配置 VLAN Trunks

9.3.1 Trunking 概述

一个 Trunk 是将一个或多个以太网交换接口和其他的网络设备（如路由器或交换机）进行连接的点对点链路，一条 Trunk 链路可以传输属于多个 VLAN 的流量。

锐捷设备的 Trunk 采用 802.1Q 标准封装。下图显示了一个采用 Trunk 连接的网络。

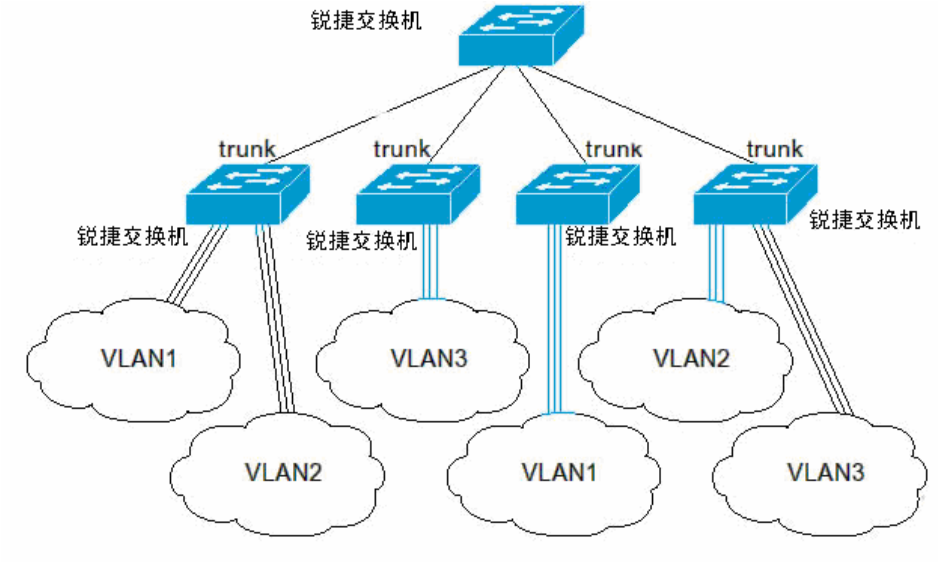


图 2

您可以把一个普通的以太网端口，或者一个 Aggregate Port 设为一个 Trunk 口（关于 Aggregate Port 的详细说明，请见配置 Aggregate Port）。

如果要把一个接口在 ACCESS 模式和 TRUNK 模式之间切换，请用 **switchport mode** 命令：

命令	作用
Ruijie(config-if)# **switchport mode access**	将一个接口设置成为 Access 模式
Ruijie(config-if)# **switchport mode trunk**	将一个接口设置成为 Trunk 模式

必须为 Trunk 口定义一个 Native VLAN。所谓 Native VLAN，就是指在这个接口上收发的 UNTAG 报文，都被认为是属于这个 VLAN 的。显然，这个接口的缺省 VLAN ID（即 IEEE 802.1Q 中的 PVID）就是 Native VLAN 的 VLAN ID。同时，在 Trunk 上发送属于 Native VLAN 的帧，则必然采用 UNTAG 的方式。每个 Trunk 口的缺省 Native VLAN 是 VLAN 1。

在配置 Trunk 链路时，请确认连接链路两端的 Trunk 口属于相同的 Native VLAN。

9.3.2 配置一个 Trunk 口

9.3.2.1 Trunk 口基本配置

在特权模式下，可以将一个接口配置成一个 Trunk 口。

命令	作用
Ruijie(config-if)#**switchport mode trunk**	定义该接口的类型为二层 Trunk 口

Ruijie(config-if)# **switchport trunk native vlan** *vlan-id*	为这个口指定一个 Native VLAN

如果想把一个 Trunk 口的所有 Trunk 相关属性都复位成缺省值，请使用 **no switchport mode** 接口配置命令。

9.3.3 定义 Trunk 口的许可 VLAN 列表

一个 Trunk 口缺省可以传输本设备支持的所有 VLAN（1－4094）的流量。但是，您也可以通过设置 Trunk 口的许可 VLAN 列表来限制某些 VLAN 的流量不能通过这个 Trunk 口。

在特权模式下，可以修改一个 Trunk 口的许可 VLAN 列表。

命令	作用
Ruijie(config-if)# **switchport trunk allowed vlan** {**all** | [**add**|**remove**| **xcept**] } *vlan-list*	（可选）配置这个 Trunk 口的许可 VLAN 列表。参数 *vlan-list* 可以是一个 VLAN，也可以是一系列 VLAN，以小的 VLAN ID 开头，以大的 VLAN ID 结尾，中间用-号连接。如：10–20。 **all** 的含义是许可 VLAN 列表包含所有支持的 VLAN； **add** 表示将指定 VLAN 列表加入许可 VLAN 列表； **remove** 表示将指定 VLAN 列表从许可 VLAN 列表中删除； **except** 表示将除列出的 VLAN 列表外的所有 VLAN 加入许可 VLAN 列表；

如果想把 Trunk 的许可 VLAN 列表改为缺省的许可所有 VLAN 的状态，请使用 **no switchport trunk allowed vlan** 接口配置命令。

下面是一个把 VLAN 2 从端口 0/15 中移出的例子：

```
Ruijie(config)# interface fastethernet 0/15
Ruijie(config-if)# switchport trunk allowed vlan remove 2
Ruijie(config-if)# end
Ruijie# show interfaces fastethernet 0/15 switchport
Interface Switchport Mode  Access Native Protected VLAN lists
--------- ---------- ----  ------ ------ -------- ----------
Fa0/15    enabled    TRUNK  1     1      Disabled   1,3－4094
```

9.3.4 配置 Native VLAN

一个 Trunk 口能够收发 TAG 或者 UNTAG 的 802.1Q 帧。其中 UNTAG 帧用来传输 Native VLAN 的流量。缺省的 Native VLAN 是 VLAN 1。

在特权模式下，可以为一个 Trunk 口配置 Native VLAN。

命令	作用
Ruijie(config-if)# **switchport trunk native vlan** *vlan-id*	配置 Native VLAN

如果想把 Trunk 的 Native VLAN 列表改回缺省的 VLAN 1，请使用 **no switchport trunk native vlan** 接口配置命令。

如果一个帧带有 Native VLAN 的 VLAN ID，在通过这个 Trunk 口转发时，会自动被剥去 TAG。

当把一个接口的 Native VLAN 设置为一个不存在的 VLAN 时，设备不会自动创建此 VLAN。此外，一个接口的 Native VLAN 可以不在接口的许可 VLAN 列表中。此时，Native VLAN 的流量不能通过该接口。

9.4 显示 VLAN

在特权模式下，才可以查看 VLAN 的信息。显示的信息包括 VLAN VID、VLAN 状态、VLAN 成员端口以及 VLAN 配置信息。以下罗列了相关的显示命令：

命令	作用
show vlan [**id** *vlan-id*]	显示所有或指定 VLAN 的参数

下面是一个显示 VLAN 的例子：

```
Ruijie#show vlan
VLAN  Name      Status             Ports
---- --------  --------   --------------------------------
   1 VLAN0001   STATIC    Gi0/1, Gi0/5, Gi0/6, Gi0/7
                          Gi0/8, Gi0/9, Gi0/10, Gi0/11
                          Gi0/12, Gi0/13, Gi0/14, Gi0/15
                          Gi0/16, Gi0/17, Gi0/18, Gi0/19
                          Gi0/20, Gi0/21, Gi0/22, Gi0/23
                          Gi0/24
  10 VLAN0010  STATIC    Gi0/2, Gi0/3
  20 VLAN0020  STATIC    Gi0/2, Gi0/3, Gi0/4
30 VLAN0030  STATIC    Gi0/3, Gi0/4

Ruijie#show vlan id 20
VLAN  Name              Status    Ports
----  --------          ------    -----------------------
  20  VLAN0020          STATIC    Gi0/2, Gi0/3, Gi0/4
```

10 Super VLAN 配置

本章描述锐捷设备的 Super VLAN 配置

10.1 概述

Super VLAN 是 VLAN 划分的一种方式。Super VLAN 又称为 VLAN 聚合，是一种专门优化 IP 地址的管理技术。其原理是将一个网段的 IP 分给不同的子 VLAN(Sub VLAN)，这些 Sub VLAN 同属于一个 Super VLAN。而每一个 Sub VLAN 都是独立的广播域，不同 Sub VLAN 之间二层相互隔离。当 Sub VLAN 内的用户需要进行三层通信时，将使用 Super VLAN 的虚接口的 IP 地址作为网关地址，这样多个 VLAN 共享一个 IP 地址，从而节省了 IP 地址资源。同时，为了实现不同 Sub VLAN 间的三层互通及 Sub VLAN 与其他网络的互通，需要利用 ARP 代理功能。通过 ARP 代理可以进行 ARP 请求和响应报文的转发与处理，从而实现了二层隔离端口间的三层互通。缺省状态下，Super VLAN 和 Sub VLAN 的 ARP 代理功能是打开的。

采用 Super VLAN 技术可以极大的节省 IP 地址，它只需对包含多个 Sub VLAN 的 Super VLAN 分配一个 IP 地址，既节省地址又方便网络管理。

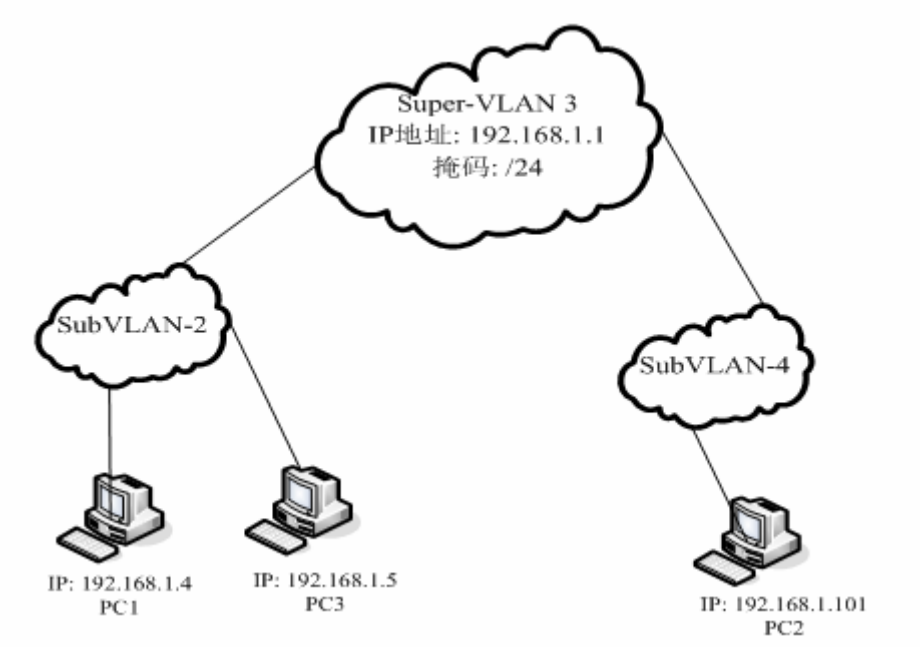


图 1

下面描述一下 VLAN 聚合情况下，两个聚合的 Sub-VLAN 之间通信的过程，如上图所示：

Sub VLAN2 和 Sub VLAN4 聚合成 Super VLAN3，为 Super VLAN3 分配一个 IP 子网， Sub VLAN2 和 Sub VLAN4 都位于此子网。假设 Sub VLAN2 中的一台主

机 PC1 要与子网中的另一台主机 PC2 通信，PC1 发现对方与自己处于同一网段，则直接发出目的 IP 的 ARP 请求报文。那么三层设备收到该 ARP 请求报文后，将其在 Sub VLAN2 的范围内直接通过二层广播此报文，并送一份给设备本身的 ARP 模块，设备的 ARP 模块首先看 ARP 请求报文中的目的 IP 地址是不是在 Sub-VLAN2 中，如果是就丢弃该报文，因为表明它和 PC1 在同一个广播域里面，那么目的主机将直接应答给 PC1；如果不是就将 SuperVLAN3 的 MAC 地址应答给 PC1，完成 ARP 的代理工作。比如 PC1 和 PC2 的通信就需要通过 ARP 代理，由设备转发 PC1 发给 PC2 的数据包； 而 PC1 和 PC3 的通信则直接进行无需通过设备转发。

限制：

- Super VLAN 不能包含任何成员口，只能包含 Sub VLAN，由 Sub VLAN 包含实际的物理接口。
- Super VLAN 不能做为其它 Super VLAN 的 Sub VLAN。
- Super VLAN 不能当正常的 1Q vlan 来使用。
- VLan 1 不能作为 SuperVLAN。
- Sub VLAN 不能配置为网络接口，不能配置 IP 地址。
- Super VLAN 不能使用 VRRP，不支持 IGMP Snooping、PIM Snooping。
- 基于 Super VLAN 接口的 ACL 和 QOS 配置不对 Sub VLAN 生效。

10.2 配置 Super VLAN

可以使用下面的命令来设置 Super VLAN。

命令	作用
Ruijie# **configure**	进入全局配置模式
Ruijie(config)# **vlan** *vlan-id*	进入 VLAN 配置模式
Ruijie(config-vlan)# **supervlan**	打开 SuperVLAN 的功能
Ruijie(config-vlan)# **end**	回到特权模式

缺省情况下，Super VLAN 功能是关闭的，使用 **no supervlan** 可以关闭已经打开得 supervlan 的功能。

10.3 配置 Super VLAN 的 Sub VLAN

必须为 SuperVLAN 配置 SubVLAN，该 SuperVLAN 才有意义。

可以使用下面的命令来使一个 VLAN 属于 Super VLAN 的 SubVLAN。

注意:

可能由于资源不足导致设置 SubVLAN 失败

命令	作用
Ruijie# **configure**	进入配置模式
Ruijie(config)# **vlan** *vlan-id*	进入 VLAN 配置模式
Ruijie(config-vlan)# **supervlan**	设置该 vlan 为 SuperVLAN
Ruijie(config-vlan)# **subvlan** *vlan-id-list*	指定若干个 sub vlan 并把它们加入 supervlan 中。
Ruijie(config-vlan)# **exit**	退出到全局模式

使用 **no subvlan** [*vlan-id-list*] 命令删除 SuperVLAN 的 SubVLAN。

注意:

不能使用 **no vlan** 命令删除 SubVLAN，必须先转成普通 VLAN 才能删除。

10.4 设置 Sub VLAN 的地址范围

用户可以为每个 SubVLAN 配置其地址空间范围，以便设备区分给定的 IP 地址属于哪个 SubVLAN. 同一个 SuperVLAN 下的 SubVLAN 配置的地址空间范围不可以有交叉重叠或者互相包含的关系.

请在全局模式下进行下列配置。

命令	作用
Ruijie# **configure**	进入配置模式
Ruijie(config)# **vlan** *vlan-id*	进入 vlan 配置模式
Ruijie(config-vlan)# **subvlan-address-range** *start-ip end-ip*	设置子 VLAN 的地址范围，start-ip 为该 SubVLAN 的 IP 起始地址 end-ip 为该 SubVLAN 的 IP 结束地址
Ruijie(config-vlan)# **end**	回到特权模式
Ruijie# **show run**	验证前面各步骤的配置

注意:

用户可以通过执行 **no subvlan-address-range** 删除之前配置

10.5 设置 Super VLAN 的虚拟接口

当 Sub VLAN 内的用户需要进行三层通信时,首先要创建 SuperVLAN 对应的虚拟三层接口进行。

使用 SuperVLAN 自身对应的 SVI 作为虚拟接口

请在全局模式下进行下列配置。

命令	作用
Ruijie**# configure**	进入配置模式
Ruijie(config)**# interface vlan** *vlan-id*	进入 SVI 模式
Ruijie(config-vlan)**# ip address** *ip mask*	设置虚拟接口的 IP 地址
Ruijie(config-vlan)**# end**	回到特权模式
Ruijie**# show run**	验证前面各步骤的配置

10.6 设置 VLAN 的代理 ARP 功能

可以使用下面的命令来设置 VLAN 的代理 ARP 功能，从而允许 SubVLAN 之间互相通信。缺省情况下该功能是打开的。

请在全局模式下进行下列配置。

命令	作用
Ruijie**# configure**	进入配置模式
Ruijie(config)**# vlan** *vlan-id*	进入 VLAN 模式
Ruijie(config-vlan)**# proxy-arp**	打开 VLAN 的 ARP 代理功能
Ruijie(config-vlan)**# end**	回到特权模式
Ruijie**# show run**	验证前面各步骤的配置

使用 **no proxy-arp** 关闭 Vlan 的代理 ARP 功能。

10.7 显示 supervlan 设置

可以使用下面的命令来显示设置 SuperVLAN 的配置

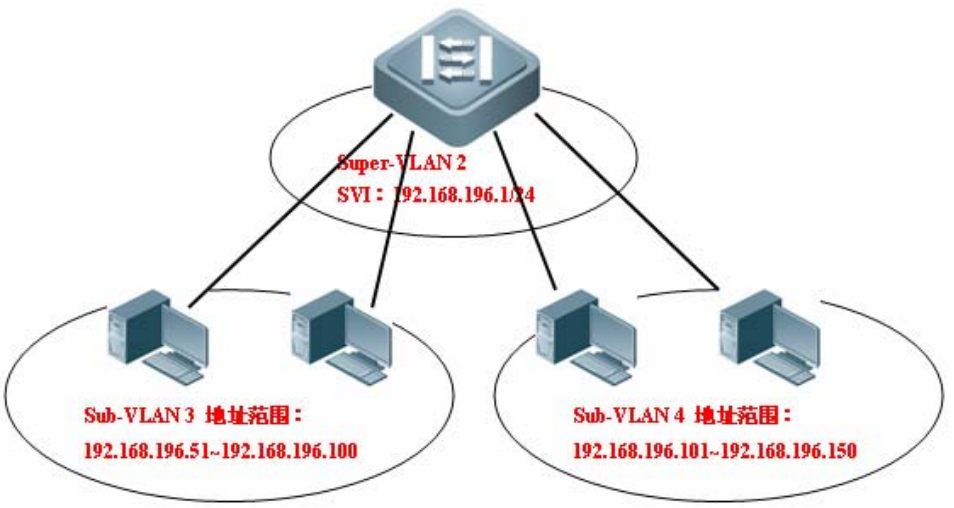
命令	作用
Ruijie**# show supervlan**	显示 supervlan 配置

10.8 Super VLAN 配置用例

10.8.1 组网需求

- 创建一个 Super-VLAN 为 VLAN 2，并配置与它关联的 Sub-VLAN 为 VLAN 3 和 VLAN ；
- Sub-VLAN 3 的地址范围为 192.168.196.51~192.168.196.100，Sub-VLAN 4 的地址范围为 192.168.196.101~192.168.196.150；
- 设置 Super-VLAN 的虚拟接口的 IP 地址为 192.168.196.1，子网掩码为 255.255.255.0；
- 设置 VLAN 的 ARP 代理功能。

10.8.2 组网拓扑



10.8.3 配置步骤

进入交换机的配置模式

```
Ruijie# configure terminal
```

创建 VLAN 2，并进入该 VLAN 的配置模式

```
Ruijie(config)# vlan 2
```

设置 VLAN 2 为 Super VLAN

```
Ruijie(config-vlan)# supervlan
```

退出到全局模式

```
Ruijie(config-vlan)# exit
```

创建 VLAN 3

```
Ruijie(config)# vlan 3
```

退出到全局模式

```
Ruijie(config-vlan)# exit
```

创建 VLAN 4

```
Ruijie(config)# vlan 4
```

退出到全局模式

```
Ruijie(config-vlan)# exit
```

进入 VLAN 2 的配置模式，并设置 VLAN 3 和 VLAN 4 为 Super VLAN 2 的 Sub-VLAN

```
Ruijie(config)# vlan 2
Ruijie(config-vlan)# subvlan 3,4
```

退出到全局模式，进入 VLAN 3 的配置模式，配置 VLAN 3 的地址范围为了 192.168.196.51~192.168.196.100

```
Ruijie(config-vlan)# exit
Ruijie(config)# vlan 3
Ruijie(config-vlan)# subvlan-address-range 192.168.196.51
192.168.196.100
```

退出到全局模式，进入 VLAN 4 的配置模式，配置 VLAN 4 的地址范围为了 192.168.196.101~192.168.196.150

```
Ruijie(config-vlan)# exit
Ruijie(config)# vlan 4
Ruijie(config-vlan)# subvlan-address-range 192.168.196.101
192.168.196.150
```

退出到全局模式

```
Ruijie(config-vlan)# exit
```

进入 SVI 模式

```
Ruijie(config)# interface vlan 2
```

为 VLAN 2 虚拟接口配置 IP 地址为 192.168.96.1，子网掩码为 255.255.255.0

```
Ruijie(config-if)# ip address 192.168.196.1 255.255.255.0
```

退出到全局模式，进入 VLAN 2 的配置模式，开启 ARP 代理功能（默认是开启的）

```
Ruijie(config-if)# exit
Ruijie(config)# vlan 2
Ruijie(config-vlan)# proxy-arp
```

退出到特权模式

```
Ruijie(config-vlan)# end
Ruijie# show supervlan
supervlan id   supervlan arp-proxy   subvlan id   subvlan
arp-proxy  subvlan ip range
--------------- ------------- -------------  ------------
2              ON              3       ON  192.168.196.51 -
192.168.196.100
                               4      ON  192.168.196.101 -
192.168.196.150
```

11 Protocol VLAN 配置

11.1 Protocol VLAN 技术

设备端口接收到的报文，都需要进行 VLAN 分类，使报文属于唯一的一个 VLAN，有以下三种可能：

1) 如果报文是空 VLAN ID 报文（UNTAG 或 Priority 报文），而设备仅支持基于端口的 VLAN 分类的话，报文所添加 TAG 的 VLAN ID 将是输入端口的 PVID。

2) 如果报文是空 VLAN ID 报文（UNTAG 或 Priority 报文），而设备支持基于报文协议类型的 VLAN 分类的话，报文所添加 TAG 的 VLAN ID 将会从输入端口上的协议组配置相对应的 VLAN ID 集中选取，而如果报文的协议类型与输入端口上的所有协议组配置都不相符的话，将按照基于端口的 VLAN 分类来分配 VLAN ID。

3) 如果报文是 TAG 报文，其所属 VLAN 分类由 TAG 中的 VLAN ID 决定。

Protocol VLAN 技术就是基于报文协议类型的 VLAN 分类技术，其可以将某一协议类型的空 VLAN ID 报文都划分到同一个 VLAN。

Protocol VLAN 配置只对 Trunk 口和 Hybrid 口生效，对于 Access 口没有作用。

锐捷产品支持全局的基于 IP 地址的 VLAN 分类和端口上的基于报文类型和以太网类型的 VLAN 分类两种 VLAN 分类技术。

基于 IP 地址的 VLAN 分类是全局配置，如果您配置了基于 IP 地址的 VLAN 分类的话，其将应用到所有的 Trunk 口和 Hybrid 口上。

1) 如果输入报文为空 VLAN ID 报文,且输入报文的 IP 地址匹配您配置的 IP 地址的话，该报文将被划分到您配置的 VLAN 内。

2) 如果输入报文为空 VLAN ID 报文,且输入报文的报文类型和以太网类型匹配您配置在输入端口上的报文类型和以太网类型的话，该报文将被划分到您配置的 VLAN 内。

基于 IP 地址的 VLAN 分类优先级高于基于报文类型和以太网类型的 VLAN 分类，所以如果您同时配置了基于 IP 地址以及基于报文类型和以太网类型的 VLAN 分类，且输入报文同时符合两者的话，将是基于 IP 地址的 VLAN 分配起作用。

您最好在配置好 VLAN、端口的 Trunk、Hybrid 、Access 和 AP 属性后，再配置 Protocol VLAN，如果您在 Trunk 或 Hybrid 口上配置了 Protocol VLAN，那么您需要报文 Trunk 和 Hybrid 口的许可 VLAN 列表包含 Protocol VLAN 相关的所有 VLAN。

11.2 配置 Protocol VLAN

11.2.1 缺省 Protocol VLAN

缺省情况下，没有 Protocol VLAN 的配置。

11.2.2 配置报文类型和以太网类型的 profile

按如下方式配置报文类型和以太网类型

命令	说明
configure terminal	进入配置模式
protocol-vlan profile *id* **frame-type** [*type*] **ether-type** [*type*]	配置报文类型和以太网类型的 profile
no protocol-vlan profile *id*	删除某 profile 配置
no protocol-vlan profile	清除所有的 profile 配置
end	退出 VLAN 模式
show protocol-vlan profile	显示所有的 profile 配置
show protocol-vlan profile *id*	显示某个 profile 的配置

举例如下

```
Ruijie# configure terminal
Ruijie(config)# protocol-vlan  profile 1 frame-type  ETHERII
ether-type EHTER_AARP
Ruijie(config)# protocol-vlan  profile 2 frame-type  SNAP
ether-type 0x809b
Ruijie(config-vlan)# end
Ruijie# show protocol-vlan profile
profile         frame-type  ether-type       Interfaces|vid
-------     ---------   ----------        -----------
1           ETHERII     EHTER_AARP       NULL|NULL
2           SNAP         ETHER_APPLETALK  NULL|NULL
```

说明：

1）、Profile 应用到端口上之后，配置才能生效；

2）、在更新某个 Profile 配置时，必须先删除该 Profile，再重新配置该 Profile；

3)、不同产品支持的 Profile 个数并不相同，S3760 支持 11 个 profile；

4)、S3760 系列交换机不支持配置协议为 0x0806（ARP）的以太网类型。

11.2.3 应用 profile

您可以通过下面的设置步骤来完成:

命令	说明
configure terminal	进入配置模式
interface [*接口 ID*]	进入接口模式
protocol-vlan profile *id* vlan *vid*	应用某 profile 到该接口
no protocol-vlan profile	清除该端口上的所有 profile
no protocol-vlan profile *id*	清除该端口上的某个 profile
end	退出接口模式

下面例子将 profile 1 和 profile 2 应用到插槽 3 的 GE 口 1,VLAN 分类为 VLAN 101 和 102:

```
Ruijie# configure terminal
Ruijie(config)# interface gi 3/1
Ruijie(config-if)# protocol-vlan profile 1 vlan 101
Ruijie(config-if)# protocol-vlan profile 2 vlan 102
Ruijie(config-if)# end
Ruijie# show protocol-vlan profile
profile          frame-type  ether-type      Interfaces|vid
-------      ---------   ----------      --------------
1            ETHERII     EHTER_AARP      gi3/1|101
2            SNAP         ETHER_APPLETALK  gi3/1|102
```

说明:

1)、每个接口上都可以应用所有的 profile；

2)、相同的 profile 在不同的接口上可以指定不同的 vid；

3)、根据不同系列产品可以指定的 VID 数目不同，S3760 系列设备可以指定 4094 个 VLAN 。

11.3 Protocol VLAN 的显示

您可以通过以下步骤显示 Protocol VLAN 内容

命令	说明
show protocol-vlan	显示 Protocol VLAN 的内容

```
Ruijie# show protocol-vlan
ip              mask            vlan
-------------   -------------   ----
192.168.100.3   255.255.255.0   100
profile          frame-type  ether-type     Interfaces|vid
-------     ---------   ----------     --------------
1           ETHERII     EHTER_AARP     gi3/1|101
2           SNAP         ETHER_APPLETALK  gi3/1|1
```

12 Private VLAN 配置

12.1 Private VLAN 技术

服务提供商如果给每个用户一个 VLAN，则由于一台设备支持的 VLAN 数最大只有 4096 而限制了服务提供商能支持的用户数；在三层设备上，每个 VLAN 被分配一个子网地址或一系列地址，这种情况导致 IP 地址的浪费，一种解决方法就是应用 Private VLAN 技术。

私有 VLAN(Private VLAN)将一个 VLAN 的二层广播域划分成多个子域，每个子域都由一个私有 VLAN 对组成：主 VLAN(Primary VLAN)和辅助 VLAN(Secondary VLAN)。

一个私有 VLAN 域可以有多个私有 VLAN 对，每一个私有 VLAN 对代表一个子域。在一个私有 VLAN 域中所有的私有 VLAN 对共享同一个主 VLAN。每个子域的辅助 VLAN ID 不同。

一个私有 VLAN 域中只有一个主 VLAN，辅助 VLAN 实现同一个私有 VLAN 域中的二层隔离，有两种类型的辅助 VLAN：

- 隔离 VLAN(Isolated VLAN)：同一个隔离 VLAN 中的端口不能互相进行二层通信。一个私有 VLAN 域中只有一个隔离 VLAN。
- 群体 VLAN(Community VLAN)：同一个群体 VLAN 中的端口可以互相进行二层通信，但不能与其它群体 VLAN 中的端口进行二层通信。一个私有 VLAN 域中可以有多个群体 VLAN。

混杂端口（Promiscuous Port），属于主 VLAN 中的端口，可以与任意端口通讯，包括同一个私有 VLAN 域中辅助 VLAN 的隔离端口和群体端口。

隔离端口(Isolated Port)，隔离 VLAN 中的端口，只能与混杂口通讯。隔离端口接收到的报文可允许转发到 Trunk Port，但 Trunk Port 接收到 vid 是隔离 VLAN 的报文不能向隔离端口转发。

隔离 TRUNK 端口（Isolated Trunk Port），可以同时是多个普通 VLAN 和多个 PVLAN 的成员端口。在隔离 VLAN 中，只能与混杂口通讯；在群体 VLAN 中，可以与同一个群体 VLAN 的群体端口通讯，也可以同混杂口通讯；在普通 VLAN 中，遵循 802.1Q 规则。隔离 TRUNK 端口接收到的隔离 VLAN ID 的报文可允许转发到 Trunk Port，但 Trunk Port 接收到 vid 是隔离 VLAN 的报文不能向隔离端口转发。

从隔离 TRUNK 端口转发出的带 TAG 报文，其 VID 如果是主 VLAN ID，会转成相应从 VLAN 的 VID 后，再输出。

群体端口(Community port)，属于群体 VLAN 中的端口，同一个群体 VLAN 的群体端口可以互相通讯，也可以与混杂通讯。不能与其它群体 VLAN 中的群体端口及隔离 VLAN 中的隔离端口通讯。

各种端口类型间的报文转发关系：

输出端口 / 输入端口	混杂端口	隔离端口	群体端口	隔离 TRUNK 端口（同 VLAN 内）	TRUNK 端口（同 VLAN 内）
混杂端口	通	通	通	通	通
隔离端口	通	不通	不通	不通	通
群体端口	通	不通	通	通	通
隔离 TRUNK 端口（同 VLAN 内）	通	不通	通	不通（同隔离 VLAN 内不通，非隔离 VLAN 通）	通
TRUNK 端口（同 VLAN 内）	通	不通	通	不通	通

各种端口类型间的报文转发后 VLAN TAG 变化关系：

输出端口 / 输入端口	混杂端口	隔离端口	群体端口	隔离 TRUNK 端口（同 VLAN 内）	TRUNK 端口（同 VLAN 内）
混杂端口	不变	不变	不变	加上从 VLAN ID	加上主 VLAN ID TAG
隔离端口	不变	NA	NA	NA	加上隔离 VLAN ID TAG
群体端口	不变	NA	不变	加上群体 VLAN ID TAG	加上群体 VLAN ID TAG
隔离 TRUNK 端口（同 VLAN 内）	去掉 VLAN TAG	NA	去掉 VLAN TAG	非隔离 VLAN 内不变。	不变
TRUNK 端口（同 VLAN 内）	去掉 VLAN TAG	NA	去掉 VLAN TAG	主 VLAN 内转成从 VLAN ID，其它非隔离 VLAN 内不变。	不变
交换机 CPU	Untag	Untag	Untag	加上从 VLAN ID TAG	加上主 VLAN ID TAG

私有 VLAN 中，只有主 VLAN 可以创建 SVI 接口，辅助 VLAN 不可以创建 SVI。

私有 VLAN 中的端口可以为 SPAN 源端口，不可以为镜像目的端口。

12.2 产品特征

S3760 系列源 MAC 地址未知的广播，未知名组播，未知名单播报文在隔离端口间能够转发。

S3760 系列 Isolate Trunk Port 输出带 TAG 报文的 VID 不支持从 PVLAN 主 VLAN ID 转成相应从 VLAN ID。

S3760 系列各种端口类型间的报文转发后 VLAN TAG 变化关系（粗体字单元为特殊特征）：

输出端口 / 输入端口	混杂端口	隔离端口	群体端口	隔离 TRUNK 端口（同 VLAN 内）	TRUNK 端口（同 VLAN 内）
混杂端口	不变	不变	不变	**加上主 VLAN ID TAG**	加上主 VLAN ID TAG
隔离端口	不变	NA	NA	NA	加上隔离 VLAN ID TAG
群体端口	不变	NA	不变	加上群体 VLAN ID TAG	加上群体 VLAN ID TAG
隔离 TRUNK 端口(同 VLAN 内)	去掉 VLAN TAG	NA	去掉 VLAN TAG	非隔离 VLAN 内不变。	不变
TRUNK 端口（同 VLAN 内）	去掉 VLAN TAG	NA	去掉 VLAN TAG	**非隔离 VLAN 内都不变。**	不变
交换机 CPU	Untag	Untag	Untag	**加上主 VLAN ID TAG**	加上主 VLAN ID TAG

12.3 Private VLAN 配置

12.3.1 缺省 Private VLAN 设置

缺省情况下，没有 Private VLAN 的配置

12.3.2 配置 VLAN 作为私有 VLAN

配置方法如下命令

命令	说明
configure terminal	进入配置模式
vlan *vid*	进入 VLAN 配置模式
private-vlan{community | isolated| primary}	配置私有 VLAN 类型
no private-vlan{community | isolated | primary}	取消私有 VLAN 配置
end	退出 VLAN 模式
show vlan private-vlan [*type*]	显示私有 VLAN

说明：

在 802.1Q Vlan 中具有成员口情况不能声明为私有 VLAN，VLAN 1 不能声明为私有 VLAN，对于具有 Trunk 口或 Uplink 口的 802.1Q VLAN 中，先将该 VLAN 从许可 VLAN 列表中删除，一对 Private VLAN 处于 ACTIVE 状态必须满足以下条件：

1) 具有 Priamry VLAN

2) 具有 Secondary VLAN

3) Secondary VLAN 与 Primary VLAN 关联

以下命令将 802.1Q VLAN 配置为 Private VLAN：

```
Ruijie# configure terminal
Ruijie(config)# vlan 303
Ruijie(config-vlan)# private-vlan community
Ruijie(config-vlan)# end
Ruijie# show vlan private-vlan community
VLAN Type  Status    Routed   Interface   Associated VLANs
--- ----   --------  ------   ---------   ------------------
303 comm   inactive  Disabled             no association
Ruijie# configure terminal
Ruijie(config)# vlan 404
Ruijie(config-vlan)# private-vlan isolated
Ruijie(config-vlan)# end
Ruijie# show vlan private-vlan
VLAN Type  Status    Routed   Interface   Associated VLANs
--- ----   --------  ------   ---------   ------------------
303 comm  inactive  Disabled             no association
404 isol  inactive  Disabled             no association
```

12.3.3 关联 Secondary VLAN 和 Primary VLAN

按如下方式关联 Secondary VLAN 和 Primary VLAN：

命令	说明
configure terminal	进入配置模式
vlan *p_vid*	进入 Primary VLAN 配置模式
private-vlan association {*svlist* | **add** *svlist* | **remove** *svlist*}	关联 Secondary VLAN
no private-vlan association	清除与所有 Secondary VLAN 的关联
end	退出 VLAN 模式
show vlan private-vlan [*type*]	显示私有 VLAN

举例如下：

```
Ruijie# configure terminal
Ruijie(config)# vlan 202
Ruijie(config-vlan)# private-vlan association 303-307,309,440
Ruijie(config-vlan)# end
Ruijie# show vlan private-vlan
VLAN Type  Status    Routed   Interface   Associated VLANs
---- ----  --------  ------   ---------   ------------------
202  prim  inactive  Disabled             303-307,309,440
303  comm  inactive  Disabled             202
304  comm  inactive  Disabled             202
305  comm  inactive  Disabled             202
306  comm  inactive  Disabled             202
307  comm  inactive  Disabled             202
309  comm  inactive  Disabled             202
440  comm  inactive  Disabled             202
```

说明：

该操作在声明为 Primary VLAN 的 VLAN 配置模式进行。

12.3.4 映射 Secondary VLAN 和 Primary VLAN 的三层接口

您可以通过下面的设置步骤来完成：

命令	说明
configure terminal	进入配置模式
interface vlan *p_vid*	进入 Primary VLAN 的接口模式
private-vlan mapping {*svlist* | **add** *svlist* | **remove** *svlist*}	映射 Secondary VLAN 到 Primary VLAN 的 SVI 三层交换。
end	退出接口模式

下面例子配置 Secondary VLAN 路由：

```
Ruijie# configure terminal
Ruijie(config)# interface vlan 202
Ruijie(config-if)# private-vlan mapping add 303-307,309,440
Ruijie(config-if)# end
Ruijie#
```

说明：

操作中的 Primary VLAN 和 Secondary VLAN 是相关联的。

12.3.5 配置二层接口作为私有 VLAN 的主机端口

按照以下步骤配置二层接口作为私有 VLAN 的主机端口(Host Port)：

命令	说明
configure terminal	进入配置模式
interface <*interface*>	进入接口配置模式 *fastethernet*, *gigabitethernet*, *tengigabitethernet*
switchport mode private-vlan host	配置为二层交换模式
no switchport mode	清除私有 VLAN 配置
end	退出 SVI 接口模式
switchport private-vlan host-association *p_vid s_vid*	关联二层口与私有 VLAN
no switchport private-vlan host-association	清除关联

举例如下：

```
Ruijie# configure terminal
Ruijie(config)# interface gigabitEthernet 0/2
Ruijie(config-if)# switchport mode private-vlan host
Ruijie(config-if)# switchport private-vlan host-association
202 203
Ruijie(config-if)# end
```

说明：

操作中的 Primary VLAN 和 Secondary VLAN 是相关联的

12.3.6 配置二层接口作为可被隔离 PVLAN TRUNK 端口

配置二层接口作为可被隔离 PVLAN TRUNK 端口，通过以下操作命令：

	命令	作用
Step 1	Ruijie# **configure terminal**	进入配置模式
Step 2	Ruijie(config-if)# **interface** <*interface*>	进入接口配置模式 *fastethernet*, *gigabitethernet*, *tengigabitethernet*
Step 3	Ruijie(config-if)# **switchport mode trunk**	配置为 trunk 模式
Step 4	Ruijie(config-if)# **switchport private-vlan association trunk** *p_vid s_vid*	关联二层口与私有 VLAN，允许配置多对。p_vid 和 s_vid 参数分别是主 VLAN ID 和从 VLAN ID。
	Ruijie(config-if)# **no switchport private-vlan association trunk** *p_vid s_vid*	清除关联
Step 5	Ruijie(config-if)# **switchport trunk allowed vlan** {**all** | [**add** | **remove** | **except**] } *vlan-list*	（可选）配置这个 Trunk 口的许可 VLAN 列表。参数 *vlan-list* 可以是一个 VLAN，也可以是一系列 VLAN，以小的 VLAN ID 开头，以大的 VLAN ID 结尾，中间用-号连接。如：10–20。 **all** 的含义是许可 VLAN 列表包含所有支持的 VLAN； **add** 表示将指定 VLAN 列表加入许可 VLAN 列表； **remove** 表示将指定 VLAN 列表从许可 VLAN 列表中删除； **except** 表示将除列出的 VLAN 列表外的所有 VLAN 加入许可 VLAN 列表；

Step 6		
	Ruijie(config-if)# **switchport trunk native vlan** *vlan-id*	配置 Native VLAN 如果想把 Trunk 的 Native VLAN 列表改回缺省的 VLAN 1，请使用 **no switchport trunk native vlan** 接口配置命令。

举例如下：

```
Ruijie# configure terminal
Ruijie(config)# interface gigabitEthernet 0/2
Ruijie(config-if)# switchport mode trunk
Ruijie(config-if)# switchport private-vlan association trunk 202
203
Ruijie(config-if)# switchport trunk allowed vlan 100
Ruijie(config-if)# switchport trunk native vlan 100
Ruijie(config-if)# end
Ruijie#
```

说明	
	操作中的 Primary VLAN 和 Secondary VLAN 是相关联的 此命令软件版本 RGOS10.3(5)以上才有

12.3.7 配置二层接口作为私有 VLAN 的混杂端口

配置二层接口作为私有 VLAN 的端口，通过以下操作命令：

命令	说明
configure terminal	进入配置模式
interface <*interface*>	进入接口配置模式 百兆,千兆，万兆
switchport mode private-vlan promiscuous	配置为私有 VLAN 二层交换模式
no switchport mode	删除端口私有 VLAN 配置
switchport private-vlan mapping *p_vid*{*svlist* | **add** *svlist* | **remove** *svlist*}	私有 VLAN 混杂端口选择所在 VLAN，及混杂的 secondary VLAN 列表
no switchport private-vlan mapping	取消混杂所有的的 seconary VLAN.

下面例子描述配置过程：

```
Ruijie# configure terminal
```

```
Ruijie(config)# interface gigabitEthernet 0/2
Ruijie(config-if)# switchport mode private-vlan promiscuous
Ruijie(config-if)# switchport private-vlan mapping 202 add 203
Ruijie(config-if)# end
Ruijie#
```

说明:

操作中的 Primary VLAN 和 Secondary VLAN 是相关联的

12.4 Private VLAN 显示

12.4.1 显示 private VLAN

您可以通过以下步骤显示 Private VLAN 内容：

命令	说明
show vlan private-vlan [*type*]	显示 private VLAN 的内容

```
Ruijie# show vlan private-vlan
VLAN Type  Status    Routed   Interface   Associated VLANs
---- ----  --------  ------   ---------   ------------------
202  prim  active    Enabled   Gi0/1      303-307,309,440
303  comm  active    Disabled  Gi0/2      202
304  comm  active    Disabled  Gi0/3      202
305  comm  active    Disabled  Gi0/4      202
306  comm  active    Disabled             202
307  comm  active    Disabled             202
309  comm  active    Disabled             202
440  comm  active    Enabled   Gi0/5      202
```

12.5 Private VLAN 典型用例

12.5.1 多台交换机上的 Private VLAN 配置

12.5.1.1 组网需求

在两台设备上实现 Private VLAN 的配置应用，创建一个 Primary VLAN，一个 Community VLAN，一个 Isolated VLAN，同一个 Community VLAN 内的主机可以进行二层通讯，Isoated VLAN 内的主机与其它主机互不通讯，但 Private VLAN 内

的主机均能与路由器通讯。

12.5.1.2 组网拓扑

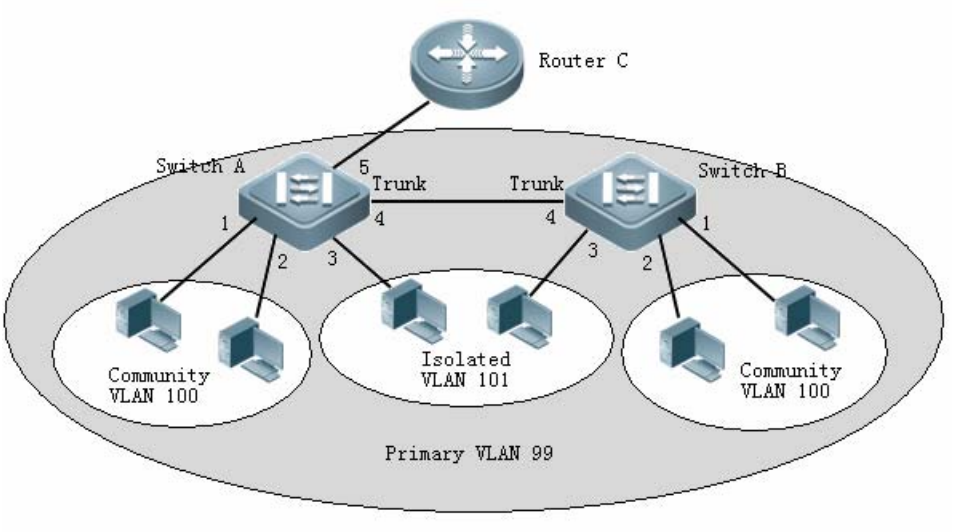


图 1

12.5.1.3 配置步骤

创建 VLAN 99 为 Primary VLAN，创建 VLAN 100 为 Community VLAN，创建 VLAN 101 为 Isolated VLAN，并关联主次 VLAN。

```
Ruijie#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Ruijie(config)#vlan 99
Ruijie(config-vlan)#private-vlan primary
Ruijie(config-vlan)#exit
Ruijie(config)#vlan 100
Ruijie(config-vlan)#private-vlan community
Ruijie(config-vlan)#exit
Ruijie(config)#vlan 101
Ruijie(config-vlan)#private-vlan isolated
Ruijie(config-vlan)#exit
Ruijie(config)#vlan 99
Ruijie(config-vlan)#private-vlan association 100,101
Ruijie(config-vlan)#exit
```

#配置端口 0／1 和 0／2 属于 Community VLAN 100，端口 0/3 属于 Isolated VLAN 101，端口 0/4 为 Promiscuous Port。

```
Ruijie(config)#interface gigabitEthernet 0/1
Ruijie(config-if)#switchport mode private-vlan host
Ruijie(config-if)#switchport private-vlan host-association 99
100
```

```
Ruijie(config-if)#exit
Ruijie(config)#interface gigabitEthernet 0/2
Ruijie(config-if)#switchport mode private-vlan host
Ruijie(config-if)#switchport private-vlan host-association 99
100
Ruijie(config-if)#exit
Ruijie(config)#interface gigabitEthernet 0/3
Ruijie(config-if)#switchport mode private-vlan host
Ruijie(config-if)#switchport private-vlan host-association 99
101
Ruijie(config-if)#exit
Ruijie(config)#interface gigabitEthernet 0/4
Ruijie(config-if)#switchport mode trunk
Ruijie(config-if)#exit
Ruijie(config)#interface gigabitEthernet 0/5
Ruijie(config-if)#switchport mode private-vlan promiscuous
Ruijie(config-if)#switchport  private-vlan  mapping  99  add
100-101
Ruijie(config-if)#show vlan private-vlan
VLAN     Type                Status       Routed       Ports
Associated VLANs
-----  ----------  --------  --------  -------------------------
99        primary        active      Disabled  Gi0/4,  Gi0/5
100-101
100   community  active   Disabled Gi0/1, Gi0/2, Gi0/4   99
101   isolated   active   Disabled Gi0/3, Gi0/4          99
```

12.5.2 单台三层交换机上的 Private VLAN 配置

12.5.2.1 组网需求

在支持 Private VLAN 的三层交换机上，可以为 Private VLAN 配置一个 SVI，一个 Private VLAN 下的所有 VLAN（包括 Primary VLAN 与的有的 Secondary VLAN）可以在同一个 SVI 内，只需在上个用例配置的基础上，为 Primary VLAN 配置一个 IP 地址，对于特定需要使用路由的 Secondary VLAN 只要在 Primary VLAN 配置相应的三层映射即可。

12.5.2.2 组网拓扑

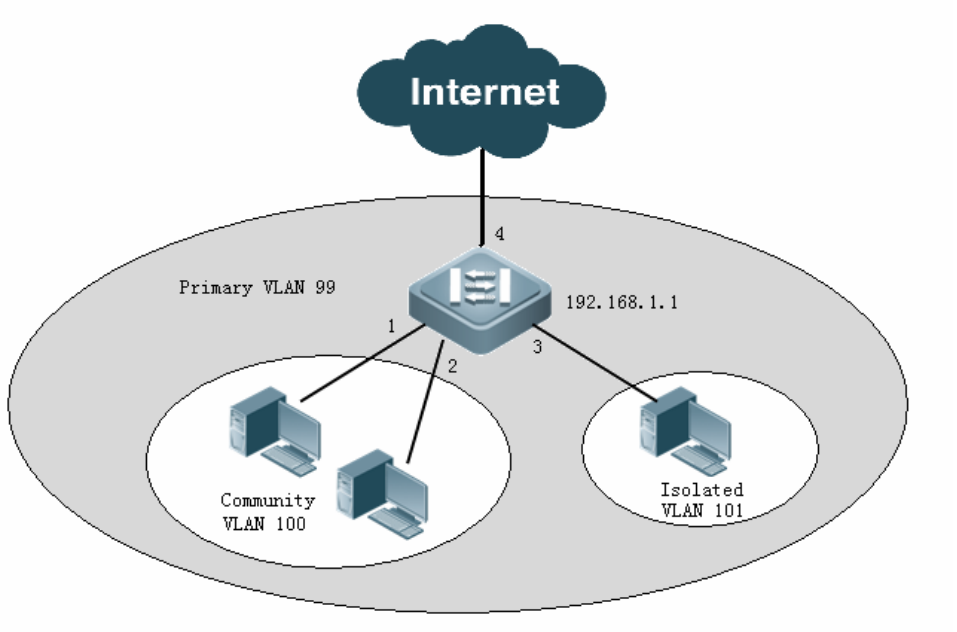


图 2

12.5.2.3 配置步骤

类似上一个用例，创建 VLAN 99 为 Primary VLAN，创建 VLAN 100 为 Community VLAN，创建 VLAN 101 为 Isolated VLAN，并关联主次 VLAN。

```
Ruijie#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Ruijie(config)#vlan 100
Ruijie(config-vlan)#private-vlan community
Ruijie(config-vlan)#exit
Ruijie(config)#vlan 101
Ruijie(config-vlan)#private-vlan isolated
Ruijie(config-vlan)#exit
Ruijie(config)#vlan 99
Ruijie(config-vlan)#private-vlan primary
Ruijie(config-vlan)#private-vlan association 100,101
Ruijie(config-vlan)#exit
```

#配置端口 0／1 和 0／2 属于 Community VLAN 100，端口 0/3 属于 Isolated VLAN 101，端口 0/4 为 Promiscuous Port。

```
Ruijie(config)#interface gigabitEthernet 0/1
Ruijie(config-if)#switchport mode private-vlan host
Ruijie(config-if)#switchport private-vlan host-association 99
100
Ruijie(config-if)#exit
Ruijie(config)#interface gigabitEthernet 0/2
```

```
Ruijie(config-if)#switchport mode private-vlan host
Ruijie(config-if)#switchport private-vlan host-association 99
100
Ruijie(config-if)#exit
Ruijie(config)#interface gigabitEthernet 0/3
Ruijie(config-if)#switchport mode private-vlan host
Ruijie(config-if)#switchport private-vlan host-association 99
101
Ruijie(config-if)#exit
Ruijie(config)#interface gigabitEthernet 0/4
Ruijie(config-if)#switchport mode private-vlan promiscuous
Ruijie(config-if)#switchport private-vlan mapping 99 add
100-101
Ruijie(config-if)#exit
```

为 Primary VLAN 配置一个 SVI (192.168.1.1)，以及映射 Secondary VLAN 和 Primary VLAN 的三层接口。

```
Ruijie(config)#interface vlan 99
Ruijie(config-if)#ip address 192.168.1.1 255.255.255.0
Ruijie(config-if)#private-vlan mapping 100-101
Ruijie(config-if)#show vlan private-vlan
VLAN     Type                Status       Routed       Ports
Associated VLANs
----- ---------- -------- -------- -------------------------
99          primary          active        Enabled     Gi0/4
100-101
100   community  active   Enabled  Gi0/1, Gi0/2         99
101   isolated   active   Enabled  Gi0/3                99
```

12.5.3 支持 Private VLAN 设备同只支持保护端口的设备配置

12.5.3.1 组网需求

支持 Private VLAN 的交换机下接多台只支持保护端口设备，要求接在任何设备的隔离端口下或保护端口下的主机间互相隔离。可以使用 Private VLAN 的 Isolate Trunk Port 同只有保护端口设备的 Trunk Port 连接实现。 如下图 1-3 所示。

12.5.3.2 组网拓扑

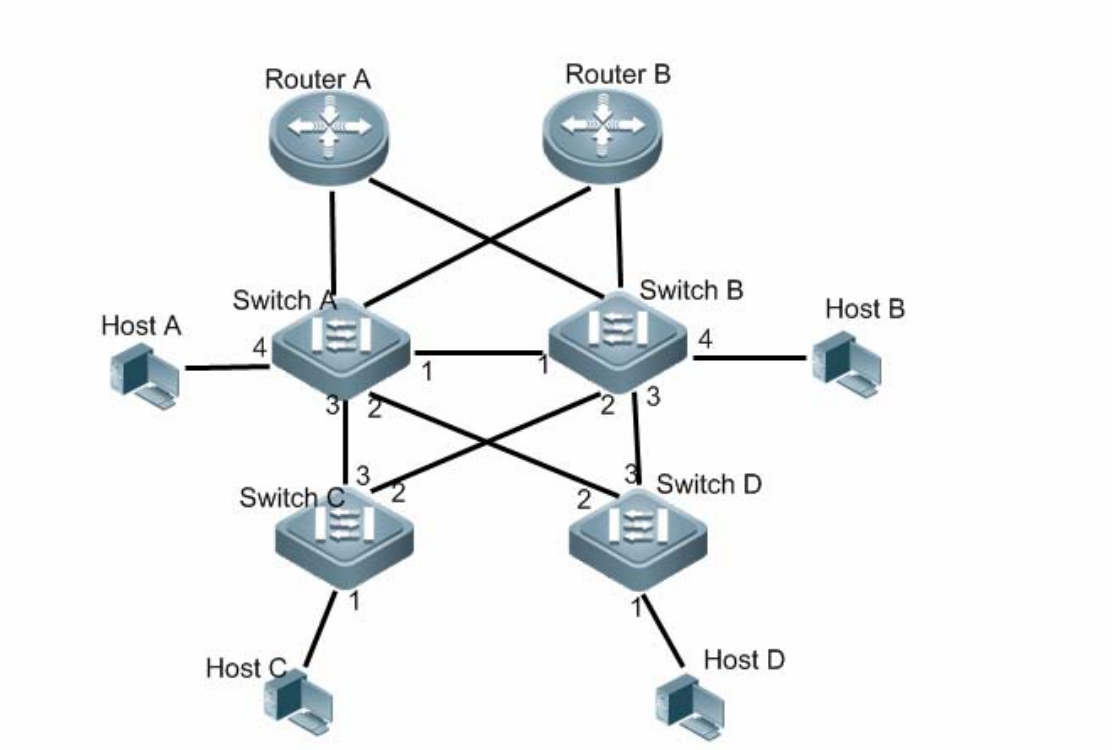


图 1-3

12.5.3.3 配置步骤

类似上一个用例，分别在 Switch A 和 Switch B 创建 VLAN 99 为 Primary VLAN，创建 VLAN 101 为 Isolated VLAN，并关联主次 VLAN。

```
Ruijie#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Ruijie(config)#vlan 101
Ruijie(config-vlan)#private-vlan isolated
Ruijie(config-vlan)#exit
Ruijie(config)#vlan 99
Ruijie(config-vlan)#private-vlan primary
Ruijie(config-vlan)#private-vlan association 101
Ruijie(config-vlan)#exit
```

＃Switch A 和 Switch B 都配置端口 0 / 1 为 Trunk Port，端口 0 / 2 和 0 / 3 为 Isolated Trunk Port，端口 0 / 4 属于 Isolated VLAN 101。

```
Ruijie(config)#interface gigabitEthernet 0/1
Ruijie(config-if)#switchport mode trunk
Ruijie(config-if)#exit
Ruijie(config)#interface gigabitEthernet 0/2
Ruijie(config-if)#switchport mode trunk
```

```
Ruijie(config-if)#switchport private-vlan association trunk
99 101
Ruijie(config-if)#exit
Ruijie(config)#interface gigabitEthernet 0/3
Ruijie(config-if)#switchport mode trunk
Ruijie(config-if)#switchport private-vlan association trunk
99 101
Ruijie(config-if)#exit
Ruijie(config)#interface gigabitEthernet 0/4
Ruijie(config-if)#switchport mode private-vlan host
Ruijie(config-if)#switchport private-vlan host-association 99
101
Ruijie(config-if)#exit
```

#Switch C 和 Switch D 创建 Vlan 101，并配置端口 0／1 为保护端口，配置端口 0／2 和 0／3 为 Trunk Port。

```
Ruijie(config)#vlan 101
Ruijie(config-vlan)#exit
Ruijie(config)#interface gigabitEthernet 0/1
Ruijie(config-if)#switchport protected
Ruijie(config-if)#exit
Ruijie(config)#interface gigabitEthernet 0/2
Ruijie(config-if)#switchport mode trunk
Ruijie(config-if)#exit
Ruijie(config)#interface gigabitEthernet 0/3
Ruijie(config-if)#switchport mode trunk
Ruijie(config-if)#exit
```

13 MAC 地址配置

以太网交换机利用 MAC 地址表的信息在数据链路层对报文进行快速寻址转发，本文是对 MAC 地址的配置方法进行描述，主要包含如下章节：

- 理解MAC地址表
- 缺省配置
- 配置动态地址
- 配置动态地址老化时间
- 配置静态地址
- 配置过滤地址
- 配置MAC地址变化通知
- 配置IP地址和MAC地址绑定
- 典型配置举例

13.1 理解 MAC 地址表

13.1.1 概述

通过识别报文的数据链路层信息对报文转发是以太网交换机的主要功能（称为二层转发功能），以太网交换机通过报文所携带的目的 MAC 地址信息将报文转发到相应的端口，以太网交换机采用 MAC 地址表存储报文转发时所需要的目的 MAC 地址与端口信息关系。

以太网交换机的 MAC 地址表中所有的 MAC 地址都和 VLAN 相关联，不同的 VLAN 允许相同的 MAC 地址。每个 VLAN 都维护它自己逻辑上的一份地址表。一个 VLAN 已学习的 MAC 地址，对于其他 VLAN 而言可能是未知的，仍然需要学习。

以太网交换机的 MAC 地址包含以下信息：

状态	VLAN	MAC地址	端口

图 3 MAC 地址表项构成

- 状态：表示地址表项为动态地址、静态地址或过滤地址。
- VLAN：MAC 地址所属的 VLAN
- MAC 地址：表项的 MAC 地址信息
- 端口：MAC 地址对应的端口信息

以太网交换机的 MAC 地址表中的表项通过以下两种方式进行更新和维护：

- 动态地址学习
- 手工配置地址

以太网交换机在转发报文时通过报文的目的 MAC 地址以及报文所属的 VLAN ID 的信息在 MAC 地址表中查找相应的转发输出端口，根据查找的结果采取单播、组播或广播的方式转发报文：

- 单播转发：以太网交换机能够在 MAC 地址表中查到与报文的目的 MAC 地址和 VLAN ID 相对应的表项并且表项中的输出端口是唯一的，报文直接从表项对应的端口输出。
- 组播转发：以太网交换机能够在 MAC 地址表中查到与报文的目的 MAC 地址和 VLAN ID 相对应的表项并且表项中对应一组输出端口，报文直接从这组端口输出。
- 广播转发：以太网交换机收到目的地址为 ffff.ffff.ffff 的报文或者在 MAC 地址表中查找不到对应的表项时，报文被送到所属的 VLAN 中除报文输入端口外的其他所有端口输出。

说明：

本文只涉及动态地址、静态地址与过滤地址的管理，组播地址的管理不在本文内描述，请参看《IGMP Snooping 配置》。

13.1.2 动态地址学习

13.1.2.1 动态地址

通过以太网交换机的自动地址学习过程产生的 MAC 地址表项被称为动态地址，只有动态地址才会被地址表的老化机制所删除。

13.1.2.2 地址学习过程

通常情况下 MAC 地址表的维护都是通过动态地址学习的方式进行，其工作原理如下：

1. 以太网交换机的 MAC 地址表为空的情况下，UserA 要与 UserB 进行通讯，UserA 首先发送报文到交换机的端口 GigabitEthernet 0/2，此时以太网交换机将 UserA 的 MAC 地址学习到 MAC 地址表中。
   由于地址表中没有 UserB 的源 MAC 地址，因此以太网交换机以广播的方式将报文发送到除了 UserA 以外的所有端口，包括 User B 与 User C 的端口，此时 UserC 能够收到 UserA 所发出的不属于它的报文。

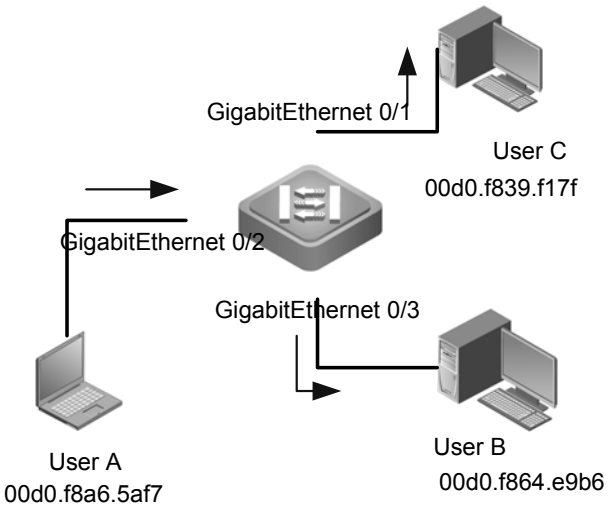


图 4 动态地址学习步骤一

Status	VLAN	MAC地址	端口
动态	1	00d0.f8a6.5af7	GigabitEthernet 0/2

图 5 以太网交换 MAC 地址表一

2. UserB 收到报文后将回应报文通过以太网交换机的端口 GigabitEthernet 0/3 发送 UserA，此时以太网交换机的 MAC 地址表中已存在 UserA 的 MAC 地址，所以报文被以单播的方式转发到 GigabitEthernet 0/2 端口，同时以太网交换机将学习 UserB 的 MAC 地址，与步骤 1 中所不同的是 UserC 此时接收不到 UserB 发送给 UserA 的报文。

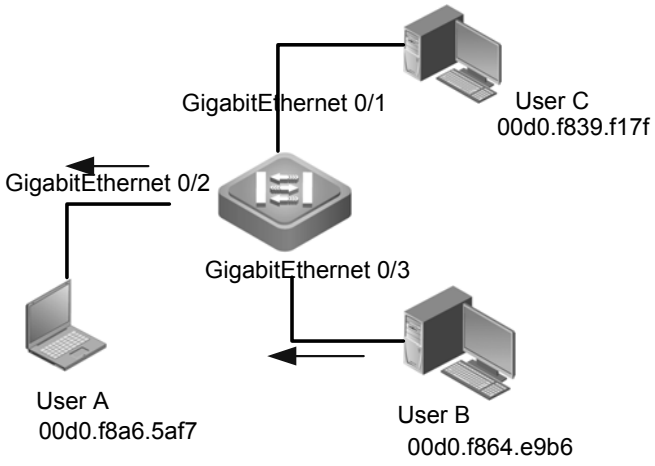


图 6 动态地址学习步骤二

Status	VLAN	MAC地址	端口
动态	1	00d0.f8a6.5af7	GigabitEthernet 0/2
动态	1	00d0.f864.e9b6	GigabitEthernet 0/3

图 7 以太网交换机 MAC 地址表二

3. 通过 UserA 与 UserB 的一次交互过程后，以太网交换机学习到了 UserA 与 UserB 的源 MAC 地址，之后 UserA 与 UserB 之间的报文交互则采用单播的方式进行转发，此后 UserC 将不再接收到 UserA 与 UserB 之间的交互报文。

13.1.2.3 地址老化

以太网交换机的 MAC 地址表是有容量限制的，以太网交换机采用地址表老化机制进行不活跃的地址表项淘汰。

以太网交换机在学习到一个新的地址的同时启动该地址的老化记时，在达到老化记时前，如果以太网交换机没有再一次收到以该地址为源 MAC 地址的报文，那该地址在达到老化时间后会从 MAC 地址表中删除。

13.1.3 静态地址

手工配置的 MAC 地址表项，用于绑定 MAC 地址与端口关系，这类地址只能通过手工配置添加和删除，保存配置后设备重启，静态地址也不会丢失。

通过手工配置静态地址的方式可以在 MAC 地址表中绑定设备下接的网络设备的 MAC 地址与端口关系。

13.1.4 过滤地址

手工配置的 MAC 地址表项，用于在以太网交换机上丢弃以所配置的 MAC 地址为源地址或目的地址的报文，这类地址只能通过手工配置添加和删除，保存配置后设备重启，过滤地址也不会丢失。

通过手工将网络中的非法接入用户的源 MAC 地址配置为过滤地址的方式可以实现过滤非法接入用户。

注意：

过滤地址对送 CPU 的报文无效。如：某个 ARP 报文的二层源 MAC 为一过滤地址，此时该 ARP 报文仍然会被送往 CPU，但其并不会被转发。

13.1.5 MAC 地址变化通知

以太网交换机的 MAC 地址通知功能通过与网络管理工作站（NMS）的协作为网络管理提供了监控网络以太网交换机下用户变化的机制。

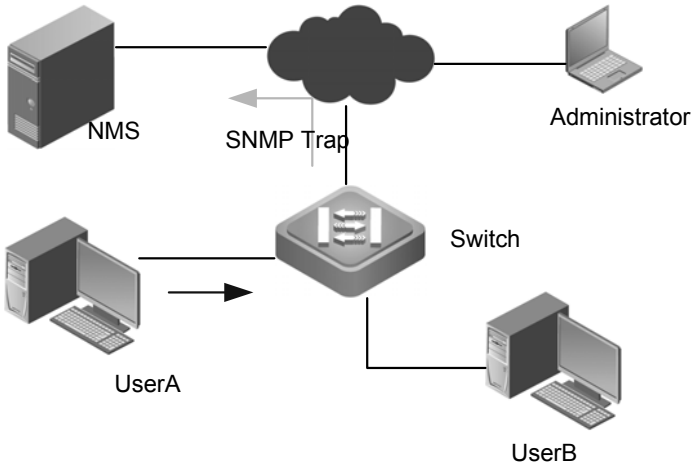


图 8 地址变化通告

打开 MAC 地址通知的功能后，当以太网交换机学习到一个新的 MAC 地址或老化掉一个已学习到的 MAC 地址时，一个反映 MAC 地址变化的通知信息就会产生，并以 SNMP Trap 的方式将通知信息发送给指定的 NMS(网络管理工作站)。

当一个 MAC 地址增加的通知产生，就可以知道一个由此 MAC 地址标识的新用户开始使用网络，当一个 MAC 地址删除的通知产生，则表示一个用户在地址老化时间内没有新的报文发送，通常可以认为此用户已经停止使用网络了。

当使用以太网交换机下接的用户较多时，可能会出现在短时间内会有大量的 MAC 地址变化产生，导致网络流量增加。为了减轻网络负担，可以设置发送 MAC 地址通知的时间间隔。在达到配置的时间间隔之后，系统将这个时间内的通知信息封装成多个通知信息，此时在每条地址通知信息中，就包含了若干个 MAC 地址变化的信息，从而可以会有效地减少网络流量。

当 MAC 地址通知产生时，通知信息同时会记录到 MAC 地址通知历史记录表中。此时即便没有配置接收 Trap 的 NMS，管理员也可以通过查看 MAC 地址通知历史记录表来了解最近 MAC 地址变化的消息。

说明：

MAC 地址通知仅对动态地址有效，对于配置的静态地址与过滤地址的变化将不会产生通知信息。

13.1.6 IP 和 MAC 地址绑定

13.1.6.1 概述

通过手动配置 IP 和 MAC 地址绑定功能，可以对输入的报文进行 IP 地址和 MAC 地址绑定关系的验证。如果将一个指定的 IP 地址和一个 MAC 地址绑定，则设备只接收源 IP 地址和 MAC 地址均匹配这个绑定地址的 IP 报文；否则该 IP 报文将被

丢弃。

利用地址绑定这个特性，可以严格控制设备的输入源的合法性。需要注意的是，通过地址绑定控制交换机的输入，将优先于 802.1X、端口安全以及 ACL 生效。

13.1.6.2 地址绑定模式

地址绑定的模式分为:兼容，宽松，严格三种模式，默认模式为严格模式。其相应的转发规则，见下表所示：

模式	IPv4 报文转发规则	IPv6 报文转发规则
严格	符合 IPV4+MAC 条件的报文转发	所有 IPV6 报文均不转发
宽松	符合 IPV4+MAC 条件的报文转发	所有 IPV6 报文均可转发
兼容	符合 IPV4+MAC 条件的报文转发	源 MAC 为绑定地址的 MAC 地址的 IPV6 报文转发

13.1.6.3 地址绑定例外端口

IP 地址和 MAC 地址绑定功能缺省对设备上的所有端口都生效，通过配置例外口的方式可以在使绑定功能在部份端口上不生效。

说明:

在应用中设备的上链端口的 IP 报文的绑定关系是不确定的，通常将设备的上链端口配置为例外口，此时上链端口则不进行 IP 地址与 MAC 地址的绑定检查。

13.1.7 协议规范

《IEEE Std 802.3™ Carrier sense multiple access with collision detection (CSMA/CD) access method and physical layer specifications》

《IEEE Std 802.1Q™ Virtual Bridged Local Area Networks》

13.2 缺省配置

功能特性	缺省值
动态地址老化时间	300 秒
MAC 地址变化通知功能	关闭
地址绑定模式	compatible

13.3 配置动态地址

- 清除动态地址
- 查看配置

13.3.1 清除动态地址

命令	作用
Ruijie**#clear mac-address-table dynamic**	删除设备上所有的动态地址
Ruijie**#clear mac-address-table dynamic address** *mac-address* **vlan** *vlan-id*	删除特定 MAC 地址 mac-address：指定要删除的 MAC 地址。 vlan-id：指定要删除的 MAC 地址所在的 VLAN。
Ruijie**#clear mac-address-table dynamic interface** *interface-id* [**vlan** *vlan-id*]	删除特定物理接口或 Aggregate Port 上的特定 VLAN 中的所有动态地址或接口上所有动态地址。 interface-id：指定的物理接口或是 Aggregate Port。 vlan-id：指定删除动态地址所属的 VLAN。
Ruijie**#clear mac-address-table dynamic vlan** vlan-id	删除特定 VLAN 上的所有动态地址 vlan-id：指定所要删除的动态地址所属的 VLAN。

下面的例子说明了如何配置删除设备接口 GigabitEthernet 0/1 下 VLAN 1 中的所有动态地下。

```
Ruijie#clear mac-address-table dynamic interface GigabitEthernet
0/1 vlan 1
```

13.3.2 查看配置

命令	作用
Ruijie**#show mac-address-table dynamic**	查看设备上所有的动态地址信息。
Ruijie**#show mac-address-table dynamic address** *mac-address* [**vlan** *vlan-id*]	查看设备上特定动态 MAC 地址信息。 mac-address：查看的 MAC 地址。 vlan-id：查看特定的 VLAN 中的特定 MAC 地址。

Ruijie**#show mac-address-table dynamic interface** *interface-id* [**vlan** *vlan-id*]	查看设备上指定物理接口或 Aggregate Port 下的动态地址信息。 interface-id：指定的物理接口或是 Aggregate Port。 vlan-id：查看特定的 VLAN 中的动态地址。
Ruijie**#show mac-address-table dynamic vlan** *vlan-id*	查看设备上指定 VLAN 下的动态地址信息。 vlan-id：查看特定的 VLAN 中的动态地址。
Ruijie**# show mac-address-table count**	查看地址表的统计信息

下面的例子说明了如何查看设备上物理接口 GigabitEthernet 0/1 下 VLAN 中的所有动态 MAC 地址信息。

```
Ruijie#show mac-address-table dynamic interface gigabitEthernet
0/1 vlan 1
Vlan        MAC Address         Type     Interface
----------  ------------------- -------- ------------------
  1         0000.5e00.010c      DYNAMIC  GigabitEthernet 0/1
  1         00d0.f822.33aa      DYNAMIC  GigabitEthernet 0/1
  1         00d0.f822.a219      DYNAMIC  GigabitEthernet 0/1
  1         00d0.f8a6.5af7      DYNAMIC  GigabitEthernet 0/1
```

下面的例子说明了如何查看设备上的地址表统计信息：

```
Ruijie# show mac-address-table count
Dynamic Address Count  : 30
Static  Address Count  : 0
Filtering Address Count: 0
Total Mac Addresses    : 30
Total Mac Address Space Available: 8159
```

13.4 配置动态地址老化时间

- 配置老化时间
- 查看配置

13.4.1 配置老化时间

命令	作用

Ruijie(config)# **mac-address-table agint-time** [0 |10-1000000]	设置一个地址被学习后将保留在动态地址表中的时间长度，单位是秒，范围是 10－1000000 秒，缺省为 300 秒。当你设置这个值为 0 时，地址老化功能将被关闭，学习到的地址将不会被老化。
Ruijie(config)**#on mac-address-table agint-time**	恢复地址老化时间为缺少值。

下面的例子说明了如何配置设备的地址老化时间为 180 秒。

```
Ruijie#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Ruijie(config)#mac-address-tabsle aging-time 180
```

13.4.2 查看配置

命令	作用
Ruijie**#show mac-address-table aging-time**	查看所有的地址老化配置信息

下面的例子说明了如何查看设备上的地址老化时间配置。

```
Ruijie#show mac-address-table aging-time
Aging time    : 180 seconds
```

注意：

地址表的实际老化时间会与设定值存在一定偏差，但不会超过设定值的 2 倍。
S3760 系列设备的老化时间范围为 10 – 630 秒

13.5 配置静态地址

- 管理静态地址
- 查看配置

13.5.1 管理静态地址

命令	作用

Ruijie(config)# **mac-address-table static** *mac-address* **vlan** *vlan-id* **interface** *interface-id*	mac-address：指定表项对应的目的 MAC 地址。 vlan-id：指定该地址所属的 VLAN。 interface-id：包将转发到的接口(可以是物理端口或 Aggregate Port)。 当设备在 vlan-id 指定的 VLAN 上接收到以 mac- address 为目的地址的报文时，这个报文将被转发到 interface-id 所指定的接口上。
Ruijie(config)# **no mac-address-table static** *mac-address* **vlan** *vlan-id* **interface** *interface-id*	删除静态地址表项，参数与添加命令一致。

下面的例子说明了如何配置添加一个静态地址 00d0.f800.073c，当在 VLAN 4 中接收到目的地址为该地址的报文时，这个报文将被转发到指定的接口 Gigabitethernet 0/3 上。

```
Ruijie#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Ruijie(config)# mac-address-table static 00d0.f800.073c vlan 4
interface gigabitethernet 0/3
```

下面的例子说明了如何配置删除上一例子中添加的静态地址 00d0.f800.073c。

```
Ruijie#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Ruijie(config)#no mac-address-table static 00d0.f800.073c vlan 4
interface gigabitethernet 0/3
```

13.5.2 查看配置

命令	作用
Ruijie**#show mac-address-table static**	查看所有的静态地址信息

下面的例子说明了如何查看设备上的所有静态地址信息。

```
Vlan        MAC Address          Type      Interface
----------  -------------------- --------  -------------------
    4       00d0.f800.073c       STATIC    GigabitEthernet 0/3
```

13.6 配置过滤地址

- 管理过滤地址
- 查看配置

13.6.1 管理过滤地址

命令	作用
Ruijie(config)# **mac-address-table filtering** *mac-address* **vlan** *vlan-id*	mac-address：指定表项对应的 MAC 地址 vlan-id：指定该地址所属的 VLAN 当设备在 vlan-id 指定的 VLAN 上接收到以 mac- address 指定的地址为源地址或目的地址的报文将被丢弃。
Ruijie(config)# **no mac-address-table filtering** *mac-address* **vlan** *vlan-id*	删除过滤地址表项，参数与添加命令一致。

下面的例子说明了如何配置添加一个过滤地址 00d0.f800.073c，当在 VLAN 4 中接受到源地址或目的地址为该地址的报文时，将丢弃此报文。

```
Ruijie#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Ruijie(config)#mac-address-table filtering 00d0.f800.073c vlan 4
```

下面的例子说明了如何配置删除上一例子中的静态地址 00d0.f800.073c。

```
Ruijie#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Ruijie(config)#no mac-address-table filtering 00d0.f800.073c
vlan 4
```

13.6.2 查看配置

命令	作用
Ruijie**#show mac-address-table filtering**	查看所有的过滤地址信息

下面的例子说明了如何查看设备上的所有静态地址信息。

```
Vlan        MAC Address          Type      Interface
----------  -------------------- --------  -------------------
    4       00d0.f800.073c       FILTER    GigabitEthernet 0/3
```

13.7 配置 MAC 地址变化通知

- 配置MAC地址变化通知
- 查看配置

13.7.1 配置 MAC 地址变化通知

缺省情况下，MAC 地址的全局开关被关闭，所有接口的 MAC 地址通知功能也均被关闭。

配置设备 MAC 地址通知功能：

命令	作用
Ruijie(config)# **snmp-server host** *host-addr* **traps** [**version** {**1**|**2c** |**3** [**auth** | **noauth** | **priv**]}] *community-string*	配置接收 MAC 地址通知的 NMS。 host-addr：指明接收者的 IP。 version：指明发送哪种版本的 snmp trap 报文，对 v3 版本还可以指定是否认证以及安全等级参数。
Ruijie (config)**#snmp-server enable traps**	使能设备发送 trap 功能 。
Ruijie(config)**#mac-address-table notification**	打开 MAC 地址通知功能开关。
Ruijie(config)**#mac-address-table notification** {**interval** *value* | **history-size** *value*}	配置 MAC 地址通知的时间间隔与历史记录容量。 interval value ：设置产生 MAC 地址通知的时间间隔(可选)。时间间隔的单位为秒，范围为 1－3600，缺省为 1 秒。 history-size value ：MAC 通知历史记录表中记录的最大个数，范围 1－200，缺省为 50。
Ruijie(config-if)**# snmp trap mac-notification** {**added** |**removed**}	打开接口的 MAC 地址通知功能。 added ： 当地址增加时通知。 removed： 当地址被删除时通知。

下面的例子说明了如何打开 MAC 地址通知功能，并以 public 为认证名向 IP 地址为 192.168.12.54 的 NMS 发送 MAC 地址变化通知的 Trap，产生 MAC 地址变化通知的间隔时间为 40 秒，MAC 地址通知历史记录表的大小为 100，打开接口 Gigabitethernet 0/1 上当 MAC 地址增加和减少时进行通知的功能：

```
Ruijie#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Ruijie(config)#snmp-server host 192.168.12.54 traps public
Ruijie(config)#snmp-server enable traps
Ruijie(config)#mac-address-table notification
Ruijie(config)#mac-address-table notification interval 40
Ruijie(config)#mac-address-table notification  history-size 100
Ruijie(config)#interface GigabitEthernet 0/1
```

```
Ruijie(config-if)#snmp trap mac-notification added
Ruijie(config-if)#snmp trap mac-notification removed
```

13.7.2 查看配置

在特权模式下，使用下表所列的命令来查看设备的 MAC 地址表信息：

命令	作用
Ruijie#show mac-address-table notification	查看 MAC 地址变化通知功能的全局配置信息。
Ruijie#show mac-address-table notification interface	查看接口的 MAC 地址变化通知的使能状况。
Ruijie#show mac-address-table notification history	查看 MAC 地址变化通知信息的历史记录表。

下面是查看 MAC 地址变化通知信息的例子。

查看 MAC 地址通知功能的全局配置信息：

```
Ruijie#show mac-address-table notification
MAC Notification Feature : Enabled
Interval(Sec): 2
Maximum History Size : 154
Current History Size : 2
Ruijie# show mac-address-table notification interface
Interface         MAC Added Trap  MAC Removed Trap
----------------- -------------- ----------------
Gi0/1              Disabled       Enabled
Gi0/2              Disabled       Disabled
Gi0/3              Enabled        Enabled
Gi0/4              Disabled       Disabled
Gi0/5              Disabled       Disabled
Gi0/6              Disabled       Disabled

Ruijie#show mac-address-table notification history
History Index:1
Entry Timestamp: 15091
MAC Changed Message :
Operation   VLAN  MAC Address    Interface
----------- ---- -------------- --------------------
Added        1    00d0.f808.3cc9  Gi0/1
Removed     1     00d0.f808.0c0c  Gi0/1
History Index:2
Entry Timestamp: 21891
MAC Changed Message :
Operation   VLAN  MAC Address    Interface
----------- ---- -------------- --------------------
```

```
Added      1     00d0.f80d.1083 Gi0/1
```

13.8 配置 IP 地址和 MAC 地址绑定

13.8.1 配置 IP 地址和 MAC 地址绑定

在全局模式下，可以通过以下步骤来设置地址绑定：

命令	作用
Ruijie(config)# **address-bind** *ip-address mac-address*	配置 IP 地址和 MAC 地址的绑定关系 ip-address：绑定的 IP 地址 mac-address：绑定的 MAC 地址
Ruijie(config)# **address-bind install**	使 IP 和 MAC 地址绑定生效

在全局配置模式下使用 **no address-bind** *ip-address mac-address* 删除一个 IP 地址和 MAC 地址的绑定项。

通过 **no address-bind install** 命令可关闭绑定功能，使地址绑定配置不生效。

下面是配置 IP 地址和 MAC 地址绑定模式的例子：

```
Ruijie#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Ruijie(config)#address-bind 192.168.5.1 00d0.f800.0001
Ruijie(config)#address-bind install
```

13.8.2 配置地址绑定模式

命令	作用
Ruijie(config)# **address-bind ipv6-mode** { **compatible**｜**loose**	**strict** }	配置地址绑定模式 compatible：兼容模式 loose：宽松模式 strict：严格模式
Ruijie(config)# **no adress-bind ipv6-mode**	恢复地址绑定模式为缺省值

下面是配置地址绑定模式为严格模式的例子：

```
Ruijie#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Ruijie(config)#address-bind ipv6-mode strict
```

注意：

设置 IPV6 模式对 **DHCP Snooping** 的地址绑定，端口安全的 MAC+IP 等多种 MAC+IP 绑定功能同时生效。对于某些产品的端口安全等功能支持 IPV6 的安全地址，IPV6 模式下的转发规则，如下表所示：

模式	Ipv4 转发规则	IPV6 转发规则
严格模式	只允许符合 IPV4+MAC 条件的报文转发。	只允许符合 IPV6 安全地址配置的 IPV6 报文转发。
宽松模式	只允许符合 IPV4+MAC 条件的报文转发。	允许所有的 IPV6 报文转发。
兼容模式	只允许符合 IPV4+MAC 条件的报文转发。	只允许源 MAC 为绑定的 MAC 地址，或符合 IPV6 安全地址配置的 IPV6 报文转发。

13.8.3 配置地址绑定例外端口

命令	作用
Ruijie(config)**#address-bind uplink** *interface-id*	配置地址绑定的例外端口 interface-id：端口或 Aggregate Port

下面是配置端口 GigabitEthernet 0/1 为例外端口的例子：

```
Ruijie#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Ruijie(config)#address-bind uplink GigabitEthernet 0/1
```

13.8.4 查看配置

在特权模式下使用下列命令查看设备上的 IP 地址和 MAC 地址绑定的相关配置

命令	作用
Ruijie**#show address-bind**	查看设备上的 IP 地址与 MAC 地址绑定配置

下面是查看设备的 IP 地址和 MAC 地址绑定配置的例子：

```
Ruijie#show address-bind
Total Bind Addresses in System : 1

IP Address        Binding MAC Addr
---------------    ----------------
   192.168.5.1    00d0.f800.0001
```

13.9 典型配置举例

13.9.1 网络配置

某信息系统组网如下：

数据库服务器通过 GigabitEthernet0/1 接到以太网交换机，WEB 服务服务器通 GigabitEthernet 0/2 接到以太网交换机，服务器管理员通过 GigabitEthernet 0/3 接到以太网交换机，其他的一般用户通过以太网交换机的 GigabitEthernet 0/10 接入访问 WEB 服务器，所有的数据均在 VLAN 1 中转发。

为保障 WEB 服务器与数据库之间交互的信息安全以及管理员与服务器之间交互的信息的安全，通过配置静态地址的方式保证 WEB 服务器与数据库服务器之间的数据转发采用单播方式，管理员与服务器之间的数据转发也采用单播方式，这样可以有效的避免这些数据被以广播的方式转发到一般用户所使用的网络中。

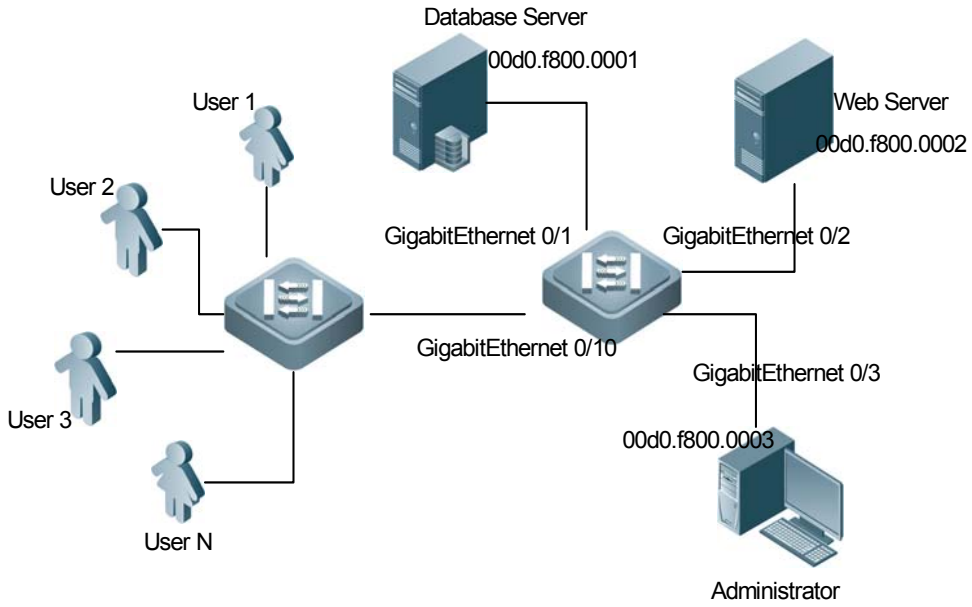


图 9 典型应用模型

13.9.2 设备配置

设备配置过程

```
Ruijie#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Ruijie(config)#mac-address-table static 00d0.f800.0001 vlan 1
interface GigabitEthernet 0/1
Ruijie(config)#mac-address-table static 00d0.f800.0002 vlan 1
interface GigabitEthernet 0/2
```

```
Ruijie(config)#mac-address-table static 00d0.f800.0003 vlan 1
interface GigabitEthernet 0/3
```

查看设备配置

```
Ruijie#show mac-address-table static
Vlan       MAC Address         Type    Interface
----------  ------------------- -------- ------------------
  1       00d0.f800.0001      STATIC   GigabitEthernet 0/1
  1       00d0.f800.0002      STATIC   GigabitEthernet 0/2
  1       00d0.f800.0003      STATIC   GigabitEthernet 0/3
```

14 DHCP Snooping 配置

14.1 DHCP Snooping 简介

14.1.1 理解 DHCP

DHCP 协议被广泛用来动态分配可重用的网络资源，如 IP 地址。一次典型的 DHCP 获取 IP 的过程如下所示：

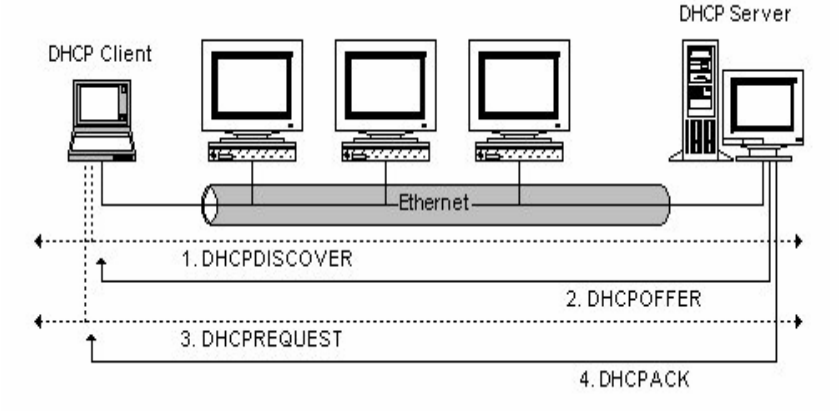


图 1

DHCP Client 发出 DHCP DISCOVER 广播报文给 DHCP Server，若 Client 在一定时间内没有收到服务器的响应，则重发 DHCP DISCOVER 报文。

DHCP Server 收到 DHCP DISCOVER 报文后，根据一定的策略来给 Client 分配资源(如 IP 地址)，然后发出 DHCP OFFER 报文。

DHCP Client 收到 DHCP OFFER 报文后，发出 DHCP REQUEST 请求，请求获取服务器租约。

服务器收到 DHCP REQUEST 报文，验证资源是否可以分配，如果可以分配，则发送 DHCP ACK 报文；如果不可分配，则发送 DHCP NAK 报文。DHCP Client 收到 DHCP ACK 报文，再次利用 ARP 验证资源可用后，开始使用服务器分配的资源。如果收到 DHCP NAK，则重新发送 DHCP DISCOVER 报文。

14.1.2 理解 DHCP Snooping

DHCP Snooping 技术：简称 DHCP 窥探，通过对客户端和服务器之间的 DHCP 交互报文进行窥探，实现对用户的监控，同时 DHCP Snooping 具有对 DHCP 报文过滤的功能，通过合理的配置可屏蔽非法 DHCP 服务器。下边对 DHCP Snooping 内使用到的一些术语及功能进行一些解释：

DHCP Snooping TRUST 口：由于 DHCP 获取 IP 的交互报文是使用广播的形式，因此可能存在非法服务器影响用户获取 IP 地址。为了防止非法服务器问题，将端口配置为两种类型，TRUST 口和 UNTRUST 口。对于 DHCP 客户端请求报文，仅将其转发到 TRUST 口。对于 DHCP 服务器响应报文，仅转发来自 TRUST 口的响应报文，而丢弃所有来自 UNTRUST 口的响应报文。通过将合法 DHCP 服务器连接的端口设置为 TURST 口，其他口设置为 UNTRUST 口，就可以实现对非法 DHCP 服务器的屏蔽。

DHCP Snooping 绑定数据库：DHCP Snooping 通过窥探客户端和服务器之间的交互报文，把用户获取到的 IP 地址以及用户 MAC、VID、PORT、租约时间等信息组成一个用户表项，从而形成一个 DHCP Snooping 绑定数据库。

DHCP Snooping 功能对经过设备的 DHCP 报文进行合法性检查，丢弃不合法的 DHCP 报文，并记录用户信息生成 DHCP Snooping 绑定数据库。以下几种类型的报文被认为是非法的 DHCP 报文：

1) UNTRUST 口收到的 DHCP 服务器响应报文，包括 DHCPACK、DHCPNAK、DHCPOFFER 等。

2) 打开源 mac 校验功能时，链路层头部源 MAC 与 DHCP 报文携带的 DHCP Client 字段不相同的报文。

3) 与 DHCP Snooping 绑定数据库中用户信息不一致的 DHCPRELEASE 报文。

14.1.3 DHCP Snooping information option

部分网络管理员在对当前的用户进行 IP 地址管理时，希望能够根据用户所处的位置来确定给用户分配 IP 地址，即希望能够根据用户所连接的网络设备信息进行分配。交换机在窥探 DHCP 报文的同时，可以把一些用户相关的设备信息以 DHCP option82 选项加入到 DHCP 请求报文中。通过 option82 选项的信息，服务器可以更准确的为用户分配 IP 地址。DHCP snooping 加入 option82 选项的格式如下：

Agent Circuit ID

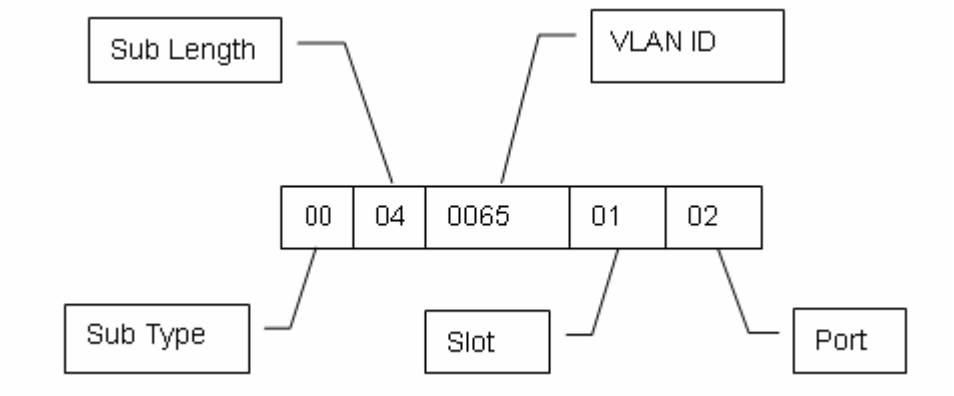


图 2

Agent Remote ID

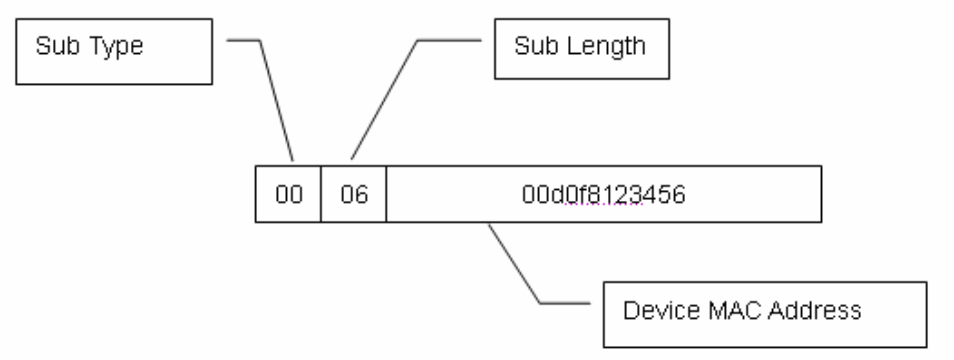


图 3

14.1.4 DHCP Snooping 地址绑定功能

DHCP snooping 地址绑定功能就是通过对 DHCP 报文交互过程进行窥探，将合法用户信息(IP 地址、MAC、VLAN、PORT、租约时间)，形成 DHCP snooping 数据库。将 DHCP snooping 数据库中的用户信息添加到 IP 报文硬件过滤表中，从而限制只有 DHCP snooping 数据库中的合法用户才能够发送 IP 报文，防止非法用户私设 IP 地址。

14.1.5 DHCP Snooping 支持 Bootp 用户绑定

DHCP Snooping 是基于 DHCP 协议设计的安全控制功能，对于 Bootp 应用的兼容性不足。主要是因为在 Bootp 报文中不存在 DHCP 相关 Option，如：报文类型、租约时间等选项。因此 DHCP Snooping 很难对 Bootp 应用环境中的用户进行有效控制。

针对市场日益增多的无盘工作站应用，我们对 DHCP Snooping 功能进行了扩展，增加了对 Bootp 协议的支持。DHCP Snooping 对 DHCP 报文交互进行窥探处理的同时，添加了对 Bootp 报文的处理，针对合法的 Bootp 用户，将会提取用户的 IP 地址、MAC 地址、端口及所属 VLAN 信息，添加一个静态绑定表项到 DHCP Snooping 数据库中。

Bootp 用户在 DHCP Snooping 数据库中相当于一个静态绑定用户，同手工配置的静态绑定用户相同。此用户在 Bootp 交互过程中动态生成，但是删除时同静态用户操作相同，需要管理员手动配置命令进行删除。

14.1.6 DHCP snooping 的相关安全功能

DHCP Snooping 地址绑定只是对 IP 报文进行过滤，不能对 ARP 报文进行过滤。为了增加安全性，防止 ARP 欺骗，需要对非法 ARP 报文进行过滤。DHCP Snooping 数据库可以为 ARP 报文过滤提供依据。（ARP 报文过滤，可以参考 ARP-CHECK、DAI 相关章节）

14.1.7 DHCP Snooping 配置的其他注意事项

1) DHCP Snooping 功能与 DHCP Relay Option 82 功能是互斥的，即不能同时打开 DHCP Snooping 和 DHCP Relay Option82 功能。

2) 将端口配置为 TRUST 口后，不再对该端口下的用户进行安全控制。

14.2 DHCP Snooping 配置

14.2.1 配置打开和关闭 DHCP Snooping

缺省情况下，设备 DHCP Snooping 功能为关闭状态。配置该功能后，DHCP Snooping 开始对经过该设备的所有 DHCP 报文进行窥探。

命令	说明
Ruijie# **configure terminal**	进入配置模式
Ruijie(config)# **[no] ip dhcp snooping**	打开和关闭 DHCP snooping

下面是打开设备的 DHCP Snooping 功能：

```
Ruijie# configure terminal
Ruijie(config)# ip dhcp snooping
Ruijie(config)# end
```

注意：

DHCP Snooping 与 Private VLAN 功能不支持共用。

14.2.2 配置 DHCP Snooping 功能生效的 VLAN

缺省情况下，设备打开 DHCP Snooping 开局功能开关后，会生效于所有 VLAN。通过配置该功能，可以在指定 VLAN 上打开和关闭 DHCP Snooping 功能。

命令	说明
Ruijie# **configure terminal**	进入配置模式
Ruijie(config)# **[no] ip dhcp snooping vlan** {*vlan-rng* | {*vlan-min* [*vlan-max*]}}	配置 DHCP Snooping 功能生效的 VLAN

下边是配置 DHCP snooping 功能在 VLAN1000 上生效：

```
Ruijie# configure terminal
Ruijie(config)# ip dhcp snooping vlan 1000
```

```
Ruijie(config)# end
```

14.2.3 配置 DHCP Snooping 支持 Bootp 用户绑定

缺省情况下，设备 DHCP Snooping 功能不支持 Bootp 用户绑定。配置该功能后，DHCP Snooping 开始对经过该设备的所有 Bootp 报文进行窥探，并将合法 Bootp 用户添加到 DHCP Snooping 静态绑定数据库中。

命令	说明
Ruijie**# configure terminal**	进入配置模式
Ruijie(config)**# [no] ip dhcp snooping bootp-bind**	打开和关闭 DHCP snooping 支持 Bootp 绑定

下面是打开设备的 DHCP Snooping 支持 Bootp 用户绑定功能：

```
Ruijie# configure terminal
Ruijie(config)# ip dhcp snooping bootp-bind
Ruijie(config)# end
```

14.2.4 配置源 MAC 检查功能

缺省情况下，源 MAC 检查功能为关闭状态。配置该功能后，DHCP Snooping 就会对 UNTRUST 口送上来的 DHCP 请求报文进行一致性检查。如果链路层头部源 MAC 字段和 DHCP 报文的 CLIENT MAC 字段相同，检查通过，则报文正常处理；否则检查失败，丢弃该报文。

命令	说明
Ruijie**# configure terminal**	进入配置模式
Ruijie(config)**# [no] ip dhcp snooping verify mac-address**	打开和关闭源 MAC 检查功能

下面是打开源 MAC 检查功能：

```
Ruijie# configure terminal
Ruijie(config)# ip dhcp snooping verify mac-address
Ruijie(config)# end
```

14.2.5 配置 DHCP snooping information option

缺省情况下，此功能关闭。配置该功能后，DHCP Snooping 在进行报文转发时，会在所有 DHCP 请求报文中添加 option82 选项；同时将所有 DHCP 响应报文中的 option82 选项删除。

命令	说明
Ruijie# **configure terminal**	进入配置模式
Ruijie(config)# **[no] ip dhcp snooping Information option**	打开和关闭 DHCP snooping information option 功能

下面是配置打开 **DHCP Snooping information option** 功能：

```
Ruijie# configure terminal
Ruijie(config)# ip dhcp snooping information option
Ruijie(config)# end
```

注意：

配置该功能后，在同一设备上配置的 DHCP relay option82 功能就会失效。

14.2.6 配置定时将 DHCP Snooping 数据库写入 Flash 文件

缺省情况下，此功能关闭。为了防止设备重新启动后，DHCP Snooping 数据库中的动态用户信息丢失，DHCP Snooping 提供定时将数据库中所有动态用户信息写入 Flash 文件的功能。

命令	说明
Ruijie# **configure terminal**	进入配置模式
Ruijie(config)# **[no] ip dhcp snooping database write-delay** [*time*]	配置定时写 Flash 文件的时间间隔；*time*：600—86400(s)，缺省为 0，不定时写。

下面是配置 DHCP Snooping 定时写 Flash 文件的时间间隔为 3600s：

```
Ruijie# configure terminal
Ruijie(config)# ip dhcp snooping database write-delay 3600
Ruijie(config)# end
```

注意：

由于不停擦写 flash，会造成 flash 的使用寿命缩短。所以在设置延迟写 flash 时间时需要注意，设置时间较短有利于设备信息更有效的保存；设置时间较长能够延长 flash 的使用寿命。

14.2.7 配置手动将 DHCP Snooping 数据库写入 Flash 文件

管理员在重新启动设备前，可以手动将当前的 DHCP Snooping 用户绑定数据库写入 Flash 文件，保证设备重新启动，与之前状态一致。

命令	说明
Ruijie**# configure terminal**	进入配置模式
Ruijie(config)**# ip dhcp snooping database write-to-flash**	手动将 DHCP snooping 数据库实时写入 flash 文件

下面是配置实时将 DHCP Snooping 数据库写入 Flash 文件：

```
Ruijie# configure terminal
Ruijie(config)# ip dhcp snooping database write-to-flash
Ruijie(config)# end
```

14.2.8 配置端口为 suppression 状态

缺省情况下，该功能关闭。通过配置该命令，将一个端口设置为 suppression 状态，可拒绝该端口下所有 DHCP 请求报文。

命令	说明
Ruijie**# configure terminal**	进入配置模式
Ruijie(config)**# interface***interface-id*	进入接口配置模式
Ruijie(config-if)**# [no] ip dhcp snooping suppression**	配置端口为 suppression 状态

下面是配置端口 **fastethernet** *0/2* 为 suppression 状态：

```
Ruijie# configure terminal
Ruijie(config)# interface fastethernet 0/2
Ruijie(config-if)# ip dhcp snooping suppression
Ruijie(config-if)# end
```

14.2.9 配置端口为 TRUST 口

缺省情况下，端口状态为 UNTRUST。通过配置该命令，将一个端口为配置为 TRUST 口，连接合法 DHCP 服务器。DHCP Snooping 功能仅转发 TRUST 口的 DHCP 响应报文，丢弃 UNTRUST 口的 DHCP 响应报文。

命令	说明
Ruijie**# configure terminal**	进入配置模式
Ruijie(config)**# interface** *interface-id*	进入接口配置模式

Ruijie(config-if)**# [no] ip dhcp snooping trust**	配置端口为 TRUST 口

下面是配置端口 **fastethernet** *0/1* 为 TRUST 口：

```
Ruijie# configure terminal
Ruijie(config)# interface fastethernet 0/1
Ruijie(config-if)# ip dhcp snooping trust
Ruijie(config-if)# end
```

14.2.10 配置端口接收 DHCP 报文的速率

缺省情况下，端口上接收的 DHCP 报文的速率没有限制。通过配置该命令，可以设置对应端口上接收 DHCP 报文的速率：

命令	说明
Ruijie**# configure terminal**	进入配置模式
Ruijie(config)**# interface fastethernet** *0/1*	进入接口配置模式
Ruijie(config-if)**# [no] ip dhcp snooping limitrate** *rate-value*	设置端口接收 DHCP 报文的速率

下边是配置端口 **fastethernet** *0/1* 接收 DHCP 报文的速率限制为 100pps：

```
Ruijie# configure terminal
Ruijie(config)# interface fastethernet 0/1
Ruijie(config-if)# ip dhcp snooping limit rate 100
Ruijie(config-if)# end
Ruijie# show ip dhcp snooping
Switch DHCP snooping status : ENABLE
Verification of hwaddr field status : DISABLE
DHCP snooping database write-delay time: 0 seconds
DHCP snooping option 82 status: ENABLE
DHCP snooping Support Bootp bind status: ENABLE
Interface                Trusted    Rate limit (pps)
-----------------------  -------    ------------
FastEthernet0/1           NO         100
```

14.2.11 清空 DHCP Snooping 数据库中所有动态用户信息

此命令用于删除 DHCP Snooping 用户绑定数据库中所有动态用户信息。

命令	说明
Ruijie**# clear ip dhcp snooping binding**	清空当前数据库中所有动态用户的信息

下边的是手动清空当前数据库中所有动态用户的信息：

```
Ruijie# clear ip dhcp snooping binding
```

14.3 DHCP Snooping 其它配置

14.3.1 显示 DHCP Snooping 配置信息

此命令用于显示 DHCP Snooping 功能当前的配置情况：

命令	说明
Ruijie# **show ip dhcp snooping**	显示 dhcp snooping 的相关配置信息

例：

```
Ruijie# show ip dhcp snooping
Switch DHCP snooping status : ENABLE
Verification of hwaddr field status : DISABLE
DHCP snooping database write-delay time: 0 seconds
DHCP snooping option 82 status: ENABLE
DHCP snooping Support Bootp bind status: ENABLE
Interface                Trusted    Rate limit (pps)
-----------------------  -------    ------------
FastEthernet0/1           NO         100
```

14.3.2 显示 DHCP Snooping 数据库信息

此命令用于显示 DHCP Snooping 数据库中的用户信息：

命令	说明
Ruijie# **show ip dhcp snooping binding**	查看 DHCP Snooping 绑定数据库中的用户信息

例如：

```
Ruijie# show ip dhcp snooping binding
Total number of bindings: 1
MacAddress     IpAddress  Lease  Type  VLAN  Interface
-------------  -------------- ---------- -------------
00d0.f801.0101 192.168.1.1   -   static 1  fastethernet 0/1
```

14.3.3 DHCP Snooping 调试开关

此命令用于打开 DHCP Snooping 的调试信息开关。

命令	说明
Ruijie# **debug ip dhcp snooping** {**event** | **packet**}	打开/关闭 DHCP Snooping 调试信息开关

例如：

```
Ruijie# debug ip dhcp snooping event
Ruijie# debug ip dhcp snooping packet
```

14.4 DHCP Snooping 典型配置用例

14.4.1 拓扑图

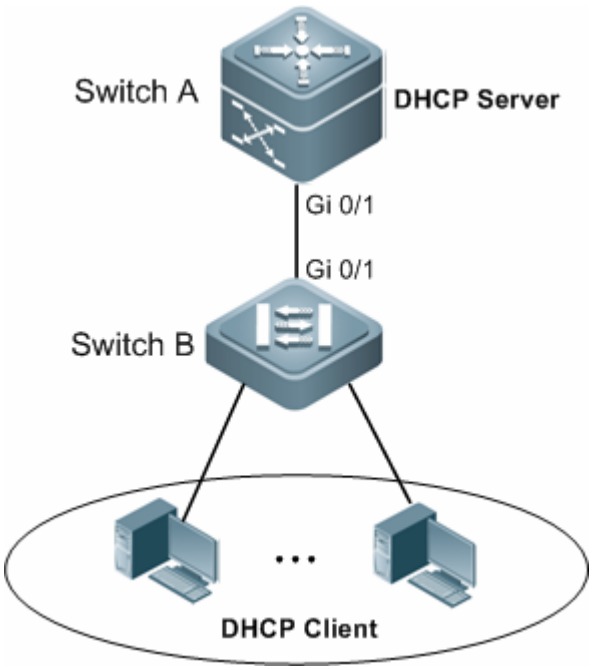


图 10

14.4.2 应用需求

1. DHCP 客户端用户通过合法 DHCP 服务器动态获取 IP 地址。
2. 避免其他用户私设 DHCP 服务器。

14.4.3 配置要点

1. 在接入设备（本例为 Switch B）上开启 DHCP Snooping 功能，将上链口（本例为端口 Gi 0/1）设置为信任口。

14.4.4 配置步骤

- 配置 Switch B

第一步，打开 DHCP Snooping 功能。

```
Ruijie#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Ruijie(config)#ip dhcp snooping
```

第二步，配置上链口为信任口。

```
Ruijie(config)#interface gigabitEthernet 0/1
Ruijie(config-if-GigabitEthernet 0/1)#ip dhcp snooping trust
```

14.4.5 配置验证

第一步，确认 Switch B 的配置，关注点：是否开启 DHCP Snooping 功能、配置的 DHCP Snooping 信任口是否为上链口。

```
Ruijie#show running-config
!
ip dhcp snooping
!
interface GigabitEthernet 0/1
 ip dhcp snooping trust
```

第二步，查看 Switch B 的 DHCP Snooping 配置情况，关注点为信任口是否正确。

```
Ruijie#show ip dhcp snooping
Switch DHCP snooping  status                  :   ENABLE
DHCP snooping  Verification of hwaddr status    :   DISABLE
DHCP snooping database write-delay time        :   0 seconds
DHCP snooping option 82 status                :   DISABLE
DHCP snooping Support bootp bind status        :   DISABLE
Interface                    Trusted      Rate limit (pps)
-------------------------      -------      ----------------
GigabitEthernet 0/1            YES          unlimited
```

第三步，查看 DHCP Snooping 地址绑定数据库信息（用户的 MAC 地址、动态分配的 IP 地址、地址租期、对应的 VLAN 和端口号等）。

```
Ruijie#show ip dhcp snooping binding
Total number of bindings: 1
MacAddress        IpAddress      Lease(sec)   Type         VLAN
Interface
------------------ --------------- ------------ -------------
----- --------------------
0013.2049.9014    172.16.1.2     86207       dhcp-snooping 1
GigabitEthernet 0/11
```

15 IGMP Snooping 配置

15.1 概述

15.1.1 理解 IGMP Snooping 的工作原理

IGMP Snooping 是 Internet Group Management Protocol（组播侦听者发现协议窥探）的简称。它是运行在 VLAN 上的 IP 组播约束机制，用于管理和控制 IP 组播流在 VLAN 内的转发，属于二层组播功能。下面描述的 IGMP Snooping 功能，都是指在 VLAN 内进行的，相关的端口也是指 VLAN 内部的成员口。

运行 IGMP Snooping 的设备通过对收到的 IGMP 报文进行分析，为端口和组播地址建立起映射关系，并根据这样的映射关系转发 IP 组播数据报文。如图 1-1 所示，当交换机没有运行 IGMP Snooping 时，IP 组播数据报文在 VLAN 内被广播；当交换机运行了 IGMP Snooping 后，已知 IP 组播组的组播数据报文不会在 VLAN 内被广播，而是发给指定的接收者。

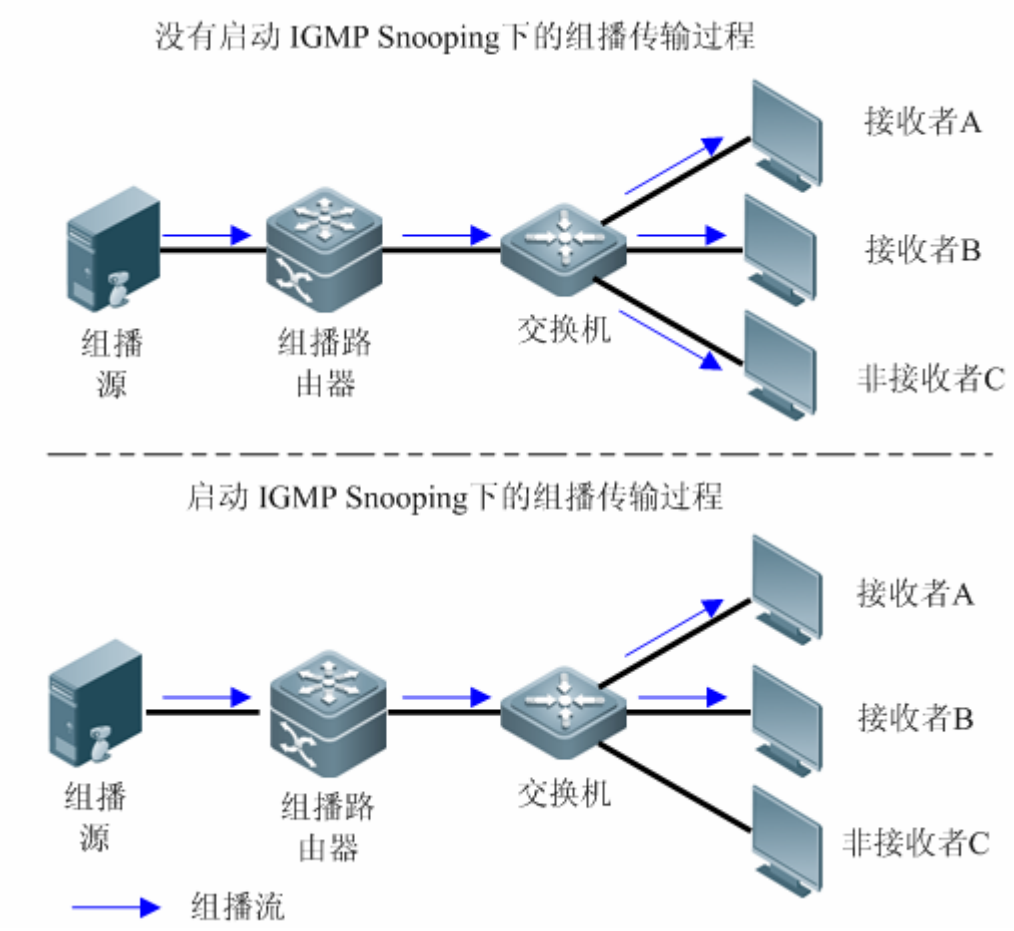


图 1-1 VLAN 上运行 IGMP Snooping 前后的对比

15.1.2 理解 IGMP Snooping 的两类端口

如图 1-2 所示，Router 连接组播源，在 Switch A 运行 IGMP Snooping，Host A 和 Host C 为接收者主机（即 IP 组播组成员）。

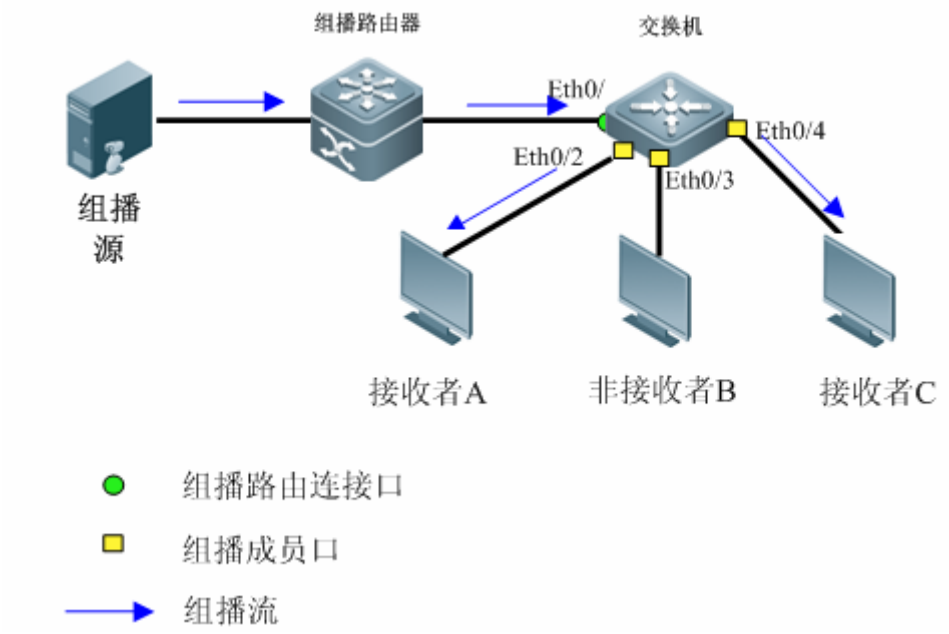


图 1-2 IGMP Snooping 的两类端口

路由连接口（Multicast Router Port）：交换机上连接组播路由器（三层组播设备），如 Switch A 的 Eth0/1 端口。交换机将本设备上的所有路由连接口（包括动态和静态端口）都记录在路由连接口列表中。路由连接口缺省情况下是对应 VLAN 内组播数据的接收者，也会被添加到 IGMP Snooping 转发表中。

成员端口（Member Port）：IP 组播组成员端口的简称，又称侦听者端口（Listener Port），表示交换机上连接 IP 组播组成员侧的端口，如 Switch A 的 Eth0/2、Eth0/3 和 Eth0/4 端口。交换机将本设备上的所有成员端口（包括动态和静态端口）都记录在 IGMP Snooping 转发表中。

15.1.3 理解动态端口的老化定时器

表 1-1 IGMP Snooping 动态端口的老化定时器

类型	描述	触发定时器启动的事件	超时后交换机的动作
动态路由连接口的老化定时器	交换机为每个动态路由连接口都启动一个定时器，其超时时间就是动态路由连接口的老化时间	收到 IGMP 普遍组查询报文或 IP PIM Hello 报文	将该端口从路由连接口列表中删除
动态成员	交换机为该端口启	收到IGMP的查询报	将该端口从 IGMP

端口的老化定时器	动一个定时器，其超时时间就是动态成员端口老化时间	文	Snooping 组播组的转发表中删除

15.1.4 理解 IGMP Snooping 的工作机制

15.1.4.1 普遍组查询和特定组查询

IGMP 查询者定期向本地网段内的所有主机与路由器（地址为 224.0.0.1）发送普通组查询报文，以查询该网段有哪些 IP 组播组的成员。在收到 IGMP 普遍组查询报文时，交换机将查询报文向本 VLAN 内的所有端口转发出去，并对该报文的接收端口做如下处理：

- 如果该端口已经在路由连接口列表中，则重置其老化定时器。
- 如果该端口不在路由连接口列表中，则将其添加到路由连接口列表中，并启动其老化定时器。
- 在收到 IGMP 的普遍组查询报文后，组播设备会对所有的成员端口启动各自的老化定时器，定时器时间为配置的对 IGMP 查询报文的最长响应时间，当定时器的值减为 0 时，则认为该端口不再有成员接收组播流，组播设备就会把该端口从 IGMP Snooping 的转发表中删除。

在收到 IGMP 的特定组查询报文后，组播设备会对该特定组的所有成员端口启动老化定时器，定时器时间为最长响应时间，当定时器超时了还没有接收到主机的应答，则认为该端口不再有成员接收组播流，组播设备就会把该端口从 IGMP Snooping 的转发表中删除。

对于 IGMP 的特定组源查询报文，不做定时器的更新处理。

15.1.4.2 报告成员关系

以下情况，主机会向 IGMP 查询者发送 IGMP 成员关系报告报文：

- 当 IP 组播组的成员主机收到 IGMP 查询(普遍组查询或特定组查询)报文后，会应答 IGMP 成员关系报告报文。
- 如果主机要加入某个 IP 组播组，它会主动向 IGMP 查询者发送 IGMP 成员关系报告报文以声明加入该 IP 组播组。

在收到 IGMP 成员关系报告报文时，交换机将其通过给 VLAN 内的所有路由连接口转发出去，从该报文中解析出主机要加入的 IP 组播组地址，并对该报文的接收端口做如下处理：

- 如果不存在该 IP 组播组所对应的转发表项，则创建转发表项，将该端口作为动态成员端口添加到出端口列表中，并启动其老化定时器；
- 如果已存在该 IP 组播组所对应的转发表项，但其出端口列表中不包含该端口，

则将该端口作为动态成员端口添加到出端口列表中，并启动其老化定时器；

- 如果已存在该 IP 组播组所对应的转发表项，且其出端口列表中已包含该动态成员端口，则重置其老化定时器。

15.1.4.3 离开组播组

当主机离开 IP 组播组时，会通过发送 IGMP 离开组报文，以通知组播路由器自己离开了某个 IP 组播组。当交换机从某动态成员端口上收到 IGMP 离开组报文时，将直接向路由连接口转发。

15.1.5 理解 IGMP Profiles

IGMP Profiles 实际上是一些组过滤器，它可以定义一系列的组播地址范围，并定义对这些组播地址的访问是 permit 或是 deny 动作，供后面的“SVGL 模式应用的组播地址范围”、“路由连接口过滤组播数据范围”、“IGMP Filtering 范围”各项功能使用。

15.1.6 理解 IGMP Snooping 的各种工作模式

DISABLE 模式：在该模式下，IGMP Snooping 不起作用，即二层组播不起作用，VLAN 内不“窥探”主机与路由设备之间的 IGMP 报文，组播帧在 VLAN 内被广播。

IVGL 工作模式(Independent VLAN Group Learning)：在该模式下，各 VLAN 间的组播流是相互独立的。主机只能朝与自己处于同一个 VLAN 的路由连接口请求接收组播流；交换机在接收到任何一个 VLAN 的组播数据流时，只能往相同 VLAN 内的成员端口转发。

15.2 配置 IGMP Snooping

我们将从以下几个章节描述如何配置 IGMP Snooping

配置功能		说明
配置 IGMP Snooping 基本功能	使能 IGMP Snooping	必选
	配置动态端口的老化定时器	可选
	配置对 IGMP 查询报文的最长响应时间	可选
配置 IGMP Snooping 端口功能	配置路由连接口	可选
	配置成员端口	可选
	配置端口快速离开	可选

	配置 IGMP 成员关系报告报文抑制	可选
配置端端口的 IP 组播组策略	配置 IP 组播组过滤器	可选

15.2.1 使能 IGMP Snooping

默认的在使能 IGMP Snooping 的时候，需要指定 IGMP Snooping 的工作模式，您可以指定 IVGL 模式。

注意：

另外，如果二层组播设备工作在 private vlan 模式下，则不支持 IGMP snooping。

15.2.2 配置 IVGL 模式

在全局模式下，按如下步骤打开并设置 IGMP Snooping 为 IVGL 模式：

命令	作用
Ruijie(config)# **ip igmp snooping ivgl**	打开并设置 IGMP Snooping 为 IVGL 模式，缺省情况下，IGMP Snooping 处于关闭状态。
Ruijie(config)# **show ip igmp snooping**	确认配置是否生效。
Ruijie(config)#**no ip igmp snooping**	全局关闭 IGMP Snooping 功能。

以下例子是打开并设置 IGMP Snooping 为 IVGL 模式：

```
Ruijie# configure terminal
Ruijie(config)# ip igmp snooping ivgl
Ruijie(config)# show ip igmp snooping
IGMP Snooping running mode: IVGL
SVGL vlan: 1
SVGL profile number: 0
Source port check: Disable
Source ip check: Disable
IGMP Fast-Leave: Disable
IGMP Report suppress: Disable
```

15.2.3 关闭 IGMP Snooping

在全局模式下，按如下步骤关闭 IGMP Snooping：

命令	作用
Ruijie(config)**# no ip igmp snooping**	关闭 IGMP Snooping，缺省情况下，IGMP Snooping 处于关闭状态。
Ruijie(config)**# show ip igmp snooping**	确认配置是否生效。

15.2.4 配置动态路由连接口的老化定时器

对于动态路由连接口，如果在其老化时间超时前没有收到 IGMP 普遍组查询报文或者 PIM Hello 报文，交换机将把该端口从路由器端口列表中删除。

在全局模式下，按如下步骤配置动态端口的老化定时器：

命令	作用
Ruijie(config)**# ip igmp snooping dyn-mr-aging-time** *time*	配置动态路由连接口老化时间， *time* ：<1-3600> 默认值为 300s。
Ruijie(config)**# no ip igmp snooping dyn-mr-aging-time**	恢复动态路由连接口的老化时间为默认值，默认值为 300s。

以下是配置动态路由连接口老化时间为 100s 的实例：

```
Ruijie# configure terminal
Ruijie(config)# ip igmp snooping dyn-mr-aging-time 100
```

15.2.5 配置对 IGMP 查询报文的最长响应时间

- 在收到 IGMP 的普遍组查询报文后，运行二层组播功能的设备会对所有的成员端口启动各自的老化定时器，定时器时间为最长响应时间，当定时器的值减为 0 时，则认为该端口不再有成员接收组播流，组播设备就会把该端口从 IGMP Snooping 的转发表中删除。
- 在收到 IGMP 的特定组查询报文后，组播设备会对该特定组的所有成员端口启动老化定时器，定时器时间为最长响应时间，当定时器的值减为 0 时，则认为该端口不再有成员接收组播流，组播设备就会把该端口从 IGMP Snooping

的转发表中删除。

- 对于 IGMP 的特定组源查询报文，不做定时器的更新处理。

命令	作用
Ruijie(config)# **ip igmp snooping query-max-response-time** *time*	配置 IGMP 普遍组查询报文的最长响应时间，范围为 1-65535，缺省值为 10s。
Ruijie(config)# **no ip igmp snooping query-max-response-time**	恢复 IGMP 普遍组查询报文的最长响应时间为缺省值，缺省值为 10s。

以下是配置 IGMP 查询报文的最长响应时间为 15s 的实例：

```
Ruijie# configure terminal
Ruijie(config)# ip igmp snooping query-max-response-time 15
```

15.2.6 配置路由连接口

缺省的情况下，VLAN 内将进行动态路由连接口的学习，您可以配置关闭组播设备动态路由连接口的学习的功能，用相应的 **no** 选项命令关闭动态学习，并清空所有动态学习到的路由连接口。

另外可以通过命令将交换机上的端口配置为静态路由连接口。

在全局模式下，按如下步骤配置路由连接口：

命令	作用
Ruijie(config)# **ip igmp snooping vlan** *vlan-id* **mrouter interface** *interface-id*	设置接口为静态路由连接口。
Ruijie(config)# **no ip igmp snooping vlan** *vlan-id* **mrouter interface** *interface-id*	取消接口为静态路由连接口。
Ruijie(config)# **ip igmp snooping vlan** *vlan-id* **mrouter learn pim-dvmrp**	设置 VLAN 上路由连接口的动态学习功能；缺省是启动动态学习功能。
Ruijie(config)# **no ip igmp snooping vlan** *vlan-id* **mrouter learn pim-dvmrp**	关闭动态学习路由连接口的功能，并清空所有动态学习到的路由连接口。

以下例子是设置以太网接口 1/1 为 VLAN1 的静态路由连接口，并启动 VLAN1 自动学习路由连接口功能：

```
Ruijie# configure terminal
Ruijie(config)# ip igmp snooping vlan 1 mrouter interface gigabitEthernet 1/1
Ruijie(config)# ip igmp snooping vlan 1 mrouter learn pim-dvmrp
Ruijie(config)# end
```

```
Ruijie# show ip igmp snooping mrouter
Vlan    Interface        State
----    ---------        ------
1   GigabitEthernet 1/1    static
1   GigabitEthernet 0/2    dynamic
Ruijie# show ip igmp snooping mrouter learn
Vlan    learn method
----    ------------------
1       pim-dvmrp
```

15.2.7 配置静态成员端口

如果某端口所连接的主机需要固定接收发往某 IP 组播组的 IP 组播数据，可以配置该端口静态加入该 IP 组播组，成为静态成员端口。

在全局模式下，按如下步骤配置 IGMP Snooping 的静态成员端口：

命令	作用
Ruijie(config)**# ip igmp snooping ivgl**	打开并设置 IGMP Snooping 为 IVGL 模式。
Ruijie(config)**# ip igmp snooping vlan** *vlan-id* **static** *ip-addr* **interface** *interface-id*	静态配置一个端口接收某个组播流。 • *vlan-id*：组播流的 vid • *ip-addr*：组播地址 • *interface-id*： 端口号
Ruijie(config)**# no ip igmp snooping vlan** *vlan-id* **static** *ip-addr* **interface** *interface-id*	删除一个静态成员端口。 • *vlan-id*：组播流的 vid • *ip-addr*：组播地址 • *interface-id*： 端口号

您可以用 **no ip igmp snooping vlan** *vlan-id* **static** *ip-addr* **interface** *interface-id* 删除组播成员的静态配置。

以下为配置 IGMP snooping 静态成员端口的实例：

```
Ruijie# configure terminal
Ruijie(config)# ip igmp snooping vlan 1 static 233.3.3.4
interface GigabitEthernet 0/7
Ruijie(config)# end
Ruijie(config)# show ip igmp snooping gda
Abbr: M - mrouter
      D - dynamic
      S - static
VLAN  Address              Member ports
---- -------------   ----------------------------
```

```
1     233.3.3.4         GigabitEthernet 0/7(S)
```

15.2.8 配置端口快速离开

端口快速离开是指当交换机从某端口收到主机发送的离开某 IP 组播组的 IGMP 离开组报文时，直接把该端口从对应转发表项的出端口列表中删除。在交换机上，如果端口下只连接有一个接收者，则可以通过使能端口快速离开功能以节约带宽和资源。该功能运用在相关端口下只存在一个点播者的情况。

在全局模式下，按如下步骤设置 **IGMP snooping** 的端口快速离开：

命令	作用
Ruijie(config)**# ip igmp snooping fast-leave enable**	打开二层组播的快速离开功能，缺省情况下是关闭的。
Ruijie(config)**# no ip igmp snooping fast-leave enable**	关闭端口快速离开功能。

使用 **no ip igmp snooping fast–leave enalbe** 命令关闭端口快速离开功能。

以下例子是打开端口快速离开功能：

```
Ruijie# configure terminal
Ruijie(config)# ip igmp snooping fast–leave enalbe
```

15.2.9 配置 IGMP Snooping 成员关系报告报文的响应抑制

当 VLAN 内的相应端口收到来自某 IP 组播组成员的 IGMP 成员关系报告报文时，会将该报文转发给路由连接口。这样，当相同 VLAN 上的其它端口也将收到属于相同 IP 组播组的 Report 报文时，路由连接口上会收到多份相同 IGMP 成员关系报告报文。

当使能了 IGMP 成员关系报告报文抑制功能后，在一个查询间隔内同一 VLAN 内只会把收到的某 IP 组播组内的第一个 IGMP 成员关系报告报文转发给路由连接口，而不继续向三层设备转发来自同一组播组的其它 IGMP 成员关系报告报文，这样可以减少网络中的报文数量。

在全局模式下，按如下步骤设置 IGMP snooping suppression：

命令	作用
Ruijie(config)**# ip igmp snooping suppression enable**	打开二层组播 Report 报文抑制功能，缺省情况下是不打开响应抑制功能的。
Ruijie(config)**# no ip igmp snooping suppression enable**	关闭二层组播 Report 报文抑制功能。

以下例子是打开 Suppression 功能：

```
Ruijie# configure terminal
Ruijie(config)# ip igmp snooping suppression enable
```

15.2.10 配置 IGMP Profiles

IGMP Profiles 实际上是一些组过滤器，供下面的“SVGL 模式应用的组播地址范围”、“路由连接口过滤组播数据范围”、“IGMP Filtering 范围”各项功能使用。

在全局模式下，按如下步骤设置一个 Profile

命令	作用
Ruijie(config)**# ip igmp profile** *profile-number*	进入 IGMP Profile 模式，分配一个数字以供标识，该数字范围为 1－1024，缺省情况下没有配置任何一条 profile。
Ruijie (config-profile)**# permit | deny**	(可选）配置是要 permit 还是 deny 这一批组播地址范围，缺省值是 deny。这个行为表示：允许/禁止以下 range 内的这些组播地址，并禁止/允许其它的组播地址。
Ruijie(config-profile)**# range** *low-address high_address*	添加组播地址范围，该值即可以是一个单个的 IP 组地址也可以是一个组地址的区间（前面的为低 IP 组地址，后面为高 IP 组地址），同时还可以配置多个 range 范围。
Ruijie(config)**# end**	退回到特权模式。

如果要删除其中一个 IGMP profile，可以用 **no ip igmp profile profile number** 来执行。如果要删除 profile 里的一个 range，可以用 **no range ip multicast address** 来执行。

以下有个例子是表示 Profile 的配置过程：

```
Ruijie(config)# ip igmp profile 1
Ruijie(config-profile)# permit
Ruijie(config-profile)# range 224.0.1.0 239.255.255.255
Ruijie(config-profile)# end
Ruijie# show ip igmp profile 1
Profile    1
       Permit
       range 224.0.1.0,   239.255.255.255
```

按以上配置，这个 IGMP Profile 的规则就是 permit 224.0.1.0 到 239.255.255.255 的组播地址，其他的组播地址都被 deny。

15.2.11 配置 IGMP Filtering

在某些情况下，您可能需要控制某个端口只能接收一批特定的组播数据流、控制该端口下最多允许动态加入多少组。IGMP Filtering 满足了这种需求。

您可以把某一个 IGMP Profile 应用在一个端口下，如果该端口收到 IGMP Report 报文，则二层组播设备就会查找这个端口所要加入的组播地址是否在 IGMP Profile 允许范围之内。若是，则允许加入，之后才进行后续处理。

您也可以在一个端口下配置最多允许加入的组的个数，超过范围，二层组播设备也不再接收、处理 IGMP Report 报文。

在全局模式下，按如下步骤配置 IGMP Filtering：

命令	作用
Ruijie(config)**# interface** *interface-id*	进入配置接口。
Ruijie(config-if)**# ip igmp snooping filter** *profile-number*	（可选）应用 Profile 于该端口，*profile number* 范围为 1- 1024。缺省情况下，一个端口不关联任何的 profile。
Ruijie(config-if)**# no ip igmp snooping filter**	（可选）删除接口上关联的 profile，接口上将允许所有组通过。
Ruijie(config-if)**# ip igmp snooping max-groups** *number*	（可选）允许最多几个组动态的加入该端口，参数范围为 0 – 1024。缺省情况下，无限制。
Ruijie(config-if)**# no ip igmp snooping max-groups**	（可选）恢复接口上允许的最大组个数为缺省值。缺省情况下，组个数无限制。

以下是配置 IGMP Filtering 的实例：

```
Ruijie# configure terminal
Ruijie(config)# interface fastEthernet 0/1
Ruijie(config-if)# ip igmp snooping filter 1
Ruijie(config-if)# ip igmp snooping max-groups 1000
Ruijie (config-if)#end
Ruijie #show ip igmp snooping interface fastEthernet 0/1
Interface            Filter profile number       max-group
---------------      ---------------------       ---------
FastEthernet 0/1            1                        1000
```

15.2.12查看 IGMP Snooping 信息

我们提供的可查看的相关 IGMP snooping 的信息如下：

- 查看当前模式
- 查看路由连接口信息
- 查看动态转发表
- 查看源端口检查状态
- 查看 IGMP Profile
- 查看 IGMP Filtering

15.2.13查看当前模式

在特权模式下使用如下命令查看 IGMP Snooping 当前的工作模式及全局配置：

命令	作用
Ruijie**# show ip igmp snooping**	查看 IGMP Snooping 当前的工作模式及全局配置。

以下例子使用 **show ip igmp snooping** 命令查看 IGMP Snooping 配置信息：

```
Ruijie# show ip igmp snooping
IGMP-snooping mode      : IVGL
SVGL vlan-id           : 1
SVGL profile number    : 0
Source port check: Disable
Source ip check: Disable
IGMP Fast-Leave: Disable
IGMP Report suppress: Disable
```

15.2.14 查看和清除 IGMP Snooping 的统计值

在特权模式下使用如下命令查看和清除 IGMP Snooping 的统计值信息：

命令	作用
Ruijie**# show ip igmp snooping statistics** [**vlan** *vlan-id*]	查看 IGMP Snooping 的统计信息。
Ruijie**# clear ip igmp snooping statistics**	清除 IGMP Snooping 的统计信息

以下例子使用 **show ip igmp snooping statistics** 命令查看 IGMP Snooping 的路

由连接口信息：

```
Ruijie# show ip igmp snooping statistics
Current number of Gda-table entries: 1
Configured Statistics database limit: 1024
Current number of IGMP Query packet received : 1957
Current number of IGMPv1/v2 Report packet received: 5
Current number of IGMPv3 Report packet received: 4
Current number of IGMP Leave packet received: 1

 GROUP  Interface   Last      Last       Report  Leave
                 report time reporter    pkts    pkts
------   -------- ---------- ---------- ---- -- -------- ----
233.3.3.3 gi1/1   00:02:40    1.1.1.1     3       1
```

15.2.15 查看路由连接口信息

在特权模式下使用如下命令查看 IGMP Snooping 的路由连接口信息：

命令	作用
Ruijie**# show ip igmp snooping mrouter**	查看 IGMP Snooping 的路由连接口信息。

以下例子使用 **show ip igmp snooping** 命令查看 IGMP Snooping 的路由连接口信息：

```
Ruijie# show ip igmp snooping mrouter
Vlan    Interface       State     IGMP profile number
----     --------         -------     -------------------
1   GigabitEthernet 0/7    static        1
1   GigabitEthernet 0/12   dynamic       0
```

15.2.16 查看转发表

在特权模式下使用如下命令查看各端口在组播组中的转发规则，即查看组目的地址 GDA（Group Destination Address）表：

命令	作用
Ruijie**# show ip igmp snooping gda-table**	查看各端口在组播组中的转发规则表。

以下例子是查看 GDA 表的各组播组信息以及一个组播组的所有成员端口的信息：

```
Ruijie# show ip igmp snooping gda-table
Abbr: M - mrouter
```

```
     D - dynamic
     S - static
VLAN  Address            Member ports
------------------------------------------------------------
1      233.3.3.3          GigabitEthernet 0/7(S)
```

15.2.17 清除转发表节点的统计信息

在特权模式下使用如下命令查看各端口在组播组中的转发规则，即查看组目的地址 GDA（Group Destination Address）表：

命令	作用
Ruijie**# clear ip igmp snooping statistics**	清除转发表中表项节点的动态统计信息。

以下例子是清除 GDA 表的动态组播组统计信息：

```
Ruijie# clear ip igmp snooping statistics
```

15.2.18 查看源端口检查状态

在特权模式下使用如下命令查看 IGMP Snooping 当前的源端口检查状态：

命令	作用
Ruijie**# show ip igmp snooping**	查看 IGMP Snooping 当前的工作模式及全局配置。

以下为查看源端口检查状态的信息：

```
Ruijie(config)# show ip igmp snooping
IGMP-snooping mode        :IVGL
SVGL vlan-id             :1
SVGL profile number      :0
Source check port        :Disable
IGMP Fast-Leave: Disable
IGMP Report suppress: Disable
```

15.2.19 查看 IGMP Profile

在特权模式下使用如下命令查看 IGMP Profile 信息：

命令	作用
Ruijie**# show ip igmp profile** *profile-number*	查看 IGMP Profile 信息。

以下为查看 IGMP Profile 的信息：

```
Ruijie# show ip igmp profile 1
Profile    1
       Permit
       range 224.0.1.0,   239.255.255.255
```

15.2.20 查看 IGMP Filtering

在特权模式下使用如下命令查看 IGMP Filtering 的配置信息：

命令	作用
Ruijie**# show ip igmp snooping interface** *interface-id*	查看 IGMP Filtering 的配置信息。

以下为查看 IGMP Filtering 的信息：

```
Ruijie# show ip igmp snooping interface GigabitEthernet 0/7
Interface         Filter Profile number     max-groups
----------       ----------------------    -----------
GigabitEthernet 0/7          1              4294967294
```

16 MSTP 配置

16.1 MSTP 概述

16.1.1 STP、RSTP

16.1.1.1 STP、RSTP 总述

本设备既支持 STP 协议，也支持 RSTP 协议，遵循 IEEE 802.1D 和 IEEE 802.1w 标准。

STP 协议是用来避免链路环路产生的广播风暴、并提供链路冗余备份的协议。

对二层以太网来说，两个 LAN 间只能有一条活动着的通路，否则就会产生广播风暴。但是为了加强一个局域网的可靠性，建立冗余链路又是必要的，其中的一些通路必须处于备份状态，如果当网络发生故障，另一条链路失效时，冗余链路就必须被提升为活动状态。手工控制这样的过程显然是一项非常艰苦的工作，STP 协议就自动地完成这项工作。它能使一个局域网中的设备起以下作用：

- 发现并启动局域网的一个最佳树型拓朴结构。
- 发现故障并随之进行恢复，自动更新网络拓朴结构，使在任何时候都选择了可能的最佳树型结构。

局域网的拓朴结构是根据管理员设置的一组网桥配置参数自动进行计算的。使用这些参数能够生成最好的一棵拓朴树。只有配置得当，才能得到最佳的方案。

RSTP 协议完全向下兼容 802.1D STP 协议，除了和传统的 STP 协议一样具有避免回路、提供冗余链路的功能外，最主要的特点就是“快”。如果一个局域网内的网桥都支持 RSTP 协议且管理员配置得当，一旦网络拓朴改变而要重新生成拓朴树只需要不超过 1 秒的时间（传统的 STP 需要大约 50 秒）。

注意：

S3760 系列交换机，若交换机的 buffer 控制处于 fc 模式，在拥塞发生时，可能导致 STP，MSTP 协议不能正常运行。建议您在运行 STP，MSTP 协议时，使交换机的 buffer 控制处于 qos 模式。
关于交换机的 buffer 控制，请参见 QOS 配置的交换机 buffer 控制章节的说明。

16.1.1.2 Bridge Protocol Data Units(简写为 BPDU)：

要生成一个稳定的树型拓朴网络需要依靠以下元素：

- 每个网桥拥有的唯一的桥 ID（Bridge ID），由桥优先级和 Mac 地址组合而成。
- 网桥到根桥的路径花费（Root Path Cost），以下简称根路径花费。
- 每个端口 ID（Port ID），由端口优先级和端口号组合而成。

网桥之间通过交换 BPDU（Bridge Protocol Data Units，网桥协议数据单元）帧来获得建立最佳树形拓朴结构所需要的信息。这些帧以组播地址 01-80-C2-00-00-00（十六进制）为目的地址。

每个 BPDU 由以下这些要素组成：

- Root Bridge ID（本网桥所认为的根桥 ID）。
- Root Path Cost（本网桥的根路径花费）。
- Bridge ID（本网桥的桥 ID）。
- Message Age（报文已存活的时间）
- Port ID（发送该报文端口的 ID）。

Forward-Delay Time、Hello Time、Max-Age Time 三个协议规定的时间参数。

其他一些诸如表示发现网络拓朴变化、本端口状态的标志位。

当网桥的一个端口收到高优先级的 BPDU（更小的 Bridge ID，更小的 Root Path Cost 等），就在该端口保存这些信息，同时向所有端口更新并传播这些信息。如果收到比自己低优先级的 BPDU，网桥就丢弃该信息。

这样的机制就使高优先级的信息在整个网络中传播开，BPDU 的交流就有了下面的结果：

- 网络中选择了一个网桥为根桥（Root Bridge）。
- 除根桥外的每个网桥都有一个根口（Root Port），即提供最短路径到 Root Bridge 的端口。
- 每个网桥都计算出了到根桥（Root Bridge）的最短路径。
- 每个 LAN 都有了指派网桥（Designated Bridge），位于该 LAN 与根桥之间的最短路径中。指派网桥和 LAN 相连的端口称为指派端口（Designated Port）。
- 根口（Root port）和指派端口（Designated Port）进入 Forwarding 状态。
- 其他不在生成树中的端口就处于 Discarding 状态

16.1.1.3 Bridge ID

按 IEEE 802.1W 标准规定，每个网桥都要有单一的网桥标识（Bridge ID），生成

树算法中就是以它为标准来选出根桥（Root Bridge）的。Bridge ID 由 8 个字节组成，后 6 个字节为该网桥的 mac 地址，前 2 个字节如果下表所示，前 4 bit 表示优先级（Priority），后 12 bit 表示 System ID，为以后扩展协议而用，在 RSTP 中该值为 0，因此给网桥配置优先级就要是 4096 的倍数。

	Priority value				System ID											
Bit 位	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1
值	32768	16384	8192	4096	2048	1024	512	256	128	64	32	16	8	4	2	1

16.1.1.4 Spanning-Tree Timers（生成树的定时器）

以下描述影响到整个生成树性能的三个定时器。

- Hello timer：定时发送 BPDU 报文的时间间隔。
- Forward-Delay timer：端口状态改变的时间间隔。当 RSTP 协议以兼容 STP 协议模式运行时，端口从 Listening 转变向 Learning，或者从 Learning 转向 Forwarding 状态的时间间隔。
- Max-Age timer：BPDU 报文消息生存的最长时间。当超出这个时间，报文消息将被丢弃。

16.1.1.5 Port Roles and Port States

每个端口都在网络中有扮演一个角色（Port Role），用来体现在网络拓朴中的不同作用。

- Root port：提供最短路径到根桥（Root Bridge）的端口。
- Designated port：每个 LAN 的通过该口连接到根桥。
- Alternate port：根口的替换口，一旦根口失效，该口就立该变为根口。
- Backup port：Designated Port 的备份口，当一个网桥有两个端口都连在一个 LAN 上，那么高优先级的端口为 Designated Port，低优先级的端口为 Backup Port。
- Disable port：当前不处于活动状态的口，即 Operation State 为 Down 的端口都被分配了这个角色。

以下为各个端口角色的示意图 1、2、3：

R = Root Port　D = Designated Port　A = Alternate Port　B = Backup Port

在没有特别说明情况下，端口优先级从左到右递减

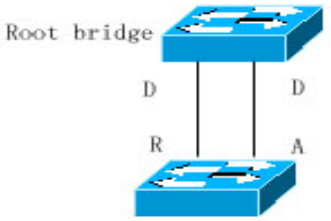


图 1

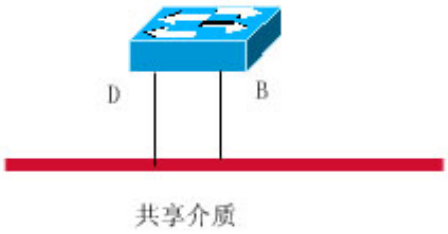


图 2

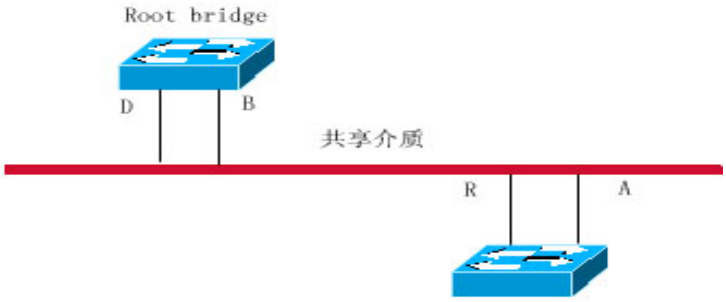


图 3

每个端口有三个状态（Port State）来表示是否转发数据包，从而控制着整个生成树拓朴结构。

- Discarding：既不对收到的帧进行转发，也不进行源 Mac 地址学习。
- Learning：不对收到的帧进行转发，但进行源 Mac 地址学习，这是个过渡状态。
- Forwarding：既对收到的帧进行转发，也进行源 Mac 地址的学习。

对一个已经稳定的网络拓朴，只有 Root Port 和 Designated Port 才会进入 Forwarding 状态，其它端口都只能处于 Discarding 状态。

16.1.1.6 网络拓朴树的生成（典型应用方案）

现在就可以说明 STP、RSTP 协议是如何把杂乱的网络拓朴生成一个树型结构了。如下图 4 所示，假设 Switch A、B、C 的 bridge ID 是递增的，即 Switch A 的优先级最高。A 与 B 间是千兆链路，A 和 C 间为百兆链路，B 和 C 间为十兆链路。Switch A 做为该网络的骨干设备，对 Switch B 和 Switch C 都做了链路冗余，显然，如果让这些链路都生效是会产生广播风暴的。

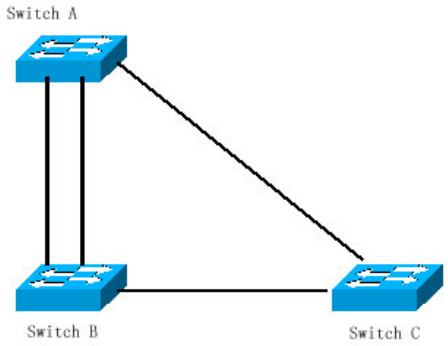


图 4

而如果这三台 Switch 都打开了 Spanning Tree 协议，它们通过交换 BPDU 选出根桥（Root Bridge）为 Switch A。Switch B 发现有两个端口都连在 Switch A 上，它就选出优先级最高的端口为 Root Port，另一个端口就被选为 Alternate Port。而 Switch C 发现它既可以通过 B 到 A，也可以直接到 A，但由于设备通过计算发现：就算通过 B 到 A 的链路花费（Path Cost）也比直接到 A 的低（各种链路对应的链路花费请查表），于是 Switch C 就选择了与 B 相连的端口为 Root port，与 A 相连的端口为 Alternate port。都选择好端口角色（Port Role）了，就进入各个端口相应的状态了，于是就生成了相应的图 5。

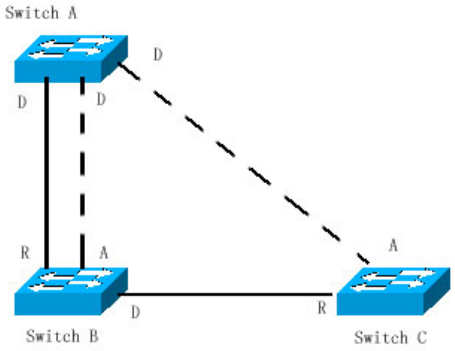


图 5

如果 Switch A 和 Switch B 之间的活动链路出了故障，那备份链路就会立即产生作用，于是就生成了相应的图 6。

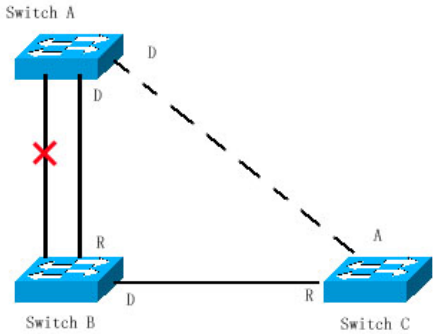


图 6

如果 Switch B 和 Switch C 之间的链路出了故障，那 Switch C 就会自动把 Alternate port 转为 Root port，就生成了图 7 的情况。

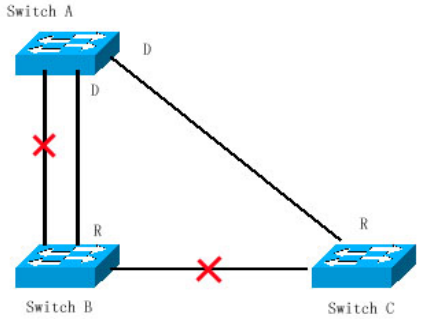


图 7

16.1.1.7 RSTP 的快速收敛

现在开始介绍 RSTP 所特有的功能，即能让端口“快速”的 Forwarding。

STP 协议是选好端口角色（Port Role）后等待 30 秒(为 Forward-Delay Time 的 2 倍，Forward-Delay Time 可配置，默认为 15 秒)后再 Forwarding 的，而且每当拓朴发生变化后，每个网桥的 Root Port 和 Designated Port 都要重新过 30 秒再 Forwarding，因此要等整个网络拓朴稳定为一个树型结构就大约需要 50 秒。

而 RSTP 端口的 Forwarding 过程就大不一样了，如图 8 所示，Switch A 发送 RSTP 特有“Proposal”报文，Switch B 发现 Switch A 的优先级比自身高，就选 Switch A 为根桥，收到报文的端口为 Root Port，立即 Forwarding，然后从 Root Port 向 Switch A 发送“Agree”报文。Switch A 的 Designated Port 得到“同意”，也就 Forwarding 了。然后 Switch B 的 Designated Port 又发送“Proposal”报文依次将生成树展开。因此在理论上，RSTP 是能够在网络拓朴发生变化的一瞬间恢复网络树型结构，达到快速收敛。

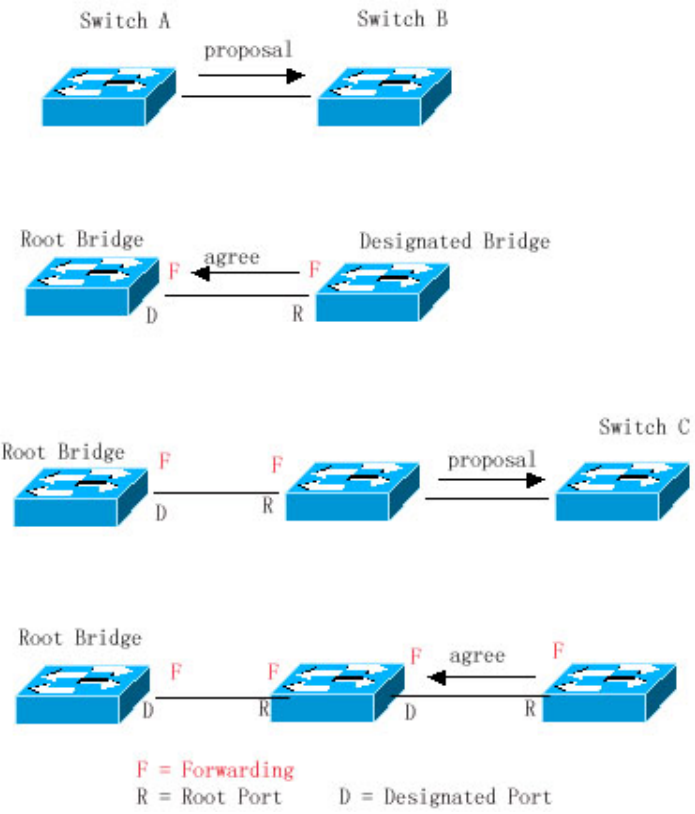


图 8

注意:

以上的“握手”过程是有条件的，就是端口间必须是“Point-to-point Connect（点对点连接）”。为了让您的设备发挥最大的功效，最好不要使设备间为非点对点连接。

本章中除图 9 外，其它示意图均为“点对点连接”，以下列出了“非点对点连接”的范例图。

非点对点连接范例：

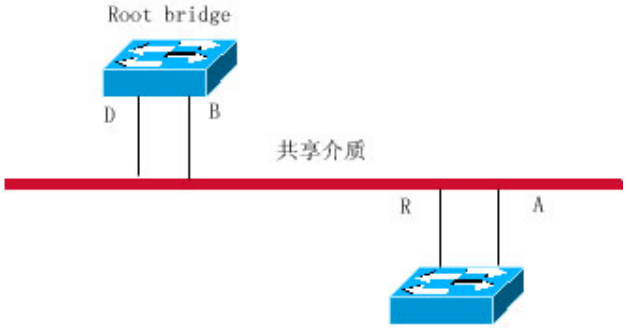


图 9

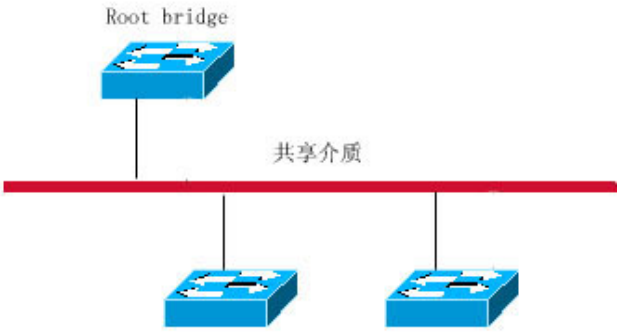


图 10

另外、下图为“点对点”连接，请用户注意区分

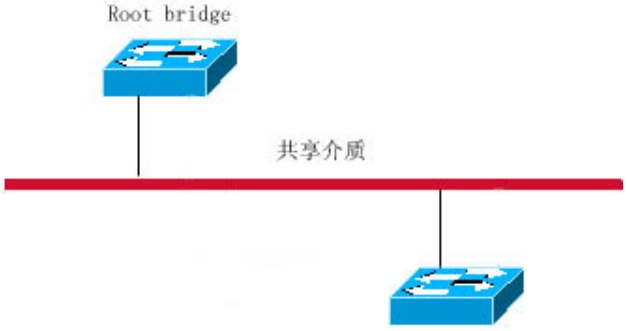


图 11

16.1.1.8 RSTP 与 STP 的兼容

RSTP 协议可以与 STP 协议完全兼容，RSTP 协议会根据收到的 BPDU 版本号来自动判断与之相连的网桥是支持 STP 协议还是支持 RSTP 协议，如果是与 STP 网桥互连就只能按 STP 的 Forwarding 方法，过 30 秒再 Forwarding，无法发挥 RSTP 的最大功效。

另外，RSTP 和 STP 混用还会遇到这样一个问题。如图 12 所示，Switch A 是支持 RSTP 协议的，Switch B 只支持 STP 协议，它们俩互连，Switch A 发现与它相连的是 STP 桥，就发 STP 的 BPDU 来兼容它。但后来如果换了台 Switch C，它支持 RSTP 协议，但 Switch A 却依然在发 STP 的 BPDU，这样使 Switch C 也认为与之互连的是 STP 桥了，结果两台支持 RSTP 的设备却以 STP 协议来运行，大大降低了效率。

为此 RSTP 协议提供了 Protocol-migration 功能来强制发 RSTP BPDU(这种情况下，对端网桥必须支持 RSTP)，这样 Switch A 强制发了 RSTP BPDU，Switch C 就发现与之互连的网桥是支持 RSTP 的，于是两台设备就都以 RSTP 协议运行了，如图 13。

Protocol Migration

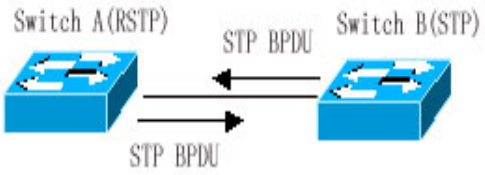


图 12

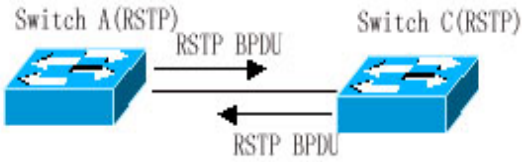


图 13

16.1.2 MSTP 概述

本设备支持 MSTP，MSTP 是在传统的 STP、RSTP 的基础上发展而来的新的生成树协议，本身就包含了 RSTP 的快速 FORWARDING 机制。

由于传统的生成树协议与 Vlan 没有任何联系，因此在特定网络拓朴下就会产生以下问题：

如图 14 所示，设备 A、B 在 Vlan1 内，设备 C、D 在 Vlan2 内，然后连成环路。

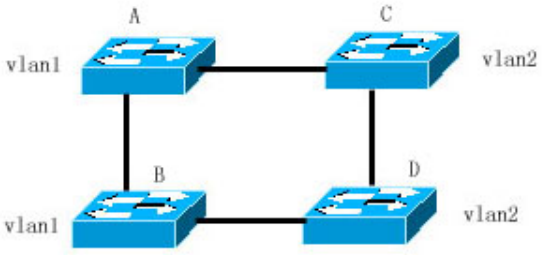


图 14

若从设备 A 依次通过设备 C、D 到达 B 的链路花费比从设备 A 直接到 B 的链路花费更少的情况下，会造成把设备 A 和 B 间的链路给 DISCARDING（如图 15 所示）。由于设备 C、D 不包含 Vlan1，无法转发 Vlan1 的数据包，这样设备 A 的 Vlan1 就无法与设备 B 的 Vlan1 进行通讯。

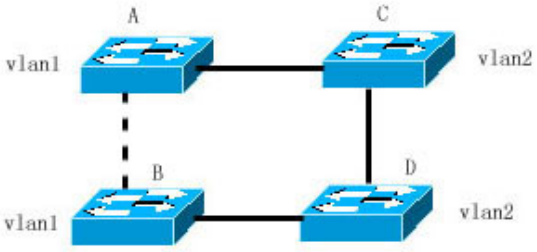


图 15

为了解决这个问题，MSTP 就产生了，它可以把一台设备的一个或多个 Vlan 划分为一个 Instance，有着相同 Instance 配置的设备就组成一个域（MST Region），运行独立的生成树（这个内部的生成树称为 IST，Internal Spanning-tree）；这个 MST region 组合就相当于一个大的设备整体，与其他 MST Region 再进行生成树算法运算，得出一个整体的生成树，称为 CST（Common Spanning Tree）。

按这种算法，以上网络就可以在 MSTP 算法下形成图 16 的拓朴：设备 A 和 B 都在 MSTP Region 1 内，MSTP Region 1 没有环路产生，所以没有链路 DISCARDING，同理 MSTP Region 2 的情况也是一样的。然后 Region 1 和 Region 2 就分别相当于两个大的设备，这两台“设备”间有环路，因此根据相关配置选择一条链路 DISCARDING。

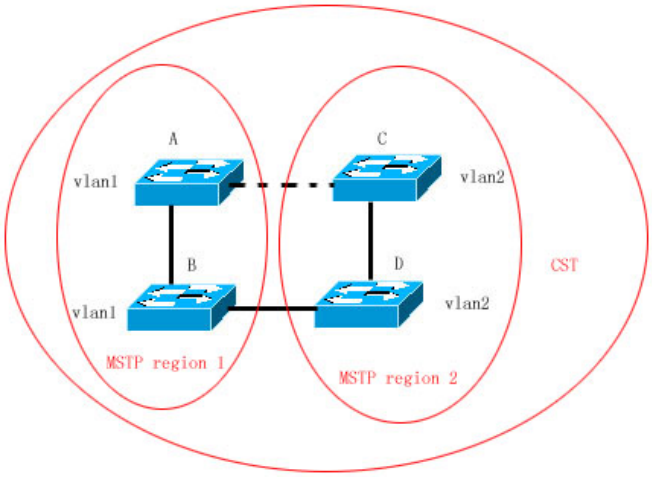


图 16

这样，既避免了环路的产生，也能让相同 Vlan 间的通讯不受影响。

16.1.2.1 如何划分 MSTP Region

根据以上描述，很明显，要让 MSTP 产生应有的作用，首先就要合理地划分 MSTP Region，相同 MSTP Region 内的设备“MST 配置信息”一定要相同。

MST 配置信息包括：

- MST 配置名称（Name）：最长可用 32 个字节长的字符串来标识 MSTP Region。

- MST Revision Number：用一个 16bit 长的修正值来标识 MSTP Region。
- MST Instance—vlan 的对应表：每台设备都最多可以创建 64 个 Instance（id 从 1 到 64），Instance 0 是强制存在的，所以系统总共支持 65 个 Instance。用户还可以按需要分配 1-4094 个 Vlan 属于不同的 Instance（0－64），未分配的 Vlan 缺省就属于 Instance 0。这样，每个 MSTI（MST Instance）就是一个“Vlan 组”，根据 BPDU 里的 MSTI 信息进行 MSTI 内部的生成树算法，不受 CIST 和其他 MSTI 的影响。

您可在用 **spanning-tree mst configuration** 全局配置命令进入“MST 配置模式”配置以上信息。

MSTP BPDU 里附带以上信息，如果一台设备收到的 BPDU 里的 MST 配置信息和自身的一样，就会认为该端口上连着的设备和自己是属于同一个 MST Region，否则就认为是从另外一个 Region 来的。

建议：

我们建议您在关闭 STP 的模式下配置 Instance—vlan 的对应表，配置好后再打开 MSTP，以保证网络拓朴的稳定和收敛。

16.1.2.2 MSTP region 内的生成树（IST）

划分好 MSTP Region 后，每个 Region 里就按各个 Instance 所设置的 Bridge Priority、Port Priority 等参数选出各个 Instance 独立的 Root Bridge，以及每台设备上各个端口的 Port Role，然后就 Port Role 指定该端口在该 Instance 内是 FORWARDING 还是 DISCARDING 的。

这样，经过 MSTP BPDU 的交流，IST(Internal Spanning Tree) 就生成了，而各个 Instance 也独立的有了自己的生成树（MSTI），其中 Instance 0 所对应的生成树与 CST 共同称为 CIST（Common Instance Spanning Tree）。也就是说，每个 Instance 都为各自的“vlan 组”提供了一条单一的、不含环路的网络拓朴。

如下图所示，在 Region 1 内，设备 A、B、C 组成环路。

在 CIST（Instance 0）中，如图 17，因 A 的优先级最高，被选为 Region Root，再根据其他参数，把 A 和 C 间的链路给 DISCARDING。因此，对 Instance 0 的“Vlan 组”来说，只有 A 到 B、B 到 C 的链路可用，打断了这个“Vlan 组”的环路。

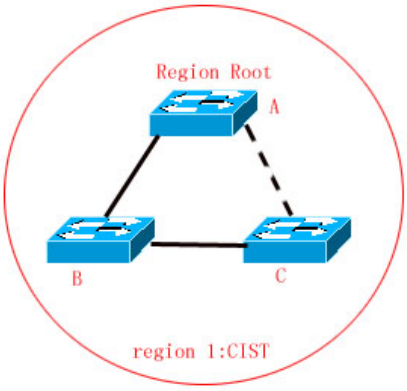


图 17

而对 MSTI 1（Instance 1）来说，如图 18，B 的优先级最高，被选为 Region Root，再根据其他参数，把 B 和 C 间的链路给 DISCARDING。因此，对 Instance 1 的“Vlan 组”来说，只有 A 到 B、A 到 C 的链路可用，打断了这个“Vlan 组”的环路。

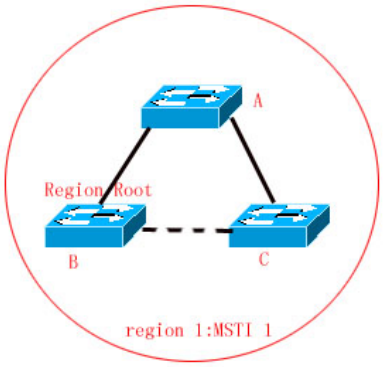


图 18

而对 MSTI 2（Instance 2）来说，图 19，C 的优先级最高，被选为 Region Root，再根据其他参数，把 A 和 B 间的链路给 DISCARDING。因此，对 Instance 2 的“Vlan 组”来说，只有 B 到 C、A 到 C 的链路可用，打断了这个“Vlan 组”的环路。

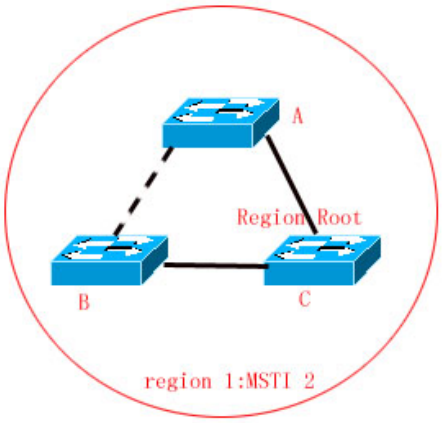


图 19

用户在这里要注意的是 MSTP 协议本身不关心一个端口属于哪个 Vlan，所以用户应该根据实际的 Vlan 配置情况来为相关端口配置对应的 Path Cost 和 Priority，以防 MSTP 协议打断了不该打断的环路。

16.1.2.3 MSTP region 间的生成树（CST）

每个 MSTP region 对 CST 来说可以相当于一个大的设备整体，不同的 MSTP Region 也生成一个大的网络拓朴树，称为 CST(Common Spanning Tree) 。如图 20 所示，对 CST 来说，Bridge ID 最小的设备 A 被选为整个 CST 的根(CST Root)，同时也是这个 Region 内的 CIST Regional Root。在 Region 2 中，由于设备 B 到 CST Root 的 Root Path Cost 最短，所以被选为这个 Region 内的 CIST Regional Root。同理，Region 3 选设备 C 为 CIST Regional Root。

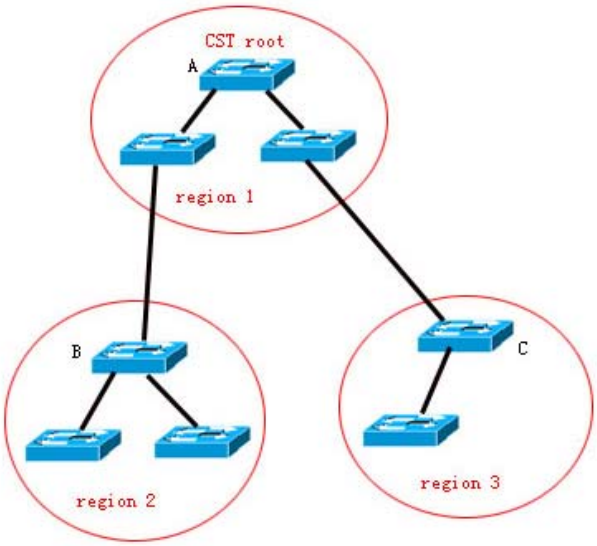


图 20

CIST Regional Root 不一定是该 Region 内 Bridge ID 最小的那台设备，它是指该 Region 内到 CST Root 的 Root Path Cost 最小的设备。

同时，CIST Regional Root 的 Root Port 对 MSTI 来说有了个新的 Port Role，为“Master port”，作为所有 Instance 对外的“出口”，它对所有 Instance 都是 FORWARDING 的。为了使拓朴更稳定，我们建议每个 Region 对 CST Root 的“出口”尽量只在该 Region 的一台设备上!

16.1.2.4 Hop Count

IST 和 MSTI 已经不用 Message Age 和 Max Age 来计算 BPDU 信息是否超时，而是用类似于 IP 报文 TTL 的机制来计算，它就是 Hop Count。

您可以用 **spanning-tree max-hops** 全局配置命令来设置。在 Region 内，从 Region Root Bridge 开始，每经过一个设备，Hop Count 就会减 1，直到为 0 则表示该 BPDU 信息超时，设备收到 Hops 值为 0 的 BPDU 就要丢弃它。

为了和 Region 外的 STP、RSTP 兼容，MSTP 依然保留了 Message Age 和 Max Age 的机制。

16.1.2.5 MSTP 和 RSTP、STP 协议的兼容

对 STP 协议来说，MSTP 会像 RSTP 那样发 STP BPDU 来兼容它，详细情况请参考“RSTP 与 STP 的兼容”章节。

而对 RSTP 协议来说，本身会处理 MSTP BPDU 中 CIST 的部分，因此 MSTP 不必专门的发 RSTP BPDU 以兼容它。

每台运行 STP 或 RSTP 协议的设备都是单独的一个 Region，不与任何一个设备组成同一个 Region。

16.2 MSTP 可选特性概述

16.2.1 理解 Port Fast

如果设备的端口直连着网络终端，那么就可以设置该端口为 Port Fast，端口直接 Forwarding，这样可免去端口等待 Forwarding 的过程（如果不配置 Port Fast 的端口，就要等待 30 秒 Forwarding）。下图表示了一个设备的哪些端口可以配置为 Port Fast enable。

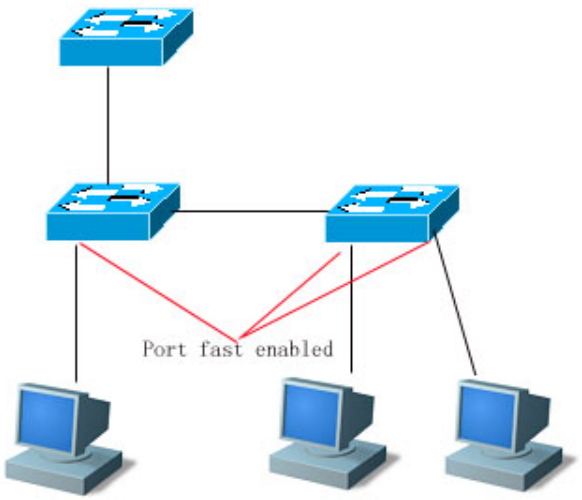


图 21

如果在设了 Port Fast 的端口中还收到 BPDU，则它的 Port Fast Operational State 为 Disabled。这时该端口会按正常的 STP 算法进行 Forwarding。

16.2.2 理解边缘口的自动识别(AutoEdge 功能)

AutoEdge 功能是指当指派口在一定的时间范围内(为 3 秒)，如果收不到下游端口发送的 BPDU，则认为该端口相连的不是一台网络设备，从而设置该端口为边缘端口，直接进入 Forwarding 状态。自动标识为边缘口的端口因收到 BPDU 而自动识别为非边缘口。

您可以通过 **spanning-tree autoedge disabled** 命令取消边缘口的自动识别功能。

该功能是缺省打开的。

注意:

1) 边缘口的自动标识功能与手工的 Port Fast 冲突时，以手工配置的为准。

2) 该功能作用于指派口与下游端口进行快速协商转发的过程中，所以 STP 协议不支持该功能。同时如果指派口已经处于转发状态，对该端口进行 Autoedge 的配置不会生效，只有在重新快速协商的过程中才生效，如拔插网线。

3) 端口如果先打开了 BPDU Filter，则该端口直接 Forwarding，不会自动识别为边缘口。

4) 该功能只适用与指派口。

5) AutoEdge 功能遵循 IEEE 802.1D 2004 版本的标准定义，在该版本中，Bridge Hello Time 参数的范围已修改为 1.0-2.0。所以在使用 AutoEdge 功能时请确认 Hello Time 时间在该范围内，超出了该范围可能会有临时环路的风险。如果确认需

要将 Hello Time 值超出该范围，建议先将端口的 AutoEdge 功能关闭。

16.2.3 理解 BPDU Guard

BPDU Guard 既能全局的 enable，也能针对单个 Interface 进行 enable。这两者有些细小的差别。

您可以在特权模式中用 **spanning-tree portfast bpduguard default** 命令打开全局的 BPDU Guard enabled 状态，在这种状态下，如果某个 Interface 打开了 Port Fast 或者该端口自动标识为边缘端口，而该 Interface 收到了 BPDU，该端口就会进入 Error-disabled 状态，以示配置错误；同时整个端口被关闭，表示网络中可能被非法用户增加了一台网络设备，使网络拓朴发生改变。

您也可以在 Interface 配置模式下用 **spanning-tree bpduguard enable** 命令来打开单个 Interface 的 BPDU Guard（与该端口是否是边缘端口无关）。在这个情况下如果该 Interface 收到了 BPDU，就进入 Error-disabled 状态。

16.2.4 理解 BPDU Filter

BPDU Filter 既能全局的 enable，也能针对单个 Interface 进行 enable。这两者有些细小的差别。

您可以在特权模式中用 spanning-tree portfast bpdufilter default 命令打开全局的 BPDU Filter enabled 状态，在这种状态下，如果某个 Interface 打开了 Port Fast 或者该端口自动标识为边缘端口，则该接口将既不收 BPDU，也不发 BPDU，这样，直连端口的主机就收不到 BPDU。而如果边缘端口因收到 BPDU 而使 Port Fast Operational 状态 disabled，BPDU Filter 也就自动失效。

您也可以在 Interface 配置模式下用 spanning-tree bpdufilter enable 命令设置单个 Interface 的 BPDU Filter enable（与该端口是否是边缘端口无关）。在这个情况下该 Interface 既不收 BPDU，也不发 BPDU，并且是直接 Forwarding 的。

16.2.5 理解 Tc-protection

TC-BPDU 报文是携带 TC 标志的 BPDU 报文，交换机收到这类报文表示网络拓扑发生了变化，会进行 MAC 地址表的删除操作，对三层交换机，还会引发路由表删除操作，并改变 ARP 表项的端口状态。为避免交换机受到伪造 TC-BPDU 报文的恶意攻击时频繁进行以上操作，负荷过重，影响网络稳定，可以使用 TC-protection 功能进行保护。

Tc-protection 只能全局的打开和关闭，缺省情况下为打开此功能。

在打开相应功能时，收到 TC-BPDU 报文后的一定时间内（一般为 4 秒），只进行一次删除操作，同时监控该时间段内是否收到 TC-BPDU 报文。如果在该时间段内收到了 TC-BPDU 报文，则设备在该时间超时后再进行一次删除操作。

16.2.6 理解 TC Guard

Tc-Protection 功能可以保证网络产生大量 tc 报文时减少动态 MAC 地址和 ARP 的删除，但在遇到 TC 报文攻击的时候还是会产生很多的删除操作，并且 TC 报文是可扩散的，将影响整个网络。使用 TC Guard 功能，我们允许用户在全局或者端口上禁止 TC 报文的扩散。当一个端口收到 TC 报文的时候，如果全局配置了 TC Guard 或者是端口上配置了 TC Guard，则该端口将屏蔽掉该端口接收或者是自己产生的 TC 报文，使得 TC 报文不会扩散到其它端口，这样能有效控制网络中可能存在的 TC 攻击，保持网络的稳定，尤其是在三层设备上，该功能能有效避免接入层设备的振荡引起核心路由中断的问题。

注意:

1) 错误的使用 tc-guard 功能会使网络之间的通讯中断。
2) 建议您在确认网络当中有非法的 tc 报文攻击的情况下再打开此功能。
3) 打开全局的 tc-guard,则所有端口都不会对外扩散 tc 报文。适用于桌面接入设备上开启。
4) 打开接口的 tc-guard,则对于该接口产生的拓扑变化以及收到的 tc 报文，将不向其它端口扩散。适合在上链口，尤其是汇聚接核心的端口开启该功能。

16.2.7 理解 BPDU 源 MAC 检查

BPDU 源 MAC 检查是为了防止通过人为发送 BPDU 报文来恶意攻击交换机而使 MSTP 工作不正常。当确定了某端口点对点链路对端相连的交换机时，可通过配置 BPDU 源 MAC 检查来达到只接收对端交换机发送的 BPDU 帧，丢弃所有其他 BPDU 帧，从而达到防止恶意攻击。你可以在 interface 模式下来为特定的端口配置相应的 BPDU 源 MAC 检查 MAC 地址，一个端口只允许配置一个过滤 MAC 地址，通过 no bpdu src-mac-check 来禁止 BPDU 源 MAC 检查，此时端口接收任何 BPDU 帧。

16.2.8 理解 BPDU 非法长度过滤

BPDU 的以太网长度字段超过 1500 时，该 BPDU 帧将被丢弃，以防止收到非法 BPDU 报文。

16.2.9 理解 ROOT Guard 功能

在网络设计中常常将根桥和备份根桥划分在同一个域内，由于维护人员的错误配置或网络中的恶意攻击，根桥有可能收到优先级更高的配置信息，从而失去当前根桥

的位置，引起网络拓扑的错误的变动。Root Guard 功能就是为了防止这种情况的出现。

接口打开 Root Guard 功能时，强制其在所有实例上的端口角色为指定端口，一旦该端口收到优先级更高的配置信息时，Root Guard 功能会将该接口置为 root-inconsistent (blocked)状态，在足够长的时间内没有收到更优的配置信息时，端口会恢复成原来的正常状态。

若由于本功能导致接口进入 blocked 状态，并需要手动恢复为正常状态，则请关闭 ROOT Guard 功能或关闭接口的保护功能（在接口层配置 **spanning-tree guard none**）。

注意:

1) 错误的使用 ROOT Guard 特性会导致网络链路的断开。
2) 在非指派口上打开 ROOT Guard 功能会强制其为指派口，同时端口会进入 BKN 状态（即 blocked 状态，只是叫法不同）。
3) 如果端口在 MST0 因收到更优的配置消息而进入 BKN 状态，会强制端口在其它所有的实例中处于 BKN 状态。
4) 端口的 ROOT Guard 和 LOOP Guard 同一时刻只能有一个生效。
5) 打开 ROOT Guard 的端口，边缘口的自动识别功能将失效。

16.2.10 理解 LOOP Guard 功能

由于单向链路的故障，根口或备份口由于收不到 BPDU 会变成指派口进入转发状态，从而导致了网络中环路的产生，LOOP Guard 功能防止了这种情况的发生。

对于配置了环路保护的端口，如果收不到 BPDU，会进行端口角色的迁移，但端口状态将一直被设成 discarding 状态。直到重新收到 BPDU 而进行生成树的重计算。

注意:

1) 您可以基于全局或接口打开 LOOP Guard 特性。
2) 端口的 ROOT Guard 和 LOOP Guard 同一时刻只能有一个生效。
3) 全局打开 LOOP Guard 功能，则所有端口的边缘口的自动识别功能将失效。
4) 接口打开 LOOP Guard 功能，则该端口的边缘口的自动识别功能将失效。

16.3 配置 MSTP

16.3.1 缺省的 Spanning Tree 设置

下面列出 Spanning Tree 的缺省配置

项目	缺省值
Enable State	Disable，不打开 STP
STP MODE	MSTP
STP Priority	32768
STP port Priority	128
STP port cost	根据端口速率自动判断
Hello Time	2 秒
Forward-delay Time	15 秒
Max-age Time	20 秒
Path Cost 的缺省计算方法	长整型
Tx-Hold-Count	3
Link-type	根据端口双工状态自动判断
Maximum hop count	20
vlan 与 实例 对应关系	所有 vlan 属于实例 0 只存在实例 0

您可通过 **spanning-tree reset** 命令让 Spanning Tree 参数恢复到缺省配置(不包括关闭 Span)

16.3.2 打开、关闭 Spanning Tree 协议

打开 Spanning-tree 协议，设备即开始运行生成树协议，本设备缺省运行的是 MSTP 协议。

设备的缺省状态是关闭 Spanning-tree 协议。

进入特权模式，按以下步骤打开 Spanning Tree 协议：

命令	作用
Ruijie**# configure terminal**	进入全局配置模式。
Ruijie(config)**# spanning-tree**	打开 Spanning Tree 协议。
Ruijie(config)**# end**	退回到特权模式。

Ruijie**# show spanning-tree**	核对配置条目。
Ruijie**# copy running-config startup-config**	保存配置。

如果您要关闭 Spanning Tree 协议，可用 **no spanning-tree** 全局配置命令进行设置。

16.3.3 配置 Spanning Tree 的模式

按 802.1 相关协议标准，STP、RSTP、MSTP 这三个版本的 Spanning Tree 协议本来就无须管理员再多做设置，版本间自然会互相兼容。但考虑到有些厂家不完全按标准实现，可能会导致一些兼容性的问题。因此我们提供这么一条命令配置，以供管理员在发现其他厂家的设备与本设备不兼容时，能够切换到低版本的 Spanning Tree 模式，以兼容之。

注意，当您从 MSTP 模式切换到 RSTP 或 STP 模式时，有关 MSTP Region 的所有的信息将被清空。

设备的缺省模式是 MSTP 模式。

进入特权模式，按以下步骤打开 Spanning Tree 协议：

命令	作用
Ruijie**# configure terminal**	进入全局配置模式。
Ruijie(config)**# spanning-tree mode mstp/rstp/stp**	切换 Spanning Tree 模式。
Ruijie(config)**# end**	退回到特权模式。
Ruijie**# show spanning-tree**	核对配置条目。
Ruijie**# copy running-config startup-config**	保存配置。

如果您要恢复 Spanning Tree 协议的缺省模式，可用 **no spanning-tree mode** 全局配置命令进行设置。

16.3.4 配置设备优先级（Switch Priority）

设置设备的优先级关系着到底哪个设备为整个网络的根，同时也关系到整个网络的拓扑结构。建议管理员把核心设备的优先级设得高些（数值小），这样有利于整个网络的稳定。您可以给不同的 Instance 分配不同的设备优先级，各个 Instance 可根据这些值运行独立的生成树协议。对于不同 Region 间的设备，它们只关心 CIST（Instance 0）的优先级。

如 Bridge ID 所讲，优先级的设置值有 16 个，都为 4096 的倍数，分别是 0，4096，8192，12288，16384，20480，24576，28672，32768，36864，40960，45056，49152，53248，57344，61440。缺省值为 32768。

进入特权模式，按以下步骤配置设备优先级：

命令	作用
Ruijie**# configure terminal**	进入全局配置模式。
Ruijie(config)**# spanning-tree** [**mst** *instance-id*] **priority** *priority*	针对不同的 instance 配置设备的优先级，当您不加 instance 参数时，即对 instance 0 进行配置。 *instance-id*，范围为 0－64 *priority*，取值范围为 0 到 61440，按 4096 的倍数递增，缺省值为 32768。
Ruijie(config)**# end**	退回到特权模式。
Ruijie**# show running-config**	核对配置条目。
Ruijie**# copy running-config startup-config**	保存配置。

您如果要恢复到缺省值，可用 **no spanning-tree mst** *instance-id* **priority** 全局配置命令进行设置。

16.3.5 配置端口优先级（Port Priority）

当有两个端口都连在一个共享介质上，设备会选择一个高优先级（数值小）的端口进入 Forwarding 状态，低优先级（数值大）的端口进入 Discarding 状态。如果两个端口的优先级一样，就选端口号小的那个进入 Forwarding 状态。您可以在一个端口上给不同的 Instance 分配不同的端口优先级，各个 Instance 可根据这些值运行独立的生成树协议。

和设备的优先级一样，可配置的优先级值也有 16 个，都为 16 的倍数，分别是 0，16，32，48，64，80，96，112，128，144，160，176，192，208，224，240。缺省值为 128。

进入特权模式，按以下步骤配置端口优先级：

命令	作用
Ruijie**# configure terminal**	进入全局配置模式。
Ruijie(config)**# interface** *interface-id*	进入该 interface 的配置模式，合法的 interface 包括物理端口和 Aggregate Link。

Ruijie(config-if)# **spanning-tree** [**mst** *instance-id*] **port-priority** *priority*	针对不同的 instance 配置端口的优先级，当您不加 instance 参数时，即对 instance 0 进行配置。 *instance-id*，范围为 0－64 *priority*，配置该 interface 的优先级，取值范围为 0 到 240，按 16 的倍数递增，缺省值为 128
Ruijie(config-if)# **end**	退回到特权模式。
Ruijie# **show spanning-tree** [**mst** *instance-id*] **interface** *interface-id*	核对配置条目。
Ruijie# **copy running-config startup-config**	保存配置。

您如果要恢复到缺省值，可用 **no spanning-tree mst** *instance-id* **port-priority** 接口配置命令进行设置。

16.3.6 配置端口的路径花费（Path Cost）

设备是根据哪个端口到根桥（Root Bridge）的 Path Cost 总和最小而选定 Root Port 的，因此 Port Path Cost 的设置关系到本设备 Root Port。它的缺省值是按 Interface 的链路速率（The Media Speed）自动计算的，速率高的花费小，如果管理员没有特别需要可不必更改它，因为这样算出的 Path Cost 最科学。您可以在一个端口上针对不同的 Instance 分配不同的路径花费，各个 Instance 可根据这些值运行独立的生成树协议。

进入特权模式，按以下步骤配置端口路径花费：

命令	作用
Ruijie# **configure terminal**	进入全局配置模式。
Ruijie(config)# **interface** *interface-id*	进入该 interface 的配置模式，合法的 interface 包括物理端口和 Aggregate Link。
Ruijie(config-if)# **spanning-tree** [**mst** *instance-id*] **cost** *cost*	针对不同的 instance 配置端口的优先级，当您不加 instance 参数时，即对 instance 0 进行配置。 *instance-id*，范围为 0－64 *cost* ，配置该端口上的花费，取值范围为 1 到 200,000,000。缺省值为根据 interface 的链路速率自动计算。
Ruijie(config-if)# **end**	退回到特权模式。

Ruijie# **show spanning-tree** [**mst** *instance-id*] **interface** *interface-id*	核对配置条目。
Ruijie# **copy running-config startup-config**	保存配置。

您如果要恢复到缺省值，可用 **no spanning-tree mst cost** 接口配置命令进行设置。

16.3.7 配置 Path Cost 的缺省计算方法（path cost method）

当该端口 Path Cost 为缺省值时，设备会自动根据端口速率计算出该端口的 Path Cost。但 IEEE 802.1d 和 IEEE 802.1t 对相同的链路速率规定了不同 Path Cost 值，802.1d 的取值范围是短整型（short）（1—65535），802.1t 的取值范围是长整型（long）(1—200,000,000)。请管理员一定要统一好整个网络内 Path Cost 的标准。缺省模式为长整型模式（IEEE 802.1t 模式）。

下表列出两种方法对不同链路速率自动设置的 Path Cost。

端口速率	Interface	IEEE 802.1d（short）	IEEE 802.1t（long）
10M	普通端口	100	2000000
	Aggregate Link	95	1900000
100M	普通端口	19	200000
	Aggregate Link	18	190000
1000M	普通端口	4	20000
	Aggregate Link	3	19000

进入特权模式，按以下步骤配置端口路径花费的缺省计算方法：

命令	作用
Ruijie# **configure terminal**	进入全局配置模式。
Ruijie(config)# **spanning-tree pathcost method long/short**	配置端口路径花费的缺省计算方法，设置值为长整型（long）或短整型（short），缺省值为长整型（long）。
Ruijie(config)# **end**	退回到特权模式。

Ruijie**# show running-config**	核对配置条目。
Ruijie**# copy running-config startup-config**	保存配置。

您如果要恢复到缺省值，可用 **no spanning-tree pathcost method** 全局配置命令进行设置。

16.3.8 配置 Hello Time

配置设备定时发送 BPDU 报文的时间间隔。缺省值为 2 秒。

进入特权模式，按以下步骤配置 Hello Time：

命令	作用
Ruijie**# configure terminal**	进入全局配置模式。
Ruijie(config)**# spanning-tree hello-time** *seconds*	配置 hello_time，取值范围为 1 到 10 秒，缺省值为 2 秒。
Ruijie(config)**# end**	退回到特权模式。
Ruijie**# show running-config**	核对配置条目。
Ruijie**# copy running-config startup-config**	保存配置。

您如果要恢复到缺省值，可用 **no spanning-tree hello-time** 全局配置命令进行设置。

16.3.9 配置 Forward-Delay Time

配置端口状态改变的时间间隔。缺省值为 15 秒。

进入特权模式，按以下步骤配置 Forward-Delay Time：

命令	作用
Ruijie**# configure terminal**	进入全局配置模式。
Ruijie(config)**# spanning-tree forward-time** *seconds*	配置 forward delay time，取值范围为 4 到 30 秒，缺省值为 15 秒。
Ruijie(config)**# end**	退回到特权模式。

Ruijie# **show running-config**	核对配置条目。
Ruijie# **copy running-config startup-config**	保存配置。

您如果要恢复到缺省值，可用 **no spanning-tree forward-time** 全局配置命令进行设置。

16.3.10 配置 Max-Age Time

配置 BPDU 报文消息生存的最长时间。缺省值为 20 秒。

进入特权模式，按以下步骤配置 Max-Age Time：

命令	作用
Ruijie# **configure terminal**	进入全局配置模式。
Ruijie(config)# **spanning-tree max-age** *seconds*	配置 max age time，取值范围为 6 到 40 秒，缺省值为 20 秒。
Ruijie(config)# **end**	退回到特权模式。
Ruijie# **show running-config**	核对配置条目。
Ruijie# **copy running-config startup-config**	保存配置。

您如果要恢复到缺省值，可用 **no spanning-tree max-age** 全局配置命令进行设置。

注意：

Hello Time、Forward-Delay Time、Max-Age Time 除了有一个自身的取值范围外，这三个之间还有一个制约关系，就是：2*(Hello Time + 1.0 seconds) <= Max-Age Time <= 2*(Forward-Delay – 1.0seconds)。您配置的这三个参数必须满足这个条件，否则有可能导致拓朴不稳定。

16.3.11 配置 Tx-Hold-Count

配置每秒钟最多发送的 BPDU 个数，缺省值为 3 个。

进入特权模式，按以下步骤配置 Tx-Hold-Count：

命令	作用

Ruijie**# configure terminal**	进入全局配置模式。
Ruijie(config)**# spanning-tree tx-hold-count** *numbers*	配置每秒最多发送 BPDU 个数，取值范围为 1 到 10 个，缺省值为 3 个。
Ruijie(config)**# end**	退回到特权模式。
Ruijie**# show running-config**	核对配置条目。
Ruijie**# copy running-config startup-config**	保存配置。

您如果要恢复到缺省值，可用 **no spanning-tree tx-hold-count** 全局配置命令进行设置。

16.3.12 配置 link-type

配置该端口的连接类型是不是“点对点连接”，这一点关系到RSTP是否能快速的收敛。请参照“ RSTP的快速收敛”。当您不设置该值时，设备会根据端口的“双工”状态来自动设置的，全双工的端口就设link type为point-to-point，半双工就设为shared。您也可以强制设置link type来决定端口的连接是不是“点对点连接”。

进入特权模式，按以下步骤配置端口的 link type：

命令	作用
Ruijie**# configure terminal**	进入全局配置模式。
Ruijie(config)**# interface** *interface-id*	进入接口配置模式
Ruijie(config-if)**# spanning-tree link-type point-to-point/shared**	配置该 interface 的连接类型，缺省值为根据端口“双工”状态来自动判断是不是“点对点连接”。全双工为“点对点连接”，即可以快速 FORWARDING。
Ruijie(config-if)**# end**	退回到特权模式。
Ruijie**# show running-config**	核对配置条目。
Ruijie**# copy running-config startup-config**	保存配置。

您如果要恢复到缺省值，可用 **no spanning-tree link-type** 接口配置命令进行设置。

16.3.13 配置 Protocol Migration 处理

该设置是让该端口强制进行版本检查。相关说明请参看 RSTP与STP的兼容。

命令	作用
Ruijie**# clear spanning-tree detected-protocols**	对所有端口强制版本检查
Ruijie**# clear spanning-tree detected-protocols interface** *interface-id*	针对一个端口进行版本检查

16.3.14 配置 MSTP Region

要让多台设备处于同一个 MSTP Region，就要让这几台设备有相同的名称（Name）、相同的 Revision Number、相同的 Instance—Vlan 对应表。

你可以配置 0－64 号 Instance 包含哪些 Vlan，剩下的 Vlan 就自动分配给 Instance 0。一个 Vlan 只能属于一个 Instance。

我们建议您在关闭 STP 的模式下配置 Instance—Vlan 的对应表，配置好后再打开 MSTP，以保证网络拓朴的稳定和收敛。

进入特权模式，按以下步骤配置 MSTP Region：

命令	作用
Ruijie**# configure terminal**	进入全局配置模式。
Ruijie(config)**# spanning-tree mst configuration**	进入 MST 配置模式。
Ruijie(config-mst)**# instance** *instance-id* **vlan** *vlan-range*	把 vlan 组添加到一个 MST instance 中 *instance-id*，范围为 0－64 *vlan-range*，范围为 1－4094 举例来说： instance 1 vlan 2-200 就是把 vlan 2 到 vlan 200 都添加到 instance 1 中。 instance 1 vlan 2,20,200 就是把 vlan 2、vlan 20，vlan 200 添加到 instance 1 中。 同样，您可以用 **no** 命令把 vlan 从 instance 中删除，删除的 vlan 自动转入 instance 0。
Ruijie(config-mst)**# name** *name*	指定 MST 配置名称，该字符串最多可以有 32 个字节。
Ruijie(config-mst)**# revision** *version*	指定 MST revision number，范围为 0－65535。缺省值为 0
Ruijie(config-mst)**# show**	核对 MST 的配置条目。

Ruijie(config-mst)# **end**	退回到特权模式。
Ruijie# **copy running-config startup-config**	保存配置。

要恢复缺省的 MST Region Configuration 配置，您可以用 **no spanning-tree mst configuration** 全局配置命令。您也可以用 **no instance** *instance-id* 来删除该 instance。同样，**no name**、**no revision** 可以分别把 MST name、MST revision number 恢复到缺省值。

以下为配置实例：

```
Ruijie(config)# spanning-tree mst configuration
Ruijie(config-mst)# instance 1 vlan 10-20
Ruijie(config-mst)# name region1
Ruijie(config-mst)# revision 1
Ruijie(config-mst)# show
Multi spanning tree protocol : Enable Name [region1]
Revision 1
Instance Vlans Mapped
-------- --------------------
0 1-9,21-4094
1 10-20
------------------------------
Ruijie(config-mst)# exit
Ruijie(config)#
```

注意：

在配置 vlan 和 instance 的映射关系前请务必确保所配置的 vlan 已经被创建，否则在部分产品上有可能会出现 vlan 和 instance 关联失败。

16.3.15 配置 Maximum-Hop Count

配置 Maximum-Hop Count，指定了 BPDU 在一个 Region 内经过多少台设备后被丢弃。它对所有 Instance 有效。

进入特权模式，按以下步骤配置 Maximum-Hop Count：

命令	作用
Ruijie# **configure terminal**	进入全局配置模式。
Ruijie(config)# **spanning-tree max-hops** *hop-count*	配置 Maximum-Hop Count，范围为 1－40，缺省值为 20

Ruijie(config)# **end**	退回到特权模式。
Ruijie# **show running-config**	核对配置条目。
Ruijie# **copy running-config startup-config**	保存配置。

您如果要恢复到缺省值，可用 **no spanning-tree max-hops** 全局配置命令进行设置。

16.3.16 配置接口的兼容性模式

配置接口的兼容性模式，可以使该端口发送 BPDU 时根据当前端口的属性有选择的携带不同的 MSTI 的信息，以实现与其它厂商之间的互连。

进入特权模式，按以下步聚配置接口的兼容性模式：

命令	作用
Ruijie# **configure terminal**	进入全局配置模式。
Ruijie(config)#**interface** *interface-id*	进入接口配置模式
Ruijie(config-if)# **spanning-tree compatible enable**	打开接口的兼容性模式
Ruijie(config-if)# **end**	退回到特权模式。
Ruijie# **show running-config**	核对配置条目。
Ruijie# **copy running-config startup-config**	保存配置。

您如果要取消该配置，可用 **no spanning-tree compatible enable** 接口配置命令进行设置。

16.4 配置 MSTP 可选特性

16.4.1 缺省的生成树可选特性设置

边缘口的自动识别功能(AutoEdge 功能)默认打开。
其它可选特性的缺省值默认都是关闭的。

16.4.2 打开 Port Fast

打开 Port Fast 后该端口会直接 Forwarding。但会因为收到 BPDU 而使 Port Fast

Operational State 为 disabled，从而正常的参与 STP 算法而 Forwarding。

进入特权模式，按以下步骤配置 Port Fast：

命令	作用
Ruijie**# configure terminal**	进入全局配置模式。
Ruijie(config)**# interface** *interface-id*	进入该 interface 的配置模式，合法的 interface 包括物理端口和 Aggregate Link。
Ruijie(config-if)**# spanning-tree portfast**	打开该 interface 的 portfast。
Ruijie(config-if)**# end**	退回到特权模式。
Ruijie**# show spanning-tree interface** *interface-id* **portfast**	核对配置条目。
Ruijie**# copy running-config startup-config**	保存配置。

您如果要关闭 Port Fast，在 Interface 配置模式下用 **spanning-tree portfast disable** 命令进行设置。

您可以用全局配置命令 **spanning-tree portfast default** 来打开所有端口的 Portfast。

16.4.3 关闭边缘口的自动识别

如果在一定的时间范围内(为 3 秒)，如果指派口没有收到 BPDU，则自动识别为边缘口。但会因为收到 BPDU 而使 Port Fast Operational State 为 disabled，该功能缺省是打开的。

进入特权模式，按以下步骤配置 Autoedge：

命令	作用
Ruijie**# configure terminal**	进入全局配置模式。
Ruijie(config)**# interface** *interface-id*	进入该 interface 的配置模式，合法的 interface 包括物理端口和 Aggregate Link。
Ruijie(config-if)**# spanning-tree autoedge**	打开该 interface 的 autoedge。
Ruijie(config-if)**# end**	退回到特权模式。
Ruijie**# show spanning-tree interface** *interface-id*	核对配置条目。

Ruijie**# copy running-config startup-config**	保存配置。

您如果要关闭 Autoedge，在 Interface 配置模式下用 **spanning-tree autoedge disabled** 命令进行设置。

16.4.4 打开 BPDU Guard

端口打开 BPDU Guard 后，如果在该端口上收到 BPDU，则会进入 Error-disabled 状态。

进入特权模式，按以下步骤配置 BPDU Guard：

命令	作用
Ruijie**# configure terminal**	进入全局配置模式。
Ruijie(config)**# spanning-tree portfast Bpduguard default**	全局的打开 BPDU guard
Ruijie(config)**# interface** *interface-id*	进入该 interface 的配置模式，合法的 interface 包括物理端口和 Aggregate Link。
Ruijie(config-if)**# spanning-tree portfast**	打开该 interface 的 portfast，全局的 bpduguard 配置才生效。
Ruijie(config-if)**# end**	退回到特权模式。
Ruijie**# show running-config**	核对配置条目。
Ruijie**# copy running-config startup-config**	保存配置。

您如果要关闭 BPDU Guard，可在全局配置命令 **no spanning-tree portfast bpduguard default** 进行设置。

您如果要针对单个 Interface 打开 BPDU Guard，您可用 Interface 配置命令 **spanning-tree bpduguard enable** 进行设置，用 **spanning-tree bpduguard disable** 关闭 BPDU guard。

16.4.5 打开 BPDU Filter

打开 BPDU Filter 后，相应端口会既不发，也不收 BPDU。

进入特权模式，按以下步骤配置端口 BPDU Filter：

命令	作用

Ruijie**# configure terminal**	进入全局配置模式。
Ruijie(config)**# spanning-tree portfast bpdufilter default**	全局的打开 BPDU filter
Ruijie(config)**# interface** *Interface-id*	进入该interface的配置模式，合法的interface包括物理端口和 Aggregate Link。
Ruijie(config-if)**# spanning-tree Portfast**	打开该 interface 的 portfast，全局的 bpdufilter 配置才生效。
Ruijie(config-if)**# end**	退回到特权模式。
Ruijie**# show running-config**	核对配置条目。
Ruijie**# copy running-config startup-config**	保存配置。

您如果要关闭 BPDU Filter，可以用全局配置命令 **no spanning-tree portfast bpdufilter default** 进行设置。

您如果要针对单个 Interface 打开 BPDU Filter，您可以用 Interface 配置命令 **spanning-tree bpdufilter enable** 进行设置，用 **spanning-tree bpdufilter disable** 关闭 BPDU Guard。

16.4.6 打开 Tc_Protection

进入特权模式，按以下步骤配置 Tc_Protection：

命令	作用
Ruijie**# configure terminal**	进入全局配置模式。
Ruijie(config)**# spanning-tree tc-protection**	打开 tc-protection
Ruijie(config)**# end**	退回到特权模式。
Ruijie**# show running-config**	核对配置条目。
Ruijie**# copy running-config startup-config**	保存配置。

您如果要关闭 Tc_Protection，可以用全局配置命令 **no spanning-tree tc-protection** 进行设置。

16.4.7 打开 TC Guard

进入特权模式，按以下步骤配置全局的 TC Guard

命令	作用
Ruijie**# configure terminal**	进入全局配置模式。
Ruijie(config)**# spanning-tree tc-protection tc-guard**	打开全局的 TC Guard
Ruijie(config)**# end**	退回到特权模式。
Ruijie**# show running-config**	核对配置条目。
Ruijie**# copy running-config startup-config**	保存配置。

进入特权模式，按以下步骤配置接口下的 TC Guard

命令	作用
Ruijie**# configure terminal**	进入全局配置模式。
Ruijie(config)**# interface** *Interface-id*	进入该 interface 的配置模式，合法的 interface 包括物理端口和 Aggregate Link。
Ruijie(config-if)**# spanning-tree tc-guard**	打开该 interface 的 TC Guard。
Ruijie(config-if)**# end**	退回到特权模式。
Ruijie**# show running-config**	核对配置条目。
Ruijie**# copy running-config startup-config**	保存配置。

16.4.8 打开 BPDU 源 MAC 检查

打开 BPDU 源 MAC 检查，将只接受源 MAC 地址为指定 MAC 的 BPDU 帧，过滤掉其它所有接收的 BPDU 帧。

进入接口模式，您可以按以下步骤配置 BPDU 源 MAC 检查：

命令	作用
Ruijie**# configure terminal**	进入全局配置模式。
Ruijie(config)**# interface** *Interface-id*	进入该interface的配置模式，合法的interface包括物理端口和 Aggregate Link。
Ruijie(config-if)**#bpdu src-mac-check** *H.H.H*	打开 bpdu 源 mac 检查。
Ruijie(config-if)**# end**	退回到特权模式。
Ruijie**# show running-config**	核对配置条目。

Ruijie**# copy running-config startup-config**	保存配置。

您如果要关闭bpdu源mac检查，可以用接口下配置命令 **no bpdu src-mac-check** 进行设置。

16.4.9 打开 Root Guard

进入特权模式，按以下步骤配置接口的 ROOT Guard

命令	作用
Ruijie**# configure terminal**	进入全局配置模式。
Ruijie(config)**# interface Interface-id**	进入该 interface 的配置模式，合法的 interface 包括物理端口和 Aggregate Link。
Ruijie(config-if)**# spanning-tree guard root**	打开接口的 ROOT Guard 特性
Ruijie(config-if)**# end**	退回到特权模式。
Ruijie**# show running-config**	核对配置条目。
Ruijie**# copy running-config startup-config**	保存配置。

16.4.10 打开 Loop Guard

进入特权模式，按以下步骤配置全局的 Loop Guard

命令	作用
Ruijie**# configure terminal**	进入全局配置模式。
Ruijie(config)**# spanning-tree Loopguard default**	打开全局的 LOOP Guard。
Ruijie(config)**# end**	退回到特权模式。
Ruijie**# show running-config**	核对配置条目。
Ruijie**# copy running-config startup-config**	保存配置。

进入特权模式，按以下步骤配置接口下的 LOOP Guard

命令	作用
Ruijie**# configure terminal**	进入全局配置模式。

Ruijie(config)# **interface** *Interface-id*	进入该 interface 的配置模式，合法的 interface 包括物理端口和 Aggregate Link。
Ruijie(config-if)# **spanning-tree guard loop**	打开该 interface 的 Loop Guard。
Ruijie(config-if)# **end**	退回到特权模式。
Ruijie# **show running-config**	核对配置条目。
Ruijie# **copy running-config startup-config**	保存配置。

16.4.11 关闭接口的保护功能

进入特权模式，按以下步骤关闭接口的根或环路保护功能。

命令	作用
Ruijie# **configure terminal**	进入全局配置模式。
Ruijie(config)# **interface** *Interface-id*	进入该 interface 的配置模式，合法的 interface 包括物理端口和 Aggregate Link。
Ruijie(config-if)# **spanning-tree guard none**	关闭接口的 guard 功能
Ruijie(config-if)# **end**	退回到特权模式。
Ruijie# **show running-config**	核对配置条目。
Ruijie# **copy running-config startup-config**	保存配置。

16.5 显示 MSTP 配置和状态

MSTP 提供了如下的显示命令命令用于查看各种配置信息及运行时信息，各命令的功能说明如下：

命令	含义
Ruijie# **show spanning-tree**	显示 MSTP 的各项参数信息及生成树的拓扑信息
Ruijie# **show spanning-tree summary**	显示 MSTP 的各 instance 的信息及其端口转发状态信息
Ruijie# **show spanning-tree inconsistentports**	显示因根保护或环路保护而 block 的端口
Ruijie# **show spanning-tree mst configuration**	显示 MST 域的配置信息
Ruijie# **show spanning-tree mst** *instance-id*	显示该 instance 的 MSTP 信息

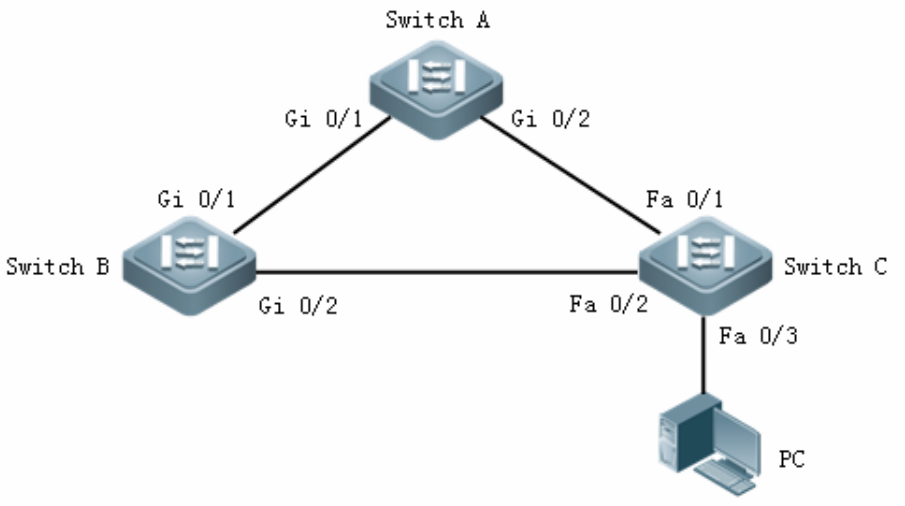
Ruijie# **show spanning-tree mst** *instance-id* **interface** *interface-id*	显示指定 interface 的对应 instance 的 MSTP 信息
Ruijie# **show spanning-tree interface** *interface-id*	显示指定 interface 的所有 instance 的 MSTP 信息
Ruijie# **show spanning-tree forward-time**	显示 forward-time
Ruijie# **show spanning-tree Hello time**	显示 Hello time
Ruijie# **show spanning-tree max-hops**	显示 max-hops
Ruijie# **show spanning-tree tx-hold-count**	显示 tx-hold-count
Ruijie# **show spanning-tree pathcost method**	显示 pathcost method

16.6 MSTP 典型配置用例

16.6.1 组网需求

1) 将三台设备连接成三角环网，配置 MSTP 模式。
2) 在设备上配置相应的 Vlan-Instance 映射，以及 MST 配置名称、MST Revision Number，并指定相应设备的实例优先级
3) 查看 MSTP 配置信息
4) 全局开启 BPDU Guard 功能，在直连 PC 的端口上配置 Port Fast 功能。

16.6.2 组网拓扑



16.6.3 配置步骤

1） **配置 Switch A**

配置端口 Gi 0/1 和 Gi 0/2 属于 Trunk 口，创建 VLAN 2 和 VLAN 3。

```
Ruijie# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Ruijie(config)# interface gigabitEthernet 0/1
Ruijie(config-if)# switchport mode trunk
Ruijie(config-if)# exit
Ruijie(config)# interface gigabitEthernet 0/2
Ruijie(config-if)# switchport mode trunk
Ruijie(config-if)# exit
Ruijie(config)# vlan 2
Ruijie(config-vlan)# exit
Ruijie(config)# vlan 3
Ruijie(config-vlan)# exit
```

配置生成树为 MSTP 模式，并将 VLAN 2 映射到 Instance 1，将 VLAN 3 映射到 Instance 2，设置 MST 配置名称为 ruijie，MST Revision Number 为 1，查看 MST 配置信息，开启生成树协议。

```
Ruijie(config)# spanning-tree mode mstp
Ruijie(config)# spanning-tree mst configuration
Ruijie(config-mst)# instance 1 vlan 2
%Warning:you must create vlans before configuring
instance-vlan relationship
Ruijie(config-mst)# instance 2 vlan 3
%Warning:you must create vlans before configuring
instance-vlan relationship
Ruijie(config-mst)# name ruijie
Ruijie(config-mst)# revision 1
Ruijie(config-mst)# show
Multi spanning tree protocol : Enable
Name    : ruijie
Revision : 1
Instance  Vlans Mapped
-------- ---------------------------------------------
0       : 1, 4-4094
1       : 2
2       : 3
---------------------------------------------------
Ruijie(config-mst)# exit
Ruijie(config)# spanning-tree
Enable spanning-tree.
```

在该设备上配置 Instance 0 的优先级为 4096

```
Ruijie(config)# spanning-tree mst 0 priority 4096
```

2) **配置 Switch B**

配置端口 Gi 0/1 和 Gi 0/2 属于 Trunk 口，创建 VLAN 2 和 VLAN 3。

```
Ruijie(config)# interface gigabitEthernet 0/1
Ruijie(config-if)# switchport mode trunk
Ruijie(config-if)# exit
Ruijie(config)# interface gigabitEthernet 0/2
Ruijie(config-if)# switchport mode trunk
Ruijie(config-if)# exit
Ruijie(config)# vlan 2
Ruijie(config-vlan)# exit
Ruijie(config)# vlan 3
Ruijie(config-vlan)# exit
```

配置生成树为 MSTP 模式，并将 VLAN 2 映射到 Instance 1，将 VLAN 3 映射到 Instance 2，设置 MST 配置名称为 ruijie，MST Revision Number 为 1，开启生成树协议。

```
Ruijie(config)# spanning-tree mode mstp
Ruijie(config)# spanning-tree mst configuration
Ruijie(config-mst)# instance 1 vlan 2
%Warning:you must create vlans before configuring
instance-vlan relationship
Ruijie(config-mst)# instance 2 vlan 3
%Warning:you must create vlans before configuring
instance-vlan relationship
Ruijie(config-mst)# name ruijie
Ruijie(config-mst)# revision 1
Ruijie(config-mst)# exit
Ruijie(config)# spanning-tree
Enable spanning-tree.
```

在该设备上配置 Instance1 的优先级为 4096

```
Ruijie(config)# spanning-tree mst 1 priority 4096
```

3) 配置 Switch C

配置端口 Fa 0/1 和 Fa 0/2 属于 Trunk 口，创建 VLAN 2 和 VLAN 3。

```
Ruijie(config)# interface fastEthernet 0/1
Ruijie(config-if)# switchport mode trunk
Ruijie(config-if)# exit
Ruijie(config)# interface fastEthernet 0/2
Ruijie(config-if)# switchport mode trunk
Ruijie(config-if)# exit
Ruijie(config)# vlan 2
```

```
Ruijie(config-vlan)# exit
Ruijie(config)# vlan 3
Ruijie(config-vlan)# exit
```

配置生成树为 MSTP 模式，并将 VLAN 2 映射到 Instance 1，将 VLAN 3 映射到 Instance 2，设置 MST 配置名称为 ruijie，MST Revision Number 为 1，开启生成树协议。

```
Ruijie(config)# spanning-tree mode mstp
Ruijie(config)# spanning-tree mst configuration
Ruijie(config-mst)# instance 1 vlan 2
%Warning:you must create vlans before configuring instance-vlan relationship
Ruijie(config-mst)# instance 2 vlan 3
%Warning:you must create vlans before configuring instance-vlan relationship
Ruijie(config-mst)# name ruijie
Ruijie(config-mst)# revision 1
Ruijie(config-mst)# exit
Ruijie(config)# spanning-tree
Enable spanning-tree.
```

在该设备上配置 Instance 2 的优先级最高

```
Ruijie(config)# spanning-tree mst 2 priority 4096
```

全局开启 BPDU Guard 功能，并配置端口 Fa 0/3 为 Port Fast。

```
Ruijie(config)# spanning-tree portfast bpduguard default
Ruijie(config)# interface fastEthernet 0/3
Ruijie(config-if)#spanning-tree portfast
%Warning: portfast should only be enabled on ports connected to a single host. Connecting hubs, Ruijiees, bridges to this interface when portfast is enabled,can cause temporary loops.
Ruijie(config-if)# end
```

查看该设备的生成树配置信息

```
Ruijie# show spanning-tree
StpVersion : MSTP
SysStpStatus : ENABLED
MaxAge : 20
HelloTime : 2
ForwardDelay : 15
BridgeMaxAge : 20
BridgeHelloTime : 2
BridgeForwardDelay : 15
MaxHops: 20
TxHoldCount : 3
PathCostMethod : Long
```

```
BPDUGuard : enabled
BPDUFilter : Disabled
LoopGuardDef  : Disabled
###### mst 0 vlans map : 1, 4-4094
BridgeAddr : 00d0.f82a.aa8e
Priority: 32768
TimeSinceTopologyChange : 0d:0h:19m:44s
TopologyChanges : 1
DesignatedRoot : 1000.00d0.f822.33aa
RootCost : 0
RootPort : 1
CistRegionRoot : 1000.00d0.f822.33aa
CistPathCost : 200000
###### mst 1 vlans map : 2
BridgeAddr : 00d0.f82a.aa8e
Priority: 32768
TimeSinceTopologyChange : 0d:0h:1m:46s
TopologyChanges : 7
DesignatedRoot : 1001.00d0.f834.56f0
RootCost : 200000
RootPort : 2
###### mst 2 vlans map : 3
BridgeAddr : 00d0.f82a.aa8e
Priority: 4096
TimeSinceTopologyChange : 0d:0h:1m:44s
TopologyChanges : 5
DesignatedRoot : 1002.00d0.f82a.aa8e
RootCost : 0
RootPort : 0
```

查看端口 Fa 0/1 的生成树配置信息

```
Ruijie# show spanning-tree interface fastEthernet 0/1
PortAdminPortFast : Disabled
PortOperPortFast : Disabled
PortAdminAutoEdge : Enabled
PortOperAutoEdge : Disabled
PortAdminLinkType : auto
PortOperLinkType : point-to-point
PortBPDUGuard : Disabled
PortBPDUFilter : Disabled
PortGuardmode  : None
###### MST 0 vlans mapped :1, 4-4094
PortState : forwarding
PortPriority : 128
PortDesignatedRoot : 1000.00d0.f822.33aa
PortDesignatedCost : 0
```

```
PortDesignatedBridge :1000.00d0.f822.33aa
PortDesignatedPort : 8002
PortForwardTransitions : 1
PortAdminPathCost : 200000
PortOperPathCost : 200000
Inconsistent states : normal
PortRole : rootPort
###### MST 1 vlans mapped :2
PortState : discarding
PortPriority : 128
PortDesignatedRoot : 1001.00d0.f834.56f0
PortDesignatedCost : 0
PortDesignatedBridge :8001.00d0.f822.33aa
PortDesignatedPort : 8002
PortForwardTransitions : 5
PortAdminPathCost : 200000
PortOperPathCost : 200000
Inconsistent states : normal
PortRole : alternatePort
###### MST 2 vlans mapped :3
PortState : forwarding
PortPriority : 128
PortDesignatedRoot : 1002.00d0.f82a.aa8e
PortDesignatedCost : 0
PortDesignatedBridge :1002.00d0.f82a.aa8e
PortDesignatedPort : 8001
PortForwardTransitions : 1
PortAdminPathCost : 200000
PortOperPathCost : 200000
Inconsistent states : normal
PortRole : designatedPort
```

17 SPAN 配置

17.1 概述

17.1.1 理解 SPAN

您可以通过使用 SPAN 将一个端口上的帧拷贝到交换机上的另一个连接有网络分析设备或 RMON 分析仪的端口上来分析该端口上的通讯。SPAN 将某个端口上所有接收和发送的帧 MIRROR 到某个物理端口上来进行分析。

例如，在图 1 中，千兆端口 5 上的所有帧都被映射到了千兆端口 10，接在端口 10 上的的网络分析仪虽然没有和端口 5 直接相连，可接收通过端口 5 上的所有帧。

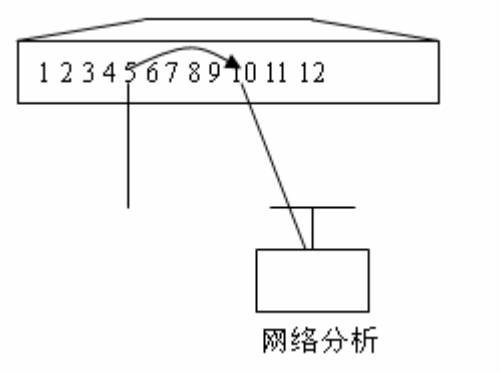


图 1 SPAN 配置实例

通过 SPAN 可以监控所有进入和从源端口输出的帧，包括路由输入帧。

SPAN 并不影响源端口和目的端口的交换，只是所有进入和从源端口输出的帧原样拷贝了一份到目的端口。然而一个流量过度的目的端口，例如一个 100Mbps 目的端口监控一个 1000Mbps 可能导致帧被丢弃。

17.2 SPAN 概念和术语

该部分描述了和 SPAN 配置相关的一些概念和术语

17.2.1 SPAN 会话

一个 SPAN 会话是一个目的端口和源端口的组合。您可以监控单个或多个接口的输入，输出和双向帧。

Switched port、routed port 和 AP 都可以配置为源端口和目的端口。SPAN 会话并不影响交换机的正常操作。

您可以将 SPAN 会话配置在一个 disabled port 上，然而，SPAN 并不马上发生作用直到您使能目的和源端口。**Show monitor session** *session number* 命令显示了 SPAN 会话的操作状态。一个 SPAN 会话在上电后并不马上生效直到目的端口处于可操作状态。

17.2.2 帧类型

SPAN 会话包含以下帧类型：

接收帧：所有源端口上接收到的帧都将被拷贝一份到目的口。在一个 SPAN 会话中，您可监控一个或几个源端口的输入帧。若由于某些原因，从源端口输入的帧可能被丢弃，如端口安全，但这不影响 SPAN 的功能，该帧仍然会发送到目的端口。

发送帧：所有从源端口发送的帧都将拷贝一份到目的端口。在一个 SPAN 会话中，您可监控一个或几个源端口的输出帧。若由于某些原因，从别的端口发送到源端口的帧可能被丢弃，同样，该帧也不会发送到目的端口。由于某些原因发送到源端口的帧的格式可能改变，例如源端口输出经过路由之后的帧，帧的源 MAC、目的 MAC、VLAN ID 以及 TTL 发生变化。同样，拷贝到目的端口的帧的格式也会变化。

双向帧：包括上面所说的两种帧。在一个 SPAN 会话中，您可监控一个或几个源端口的输入和输出帧。

17.2.3 源端口

源端口(也叫被被监控口)是一个 switched port、routed port 或 AP，该端口被监控用做网络分析。在单个的 SPAN 会话中，您可以监控输入，输出和双向帧，对于源端口的最大个数不做限制。

一个源端口有以下特性：

1) 它可以是 switched port，routed port 或 AP。

2) 它不可以同时为目的口。

3) 它可以指定被监控帧的输入或输出方向。

4) 源端口和目的端口可以处于一个 VLAN 或不处于一个 VLAN 中。

17.2.4 目的端口

SPAN 会话有一个目的口(也叫监控口)，用于接收源端口的帧拷贝。

目的端口有以下特性：

- 它可以是 switched port、routed port 和 AP。

注意:

S3760 系列交换机中，目的端口接收到的源端口的帧拷贝不携带帧原有的 tag。

17.2.5 SPAN Traffic

您可以使用 SPAN 监控所有网络通讯，包括多播帧，BPDU 帧等。

17.2.6 SPAN 和其他功能的接口

SPAN 和以下功能交互：

- Spanning Tree Protocol(STP)—SPAN 的目的口参与 STP。

17.3 配置 SPAN

本部分描述了如何在您的交换机上配置 SPAN，它包含以下配置:

17.3.1 SPAN 缺省状态

功能	缺省配置
SPAN 状态	禁止

17.3.2 SPAN 配置指南

请遵循以下规则进行 SPAN 的配置：

请将网络分析仪连接到监控口。

目的端口不能为源端口，源端口不能为目的端口。

您可以配置一个 disabled port 为目的端口或源端口，但此时 SPAN 功能并不起作用，直到目的端口和源端口被重新使能。

no monitor session *session_number* 全局配置命令可从 SPAN 会话中删除源或目的端口。

SPAN 目的端口参与 STP。

当 SPAN 处于使能状态，配置的变更有以下结果：

1) 如果您改变了源端口的 VLAN 配置，该配置将马上生效。

2) 如果您改变了目的端口的 VLAN 配置，该配置马上生效。

3) 如果您禁止了源端口或目的端口，SPAN 将不起作用。

4) 如果您将源端口或者目的端口加入到一个 AP 中，该配置将使 SPAN 的源端口或者目的端口被取消。

17.3.3 创建一个 SPAN 会话并指定监控口和被监控口

创建一个 SPAN 会话并指定目的端口(监控口)和源端口(被监控口)。

命令	作用
Ruijie(config)**# monitor session** *session_number* **source interface** *interface-id* [**,** | **-**] {**both** | **rx** | **tx**}	指定源端口。对于 *interface-id*，请指定相应的接口号。
Ruijie(config)**# monitor session** *session_number* **destination interface** *interface-id* [**switch**]	指定目的端口。对于 *interface-id*，请指定相应的接口号 添加 **switch** 参数将支持镜像目的口交换功能。

想要删除 SPAN 会话，可使用 **no monitor session** *session_number* 全局配置命令。想要删除所有 SPAN 会话，可使用 **no monitor session all** 全局配置命令。使用 **no monitor session session_number source interface** *interface-id* 全局配置命令或 **no monitor session** *session_number* **destination interface** *interface-id* 可删除源端口或目的端口

下面这个例子说明了如何创建一个 SPAN 会话：会话 1。首先，将当前会话 1 的配置清除掉，然后设置端口 1 的帧 MIRROR 到端口 8。**Show monitor session** 特权命令用于确认配置。

```
Ruijie(config)# no monitor session 1
Ruijie(config)# monitor session 1 source interface
gigabitEthernet 3/1 both
Ruijie(config)# monitor session 1 destination interface
gigabitEthernet 3/8
Ruijie(config)# end
Ruijie# show monitor session 1
sess-num: 1
src-intf:
GigabitEthernet 3/1 frame-type Both
dest-intf:
GigabitEthernet 3/8
```

注意:

S3760 系列交换机仅支持一个会话。

17.3.4 从 SPAN 会话中删除一个口

从一个 SPAN 会话中删除源端口。

命令	作用
Ruijie(config)# **no monitor session** *session_number* **source interface** *interface-id* [, | -] [**both** | **rx** | **tx**]	指定需删除的源端口。对于 interface-id，请指定相应的接口号。

使用 **no monitor session** *session_number* **source interface** *interface-id* 全局配置命令可从一个 SPAN 会话中删除源端口。下面这个例子显示了如何将端口 1 从会话 1 中删除并确认配置。

```
Ruijie(config)# no monitor session 1 source interface
gigabitethernet 1/1 both
Ruijie(config)# end
Ruijie# show monitor session 1
sess-num: 1
dest-intf:
GigabitEthernet 3/8
```

17.4 显示 SPAN 状态

使用 **show monitor** 特权命令可显示当前 SPAN 配置的状态，下面这个例子说明了如何通过 **show monitor** 特权命令显示 SPAN 会话 1 的当前状态

```
Ruijie# show monitor session 1
sess-num: 1
src-intf:
GigabitEthernet 3/1 frame-type Both
dest-intf:
GigabitEthernet 3/8
```

18 RSPAN 配置

本章描述锐捷设备的 RSpan 配置

18.1 概述

RSPAN 是 SPAN 的扩展，能够远程监控多台设备，每个 RSPAN Session 建立于用户指定的 RSPAN Vlan 内。远程镜像突破了被镜像端口和镜像端口必须在同一台设备上的限制，使被镜像端口和镜像端口间可以跨越多个网络设备，这样维护人员就可以坐在中心机房通过分析仪观测远端被镜像端口的数据报文了。

RSPAN 实现的功能是将所有的被镜像报文通过一个特殊的 RSPAN Vlan 传递到远端的镜像端口，设备组网图如下图所示：

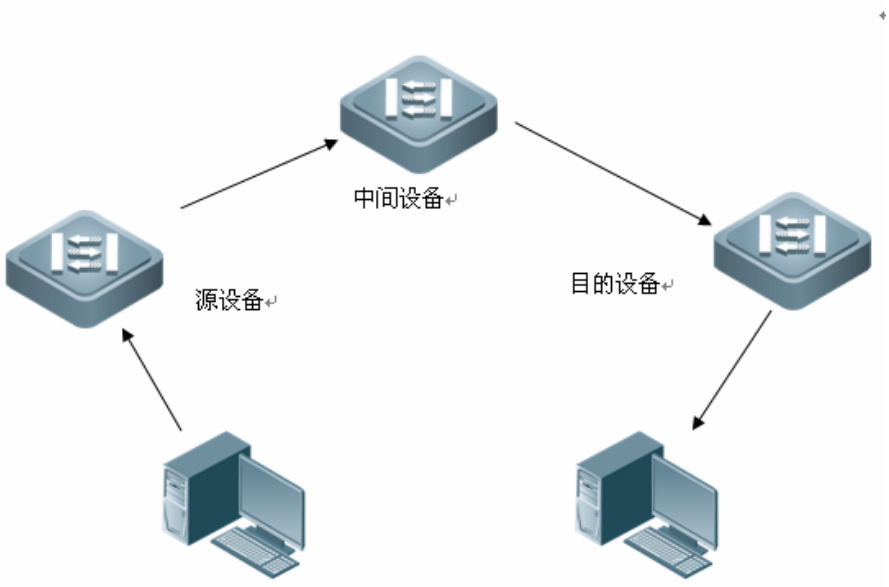


图 1

实现了远程端口镜像功能的交换机分为三种：

- 源交换机：被监测的端口所在的交换机，负责将镜像流量复制到 Remote VLAN 中，然后转发给中间交换机或目的交换机。
- 中间交换机：网络中处于源交换机和目的交换机之间的交换机，通过 Remote VLAN 把镜像流量传输给下一个中间交换机或目的交换机。如果源交换机与目的交换机直接相连，则不存在中间交换机。
- 目的交换机：远程镜像目的端口所在的交换机，将从 Remote VLAN 接收到的镜像流量通过镜像目的端口转发给监控设备。

各个交换机上参与镜像的端口如下所示。

交换机	参与镜像的端口	作用
源交换机	源端口（Source Port）	被监测的用户端口，通过本地端口镜像把用户数据报文复制到指定的输出端口或者反射端口（Reflector port），源端口可以有多个
	反射端口（Reflector port）	接收本地端口镜像的用户数据报文
	输出端口	将镜像报文发送到中间交换机或者目的交换机
中间交换机	普通端口	将镜像报文发送到目的交换机。 建议中间交换机上配置两个 Trunk 端口，和两侧的设备相连
目的交换机	源端口	接收远程镜像报文
	镜像目的端口（Destination port）	远程镜像报文的监控端口

为了实现远程端口镜像功能，需要定义一个特殊的 VLAN，称之为 Remote VLAN。这个 VLAN 只传输镜像报文，不能用来承载正常的业务数据。所有被镜像的报文通过该 VLAN 从源交换机传递到目的交换机的指定端口，实现在目的交换机上对源交换机的远程端口的报文进行监控的功能。

注意：

1) 推荐用户将镜像源端口和反射口设置于不同 Vlan 内

2) 不支持 AP 口设置为反射口

3) Remote-span Vlan 不能是 Vlan 1，也不能是 Private Vlan

4) Remote-span Vlan 不参与 GVRP

5) 若设备上开启 STP，建议在属于 remote vlan 的端口上配置 bpdu filter，以防止端口变成 block 状态，影响镜像报文转发。由于配置 bpdu filter 后将无法收发 bpdu 报文，故请用户注意避免在 Remote vlan 上产生环路。

18.2 配置 RSPAN 会话

18.2.1 配置准备

- 确定了源交换机、中间交换机、目的交换机
- 确定了镜像源端口、反射端口、镜像目的端口、Remote VLAN
- 通过配置保证了 Remote VLAN 内从源交换机到目的交换机的二层互通性

- 确定了被监控报文的方向
- 启动了 Remote VLAN

18.2.2 源交换机上的配置过程

命令	作用
Ruijie# **configure terminal**	进入全局配置模式
Ruijie(config)# **vlan** *vlan-id*	进入 Vlan 配置模式
Ruijie(config-van)# **remote-span**	设置 Vlan 为 remote-span Vlan
Ruijie(config-vlan)# **exit**	退到全局配置模式
Ruijie(config)# **monitor session** *session_num* **remote-source**	配置远程源镜像
Ruijie(config)# **monitor session** *session-num* **source interface** *interface-name* [**rx** | **tx** | **both**]	配置远程镜像源端口（源口的 rx，tx 可以配置到同一个目的口，也可以配置到不同的目的口，但每一个只能配置到一个目的口）
Ruijie(config)#**monitor session** *session_num* **destination remote vlan** *remote_vlan-id* [**reflector-port**] **interface** *interface-name* [**switch**] Ruijie(config)# **monitor session** *session_number* **source interface** *interface-id* **rx acl** *name*	配置远程源镜像组的 Remote VLAN 和远程反射端口 Switch 关键字表示目的口参与交换 设定需要镜像的流所匹配的 acl name

注意:

- 反射端口无法作为正常的端口转发流量，所以建议用户将没有使用的处于 DOWN 状态的端口（没有接网线的端口）配置为反射端口，且不要在该端口上添加其它配置。
- 建议用户不要将普通端口加入 Remote VLAN。
- 不要在与中间交换机或目的交换机相连的端口上配置镜像源端口，否则可能引起网络内的流量混乱。
- 在一个 RSPAN 会话中，中间交换机如果有使用非-E 线卡的端口作为转发口，源交换机在同一时刻只能配置 RX 镜像或者 TX 镜像。此时，如果源设备进行 RX 镜像和 TX 镜像的交替配置时，需要去中间交换机上执行清除 remote-span vlan 内 mac 地址的操作。
- 若要删除大量连续的 RSPAN（如超过 200 个），不建议进行批量删除（如在 VLAN RANGE 100 – 1500 的模式下进行删除），这会严重影响系统的处理效率。
- S3760 系列交换机中，源交换机的输出端口所发送的源端口的帧拷贝不携带

帧原有的 tag。

- 在 S3760 系列产品中，对只送到 CPU 的一些报文（如 RLDP 报文，DHCP 报文等），不能进行远程端口镜像。

18.2.3 中间交换机上的配置过程

命令	作用
Ruijie# **configure**	进入全局配置模式
Ruijie(config)# **vlan** *vlan-id*	进入 Vlan 配置模式
Ruijie(config-vlan)# **remote-span**	设置 Vlan 为 remote-span Vlan
Ruijie(config-vlan)# **exit**	退到全局配置模式

18.2.4 目的交换机上的配置过程

命令	作用
Ruijie# **configure**	进入全局配置模式。
Ruijie(config)# **vlan** *vlan-id*	进入 Vlan 配置模式。
Ruijie(config-vlan)# **remote-span**	设置 Vlan 为 remote-span Vlan。
Ruijie(config-vlan)# **exit**	退到全局配置模式。
Ruijie(config)# **monitor session** *session_num* **remote-destination**	配置远程目的镜像。
Ruijie(config)# **monitor session** *session-num* **destination remote vlan** *vlan-id* **interface** *interface-name* [**switch**]	配置 Remote VLAN 和远程镜像目的端口，Switch 关键字表示目的口参与交换。
Ruijie(config)# **interface** *interface-name*	进入远程镜像目的端口。
Ruijie(config-if)#{**switchport access vlan** *vid* | **switchport trunk native vlan** *vid* }	vid 表示 remote-span vlan 的 vid，如果目的口是 access 口，则把它加入 remote-span vlan，如果目的口是 trunk 口，则把它加入 remote-span vlan，并且将 remote-span vlan 设置成它的 native vlan。

注意：

S3760 系列交换机的端口作为 RSPAN 的目的端口时，无论是否配置端口的 switch 属性(见命令：**monitor session** *session-num* **destination remote vlan** *vlan-id*

interface *interface-name* [**switch**])，该端口总是参与交换。

18.3 显示 RSPAN 会话

命令	作用
Ruijie**# show monitor**	显示镜像

示例：

```
Ruijie# show monitor
sess-num: 1
src-intf:
GigabitEthernet 0/4 frame-type Both
dest-intf:
GigabitEthernet 0/6
remote vlan 3
```

18.4 配置举例

设备拓扑如下所示：

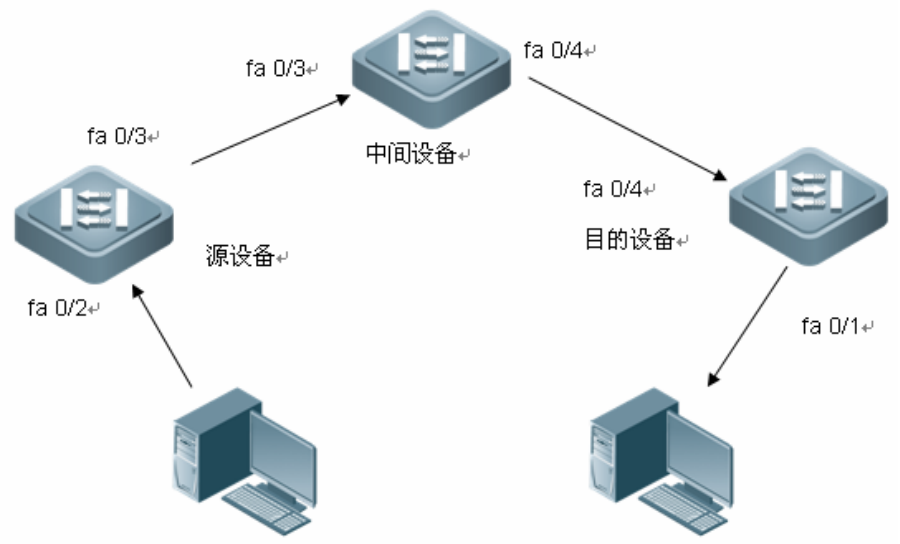


图 2：

源交换机配置：

```
Ruijie# configure
Ruijie(config)# vlan 7
Ruijie(config-vlan)# remote-span
Ruijie(config-vlan)# exit
```

```
Ruijie(config)#Interface fastEthernet 0/3
Ruijie(config-if)#switchport mode trunk
Ruijie(config-if)#switchport trunk allowed vlan add 7
Ruijie(config-if)# exit
Ruijie(config)# monitor session 1 remote-source
Ruijie(config)# monitor session 1 source interface
fastEthernet 0/2
Ruijie(config)#Interface fastEthernet 0/1
Ruijie(config-if)#switchport access vlan 7
Ruijie(config)#monitor session 1 destination remote vlan 7
reflector-port interface fastEthernet 0/1 switch
```

中间交换机配置：

```
Ruijie# configure
Ruijie(config)# vlan 7
Ruijie(config-vlan)# remote-span
Ruijie(config-vlan)# exit
Ruijie(config)#Interface  fastEthernet 0/3
Ruijie(config-if)#switchport mode trunk
Ruijie(config-if)#switchport trunk allowed vlan add 7
Ruijie(config-if)#exit
Ruijie(config)#Interface  fastEthernet 0/4
Ruijie(config-if)#switchport mode trunk
Ruijie(config-if)#switchport trunk allowed vlan add 7
```

目的交换机配置：

```
Ruijie# configure
Ruijie(config)# vlan 7
Ruijie(config-vlan)# remote-span
Ruijie(config-vlan)# exit
Ruijie(config)#Interface fastEthernet 0/4
Ruijie(config-if)#switchport mode trunk
Ruijie(config-if)#switchport trunk allowed vlan add 7
Ruijie(config-if)# exit
Ruijie(config)# monitor session 1 remote- destination
Ruijie(config)#monitor session 1 destination remote vlan 7
interface fastEthernet 0/1 switch
```

19 IP 地址与服务配置

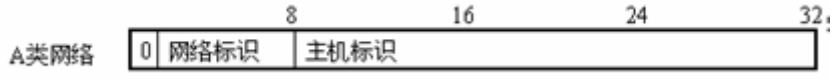
19.1 IP 地址配置

19.1.1 IP 地址简介

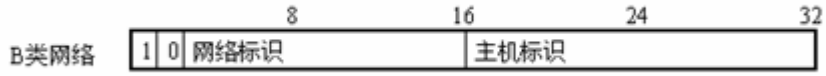
IP 地址由 32 位二进制组成，为了书写和描述方便，一般用十进制表示。十进制表示时，分为四组，每组 8 位，范围从 0~255，组之间用“.”号隔开，比如“192.168.1.1”就是用十进制表示的 IP 地址。

IP 地址顾名思义，自然是 IP 层协议的互连地址。32 位的 IP 地址由两个部分组成：1）网络部分；2）本地地址部分。根据网络部分的头几个比特位的值，目前使用中的 IP 地址可以划分成四大类。

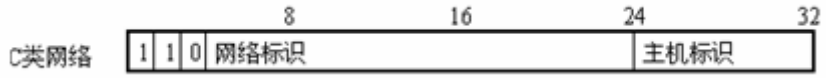
A 类地址，最高比特位为“0”，有 7 个比特位表示网络号，24 个比特位表示本地地址。这样总共有 128 个 A 类网络。



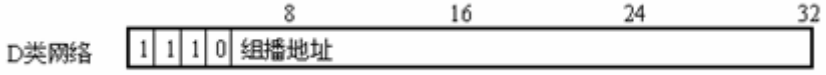
B 类地址，前两个最高比特位为“10”，有 14 个比特位表示网络号，16 个比特位表示本地地址。这样总共有 16,348 个 B 类网络。



C 类地址，前三个最高比特位为“110”，有 22 个比特位表示网络号，8 个比特位表示本地地址。这样总共有 2,097,152 个 C 类网络。



D 类地址，前四个最高比特位为“1110”，其余比特位为组播地址。



说明：

前四个最高比特位为“1111”的地址是不允许分配的，这些地址称为 E 类地址，属于保留地址。

在建设网络过程中，进行 IP 地址规划时，一定要根据建设网络的性质进行 IP 地址分配。如果建设的网络需要与互联网连接，则需要到相应的机构申请分配 IP 地址。中国地区可以向中国互联网信息中心（CNNIC）申请，负责 IP 地址分配的最终机构为国际互联网名字与编号分配公司（ICANN,Internet Corporation for Assigned Names and Numbers）。如果建设的网络为内部私有网络，就不需要申请 IP 地址，但是也不能随便分配，最好分配专门的私有网络地址。

下表为保留与可用的地址列表：

类别	地址空间	状态
A 类网络	0.0.0.0	保留
	1.0.0.0~126.0.0.0	可用
	127.0.0.0	保留
B 类网络	128.0.0.0~191.254.0.0	可用
	191.255.0.0	保留
C 类网络	192.0.0.0	保留
	192.0.1.0~223.255.254.0	可用
	223.255.255.0	保留
D 类网络	224.0.0.0~239.255.255.255	可用
E 类网络	240.0.0.0~255.255.255.254	保留
	255.255.255.255	组播

其中专门有三个地址块提供给私有网络，这些地址是不会在互联网中使用的，如果分配了这些地址的网络需要连接互联网，则需要将这些 IP 地址转换成有效的互联网地址。下表为私有网络地址空间，私有网络地址由 RFC 1918 文档定义：

类别	IP 地址范围	网络数
A 类网络	10.0.0.0~10.255.255.255	1 个 A 类网络
B 类网络	172.16.0.0~172.31.255.255	16 个 B 类网络
C 类网络	192.168.0.0~192.168.255.255	256 个 C 类网络

关于 IP 地址、TCP/UDP 端口及其它编码的分配情况，请参考 RFC 1166 文档。

19.1.2 IP 地址配置任务列表

IP 地址配置任务包括以下各项，但只有第一项配置是必须要做，其它任务可以根

据网络的具体需要决定是否要执行。

- 接口 IP 地址配置（要求）
- 地址解析协议（ARP）配置（可选）
- IP 到广域网地址映射配置（可选）
- 关闭 IP 路由（可选）
- 广播包处理配置（可选）

19.1.2.1 接口 IP 地址配置

一个设备只有配置了 IP 地址，才可以接收和发送 IP 数据报，接口配置了 IP 地址，说明该接口允许运行 IP 协议。

要分配一个接口的 IP 地址，在接口配置模式中执行以下命令：

命令	作用
Ruijie(config-if)# **ip address** *ip-address mask*	设置一个接口的 IP 地址
Ruijie(config-if)# **no ip address**	取消一个接口的 IP 地址配置

网络掩码也是一个 32 比特的数值，标识着该 IP 地址的哪几个比特为网络部分。网络掩码中，值为“1”的比特对应的 IP 地址比特位就是为网络部分，值为“0”的比特对应的 IP 地址比特位就是为主机地址部分。如 A 类网络对应的网络掩码为“255.0.0.0”。您可以利用网络掩码对一个网络进行子网划分，子网划分就是将一个将主机地址部分的一些比特位也作为网络部分，缩小主机容量，增加网络的数量，这时的网络掩码就称为子网掩码。

说明：

理论上，子网掩码的比特位可以是主机地址部分中的任何一段比特位。锐捷产品只支持从网络部分开始的从左到右连续的子网掩码。

与接口 IP 地址相关的特性配置见下列任务列表，以下任务作为可选配置，可以根据实际情况决定是否需要配置：

- 接口配置多个 IP 地址

19.1.2.1.1 接口配置多个 IP 地址

锐捷产品可以支持一个接口配置多个 IP 地址，其中一个为主 IP 地址，其余全部为次 IP 地址。次 IP 地址的配置理论上没有数目限制，但是次 IP 地址与主 IP 以及次 IP 之间地址必须属于不同网络。在网络建设中，会经常使用到次 IP 地址，通常在以下情况下应该考虑试用次 IP 地址：

- 一个网络没有足够多的主机地址。例如，现在一般局域网需要一个 C 类网络，可分配 254 台主机。但是当局域网主机超过 254 台时，一个 C 类网络将不够分配，有必要分配另一个 C 类网络地址。这样设备就需要连接两个网络，所以就需要配置多个 IP 地址。
- 许多旧的网络是基于第二层的桥接网络，没有进行子网的划分。次 IP 地址的使用可以使该网络很容易升级到基于 IP 层的路由网络。对于每个子网，设备都配置一个 IP 地址。
- 一个网络的两个子网被另外一个网络隔离开，您可以创建一个被隔离网络的子网，通过配置次 IP 地址的方式，将隔离的子网连接起来。一个子网不能在设备的两个或两个以上接口出现。

说明：

配置次 IP 地址之前，需要确定已经配置了主 IP 地址。如果网络上的一台设备配置了次 IP 地址，则其它设备也必须配置同一网络的次 IP 地址。当然如果其它设备原先没有分配 IP 地址，可以配置为主地址。

要配置次 IP 地址，在接口配置模式中执行以下命令：

命令	作用
Ruijie(config-if)# **ip address** *ip-address mask* **secondary**	设置接口次 IP 地址
Ruijie(config-if)# **no ip address** *ip-address mask* **secondary**	取消接口次 IP 地址配置

19.1.2.2 地址解析协议（ARP）配置

在局域网中，每个 IP 网络设备都有两个地址：1）本地地址，由于它包含在数据链路层的帧头中，更准确地说应该是数据链路层地址，但实际上对本地地址进行处理的是数据链路层中的 MAC 子层，因此习惯上称为 MAC 地址，MAC 地址在局域网上代表着 IP 网络设备；2）网络地址，在互联网上代表着 IP 网络设备，同时它也说明了该设备所属的网络。

局域网上两台 IP 设备之间需要通信，必须要知道对方的 48 比特的 MAC 地址。根据 IP 地址来获知 MAC 地址的过程称为地址解析（ARP）。而根据 MAC 地址获知 IP 地址的过程称为反向地址解析（RARP）。地址解析的方式有两类：1）地址解析协议（ARP）；2）代理地址解析协议（Proxy ARP）。关于 ARP 、Proxy ARP、RARP，分别在 RFC 826，RFC 1027，RFC 903 文当中描述。

ARP 是用来绑定 MAC 地址和 IP 地址的，以 IP 地址作为输入，ARP 能够知道其关联的 MAC 地址。一旦知道了 MAC 地址，IP 地址与 MAC 地址对应关系就会保存在设备的 ARP 缓冲中。有了 MAC 地址，IP 设备就可以封装链路层的帧，然后将数据帧发送到局域网上去。缺省配置下，以太网上 IP 和 ARP 的封装为 Ethernet

II 类型，也可以封装成其它类型的以太网帧类型如 SNAP。

RARP 的工作原理与 ARP 类似，不过 RARP 是以 MAC 地址作为输入，然后得到关联的 IP 地址。RARP 通常应用在无盘工作站上。

通常情况下，不需要特别配置设备的地址解析已经可以工作了。除非有特殊情况，否则不需要额外配置，锐捷产品还能通过以下配置来管理地址解析：

- 静态配置 ARP
- ARP 封装设置
- ARP 超时设置

19.1.2.2.1 静态配置 ARP

ARP 协议提供了 IP 地址和 MAC 地址动态映射的功能，通常情况下不需要进行静态配置。锐捷产品通过配置静态 ARP，还可以响应不是属于自己 IP 地址的 ARP 请求。

要配置静态 ARP，在全局配置模式中执行以下命令：

命令	作用
Ruijie(config)# **arp** *ip-address mac-address arp-type*	定义静态 ARP，其中 arp-type 目前只支持 arpa 类型。
Ruijie(config)# **no arp** *ip-address*	取消静态 ARP

19.1.2.2.2 ARP 封装设置

目前 ARP 封装只支持 Ethernet II 类型，在锐捷产品中也表示为 ARPA 关键字。

19.1.2.2.3 ARP 超时设置

ARP 超时设置只对动态学习到的 IP 地址和 MAC 地址映射起作用。超时时间设置得越短，ARP 缓冲中保存的映射表就越真实，但是 ARP 消耗网络带宽也越多，所以需要权衡利弊。除非有特别的需要，否则一般不需要配置 ARP 超时时间。

要配置 ARP 超时时间，在接口配置模式中执行以下命令：

命令	作用
Ruijie(config-if)# **arp timeout** *seconds*	配置 ARP 超时时间，范围 0-2147483，其中 0 表示不老化
Ruijie(config-if)# **no arp timeout**	恢复缺省配置

缺省情况下，超时时间为 3600 秒，即 1 个小时。

19.1.2.3 关闭 IP 路由

IP 路由功能缺省情况下是启动的，除非确定不需要 IP 路由功能，否则不要执行该操作。关闭 IP 路由将使设备丢失所有的路由，而且没有路由转发的功能。

要关闭 IP 路由功能，在全局配置模式中执行以下命令：

命令	作用
Ruijie(config)# **no ip routing**	关闭 IP 路由功能
Ruijie(config)# **ip routing**	启动 IP 路由功能

19.1.2.4 广播包处理配置

广播包是指目标地址为某个物理网络上所有的主机。锐捷产品支持两种类型广播包：1）定向广播，是指数据报接收者为一个指定网络的所有主机，目标地址的主机部分全为“1”；2）淹没广播，是指数据报接收者为所有网络的主机，目标地址 32 比特位全为“1”时。广播包目前被一些 IP 协议的泛滥使用，其中包括十分重要的 IP 协议。所以如何控制和使用广播包是一个网络管理人员的基本职责。

如果 IP 网络设备转发淹没广播，可能会引起网络的超负载，严重影响网络的运行，这种情况称为广播风暴。设备提供了一些办法能够将广播风暴限制在本地网络，阻止其继续扩张。但对于桥和交换机等基于二层网络设备，将转发和传播广播风暴。

解决广播风暴最好的办法就是给每个网络指定一个广播地址，这就是定向广播，这要求使用广播包的 IP 协议尽可能应用定向广播而不是淹没广播进行数据传播。

关于广播问题的详细描述，请参见 RFC 919 和 RFC 922。

如何对广播包进行处理，按照以下任务任务表，根据网络实际的需求进行配置。

- 定向网络广播到物理广播转换
- 创建 IP 广播地址

19.1.2.4.1 定向广播到物理广播转换

IP 定向广播报文是指目标地址为某个 IP 子网广播地址的 IP 报文，如目标地址为 172.16.16.255 的报文就称为定向广播报文。但是产生该报文的节点又不是目标子网的成员。

没有与目标子网直连的设备接收到 IP 定向广播报文，跟转发单播报文一样处理定向广播报文。当定向广播报文到达直连该子网的设备后，设备将把定向广播报文转换为淹没广播报文（一般指目标 IP 地址为全“1”的广播报文），然后以链路层广播方式发送给目标子网上的所有主机。

您可以在指定的接口上，启动定向广播到物理广播转换的功能，这样该接口就可以转发到直连网络的定向广播了。该命令只影响到达最终目标子网的定向广播报文的

最后传输，而不影响其它定向广播报文的正常转发。

在接口上，您还可以通过定义访问控制列表来控制转发某些定向广播。当定义了访问列表，只有符合访问列表中定义的数据包才进行定向广播到物理广播的转换。

要配置定向广播到物理广播的转换，在接口配置模式中执行以下命令：

命令	作用
Ruijie(config-if)# **ip directed-broadcast** [*access-list-number*]	在接口上，启动定向广播到物理广播的转换
Ruijie(config-if)# **no ip directed-broadcast**	取消转换

19.1.2.4.2 创建 IP 广播地址

当前使用最多的广播包，其目标地址为全“1”，表示为 255.255.255.255。锐捷产品可以通过软件定义产生其它地址的广播包，而且可以接收所有类型的广播包。

要配置有别于 255.255.255.255 的广播地址，在接口配置模式中执行以下命令：

命令	作用
Ruijie(config-if)# **ip broadcast-address** *ip-address*	创建新的广播地址
Ruijie(config-if)# **no ip broadcast-address**	取消新的广播地址

19.1.3 监视和维护 IP 地址

要监视和维护网络，根据以下任务描述，根据实际需要进行操作。

- 清除缓冲和表内容
- 显示系统和网络状态

19.1.3.1 清除缓冲和表内容

您可以删除一些特定缓冲、表、数据库的全部内容，主要包括三个方面：

1) 清除 ARP 缓冲；
2) 清除主机名到 IP 地址的映射表；
3) 清除路由表。

命令	作用
Ruijie# **clear arp-cache**	清除 ARP 缓冲
Ruijie# **clear ip route** {*network* [*mask*] | ***}**	清除 IP 路由表

19.1.3.2 显示系统和网络状态

您可以查看 IP 路由表、缓冲、数据库的所有内容，通过这些信息对网络故障的排除十分有帮助。通过测试本地设备网络的可达到性，您可以知道数据包在离开本设备后将往那条路径发送。

要显示系统和网络统计量，在特权用户模式中执行以下命令：

命令	作用
Ruijie# **show arp**	显示 ARP 缓冲表
Ruijie# **show ip arp**	显示 IP ARP 缓冲表
Ruijie# **show ip interface** [*interface-type interface-number*]	显示接口 IP 信息
Ruijie# **show ip route** [*network* [*mask*]]	显示路由表
Ruijie#**show ip route**	显示路由表摘要
Ruijie# **ping** *ip-address* [**length** *bytes*] [**ntimes** *times*] [**timeout** *seconds*]	测试网络可达性

19.1.4 IP 地址配置范例

本章提供了以下 IP 地址配置范例：

- 次 IP 地址配置范例

19.1.4.1 次 IP 地址配置范例

- **配置要求**

IP 地址分配和网络设备连接图见图 1。

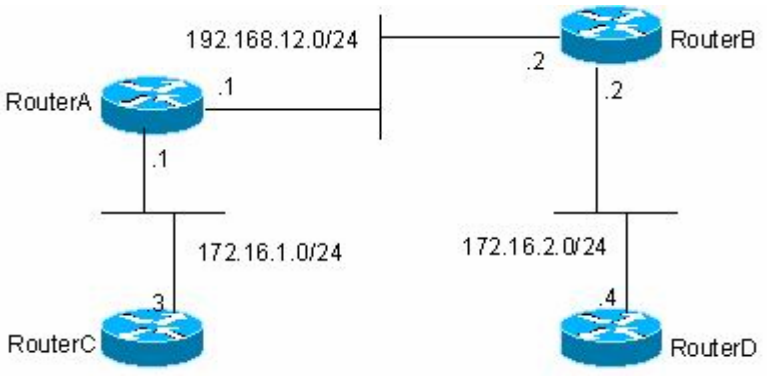


图 1　次 IP 地址配置范例

要求配置 RIP 路由协议，但只允许将版本设定为 RIPv1，在路由器 C 上可以看到 172.16.2.0/24 的路由，在路由器 D 上可以看到 172.16.1.0/24 的路由。

- **路由器具体配置**

RIPv1 路由协议是不支持无类路由的，即在路由通告中不携带掩码信息，172.16.1.0/24 和 172.16.2.0/24 两个同网络的子网又被 C 类网络 192.168.12.0/24 分割开，按照通常配置，路由器 C 和路由器 D 上是不可能学到对方网络的详细路由。但是 RIP 路由协议有个特性，如果接口网络与接收的路由同属一个网络，该路由的网络掩码与该接口网络掩码将设为一致。根据这个特性，可以通过配置路由器 A 和路由器 B，在 192.168.12.0/24 网络上构建一个次网络 172.16.3.0/24，这样两个被分割的子网就连接起来了。以下只写出路由器 A 和路由器 B 的配置。

路由器 A 的配置：

```
interface FastEthernet 0/1
ip address 172.16.3.1 255.255.255.0 secondary
ip address 192.168.12.1 255.255.255.0
!
interface FastEthernet 0/2
ip address 172.16.1.1 255.255.255.0
!
router rip
network 172.16.0.0
network 192.168.12.0
```

路由器 B 的配置:

```
interface FastEthernet 0/1
ip address 172.16.3.2 255.255.255.0 secondary
ip address 192.168.12.2 255.255.255.0
!
interface FastEthernet 0/2
ip address 172.16.2.1 255.255.255.0
!
router rip
network 172.16.0.0
network 192.168.12.0
```

19.2 IP 服务配置

19.2.1 IP 连接管理

IP 协议栈提供了许多服务用来控制和管理 IP 连接，ICMP 就提供了许多这些服务。当网络发生任何问题，设备或接入服务器将发送 ICMP 消息给主机或其它设备。详细的 ICMP 消息定义，请参见 RFC 792。

要管理各种各样的 IP 连接问题，可选择性地执行以下各项任务：

- 启用 ICMP 协议不可达消息
- 启用 ICMP 重定向消息
- 启用 ICMP 掩码应答消息
- 设置 IP MTU
- 配置 IP 源路由

19.2.1.1 启用 ICMP 目标不可达消息

当设备接收到目标为自己的非广播包，该数据包中采用了设备不能处理的 IP 协议，设备就向源地址发送 ICMP 协议不可达消息。另外，如果设备由于不知道路由而不能转发数据包时，也会发送 ICMP 主机不可达消息。这种特性缺省是启用的。

如果要重新启用 ICMP 协议不可达消息，在接口配置模式中执行以下命令：

命令	作用
Ruijie(config-if)**# ip unreachables**	启用 ICMP 协议不可达和主机不可达消息
Ruijie(config-if)**# no ip unreachables**	关闭 ICMP 协议不可达和主机不可达消息

19.2.1.2 启用 ICMP 重定向消息

路由有时会不够优化，使得设备从一个接口接收到的数据包，还要从该接口发送出去。如果设备将数据包从接收接口重新发送出去，设备就会给数据源发送一个 ICMP 重定向消息，告诉数据源到该目标地址的网关为同一子网上的另外一台设备。这样数据源就会将后续的数据包按照最佳的路径进行发送。该特性缺省是启用。

要配置 ICMP 重定向消息，在接口配置模式中执行以下命令：

命令	作用
Ruijie(config-if)**# ip redirects**	启用 ICMP 重定向消息，缺省启用
Ruijie(config-if)**# no ip redirects**	关闭 ICMP 重定向消息

19.2.1.3 启用 ICMP 掩码应答消息

网络设备有时需要知道互联网上某个子网的子网掩码，为了获取该信息，网络设备可以发送 ICMP 掩码请求消息，接收到 ICMP 掩码请求消息的网络设备就会发送掩码应答消息。锐捷产品可以响应 ICMP 掩码请求消息，缺省情况下是启用该特性的。

要配置 ICMP 掩码应答消息，在接口配置模式中执行以下命令：

命令	作用
Ruijie(config-if)# **ip mask-reply**	启用掩码应答消息
Ruijie(config-if)# **no ip mask-reply**	关闭掩码应答消息

19.2.1.4 设置 IP MTU

设备所有的接口都有缺省的 MTU（最大传输单元）值，所有大于 MTU 的数据包要从该接口转发出去必须分段，否则发送失败。

锐捷产品允许调整接口的 MTU 值，而且 MTU 的变化会引起 IP MTU 的变化，IP MTU 总是会自动与接口 MTU 保持一致。但是反之不行，如果调整了 IP MTU 值，接口 MTU 不会跟着改变。

一个物理网络上的设备接口，相同协议 MTU 值必须保持一致。

要设置 IP MTU 值，在接口配置模式中执行以下命令：

命令	作用
Ruijie(config-if)# **ip mtu** *bytes*	设置 MTU 值，范围 68~1500
Ruijie(config-if)# **no ip mtu**	恢复缺省值

19.2.1.5 配置 IP 源路由

锐捷产品支持 IP 源路由。当设备接收到 IP 数据包时，会对 IP 报头的严格源路由、宽松源路由、记录路由等选项进行检查，这些选项在 RFC 791 中有详细描述。如果检测到该数据包启用了其中一个选项，就会执行响应的动作；如果检测到无效的选项，就会给数据源发送一个 ICMP 参数问题消息，然后丢弃该数据包。锐捷产品缺省情况下支持 IP 源路由特性。

要配置 IP 源路由，在全局模式中执行以下命令：

命令	作用
Ruijie(config)# **ip source-route**	启用 IP 源路由
Ruijie(config)# **no ip source-route**	关闭 IP 源路由

20 DHCP 配置

20.1 DHCP 介绍

DHCP(Dynamic Host Configuration Protocol，动态主机配置协议)在 RFC 2131 中有详细的描述，DHCP 为互联网上主机提供配置参数。DHCP 是基于 Client/Server 工作模式，DHCP 服务器为需要动态配置的主机分配 IP 地址和提供主机配置参数。

DHCP 有三种机制分配 IP 地址：

- 自动分配，DHCP 给客户端分配永久性的 IP 地址；
- 动态分配，DHCP 给客户端分配过一段时间会过期的 IP 地址（或者客户端可以主动释放该地址）；
- 手工配置，由网络管理员给客户端指定 IP 地址。管理员可以通过 DHCP 将指定的 IP 地址发给客户端。

三种地址分配方式中，只有动态分配可以重复使用客户端不再需要的地址。

DHCP 消息的格式是基于 BOOTP(Bootstrap Protocol)消息格式的，这就要求设备具有 BOOTP 中继代理的功能，并能够与 BOOTP 客户端和 DHCP 服务器实现交互。BOOTP 中继代理的功能，使得没有必要在每个物理网络都部署一个 DHCP 服务器。RFC 951 和 RFC 1542 对 DHCP 协议进行了详细描述。

20.2 DHCP 服务器简介

锐捷产品的 DHCP 服务器完全根据 RFC 2131 来实现的，主要功能就是为主机分配和管理 IP 地址。DHCP 工作的基本流程如图 3-1 所示。

图 1 DHCP 基本工作流程

DHCP 请求 IP 地址的过程如下：

1) 主机发送 DHCPDISCOVER 广播包在网络上寻找 DHCP 服务器；
2) DHCP 服务器向主机发送 DHCPOFFER 单播数据包，包含 IP 地址、MAC 地址、域名信息以及地址租期；

3) 主机发送 DHCPREQUEST 广播包，正式向服务器请求分配已提供的 IP 地址；

4) DHCP 服务器向主机发送 DHCPACK 单播包，确认主机的请求。

说明：

DHCP 客户端可以接收到多个 DHCP 服务器的 DHCPOFFER 数据包，然后可能接受任何一个 DHCPOFFER 数据包，但客户端通常只接受收到的第一个 DHCPOFFER 数据包。另外，DHCP 服务器 DHCPOFFER 中指定的地址不一定为最终分配的地址，通常情况下，DHCP 服务器会保留该地址直到客户端发出正式请求。

正式请求 DHCP 服务器分配地址 DHCPREQUEST 采用广播包，是为了让其它所有发送 DHCPOFFER 数据包的 DHCP 服务器也能够接收到该数据包，然后释放已经 OFFER（预分配）给客户端的 IP 地址。

如果发送给 DHCP 客户端的 DHCPOFFER 信息包中包含无效的配置参数，客户端会向服务器发送 DHCPDECLINE 信息包拒绝接受已经分配的配置信息。

在协商过程中，如果 DHCP 客户端没有及时响应 DHCPOFFER 信息包，DHCP 服务器会发送 DHCPNAK 消息给 DHCP 客户端，导致客户端重新发起地址请求过程。

在网络建设中，应用锐捷产品 DHCP 服务器，可以带来以下好处：

- 降低网络接入成本。一般采用静态地址分配的接入费用比较昂贵，应用动态地址分配的接入成本较低。
- 简化配置任务，降低网络建设成本。采用动态地址分配，大大简化了设备配置，对于在没有专业技术人员的地方部署设备，更是降低了部署成本。
- 集中化管理。在对几个子网进行配置管理时，有任何配置参数的变动，只需要修改和更新 DHCP 服务器的配置即可。

20.3 DHCP 客户端简介

DHCP 客户端可以让设备自动地从 DHCP 服务器获得 IP 地址以及其它配置参数。DHCP 客户端可以带来如下好处：

- 降低了配置和部署设备时间。
- 降低了发生配置错误的可能性。
- 可以集中化管理设备的 IP 地址分配。

注意：

锐捷产品目前版本支持以太网接口以及 FR、PPP、HDLC 接口上的 DHCP 客户端。

20.4 DHCP 中继代理简介

DHCP 中继代理，就是在 DHCP 服务器和客户端之间转发 DHCP 数据包。当 DHCP 客户端与服务器不在同一个子网上，就必须有 DHCP 中继代理来转发 DHCP 请求和应答消息。DHCP 中继代理的数据转发，与通常路由转发是不同的，通常的路由转发相对来说是透明传输的，设备一般不会修改 IP 包内容。而 DHCP 中继代理接收到 DHCP 消息后，重新生成一个 DHCP 消息，然后转发出去。

在 DHCP 客户端看来，DHCP 中继代理就像 DHCP 服务器；在 DHCP 服务器看来，DHCP 中继代理就像 DHCP 客户端。

20.5 DHCP 配置

要配置 DHCP，请按照下面任务列表进行配置，其中前三个配置任务是必须的。

- 启用 DHCP 服务器与中继代理（要求）
- DHCP 排斥地址配置（要求）
- DHCP 地址池配置（要求）
- 配置 DHCP 服务器强制回复 NAK（可选）手工地址绑定（可选）
- 配置 Ping 包次数（可选）
- 配置 Ping 包超时时间（可选）
- 以太网接口 DHCP 客户端配置（可选）

20.5.1 启用 DHCP 服务器与中继代理

要启用 DHCP 服务器、中继代理，全局配置模式中执行以下命令：

命令	作用
Ruijie(config)# **service dhcp**	启用 DHCP 服务器和 DHCP 中继代理功能
Ruijie(config)# **no service dhcp**	关闭 DHCP 服务器和中继代理功能

20.5.2 DHCP 排斥地址配置

如果没有特别配置，DHCP 服务器会试图将在地址池中定义的所有子网地址分配给 DHCP 客户端。因此，如果你想保留一些地址不想分配，比如已经分配给服务器或者设备了，你必须明确定义这些地址是不允许分配给客户端的。

要配置哪些地址不能分配给客户端，在全局配置模式中执行以下命令：

命令	作用
Ruijie(config)# **ip dhcp excluded-address** *low-ip-address* [*high-ip-address*]	定义 IP 地址范围，这些地址 DHCP 不会分配给客户端
Ruijie(config)# **no ip dhcp excluded-address** *low-ip-address* [*high-ip-address*]	取消配置地址排斥

配置 DHCP 服务器，一个好的习惯是将所有已明确分配的地址全部不允许 DHCP 分配，这样可以带来两个好处：1）不会发生地址冲突；2）DHCP 分配地址时，减少了检测时间，从而提高 DHCP 分配效率。

20.5.3 DHCP 地址池配置

DHCP 的地址分配以及给客户端传送的 DHCP 各项参数，都需要在 DHCP 地址池中进行定义。如果没有配置 DHCP 地址池，即使启用了 DHCP 服务器，也不能对客户端进行地址分配；但是如果启用了 DHCP 服务器，不管是否配置了 DHCP 地址池，DHCP 中继代理的总是起作用的。

你可以给 DHCP 地址池起个有意义、易记忆的名字，地址池的名字由字符和数字组成。锐捷产品可以定义多个地址池，根据 DHCP 请求包中的中继代理 IP 地址来决定分配哪个地址池的地址。

- 如果 DHCP 请求包中没有中继代理的 IP 地址，就分配与接收 DHCP 请求包接口的 IP 地址同一子网或网络的地址给客户端。如果没定义这个网段的地址池，地址分配就失败；
- 如果 DHCP 请求包中有中继代理的 IP 地址，就分配与该地址同一子网或网络的地址给客户端。如果没定义这个网段的地址池，地址分配就失败。

要进行 DHCP 地址池配置，请根据实际的需要执行以下任务，其中前三个任务要求执行：

- 配置地址池并进入其配置模式（要求）
- 配置地址池子网及其掩码（要求）
- 配置客户端缺省网关（要求）
- 配置地址租期（可选）
- 配置客户端的域名（可选）
- 配置域名服务器（可选）
- 配置 NetBIOS WINS 服务器（可选）
- 配置客户端 NetBIOS 节点类型（可选）

20.5.3.1 配置地址池名并进入其配置模式

要配置地址池名并进入地址池配置模式，在全局配置模式中执行以下命令：

命令	作用
Ruijie(config)# **ip dhcp pool** *dhcp-pool*	配置地址池名并进入地址池配置模式

地址池的配置模式显示为“Ruijie(dhcp-config)#”。

20.5.3.2 配置客户端启动文件

客户端启动文件是客户端启动时要用到的启动映像文件。启动映像文件通常是 DHCP 客户端需要下载的操作系统。

要配置客户端的启动文件，在地址池配置模式中执行以下命令：

命令	作用
Ruijie (dhcp-config)# **bootfile** *filename*	配置客户端启动文件名

20.5.3.3 配置客户端缺省网关

配置客户端默认网关，这个将作为服务器分配给客户端的默认网关参数。缺省网关的 IP 地址必须与 DHCP 客户端的 IP 地址在同一网络。

要配置客户端的缺省网关，在地址池配置模式中执行以下命令：

命令	作用
Ruijie(dhcp-config)# **default-router** *address* [*address2…address8*]	配置缺省网关

20.5.3.4 配置地址租期

DHCP 服务器给客户端分配的地址，缺省情况下租期为 1 天。当租期快到时客户端需要请求续租，否则过期后就不能使用该地址。

要配置地址租期，在地址池配置模式中执行以下命令：

命令	作用
Ruijie(dhcp-config)# **lease** {*days* [*hours]* [*minutes*] | **infinite**}	配置地址租期

20.5.3.5 配置客户端的域名

可以指定客户端的域名，这样当客户端通过主机名访问网络资源时，不完整的主机名会自动加上域名后缀形成完整的主机名。

要配置客户端的域名，在地址池配置模式中执行以下命令：

命令	作用
Ruijie(dhcp-config)# **domain-name** *domain*	配置域名

20.5.3.6 配置域名服务器

当客户端通过主机名访问网络资源时，需要指定 DNS 服务器进行域名解析。要配置 DHCP 客户端可使用的域名服务器，在地址池配置模式中执行以下命令：

命令	作用
Ruijie(dhcp-config)# **dns-server** *address* [*address2…address8*]	配置 DNS 服务器

20.5.3.7 配置 NetBIOS WINS 服务器

WINS 是微软 TCP/IP 网络解析 NetNBIOS 名字到 IP 地址的一种域名解析服务。WINS 服务器是一个运行在 Windows NT 下的服务器。当 WINS 服务器启动后，会接收从 WINS 客户端发送的注册请求，WINS 客户端关闭时，会向 WINS 服务器发送名字释放消息，这样 WINS 数据库中与网络上可用的计算机就可以保持一致了。

要配置 DHCP 客户端可使用的 NetBIOS WINS 服务器，在地址池配置模式中执行以下命令：

命令	作用
Ruijie(dhcp-config)# **netbios-name-server** *address* [*address2…address8*]	配置 DNS 服务器

20.5.3.8 配置客户端 NetBIOS 节点类型

微软 DHCP 客户端 NetBIOS 节点类型有四种：1）Broadcast，广播型节点，通过广播方式进行 NetBIOS 名字解析；2）Peer-to-peer，对等型节点，通过直接请求 WINS 服务器进行 NetBIOS 名字解析；3）Mixed，混合型节点，先通过广播方式请求名字解析，后通过与 WINS 服务器连接进行名字解析；4）Hybrid，复合型节点，首先直接请求 WINS 服务器进行 NetBIOS 名字解析，如果没有得到应答，就通过广播方式进行 NetBIOS 名字解析。

缺省情况下，微软操作系统的节点类型为广播型或者复合型。如果没有配置 WINS 服务器，就为广播型节点；如果配置了 WINS 服务器，就为复合型节点。

要配置 DHCP 客户端 NetBIOS 节点类型，在地址池配置模式中执行以下命令：

命令	作用
Ruijie(dhcp-config)# **netbios-node-type** *type*	配置 NetBIOS 节点类型

20.5.3.9 配置 DHCP 地址池的网络号和掩码

进行动态地址邦定的配置，必须配置新建地址池的子网及其掩码，为 DHCP 服务器提供了一个可分配给客户端的地址空间。除非有地址排斥配置，否则所有地址池中的地址都有可能分配给客户端。DHCP 在分配地址池中的地址，是按顺序进行的，如果该地址已经在 DHCP 绑定表中或者检测到该地址已经在该网段中存在，就检查下一个地址，直到分配一个有效的地址。

要配置地址池子网和掩码，在地址池配置模式中执行以下命令:

命令	作用
Ruijie(dhcp-config)# **network** *network-number mask*	配置 DHCP 地址池的网络号和掩码

注意:

锐捷产品的 DHCP 动态地址池中，地址的分配是以客户端的物理地址和客户端 ID 为索引的，这就意味着 DHCP 动态地址池中不可能存在相同客户端的两份租约；如果客户端和服务器之间的网络拓扑存在路径上的冗余[客户端可以通过直连路径，同时也可以通过中继路径到达服务器]，就会导致服务器分配地址出现问题，可能导致地址分配失败；

因此，为了避免上述问题，要求网络管理员在构建网络的时候，通过其它的方式，如调整物理链路或者网络路径，来避免这种客户端到服务器的路径冗余。

20.5.4 配置 DHCP 服务器强制回复 NAK

根据 RFC2131 协议介绍，当 DHCP 服务器在收到客户端的 Request 续租报文时，发现客户端的网段发生更改或者是租约超时时会给予回复 NAK，要求客户端重新获取 IP 地址，避免客户端不断发送 Request 报文直至超时后重新获取 IP 地址，延长 IP 地址获取时间。

但是，DHCP 服务器发送 NAK 报文的前提是该 DHCP 客户端在自己的管理范围之内，也就是可以查找到对应的租约记录信息。当 DHCP 客户端从另一个网络环境中移入时，DHCP 服务器将无法在本地查找到对应的租约记录信息，不予回复 NAK，此时 DHCP 客户端需要不断发送 Request 报文直至超时后重新获取 IP 地址，导致 IP 地址获取时间变长。在 DHCP 服务器重启时丢失客户端租约，而客户端要求续租时也会遇到类似情况。在这种情况下，可以通过配置命令强制让 DHCP 服务器在查找不到租约记录时也给予回复 NAK 报文，触发客户端快速获取到 IP 地址。

要配置 DHCP 服务器强制回复 NAK，在全局模式中执行以下命令：

命令	作用

Ruijie(dhcp-config)# **ip dhcp force-send-nak**	配置 DHCP 服务器强制回复 NAK

注意:

1、该命令默认关闭；

2、在开启该命令的情况下，同一广播域中仅能部署一台 DHCP 服务器，否则 DHCP 客户端可能会收到不正确的 NAK 响应，而导致续租失败。

20.5.5 手工地址绑定

地址绑定是指 IP 地址和客户端 MAC 地址的映射关系。地址绑定有两种：1）手工绑定，就是在 DHCP 服务器数据库中，通过手工定义将 IP 地址和 MAC 地址进行静态映射，手工绑定其实是一个特殊地址池；2）动态绑定，DHCP 服务器接收到 DHCP 请求时，动态地从地址池中分配 IP 地址给客户端，而形成的 IP 地址和 MAC 地址映射。

要定义手工地址绑定，首先需要为每一个手动邦定定义一个主机地址池，然后定义 DHCP 客户端的 IP 地址和硬件地址或客户端标识。硬件地址就是 MAC 地址。客户端标识，微软客户端一般定义客户端标识，而不定义 MAC 地址，客户端标识包含了网络媒介类型和 MAC 地址。关于媒介类型的编码，请参见 RFC 1700 中关于“Address Resolution Protocol Parameters”部分内容。以太网类型为“01”。

要配置手工地址绑定，在地址池配置模式中执行以下命令：

命令	作用
Ruijie(config)# **ip dhcp pool** *name*	定义地址池名，进入 DHCP 配置模式
Ruijie(dhcp-config)# **host** *address*	定义客户端 IP 地址
Ruijie(dhcp-config)# **hardware-address** *hardware-address type* Ruijie(dhcp-config)# **client-identifier** *unique-identifier*	定义客户端硬件地址，如 aabb.bbbb.bb88 定义客户端的标识，如 01aa.bbbb.bbbb.88
Ruijie(dhcp-config)# **client-name** *name*	（可选）用标准的 ASCII 字符定义客户端的名字，名字不要包括域名。如定义 mary 主机名，不可定义成 mary.rg.com

20.5.6 配置 Ping 包次数

缺省情况，当 DHCP 服务器试图从地址池中分配一个 IP 地址时，会对该地址执行

两次 Ping 命令(一次一个数据包)。如果 Ping 没有应答，DHCP 服务器认为该地址为空闲地址，就将该地址分配给 DHCP 客户端；如果 Ping 有应答，DHCP 服务器认为该地址已经在使用，就试图分配另外一个地址给 DHCP 客户端，直到分配成功。

要配置 Ping 包次数，在全局配置模式中执行以下命令：

命令	作用
Ruijie(config)**# ip dhcp ping packets** *number*	配置 DHCP 服务器在分配地址之前的 Ping 包次数，如果设为 0 则不进行 Ping 操作，缺省为 2。

20.5.7 配置 Ping 包超时时间

缺省情况下，DHCP 服务器 Ping 操作如果 500 毫秒没有应答，就认为没有该 IP 地址主机存在。你可以通过调整 Ping 包超时时间，改变服务器 Ping 等待应答的时间。

要配置 Ping 包超时时间，在全局配置模式中执行以下命令：

命令	作用
Ruijie(config)**# ip dhcp ping timeout** *milliseconds*	配置 DHCP 服务器 Ping 包超时时间，缺省为 500ms。

20.5.8 以太网接口 DHCP 客户端配置

锐捷产品支持以太网端口通过 DHCP 获得动态分配的 IP 地址。要配置以太网接口 DHCP 客户端，在接口配置模式中执行以下命令：

命令	作用
Ruijie(config-if)**# ip address dhcp**	配置通过 DHCP 得到 IP 地址

20.6 监视和维护信息

20.6.1 监视和维护 DHCP 服务器

监视和维护 DHCP 服务器有三类命令：

1）清除命令，可以清除 DHCP 地址绑定、地址冲突、服务器统计状态等信息；

2）调试（debug）命令，输出必要的调试信息，主要用于故障诊断和排除；

3）显示命令，显示 DHCP 相关信息。

锐捷产品提供了三条清除命令，要进行相关清除信息操作，在命令执行模式中执行以下命令：

命令	作用
Ruijie**# clear ip dhcp binding** { *address* | ***}**	清除 DHCP 地址绑定信息
Ruijie**# clear ip dhcp conflict** { *address* | ***}**	清除 DHCP 地址冲突信息
Ruijie**# clear ip dhcp server statistics**	清除 DHCP 服务器统计状态

要进行 DHCP 服务器的调试，在命令执行模式中执行以下命令：

命令	作用
Ruijie**# debug ip dhcp server** [**events** | **packet**]	调试 DHCP 服务器

要显示 DHCP 服务器的工作状态，在命令执行模式中执行以下命令：

命令	作用
Ruijie**# show ip dhcp binding** [**address**]	显示 DHCP 地址绑定信息
Ruijie**# show ip dhcp conflict**	显示 DHCP 地址冲突信息
Ruijie**# show ip dhcp server statistics**	显示 DHCP 服务器统计信息

20.6.2 监视和维护 DHCP 客户端

监视和维护 DHCP 客户有两类命令，可以通过在客户端上进行以下操作：

1）调试（debug）命令，输出必要的调试信息，主要用于故障诊断和排除.

2）显示命令，显示 DHCP 相关信息。

要进行 DHCP 客户的调试，在命令执行模式中执行以下命令：

命令	作用
Ruijie**# debug ip dhcp client**	调试 DHCP 客户

要显示 DHCP 客户获得的租约信息，在命令执行模式中执行以下命令：

命令	作用
Ruijie**# show dhcp lease**	显示 DHCP 租约信息

20.7 配置范例

本节提供了 3 个配置例子：

- 地址池配置例子
- 手工绑定配置例子
- DHCP 客户端配置例子

20.7.1 地址池配置例子

在以下配置中，定义了一个地址池 net172，地址池网段为 172.16.1.0/24，缺省网关为 172.16.16.254，域名为 rg.com，域名服务器为 172.16.1.253，WINS 服务器为 172.16.1.252，NetBIOS 节点类型为复合型，地址租期为 30 天。该地址池中除了 172.16.1.2~172.16.1.100 地址外，其余地址均为可分配地址。

```
ip dhcp excluded-address 172.16.1.2 172.16.1.100
!
ip dhcp pool net172
network 172.16.1.0 255.255.255.0
default-router 172.16.1.254
domain-name rg.com
dns-server 172.16.1.253
netbios-name-server 172.16.1.252
netbios-node-type h-node
lease 30
```

20.7.2 手工绑定配置

在以下配置中，对 MAC 地址为 00d0.df34.32a3 的 DHCP 客户端分配的 IP 地址为 172.16.1.101，掩码为 255.255.255.0，主机名为 Billy.rg.com，缺省网关为 172.16.1.254，WINS 服务器为 172.16.1.252，NetBIOS 节点类型为复合型。

```
ip dhcp pool Billy
host 172.16.1.101 255.255.255.0
hardware-address 00d0.df34.32a3 ethernet
client-name Billy
default-router 172.16.1.254
domain-name rg.com
dns-server 172.16.1.253
netbios-name-server 172.16.1.252
netbios-node-type h-node
```

20.7.3 DHCP 客户端配置

以下配置中，为设备接口 FastEthernet 0/0 配置 DHCP 自动分配地址。

```
interface FastEthernet0/0
ip address dhcp
```

21 DHCP Relay 配置

21.1 概述

21.1.1 理解 DHCP

DHCP 协议被广泛用来动态分配可重用的网络资源，如 IP 地址。

DHCP Client 发出 DHCP DISCOVER 广播报文给 DHCP Server。DHCP Server 收到后 DHCP DISCOVER 报文后，根据一定的策略来给 Client 分配资源,如 IP 地址，发出 DHCP OFFER 报文。DHCP Client 收到 DHCP OFFER 报文后，验证资源是否可用。如果资源可用发送 DHCP REQUEST 报文；如果不可用，重新发送 DHCP DISCOVER 报文。服务器收到 DHCP REQUEST 报文，验证 IP 地址资源（或其他有限资源）是否可以分配，如果可以分配，则发送 DHCP ACK 报文；如果不可分配，则发送 DHCP NAK 报文。DHCP Client 收到 DHCP ACK 报文，就开始使用服务器分配的资源；如果收到 DHCP NAK，则可能重新发送 DHCP DISCOVER 报文再次请求另一个 IP 地址。

21.1.2 理解 DHCP 中继代理（DHCP Relay Agent）

DHCP 请求报文的目的 IP 地址为 255.255.255.255，这种类型报文的转发局限于子网内，不会被设备转发。为了实现跨网段的动态 IP 分配，DHCP Relay Agent 就产生了。它把收到的 DHCP 请求报文封装成 IP 单播报文转发给 DHCP Server，同时，把收到的 DHCP 响应报文转发给 DHCP Client。这样 DHCP Relay Agent 就相当于一个转发站，负责沟通位于不同网段的 DHCP Client 和 DHCP Server。这样就实现了只要安装一个 DHCP Server 就可对所有网段的动态 IP 管理，即 Client—Relay Agent—Server 模式的 DHCP 动态 IP 管理。

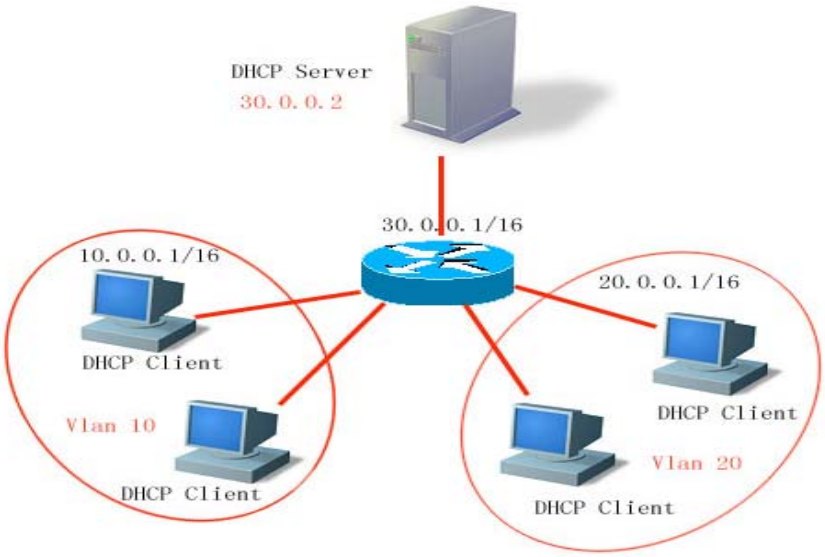


图 1

VLAN 10 和 VLAN 20 分别对应 10.0.0.1/16 和 20.0.0.1/16 的网络，而 DHCP Server 在 30.0.0.1/16 的网络上，30.0.0.2 的 DHCP Server 要对 10.0.0.1/16 和 20.0.0.1/16 的网络进行动态 IP 管理，只要在作为网关的设备上打开 DHCP Relay Agent，并指定 DHCP Server IP 为 30.0.0.2 就可以了

21.1.3 理解 DHCP Relay Agent Information(option 82)

根据 RFC3046 的定义，中继设备进行 DHCP relay 时，可以通过添加一个 option 的方式来详细的标明 DHCP client 的一些网络信息，从而使服务器可以根据更精确的信息给用户分配不同权限的 IP，根据 RFC3046 的定义，所使用 option 选项的选项号为 82，故也被称作 option82，该 option 可以继续分解成多个子选项，现阶段经常使用的子选项有 Circuit ID 和 Remote ID。本公司实现的 relay agent information 现阶段存在两种，一种是与 802.1x/SAM 应用方案结合 relay agent information option dot1x，另一种是结合用户所属的端口 vid，slot，port，以及设备 mac 信息的 relay agent information option82，下边对两种方案应用时 option 携带的内容及格式以及一些典型的应用方案进行一些说明：

1. relay agent information option dot1x：此种应用方案需要结合 802.1x 认证以及锐捷产品 RG-SAM。通过 RG-SAM 在 802.1x 认证过程中给设备下放不同的 IP 权限，结合 DHCP client 所属的 vid 组合成 Circuit ID 子选项。在 DHCP relay 上传到 DHCP server 时，结合 DHCP server 的配置，就可以实现给不同权限用户分配不同权限 IP 的应用。组合成 Circuit ID 格式如下，其中 priviliage 和 vid 字段各占两个字节：

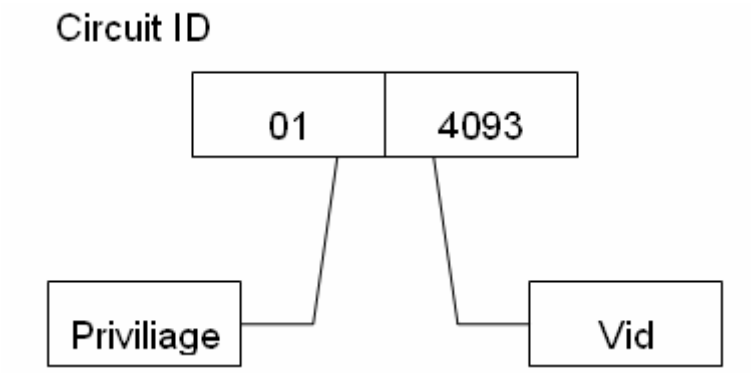


图 2

2. relay agent information option82：此种 option 的应用不需要结合其他协议模块的运行，设备在 DHCP relay 的过程中，根据收到 DHCP 请求的实体端口，以及设备自身的物理地址信息，组合构成 option82 信息上传到服务器，option 选贤得格式如下：

Agent Circuit ID

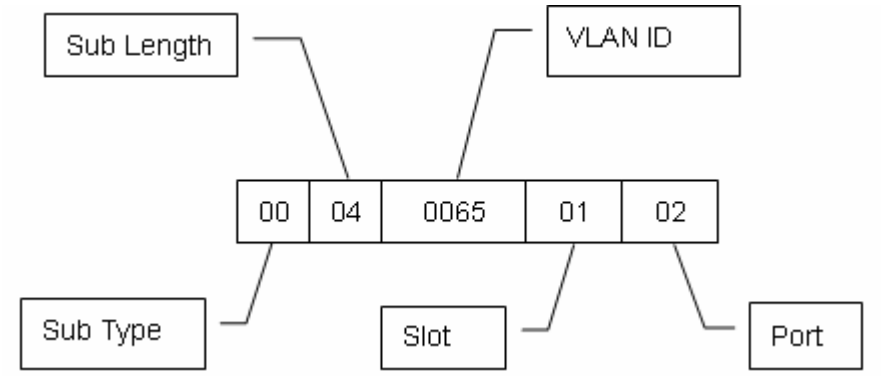


图 3

Agent Remote ID

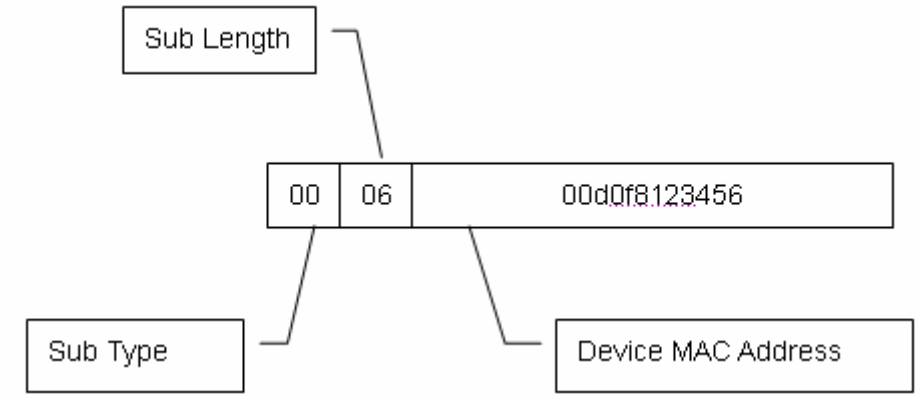


图 4

21.1.4 理解 DHCP relay Check Server-id 功能

在 DHCP 应用时，通常会为每一个网络配备多个 DHCP 服务器，从而进行备份，防止因为一台服务器的工作不正常影响网络的正常使用。在 DHCP 获取的四个交互过程中，当 DHCP client 在发送 DHCP REQUEST 时已经选定了服务器，此时会在请求的报文中携带一个 server-id 的 option 选项，在某些特定的应用环境中为了减轻网络服务器压力，需要我们 relay 能够使能此选项，只把请求报文发给此选项里的 server，而不是发送给每一个配置的 DHCP server，上述就是 DHCP check server-id 功能

21.2 配置 DHCP

21.2.1 配置 DHCP 中继代理

在全局配置模式下，请按如下步骤配置 DHCP 中继代理：

命令	作用
Ruijie (config)**# service dhcp**	启用 DHCP 代理
Ruijie(config)**# no service dhcp**	关闭 DHCP 代理。

21.2.2 配置 DHCP Server 的 IP 地址

在配置 DHCP Server 的 IP 地址后，设备所收到的 DHCP 请求报文将转发给它，同时，收到的来自 Server 的 DHCP 响应报文也会转发给 Client。

DHCP server 地址可以全局配置，也可以在三层接口上配置，每种配置模式都可以配置多个服务器地址最多可以配置 20 个服务器地址，。在某接口收到 DHCP 请求，则首先使用接口 DHCP 服务器；如果接口上面没有配置服务器地址，则使用全局配置的 DHCP 服务器。

DHCP 中继支持基于 vrf 的中继功能，配置方法就是在对应的服务器地址前面添加 vrf 参数。配置 DHCP 服务器地址请按如下方式进行：

命令	作用
Ruijie(config)**# IP helper-address** [**vrf**] *A.B.C.D*	添加一个全局的 DHCP 服务器地址
Ruijie(config-if)**# IP helper-address** [**vrf**] *A.B.C.D*	添加一个接口的 DHCP 服务器地址。此命令必须在三层接口下设置。
Ruijie(config)**# no IP helper-address** [**vrf**] *A.B.C.D*	删除一个全局的 DHCP 服务器地址

Ruijie(config-if)**# no IP helper-address** [**vrf**] *A.B.C.D*	删除一个接口的 DHCP 服务器地址

21.2.3 配置 DHCP option dot1x

通过理解 DHCP Relay Agent Information 的描述可知，在网络如果需要根据用户权限的不同而给用户分配不同权限 IP 时，我们就可以通过配置 **ip dhcp relay information option dot1x** 来配置打开 DHCP relay 的 option dot1x 功能，当打开此功能时，设备在进行 relay 时就会结合 802.1x 添加对应的 option 信息到服务器，配置此功能时需要和 dot1x 功能结合使用。

在全局配置模式下，请按如下步骤配置 DHCP option dot1x：

命令	作用
Ruijie(config)**# ip dhcp relay information option dot1x**	启用 DHCP option dot1x 功能
Ruijie(config)**# no ip dhcp relay information option dot1x**	关闭 DHCP option dot1x 功能。

21.2.4 配置 DHCP option 82

当配置命令 **ip dhcp relay information option82** 命令时，设备就会在 DHCP relay 的过程中添加如理解 **DHCP Relay Agent Information** 中所述格式的 **option** 到服务器。

在全局配置模式下，请按如下步骤配置 DHCP option82：

命令	作用
Ruijie(config)**# ip dhcp relay information option82**	启用 DHCP option82 功能
Ruijie(config)**# no ip dhcp relay information option82**	关闭 DHCP option82 功能。

21.2.5 配置 DHCP relay check server-id

当配置命令 **ip dhcp relay check** *server-id* 后，设备在收到 DHCP relay 时就会去解析 DHCP SERVER-ID option，如果此选项不为空，则只对此 server 发送请求，而不对其他配置的服务器发送请求。

在全局配置模式下，请按如下步骤配置 **DHCP relay check** *server-id* 功能：

命令	作用

Ruijie(config)**# ip dhcp relay check** *server-id*	启用 DHCP relay check server-id 功能
Ruijie(config)**# no ip dhcp relay check** *server-id*	关闭DHCP relay check server-id功能。

21.2.6 配置 DHCP relay suppression

当配置命令 **ip dhcp relay suppression** 后，配置了 DHCP realy suppression 的接口不把收到的 DHCP 广播请求转为单播 relay 出去，而对于端口收到的广播报文的正常广播转发不做抑制。

在接口配置模式下，请按如下步骤功能：

命令	作用
Ruijie(config-if)**# ip dhcp relay suppression**	启用 DHCP relay suppresson 功能
Ruijie(config-if)**# no ip dhcp relay suppression**	关闭 DHCP relay suppresson 功能。

21.2.7 DHCP 配置实例

如下命令打开了 dhcp relay 功能、添加了两组服务器地址的例子：

```
Ruijie# configure  terminal
Ruijie(config)# service dhcp                    //打开 dhcp relay 功能
Ruijie(config)# ip helper-address 192.18.100.1  //添加全局服务器地址
Ruijie(config)# ip helper-address192.18.100.2   //添加全局服务器地址
Ruijie(config)# interface GigabitEthernet 0/3
Ruijie(config-if)# ip helper-address 192.18.200.1  //添加接口服务器地址
Ruijie(config-if)# ip helper-address 192.18.200.2  //添加接口服务器地址
Ruijie(config-if)# end
```

21.3 配置 DHCP relay 的其他注意事项

对于二层的网络设备来说，需要实现跨管理 vlan relay 功能时就必须打开 option dot1x、动态地址绑定和 option82 的至少一个功能，否则在二层设备上只能实现管理 vlan 的 relay 功能。

21.3.1 配置 DHCP option dot1x 需要的注意事项

1. 此命令的实际生效需要在 AAA/802.1x 相关的配置正确的情况下。
2. 在应用此方案时需要启用 802.1x 的 DHCP 模式的 IP 授权。
3. 此命令与 **dhcp option82** 命令互斥，不能同时使用。
4. 在启用了 802.1x 的 DHCP 模式的 IP 授权的模式下，也会设置 MAC + IP 的绑定，所以不能与 DHCP 动态绑定功能不能同时启用。

21.3.2 配置 DHCP option82 需要的注意事项

DHCP option82 功能于 **dhcp option dot1x** 功能互斥，不能同时使用

21.4 显示 DHCP 配置

请在特权模式下用 **show running-config** 命令显示 DHCP 配置。

```
Ruijie# show running-config
Building configuration...
Current configuration : 1464 bytes
version RGOS 10.1.00(1), Release(11758)(Fri Mar 30 12:53:11 CST
2007 -nprd
hostname Ruijie
vlan 1
ip helper-address 192.18.100.1
ip helper-address 192.18.100.2
ip dhcp relay information option dot1x
interface GigabitEthernet 0/1
interface GigabitEthernet 0/2
interface GigabitEthernet 0/3
no switchport
ip helper-address 192.168.200.1
ip helper-address 192.168.200.2
interface VLAN 1
ip address 192.168.193.91 255.255.255.0
line con 0
exec-timeout 0 0
line vty 0
exec-timeout 0 0
login
password 7 0137
line vty 1 2
login
```

```
password 7 0137
line vty 3 4
login
end
```

22 DNS 配置

22.1 DNS 概述

每个 IP 地址都可以有一个主机名，主机名由一个或多个字符串组成，字符串之间用小数点隔开。有了主机名，就不要死记硬背每台 IP 设备的 IP 地址，只要记住相对直观有意义的主机名就行了。这就是 DNS 协议所要完成的功能。

主机名到 IP 地址的映射有两种方式：1）静态映射，每台设备上都配置主机到 IP 地址的映射，各设备独立维护自己的映射表，而且只供本设备使用；2）动态映射，建立一套域名解析系统（DNS），只在专门的 DNS 服务器上配置主机到 IP 地址的映射，网络上需要使用主机名通信的设备，首先需要到 DNS 服务器查询主机所对应的 IP 地址。

通过主机名，最终得到该主机名对应的 IP 地址的过程叫做域名解析（或主机名解析）。锐捷设备支持在本地进行主机名解析，也支持通过 DNS 进行域名解析。在解析域名时，可以首先采用静态域名解析的方法，如果静态域名解析不成功，再采用动态域名解析的方法。可以将一些常用的域名放入静态域名解析表中，这样可以大大提高域名解析效率。

22.2 配置域名解析

22.2.1 缺省的 DNS 配置

DNS 的缺省配置如下表：

属性	缺省值
DNS 域名解析功能开关	打开
DNS 服务器 IP 地址	空
静态主机列表	空
DNS 服务器最大个数	6

22.2.2 打开 DNS 域名解析服务

本节描述如何打开 DNS 域名解析功能开关。

命令	作用
Ruijie(config)# **ip domain-lookup**	打开 DNS 域名解析功能开关

使用 **no ip domain-lookup** 命令关闭 DNS 域名解析的功能

```
Ruijie(config)# ip domain-lookup
```

22.2.3 配置 DNS Server

本节描述如何配置 DNS 服务器。只有配置了 DNS 服务器，才能进行动态域名解析。

您如果要删除 DNS 服务器，可以使用 **no ip name-server** [*ip-address*] 命令。其中参数 **ip-address** 表示删除指定的域名服务器，否则删除所有的域名服务器。

命令	作用
Ruijie(config)# **ip name-server** *ip-address*	添加 DNS Server 的 IP 地址。每次执行这条命令，设备都会添加一个 DNS Server。当无法从第一个 Server 获取到域名时，设备会尝试向后续几个 Server 发送 DNS 请求，直到正确收到回应为止。系统最多支持 6 个域名服务器。

22.2.4 静态配置主机名和 IP 地址的映射

本节描述如何配置主机名和 IP 地址的映射。本地维护了一张主机名和 IP 地址的对应表，也叫主机名到 IP 地址的映射表。主机名到 IP 地址的映射表内容有两个来源：手工配置和动态学习。在不能动态学习的情况下，手工配置就有必要了。

命令	作用
Ruijie(config)# **ip host** *host-name ip-address*	手工配置主机名和 IP 地址映射

使用该命令的 **no** 形式就可以删除主机名和 IP 地址的映射。

22.2.5 清除动态主机名缓存表

本节描述如何清除动态主机名缓存表。如果输入 **clear host** 或 **clear host** *命令将清除动态缓存表。否则只删除指定域名的表项。

命令	作用
Ruijie# **clear host** [*word*]	清除动态主机名缓存表。 该命令不能删除静态配置的主机名。

22.2.6 域名解析信息显示

本节描述如何显示 DNS 的相关配置信息：

命令	作用
Ruijie# **show hosts**	查看 DNS 的相关参数

```
Ruijie# show hosts
DNS name server    :
192.168.5.134    static
    host                type          address
www.163.com             static       192.168.5.243
www.ruijie.com          dynamic      192.168.5.123
```

22.2.7 应用举例

Ping 指定域名的主机：

```
Ruijie# ping www.ietf.org
Resolving host[www.ietf.org]。。。。。。
Sending 5,100-byte ICMP Echos to 192.168.5.123,
timeout is 2000 milliseconds.
!!!!!
Success rate is 100 percent(5/5)
Minimum = 1ms Maximum = 1ms, Average = 1ms
```

23 简单网络时间协议(SNTP)

23.1 概述

目前，因特网上普遍采用了通讯协议来实现网络时间同步，即 NTP（Network Time Protocol--网络时间协议），还有一种协议是 NTP 协议的简化版，即 SNTP（Simple Network Time Protocol，简单网络时间协议）。

NTP 协议可以跨越各种平台和操作系统，用非常精密的算法，因而几乎不受网络的延迟和抖动的影响，可以提供 1-50 ms 精度。NTP 同时提供认证机制，安全级别很高。但是 NTP 算法复杂，对系统要求较高。

SNTP（简单网络时间协议）是 NTP 的简化版本，在实现时，计算时间用了简单的算法，性能较高。而精确度一般也能达到 1 秒左右，也能基本满足绝大多数场合的需要。

由于 SNTP 的报文和 NTP 的报文是完全一致的，所以本设备实现的 SNTP Client 能完全兼容 NTP Server。

23.1.1 SNTP 原理

SNTP 协议采用客户/服务器工作方式，服务器通过接收 GPS 信号或自带的原子钟作为系统的时间基准，客户机通过定期访问服务器提供的时间服务获得准确的时间信息，并调整自己的系统时钟，达到网络时间同步的目的。

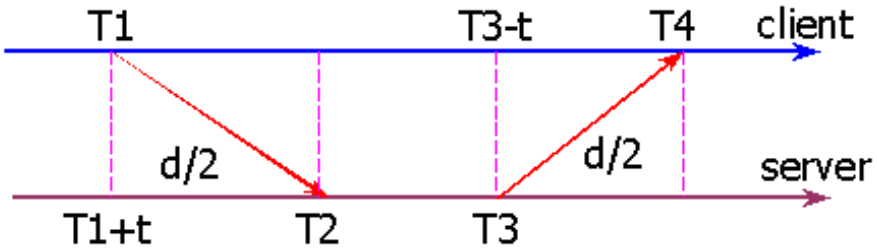


图 1

Originate Timestamp	T1	time request sent by client
Receive Timestamp	T2	time request received at server
Transmit Timestamp	T3	time reply sent by server
Destination Timestamp	T4	time reply received at client

T1：客户方发送查询请求时间(以客户方时间系统为参照)，标记为 Originate Timestamp；

T2：服务器收到查询请求时间(以服务器时间系统为参照)，标记为 Receive

Timestamp；

T3：服务器回复时间信息包时间(以服务器时间系统为参照)，标记为 Transmit Timestamp；

T4：客户方收到时间信息包时间(以客户方时间系统为参照)，标记为 Destination Timestamp；

T：服务器和客户端之间的时间偏差；

d：两者之间的往返时间；

时间求解过程：

因为

T2 = T1 + t + d / 2;

所以

T2 - T1 = t + d / 2;

又因为

T4 = T3 – t + d / 2;

所以

T3 - T4 = t – d / 2;

求得

d = (T4 - T1) - (T3 - T2);

t = ((T2 - T1) + (T3 - T4)) / 2;

求出了 t 和 d，SNTP Client 就可据此算出当前的时间。

即，当前时间为 T4 + t

23.2 配置 SNTP

本章节将介绍如何配置 SNTP

23.2.1 缺省的 SNTP 设置

缺省情况下，SNTP 的配置如下：

项目	缺省值
SNTP 状态	Disable，关闭 SNTP 服务
NTP Server 的 IP 地址	0

SNTP 的同步时间的间隔	1800s
本地时区	+8，即东八区

23.2.2 打开 SNTP

进入特权模式，按以下步骤打开 SNTP：

1) 进入全局配置模式。

```
Ruijie# config
```

2) 打开 SNTP，即时同步一次时钟。后续如果输入这个命令， 都会“即时同步”时钟，不必等待定时同步。(为了避免频繁地同步时间，每两次的“即时同步”时间请勿小于 5 秒)。

```
Ruijie(config)# sntp enable
```

3) 退回到特权模式

```
Ruijie(config)# end
```

4) 显示当前配置

```
Ruijie# show running-config
```

5) 保存配置。

```
Ruijie# copy running-config startup-config
```

您如果要关闭 SNTP，可以用 **no sntp enable** 全局配置命令来关闭 SNTP。

23.2.3 配置 NTP Server 的地址

由于 SNTP 的报文和 NTP 的报文是完全一致的，所以 SNTP Client 能完全兼容 NTP Server。网络上存在着较多的 NTP Server，您可以选择一个网络延迟较少的一个做为交换机上的 NTP Server。

具体的NTP server地址可以登录 http://www.time.edu.cn/或 http://www.ntp.org/上获取。

如 192.43.244.18(time.nist.gov)。

进入特权模式，按以下步骤配置 SNTP Server IP 地址：

1) 进入全局配置模式。

```
Ruijie# config
```

2) 设置 SNTP Server 的 IP 地址。

```
Ruijie(config)# sntp server <ip-addr>
```

3) 退回到特权模式

```
Ruijie(config)# end
```

4) 显示当前配置

```
Ruijie# show running-config
```

5) 保存配置。

```
Ruijie# copy running-config startup-config
```

23.2.4 配置SNTP同步时钟的间隔

SNTP Client 需要设置一定的时间间隔定期访问 NTP Server，以便定时校正时钟。以下步骤将配置交换机和 NTP Server 同步时钟的间隔：

1) 进入全局配置模式。

```
Ruijie# config
```

2) 设置定时同步时钟的间隔，单位为秒。范围为 60 秒-65535 秒，缺省值为 1800 秒。

```
Ruijie(config)# sntp interval <seconds>
```

3) 退回到特权模式

```
Ruijie(config)# End
```

4) 显示当前配置

```
Ruijie# show running-config
```

5) 保存配置。

```
Ruijie# copy running-config startup-config
```

注意：

这里设置的时间间隔不会立即生效，如果要立即生效，请配置完时间间隔后执行 sntp enable 命令。

23.2.5 配置本地时区

通过 SNTP 协议通讯后获取的时间都是格林威治标准时间 (GMT)，为了准确的获取本地时间，需要设置本地时区来对标准时间进行调正。

1） 进入全局配置模式。

```
Ruijie# config
```

2） 配置时区，范围为-23 至 23，负数表示西区，正数表示东区。如 8 表示东八区，-8 表示西 8 区，0 表示格林威治标准时间。缺省值为北京时间东八区。

```
Ruijie(config)# clock timezone <timezone>
```

3） 退回到特权模式

```
Ruijie(config)# End
```

4） 显示当前配置

```
Ruijie# show running-config
```

5） 保存配置

```
Ruijie# copy running-config startup-config
```

您可以通过 **no clock timezone** 来恢复缺省值。

23.3 显示 SNTP

步骤如下：

1) 查看 SNTP 的相关参数

```
Ruijie# show sntp
```

2) 使用 **show sntp** 查看系统保护的配置参数：

```
Ruijie# show sntp
SNTP state               : ENABLE              //SNTP 是否打开
SNTP server              : 192.168.4.12          //NTP Server
SNTP sync interval       : 60              //定时同步的时间间隔
Time zone             : 8(east)                          //本地时区
```

24 NTP 配置

24.1 理解 NTP

Network Time Protocol（NTP）是用来使网络设备时间同步化的一种协议，它可以使网络设备对其服务器或时钟源做同步化，它可以提供高精准度的时间校正（LAN 上与标准间差小于 1 毫秒，WAN 上几十毫秒），且可使用加密确认的方式来防止攻击。

NTP 提供准确时间，首先要有准确的时间来源，这一时间应该是国际标准时间 UTC。NTP 获得 UTC 的时间来源可以是原子钟、天文台、卫星，也可以从 Internet 上获取。这样就有了准确而可靠的时间源。

为防止对时间服务器的恶意破坏，NTP 使用了识别(Authentication)机制，检查时间同步信息是否是真正来自所宣称的服务器并检查资料的返回路径，以提供对抗干扰的保护机制。

目前我司设备支持 NTP 的客户端与服务器功能，即设备既可以从时间服务器上同步时间，也能够作为时间服务器对其他设备进行时间同步。在作为服务器工作时设备仅支持单播 Server 模式。

24.2 配置 NTP

本章介绍怎样在我们的系统实现中配置 NTP 客户端和服务器。

- 配置 NTP 全局安全识别机制
- 配置 NTP 全局认证密钥
- 配置 NTP 全局信任密钥 ID
- 配置 NTP 服务器
- 关闭接口接收 NTP 报文
- NTP 功能开关
- 配置 NTP 实时同步
- 配置 NTP 更新硬件时钟
- 设置 NTP 主时钟
- 配置 NTP 服务的访问控制权限

24.2.1 配置 NTP 全局安全识别机制

锐捷的 NTP 客户端支持与服务器进行加密通信，加密方式为密钥加密机制。

配置 NTP 客户端通过加密方式与服务器通信分两步：第一，对 NTP 客户端进行全局安全识别以及全局密钥相关设置；第二，设置通信服务器的信任密钥设置；NTP 全局安全设置机制属于第一个设置步骤，但是要真正发起与服务器之间的加密通信，还要对对应的服务器设置认证密钥。

在缺省情况下，客户端不使用全局安全识别机制。如果未使用安全识别机制则不对通信进行加密处理。但是仅仅设置了全局安全标志，并不代表一定采用了加密方式完成服务器与客户端的通信，还必须完成其他全局密钥配置并设置服务器加密密钥才可能发起和服务器的加密通信。

要配置全局安全识别机制，在全局配置模式中执行以下命令：

命令	作用
ntp authenticate	配置 NTP 全局安全识别机制。
no ntp authenticate	关闭 NTP 全局安全识别机制。

报文的验证是通过 **ntp authentication-key**、**ntp trusted-key** 指定的信任密钥进行验证的。

24.2.2 配置 NTP 全局认证密钥

进行 NTP 安全认证全局配置，接下来的配置就是设置全局认证密钥。

配置全局认证密钥，每个密钥有一个唯一的 key-id 标示，客户可以用 **ntp trusted-key** 将该 key-id 对应的密钥设置为全局信任密钥。

要配置全局认证密钥，在全局配置模式中执行以下命令：

命令	作用
ntp authentication-key *key-id* **md5** *key-string* [*enc-type*]	配置 NTP 全局认证密钥。 key-id：1-4294967295 key-string：长度范围任意 enc-type：有 0 和 7 两种类型。
no ntp authentication-key *key-id* **md5** *key-string* [*enc-type*]	删除 NTP 全局认证密钥。

配置了全局认证密钥并不说明该密钥一定有效，在使用该密钥之前必须将该密钥配置成为全局信任密钥。

注意:

我司目前的版本只支持最大 1024 认证密钥，每个服务器允许设置唯一一个密钥进行安全通信。

24.2.3 配置 NTP 全局信任密钥 ID

进行全局安全认证配置的最后一个阶段，就是将一个全局密钥配置成全局信任密钥。通过该信任密钥，用户才可以发送加密数据并检验收到数据报的合法性。

要指定全局信任密钥，在全局配置模式中执行以下命令：

命令	作用
ntp trusted-key *key-id*	配置 NTP 全局信任密钥 ID。
no ntp trusted-key *key-id*	删除 NTP 全局信任密钥 ID。

以上三步设置仅仅是完成安全认证机制的第一个步骤，真正发起与客户服务器的加密通信必须对相应的服务器设置信任密钥。

注意:

直接删除全局认证密钥，则该密钥对应的信任信息也会被删除。

24.2.4 配置 NTP 服务器

在缺省情况下，没有配置 NTP 服务器。锐捷的客户端系统支持最多同时与 20 个 NTP 服务器交互，（在全局认证以及密钥相关设置完成后）可以为每一个服务器设置一个认证密钥，发起与服务器的加密通信。

与服务器的默认通信版本为 NTP 版本 3，同时可以配置发送 NTP 报文的源接口，并只在发送接口上接收对应服务器的 NTP 报文。

配置一个 NTP 服务器，在全局配置模式下执行以下命令：

命令	作用

ntp server *ip-addr* [**version** *version*][**source** *if-name number*][**key** *keyid*][**prefer**]	配置 NTP 服务器 version（NTP 版本号）：1-3 if-name（接口类型）：包括 Aggregateport、Dialer GigabitEthernet、Loopback、Multilink、Null、Tunnel、Virtual-ppp、Virtual-template、Vlan 类型。 keyid：1-4294967295
no ntp server *ip-addr*	删除 NTP 服务器

只有完成了全局安全识别以及密钥设置机制，这时候设置服务器通信的信任密钥，才能发起于服务器的加密通信，而加密通信的完成需要服务器端信任相同的密钥。

24.2.5 关闭接口接收 NTP 报文

该命令的功能是关闭对应接口上接收 NTP 报文。

在缺省情况下，任意接口上接收的 NTP 报文都可以提供给客户端进行时钟调整，通过设置这个功能，可以屏蔽对应接口上收到的 NTP 报文。

注意：

能够进行该功能命令配置的接口肯定是能够配置 IP 收发报文的接口，在其他接口上没有该命令。

在接口配置模式下执行以下命令，配置关闭接口接收 NTP 报文：

命令	作用
interface *interface-type number*	进入接口配置模式
ntp disable	关闭接口接收 NTP 报文的功能。

要打开接口接收 NTP 报文的功能，在接口模式下使用 **no ntp disable** 命令

24.2.6 NTP 功能开关

no ntp 命令的功能是关闭 NTP 同步服务，停止时间同步，同时清空相关的 NTP 配置信息。

在缺省情况下，NTP 功能是关闭的，但只要配置了 NTP 服务器或 NTP 安全识别机制，NTP 功能就会被打开。

要关闭 NTP，在全局配置模式下执行以下命令：

命令	作用
no ntp	关闭 NTP 功能。
ntp authenticate 或 **ntp server** *ip-addr* [**version** *version*][**source** *if-name number*][**key** *keyid*][**prefer**]	打开 NTP 功能

24.2.7 配置 NTP 更新硬件时钟

使用此功能可以让 NTP 客户端使用从外部时钟源同步得来的时钟值更新设备的硬件时钟。

在全局配置模式下执行以下命令配置更新硬件时钟：

命令	作用
ntp update-calendar	配置更新硬件时钟
no ntp update-calendar	取消配置更新硬件时钟

在缺省情况下没有配置 NTP 更新硬件时钟。配置之后，NTP 客户端会在每次与外部时钟源同步成功时也同时更新设备的硬件时钟。一般情况下建议启用此功能，使设备的硬件时钟也能同时保持精准。

24.2.8 设置 NTP 主时钟

该功能用来设置本地时钟作为 NTP 主时钟（本地时钟参考源可靠），为其它设备提供同步时间。

在通常情况下，本地系统都会直接或间接地与外部的时钟源进行同步。但若由于网络连接故障等原因而导致本地系统无法与外部时钟源同步时，可以通过该命令设置本地时钟参考源可靠，为其他设备提供同步时间。

一旦进行了此设置，系统便不会与比其时钟层数数值更高的的时钟源进行同步。

说明：

NTP 使用“层数（stratum）”的概念来描述设备距离权威时钟源的“跳数（hops）”。一个层数为 1 的时间服务器应当有个直连的原子钟或电波钟；层数为 2 的时间服务器就从层数为 1 的服务器获取时间；层数为 3 的服务器就从层数为 2 的获取时间……如此递推。因此时钟层数数值更低的时钟源即被认为拥有更高的时钟精度。

在全局配置模式下执行以下命令配置 NTP 主时钟功能：

命令	作用
ntp master [*stratum*]	设置本地时钟作为 NTP 主时钟并指定相应时钟层数。时钟层数取值范围为 1～15；若不指定该参数则默认值为 8。
no ntp master	取消 NTP 主时钟设置

如下可以设置本地时钟参考源可靠，并设置其时钟层数为 12。

```
Ruijie(config)# ntp master 12
```

注意：

使用此命令时必须特别小心。将本地时钟设置为主时钟（尤其是指定了较低的时钟层数值时）很有可能将真正的有效时钟源覆盖。如果对同一网络中的多个设备都使用了该命令，则可能由于设备之间的时钟差异导致网络的时钟同步不稳定。

另外，使用该命令前若系统从未与外部时钟源同步过，则有可能需要手动校准系统时钟以保证其不会有过大的偏差（关于如何手动校准系统时钟请参考《交换机基础管理配置指南》中的系统时间配置部分）。

此命令不受 ntp 访问控制限制（即使 NTP 访问控制功能有对应的匹配限制，此命令仍然生效）。

24.2.9 配置 NTP 服务的访问控制权限

NTP 服务的访问控制功能提供了一种最小限度的安全措施（更安全的方法是使用 NTP 身份验证机制）。系统在缺省情况下未配置任何 NTP 访问控制规则。

在全局配置模式下执行以下命令配置 NTP 服务的访问控制权限：

命令	作用
ntp access-group { peer | serve | serve-only | query-only } *access-list-number|access-list-name*	设置对本地服务的访问控制权限
no ntp access-group { peer | serve | serve-only | query-only } *access-list-number|access-list-name*	取消对本地服务的访问控制权限的设置

其中：

- **peer**：既允许对本地 NTP 服务进行时间请求和控制查询，也允许本地设备与远程系统同步时间（完全访问权限）。
- **serve**：允许对本地 NTP 服务进行时间请求和控制查询，但不允许本地设备与远程系统同步时间。
- **serve-only**：仅允许对本地 NTP 服务进行时间请求。
- **query-only**：仅允许对本地 NTP 服务进行控制查询。
- *access-list-number*：IP 访问控制列表标号；范围为 1～99 和 1300～1999。关于如何创建 IP 访问控制列表请参考《访问控制列表配置指南》中的相关描述。
- *access-list-name*：IP 访问控制列表名。关于如何创建 IP 访问控制列表请参考《访问控制列表配置指南》中的相关描述。

当一个访问请求到达时，NTP 服务按照从最小访问限制到最大访问限制的顺序依次匹配规则，以第一个匹配到的规则为准。匹配顺序为 peer、serve、serve-only、query-only。

注意：

目前系统暂未支持控制查询功能（用于通过网络管理设备对 NTP 服务器进行控制，如设置闰秒标记或监控其工作状态等）。虽然是按照上述顺序进行规则匹配，但涉及到与控制查询相关的请求都无法支持。

如果未配置任何访问控制规则，则所有访问都是允许的。但一旦配置了访问控制规则，则仅有规则中所允许的访问才能进行。

可以如下配置以允许第 1 号访问列表中的对端设备对本地设备进行时间请求、查询控制和时间同步；并限制第 2 号访问列表中的对端设备仅能对本地设备进行时间请求：

```
Ruijie(config)# ntp access-group peer 1
Ruijie(config)# ntp access-group serve-only 2
```

24.3 NTP 信息显示

24.3.1 调试 NTP

要进行 NTP 功能调试，可以通过该命令输出必要的调试信息，进行故障诊断和排除。

调试 NTP 功能，在特权模式下执行以下命令：

命令	作用
debug ntp	打开调试功能。
no debug ntp	关闭调试功能。

24.3.2 显示 NTP 信息

在特权模式下，您可以使用 **show ntp status** 命令来显示当前的 NTP 信息。

显示 NTP 状态信息，在特权模式下执行以下命令：

命令	作用
show ntp status	显示当前的 NTP 信息

只有在配置了相关的通信服务器之后该命令才能打印出显示信息。

```
Ruijie# show ntp status
Clock is synchronized, stratum 9, reference is 192.168.217.100
nominal freq is 250.0000 Hz, actual freq is 250.0000 Hz,
precision is 2**18
reference time is AF3CF6AE.3BF8CB56 (20:55:10.000 UTC Mon Mar
1 1993)
clock offset is 32.97540 sec, root delay is 0.00000 sec
root dispersion is 0.00003 msec, peer dispersion is 0.00003 msec
```

注：starum 代表当前时钟的等级，reference 为同步服务器的地址，frep 当前系统时钟频率，precision 为当前系统时钟精度，reference time 为同步服务器参考时钟的 UTC 时间值，clock offset 为当前时钟偏移，root delay 为当前时钟延迟，root dispersion 为顶级服务器精度，peer dispersion 为同步服务器精度。

24.4 配置范例

在以下配置中，网络中有一个指定为 master 的 NTP 服务器，并打开了相应的认证机制，配置了一个 key-id 为 6、key-string 为 wooooop 的密钥，并设置该密钥为服务器信任密钥；为了配置锐捷的客户机可以和网络上的 NTP 服务器进行时间同步，我们将对客户进行如下配置，启动安全认证，并配置一个和 NTP 服务器端一样的密钥，然后将我们的同步服务器设定为网络上的这个 NTP 服务器，开始时间同步。

IPv4 配置范例：

```
Ruijie(config)# no ntp
Ruijie(config)# ntp authentication-key 6 md5 wooooop
Ruijie(config)# ntp authenticate
```

```
Ruijie(config)# ntp trusted-key 6
Ruijie(config)# ntp server 192.168.210.222 key 6
Ruijie(config)# interface gigabitEthernet 0/1
Ruijie(config-if)# ntp disable
Ruijie(config-if)# no ntp disable
```

25 UDP-Helper 配置

25.1 UDP-Helper 配置

25.1.1 UDP-Helper 简介

UDP-Helper 的主要功能是实现 UDP 广播报文的中继转发。通过配置需要转发的目的服务器，它能将 UDP 广播报文转换成单播报文后发往指定的目的服务器，起到中继的作用。

使能 UDP-Helper 功能后，将对接收到的广播报文的目的 UDP 端口号进行判断，当这个目的 UDP 端口号与需要转发的端口号相匹配时，它将修改报文的目的 IP 地址为配置的目的服务器的 IP 地址，以单播的方式发送指定的目的服务器。

使能 UDP-Helper 功能后，默认对 69、53、37、137、138、49 端口的广播报文进行中继转发。

说明:

BOOTP/DHCP 广播报文的中继由 DHCP Relay 模块使用 UDP 端口 67、68 来实现，因此端口 67、68 不能配置成 UDP-Helper 中继端口。

25.2 配置 UDP-Helper

25.2.1 缺省的 UDP-Helper 配置

属 性	缺 省 值
中继转发功能	关闭
中继转发的 UDP 端口号	打开 UDP-Helper 功能后默认对 69、53、37、137、138、49 端口的 UDP 广播报文进行中继转发。
中继转发的目的服务器	没有配置

25.2.2 启动 UDP-Helper 的中继转发功能

命 令	功 能

Ruijie(config)# **udp-helper enable**	**udp-helper enable** 命令用来启动 UDP 广播报文的中继转发功能。缺省 UDP 广播报文的中继转发处于关闭状态

使用 **no udp-helper enable** 命令用来关闭 UDP 的中继转发功能。

说明:

1) 缺省情况下中继转发功能处于关闭状态。

2) 启动 UDP 广播报文的中继转发功能，则默认对 69、53、37、137、138、49 的 UDP 端口的广播报文进行中继转发。

3) 当关闭 DUP 广播报文的中继转发功能后，所有已配置的 UDP 端口都被取消，包括默认端口。

25.2.3 配置中继转发的目的服务器

命 令	功 能
Ruijie(config-if)# **ip helper-address** *ip-address*	配置 UDP 广播报文中继转发的目的服务器。缺省情况下没有配置中继转发的目的服务器。

使用 **no ip helper-address** 命令删除中继转发的目的服务器。

说明:

1) 一个接口最多对应 20 个目的服务器

2) 如果中继转发的目的服务器是在某指定接口上配置的，在启动 UDP-Helper 功能后，从该接口接收的指定 UDP 端口的广播报文，将以单播形式发送到该接口配置的目的服务器上。

25.2.4 配置需要中继转发的 UDP 端口

命 令	功 能
Ruijie(config)# **ip forward-protocol udp** *ID*	配置需要中继转发的 UDP 端口。 如果只指定了 UDP 参数，则默认对缺省的端口进行中继转发，否则根据需要进行配置。 使能 UDP-Helper 功能后，缺省对 69、53、37、137、138、49 端口的广播报文将进行中继转发

使用 **no ip forward-protocol udp port** 命令关闭需要中继转发的 UDP 端口。

说明:

- 只有打开 UDP-Helper 的中继转发功能并配置了中继转发的目的服务器后，才能配置需要中继转发的 UDP 端口。否则，将会有错误提示信息。
- 当启动 UDP 中继转发功能后， 69、53、37、137、138、49 默认端口的广播 UDP 报文转发功能将会立即启动，不需要用户再配置。
- 设备最多支持配置 256 个需要中继转发的 UDP 端口。
- 对默认端口的配置有两种方式，如 **ip forward-protocol udp domain** 和 **ip forward-protocol udp** *53* 的配置是相同的。

26 SNMP 配置

26.1 SNMP 相关知识

26.1.1 概述

SNMP 是 Simple Network Manger Protocol（简单网络管理协议）的缩写，在 1988 年 8 月就成为一个网络管理标准 RFC1157。到目前，因众多厂家对该协议的支持，SNMP 已成为事实上的网管标准，适合于在多厂家系统的互连环境中使用。利用 SNMP 协议，网络管理员可以对网络上的节点进行信息查询、网络配置、故障定位、容量规划，网络监控和管理是 SNMP 的基本功能。

SNMP 是一个应用层协议，为客户机/服务器模式，包括三个部分：

- SNMP 网络管理器
- SNMP 代理
- MIB 管理信息库

SNMP 网络管理器，是采用 SNMP 来对网络进行控制和监控的系统，也称为 NMS (Network Management System)。常用的运行在 NMS 上的网管平台有 HP OpenView 、CiscoView、CiscoWorks 2000，锐捷网络针对自己的网络设备，开发了一套网管软件－－Star View。这些常用的网管软件可以方便的对网络设备进行监控和管理。

SNMP 代理（SNMP Agent）是运行在被管理设备上的软件，负责接受、处理并且响应来自 NMS 的监控和控制报文，也可以主动发送一些消息报文给 NMS。

NMS 和 Agent 的关系可以用如下的图来表示：

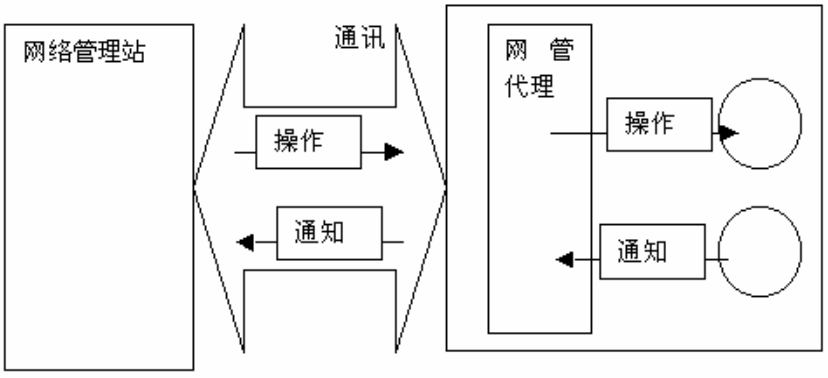


图 1 网络管理站（NMS）与网管代理（Agent）的关系图

MIB（Management Information Base）是一个虚拟的网络管理信息库。被管理的网络设备中包含了大量的信息，为了能够在 SNMP 报文中唯一的标识某个特定的管理单元，MIB 采用树形层次结构来描述网络设备中的管理单元。树的节点表示某

个特定的管理单元。如下图 MIB 对象命名树，为了唯一标识网络设备中的某个管理单元 System，可以采用一串的数字来表示，如{1.3.6.1.2.1.1}这一串数字即为管理单元的 Object Identifier（单元标识符），MIB 则是网络设备的单元标识符的集合。

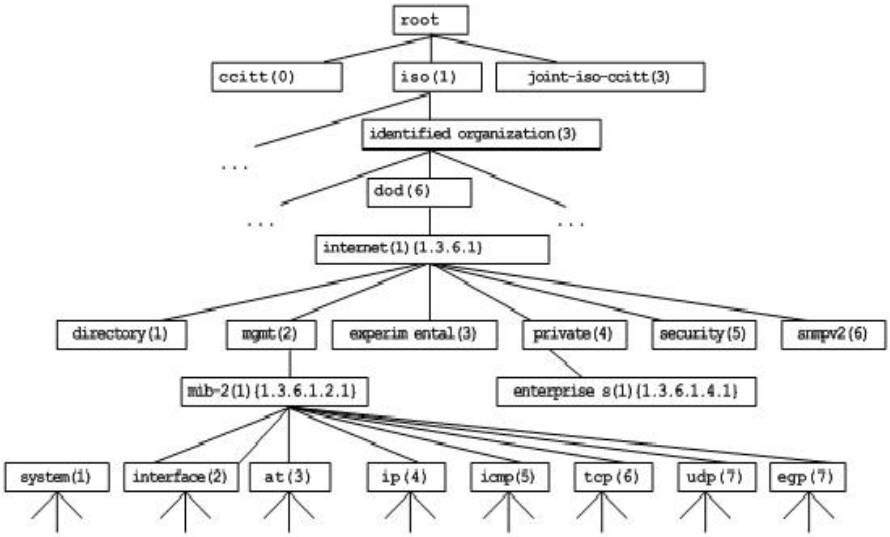


图 2　MIB 树形层次结构

26.1.2 SNMP 协议版本

目前 SNMP 支持以下版本：

- SNMPv1 ：简单网络管理协议的第一个正式版本，在 RFC1157 中定义。
- SNMPv2C：基于共同体（Community-Based）的 SNMPv2 管理架构，在 RFC1901 中定义的一个实验性协议。
- SNMPv3 ：通过对数据进行鉴别和加密，提供了以下的安全特性：

1. 确保数据在传输过程中不被篡改；
2. 确保数据从合法的数据源发出；
3. 加密报文，确保数据的机密性；

SNMPv1 和 SNMPv2C 都采用基于共同体（Community-based）的安全架构。通过定义主机地址以及认证名(Commumity String)来限定能够对代理的 MIB 进行操作的管理者。

SNMPv2C 增加了 Get-bulk 操作机制并且能够对管理工作站返回更加详细的错误信息类型。Get-bulk 操作能够一次性地获取表格中的所有信息或者获取大批量的数据，从而减少请求-响应的次数。SNMPv2C 错误处理能力的提高包括扩充错误代码以区分不同类型的错误，而在 SNMPv1 中这些错误仅有一种错误代码。现在通过错误代码可以区分错误类型。由于网络上可能同时存在支持 SNMPv1 和 SNMPv2C 的管理工作站，因此 SNMP 代理必须能够识别 SNMPv1 和 SNMPv2C 报文，并且能返回相应版本的报文。

26.1.3 SNMP 管理操作

SNMP 协议中的 NMS 和 Agent 之间的交互信息，定义了 6 种操作类型：

1. Get-request 操作：NMS 从 Agent 提取一个或多个参数值。
2. Get-next-request 操作：NMS 从 Agent 提取一个或多个参数的下一个参数值。
3. Get-bulk 操作：NMS 从 Agent 提取批量的参数值；
4. Set-request 操作：NMS 设置 Agent 的一个或多个参数值。
5. Get-response 操作：Agent 返回的一个或多个参数值，是 Agent 对 NMS 前面 3 个操作的响应操作。
6. Trap 操作：Agent 主动发出的报文，通知 NMS 有某些事情发生。

前面的 4 个报文是由 NMS 向 Agent 发出的，后面两个是 Agent 发给 NMS 的（注意：SNMPv1 版本不支持 Get-bulk 操作）。下图描述了这几种种操作。

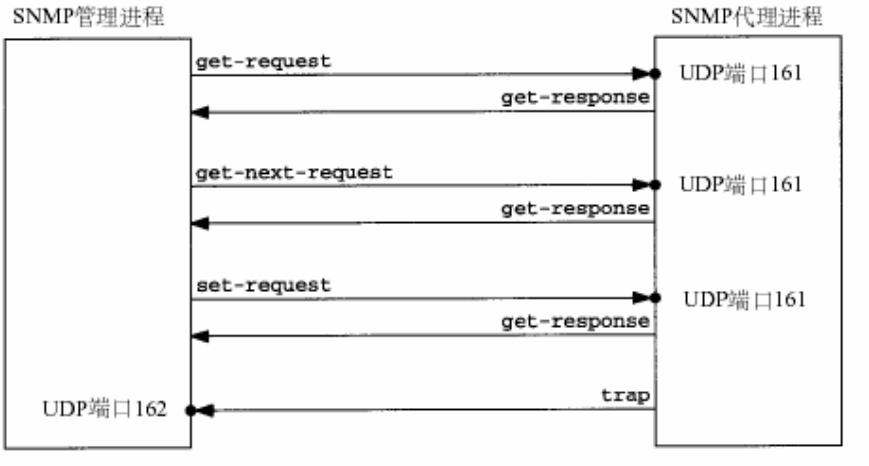


图 3　SNMP 的报文类型

NMS 向 Agent 发出的前面 3 种操作和 Agent 的应答操作采用 UDP 的 161 端口。Agent 发出的 Trap 操作采用 UDP 的 162 端口。

26.1.4 SNMP 安全

SNMPv1 和 SNMPv2 版本使用认证名用来鉴别是否有权使用 MIB 对象。为了能够管理设备，网络管理系统 (NMS)的认证名必须同设备中定义的某个认证名一致。

一个认证名可以有以下属性:

- 只读(Read-only)：为被授权的管理工作站提供对所有 MIB 变量的读权限。
- 读写(Read-write)：为被授权的管理工作站提供对所有 MIB 变量的读写权限。

在 SNMPv2 的基础上， SNMPv 3 通过安全模型以及安全级别来确定对数据采用哪种安全机制进行处理；目前可用的安全模型有三种类别：SNMPv1、SNMPv2C、

SNMPv3。

下表为目前可用的安全模型以及安全级别

安全模型	安全级别	鉴别	加密	说明
SNMPv1	noAuthNoPriv	认证名	无	通过认证名确认数据的合法性
SNMPv2c	noAuthNoPriv	认证名	无	通过认证名确认数据的合法性
SNMPv3	noAuthNoPriv	用户名	无	通过用户名确认数据的合法性
SNMPv3	authNoPriv	MD5 或者 SHA	无	提供基于 HMAC-MD5 或者 HMAC-SHA 的数据鉴别机制
SNMPv3	authPriv	MD5 或者 SHA	DES	提供基于 HMAC-MD5 或者 HMAC-SHA 的数据鉴别机制 提供基于 CBC-DES 的数据加密机制

26.1.5 SNMP 引擎标识

引擎标识用于唯一标识一个 SNMP 引擎。由于每个 SNMP 实体仅包含一个 SNMP 引擎，它将在一个管理域中唯一标识一个 SNMP 实体。因此，作为一个实体的 SNMPv3 代理器必须拥有一个唯一的引擎标识，即 SnmpEngineID。

引擎标识为一个 OCTET STRING，长度为 5～32 字节长。在 RFC3411 中定义了引擎标识的格式:

- 前 4 个字节标识厂商的私有企业号（由 IANA 分配），用 HEX 表示。
- 第 5 个字节表示剩下的字节如何标识：

0：保留

1：后面 4 个字节是一个 Ipv4 地址。

2：后面 16 个字节是一个 Ipv6 地址。

3：后面 6 个字节是一个 MAC 地址。

4：文本，最长 27 个字节，由厂商自行定义。

5：16 进制值，最长 27 个字节，由厂商自行定义。

6-127：保留。

128-255：由厂商特定的格式。

26.2 SNMP 的配置

SNMP 的配置工作在网络设备的全局配置模式下完成，在进行 SNMP 配置前，请先进入全局配置模式。

26.2.1 设置认证名及访问权限

SNMPv1/SNMPv2C 采用基于共同体（Community-based）的安全方案，SNMP 代理只接受来自相同认证名（Community-String）的管理操作，与网络设备的认证名不符的 SNMP 报文将不被响应，直接丢弃。认证名相当于 NMS 和 Agent 之间的密码。

- 可以设置访问列表关联，只有指定的 IP 地址的 NMS 可以管理；
- 可以设定该共同体的操作权限，是 ReadOnly（只读）还是 ReadWrite（读写）；
- 指定视图的名称，用于基于视图的管理。默认没有指定视图，即允许访问 所有 MIB 对象；
- 可以指明能够使用该认证名的管理者的 IP。 若不指明，则表示不限制使用该认证名的管理者的 IP 地址。缺省为不限制使用该认证名的管理者的 IP 地址；

要配置 SNMP 认证名，在全局配置模式下执行如下命令：

命令	作用
Ruijie(config)# **snmp-server community** *string* [**view** *view-name*] [**ro** | **rw**] [**host** *host-ip*] [*num*]	设置认证名和权限。

可以配置一条或者多条指定，来指定多个不同的共同体名称，使得网络设备可以供不同的权限的 NMS 的管理，要删除共同体名称和权限，在全局配置模式下，执行 **no snmp-server community** 命令。

26.2.2 配置 MIB 视图和组

可以使用基于视图的访问控制模型来判定一个操作关联的管理对象是否在视图允许之内或被排除在外，只有在视图允许之内的管理对象才被允许访问。在进行控制时，一般是将某些用户和一个组关联，再将某个组与某个视图关联。一个组内的用户具有相同的访问权限。

- 可以设置包含视图和排除视图；
- 可以为一组用户设置只读的视图和可写的视图；
- 如果是 SNMPv3 的用户，可以为其指定使用的安全级别，是否需要进行认证、是否需要进行加密；

要配置 MIB 视图和组，在全局配置模式下执行如下命令：

命令	作用
Ruijie(config)# **snmp-server view** *view-name oid-tree* {**include** | **exclude**}	创建一个 MIB 视图，包含或排除关联的 MIB 对象。

Ruijie(config)# **snmp-server group** *groupname* {**v1**|**v2c** |**v3**{**auth**|**noauth**|**priv**}}[**read** *readview*][**write** *writeview*][**access**{*num*|*name*}]	创建一个组，并和视图关联。

使用 **no snmp-server view** *view-name* 命令来删除一个视图，或者使用 **no snmp-server view** *view-name oid-tree* 命令在一个视图中删除一棵子树。也可以使用 **no snmp-server group** *groupname* 命令来删除一个组。

26.2.3 配置 SNMP 用户

可以使用基于用户的安全模型来进行安全管理，基于用户的管理必须事先配置用户的信息，NMS 只有使用合法的用户才能同代理进行通信。

对于 SNMPv3 用户，可以指定安全级别、认证算法（MD5 或 SHA）、认证口令、加密算法（目前只有 DES）和加密口令；

要配置 SNMP 用户，在全局配置模式下执行如下命令：

命令	作用
Ruijie(config)# **snmp-server user** *username roupname* {**v1** | **v2** | **v3** [**encrypted**] [**auth** { **md5**|**sha** } *auth-password*] [**priv des56** *priv-password*] } [**access** {*num* | *name*}]	设置用户信息。

通过 **no snmp-server user** *username groupname* 删除指定用户。

26.2.4 配置 SNMP 主机地址

Agent 在特定的情况下，也会主动的向 NMS 发送消息，要配置 Agent 主动发送消息的 NMS 主机地址，在全局配置模式下，执行如下指令：

命令	作用
Ruijie(config)# **snmp-server host**{ *host-addr* | **ipv6** ipv6-addr} [**vrf** *vrfname*] [**traps**] [**version**{**1**|**2c**|**3**[**auth**|**noauth**|**priv**]}]*community-string* [**udp-port** *port-num*] [**type**]	设置 SNMP 主机地址，主机端口，vrf 选项，消息类型，认证名（在 SNMPv3 下是用户名）、安全级别（仅 SNMPv3 支持）等

26.2.5 设置 SNMP 代理参数

可以对 SNMP 的 Agent 的基本参数进行配置，设置设备的联系方式、设备位置、序列号的信息，NMS 通过访问设备的这些参数，便可以得知设备的联系人，设备所在的物理位置等信息。

要配置 SNMP 代理参数，在全局配置模式下，执行如下指令：

命令	作用
Ruijie(config)# **snmp-server contact** *text*	设置系统的联系方式
Ruijie(config)# **snmp-server location** *text*	设置系统的位置
Ruijie(config)# **snmp-server chassis-id** *number*	设置系统的序列码

26.2.6 定义 SNMP 代理最大数据报文长度

为了减少对网络带宽的影响，可以定义 SNMP 代理的数据包的最大长度。在全局配置模式下，执行如下指令：

命令	作用
Ruijie(config)# **snmp-server packetsize** *byte-count*	设置最大代理数据包大小

26.2.7 屏蔽 SNMP 代理

SNMP 代理服务是锐捷产品提供的一个服务，默认是启动的，在不需要代理服务的时候，可以通过如下方式屏蔽 snmp 代理功能以及相关配置信息；屏蔽 snmp 代理功能，在全局配置模式下，执行如下指令：

命令	作用
Ruijie(config)# **no snmp-server**	屏蔽 SNMP 代理服务

26.2.8 关闭 SNMP 代理

不同于屏蔽命令，锐捷产品提供了关闭 snmp 代理的命令，该命令会直接 snmp 所有服务，但是不会屏蔽代理的配置信息；要关闭 SNMP 代理服务，在全局配置模式下，执行如下指令：

命令	作用
Ruijie(config)# **no enable service snmp-agent**	关闭 SNMP 代理服务

26.2.9 配置 Agent 主动向 NMS 发送 Trap 消息

Trap 是 Agent 不经请求主动向 NMS 发送的消息，用于报告一些紧急而重要的事件的发生。缺省是不允许 Agent 主动发送 Trap 消息，如果要允许，在全局配置模式下，执行如下指令：

命令	作用

Ruijie(config)# **snmp-server enable traps** [*type*] [*option*]	允许 Agent 主动发送 Trap 消息
Ruijie(config)# **no snmp-server enable traps** [*type*] [*option*]	禁止 Agent 主动发送 Trap 消息

26.2.10 Link Trap 策略配置

在设备中可以基于接口配置是否发送该接口的 LinkTrap，当功能打开时，如果接口发生 Link 状态变化，SNMP 将发出 LinkTrap,反之则不发。缺省情况下，该功能打开。

命令	作用
Ruijie(config)# **interface** *interface-id*	进入端口配置模式
Ruijie(config-if)# [**no**] **snmp trap link-status**	打开或者关闭发送该接口 link trap 的功能.

下面配置将配置接口为不发送 Link trap:

```
Ruijie(config)# interface fastEthernet 0/1
Ruijie(config-if)# no snmp trap link-status
```

26.2.11 配置发送消息操作的参数

可以指定 Agent 发送 Trap 消息的一些参数，执行如下指令来设置：

命令	作用
Ruijie(config)# **snmp-server trap-source** *interface*	指定发送 Trap 消息的源接口
Ruijie(config)# **snmp-server queue-length** *length*	指定每个 Trap 消息报文的队列长度
Ruijie(config)# **snmp-server trap-timeout** *seconds*	指定发送 Trap 消息的时间间隔

26.2.12 配置接口索引维持功能

若每次重新初始化（重启）后的接口索引值维持不变，将有利于网络管理。当要求重启后设备的接口索引值不发生变化，执行如下命令：

命令	作用
Ruijie(config)# **snmp-server if-index persist**	打开接口索引维持功能

若要关闭该功能，执行 no snmp-server if-index persist 命令。缺省情况下，该功能是关闭的。

下面配置打开接口索引维持功能：

```
Ruijie(config)# snmp-server if-index persist
```

26.3 SNMP 的监控与维护

26.3.1 查看当前的 SNMP 状态

为了监控 SNMP 状态和排除 SNMP 配置中的一些故障，锐捷产品提供了 SNMP 的监控指令，可以方便的查看当前网络设备的 SNMP 的状态，在特权用户模式下，执行 **show snmp** 来查看当前的 SNMP 状态。

```
Ruijie# show snmp
Chassis: 1234567890 0987654321
Contact: wugb@i-net.com.cn
Location: fuzhou
2381 SNMP packets input
5 Bad SNMP version errors
6 Unknown community name
0 Illegal operation for community name supplied
0 Encoding errors
9325 Number of requested variables
0 Number of altered variables
31 Get-request PDUs
2339 Get-next PDUs
0 Set-request PDUs
2406 SNMP packets output
0 Too big errors (Maximum packet size 1500)
4 No such name errors
0 Bad values errors
0 General errors
2370 Get-response PDUs
36 SNMP trap PDUs
SNMP global trap: disabled
SNMP logging: enabled
SNMP agent: enabled
.
```

对于上述的统计报文信息的解释见下表：

显示信息	描述
Bad SNMP version errors	SNMP 版本不对

Unknown community name	不能识别的认证名称
Illegal operation for community name supplied	非法操作
Encoding errors	编码错误
Get-request PDUs	Get-request 报文
Get-next PDUs	Get-next 报文
Set-request PDUs	Set-request 报文
Too big errors (Maximum packet size 1500)	响应报文太大
No such name errors	不存在指定的管理单元
Bad values errors	设定值类型错误
General errors	一般性错误
Get-response PDUs	Get-response 报文
SNMP trap PDUs	SNMP trap 报文

26.3.2 查看当前 SNMP 代理支持的 MIB 对象

在特权用户模式下，执行 **show snmp mib** 来查看当前的代理支持的 MIB 对象。

```
Ruijie# show snmp mib
sysDescr
sysObjectID
sysUpTime
sysContact
sysName
sysLocation
sysServices
sysORLastChange
snmpInPkts
snmpOutPkts
snmpInBadVersions
snmpInBadCommunityNames
snmpInBadCommunityUses
snmpInASNParseErrs
snmpInTooBigs
snmpInNoSuchNames
snmpInBadValues
snmpInReadOnlys
snmpInGenErrs
snmpInTotalReqVars
snmpInTotalSetVars
snmpInGetRequests
```

```
snmpInGetNexts
snmpInSetRequests
snmpInGetResponses
snmpInTraps
snmpOutTooBigs
snmpOutNoSuchNames
snmpOutBadValues
snmpOutGenErrs
snmpOutGetRequests
snmpOutGetNexts
snmpOutSetRequests
snmpOutGetResponses
snmpOutTraps
snmpEnableAuthenTraps
snmpSilentDrops
snmpProxyDrops
entPhysicalEntry
entPhysicalEntry.entPhysicalIndex
entPhysicalEntry.entPhysicalDescr
entPhysicalEntry.entPhysicalVendorType
entPhysicalEntry.entPhysicalContainedIn
entPhysicalEntry.entPhysicalClass
entPhysicalEntry.entPhysicalParentRelPos
entPhysicalEntry.entPhysicalName
entPhysicalEntry.entPhysicalHardwareRev
entPhysicalEntry.entPhysicalFirmwareRev
entPhysicalEntry.entPhysicalSoftwareRev
entPhysicalEntry.entPhysicalSerialNum
entPhysicalEntry.entPhysicalMfgName
entPhysicalEntry.entPhysicalModelName
entPhysicalEntry.entPhysicalAlias
entPhysicalEntry.entPhysicalAssetID
entPhysicalEntry.entPhysicalIsFRU
entPhysicalContainsEntry
entPhysicalContainsEntry.entPhysicalChildIndex
entLastChangeTime
```

26.3.3 查看 SNMP 用户

在特权用户模式下，执行 **show snmp user** 来查看当前代理上配置的 SNMP 用户。

```
Ruijie# show snmp user
```

```
User name: test
Engine ID: 800013110300000000000
storage-type: permanent    active
```

```
Security level: auth priv
Auth protocol: SHA
Priv protocol: DES
Group-name: g1
```

26.3.4 查看 SNMP 视图和组

在特权用户模式下，执行 **show snmp group** 来查看当前代理上配置的组。

```
Ruijie# show snmp group
groupname: g1
securityModel: v3
securityLevel:authPriv
readview: default
writeview: default
notifyview:
groupname: public
securityModel: v1
securityLevel:noAuthNoPriv
readview: default
writeview: default
notifyview:
groupname: public
securityModel: v2c
securityLevel:noAuthNoPriv
readview: default
writeview: default
notifyview:
```

在特权用户模式下，执行 **show snmp view** 来查看当前代理上配置的视图。

```
Ruijie# show snmp view
default(include) 1.3.6.1
test-view(include) 1.3.6.1.2.1
```

26.4 SNMP 配置举例

26.4.1 典型配置实例

- 配置要求

如图，网络设备和网管工作站 NMS 通过以太网连接，NMS 的 IP 地址为 192.168.12.181，网络设备的 IP 地址为 192.168.12.1，在网管工作站上运行了网管软件（以 HP OpenView 为例）。

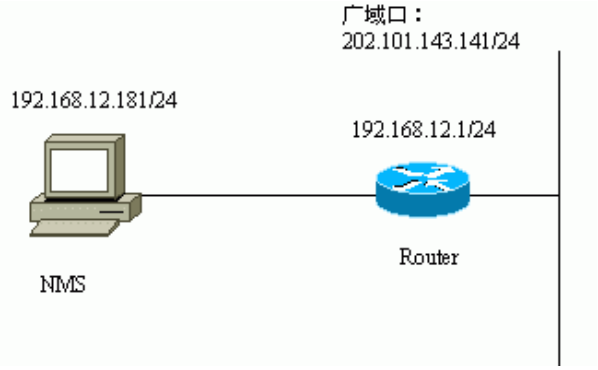


图 5　SNMP 典型配置组网图

- 网络设备具体配置

启动 SNMP 代理服务：

```
Ruijie(config)# snmp-server community public RO
```

只需要在全局配置模式下，配置以上指令，网络设备便启动了 SNMP 代理服务功能，这时 NMS 便可以对网络设备进行 SNMP 的监控了，不过只配置了只读权限，不能修改网络设备的配置，只能是监控网络设备。其他的配置都是可选的。

如果需要有读写的功能，可以采用如下的配置：

```
Ruijie(config)# snmp-server community private RW
```

以下是配置网络设备 SNMP 的一些代理基本参数，NMS 可以通过这些参数得知网络设备的一些基本系统信息，该配置为可选配置：

```
Ruijie(config)# snmp-server location fuzhou
Ruijie(config)# snmp-server contact wugb@i-net.com.cn
Ruijie(config)# snmp-server chassis-id 1234567890
0987654321
```

以下的配置，是允许网络设备主动向 NMS 发送一些 Trap 消息，该配置为可选配置：

```
Ruijie(config)# snmp-server enable traps
Ruijie(config)# snmp-server host 192.168.12.181 public
```

通过如上配置，网络设备的 SNMP 代理已经配置完毕，NMS 便可以对网络设备进行监控和管理了，以 HP OpenView 为例，可以产生网络拓扑结构图，如下图：

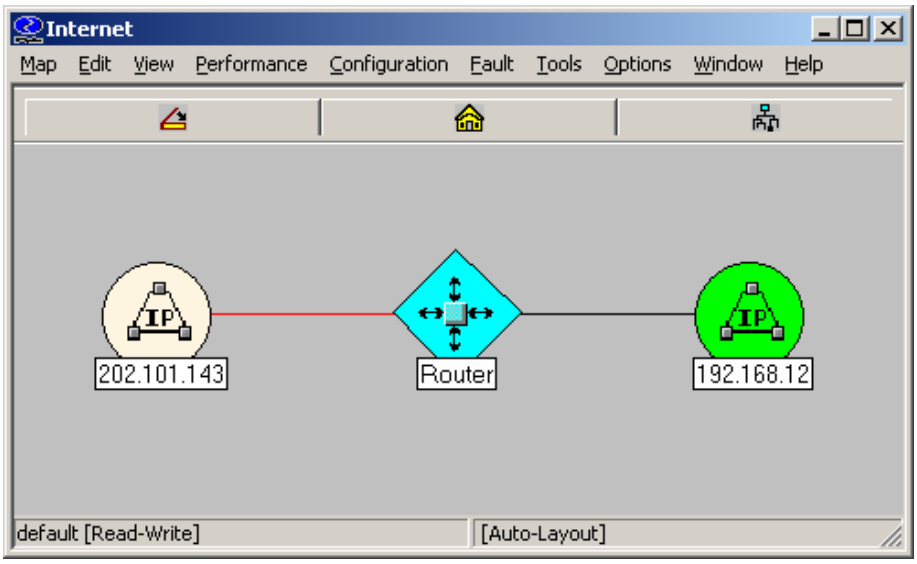


图 6　网络拓扑结构图

您可以对网络设备中的管理单元进行查询和设置，点击 HP OpenView 的 TOOL->SNMP MIB Brower 菜单，出现如下的对话框，在 Name 中输入 IP 地址 192.168.12.1，在 Community Name 中输入 Public，选择要查询的 MIB 的具体管理单元，比如下图的 System。点击 Start Query，便开始对网络设备进行 MIB 的查询了，具体的查询结果见对话框的 MIB Values 窗口：

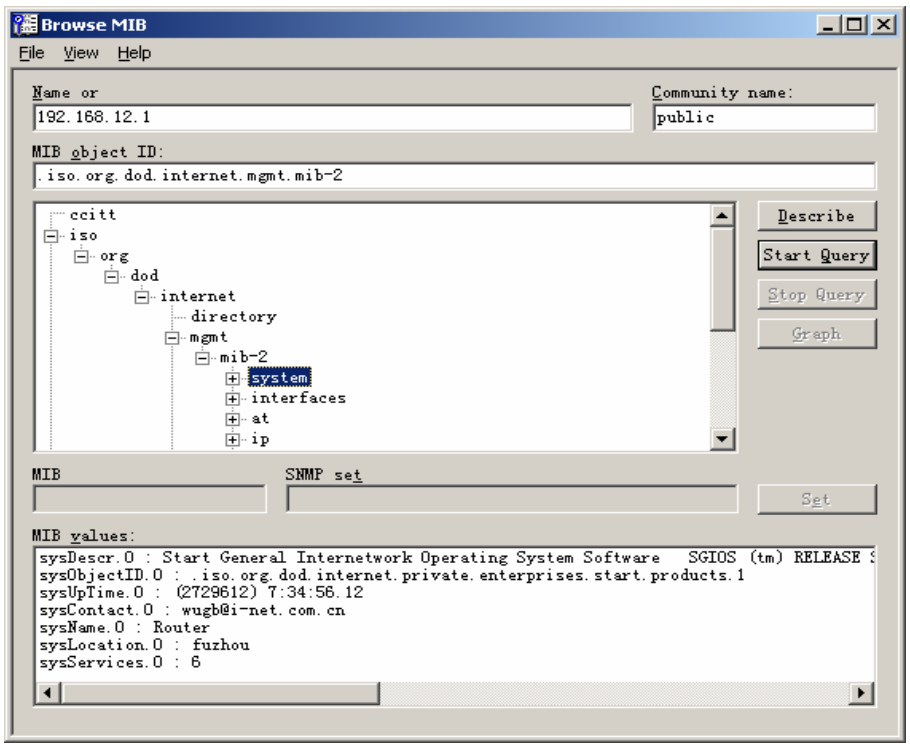


图 7　MIB 查询界面

HP OpenView 有很强大的网络管理的功能，比如，还可以用图表示出网络接口的流量统计图，其他的具体的各项 SNMP 的功能，见网管软件的文档，这里不在详述：

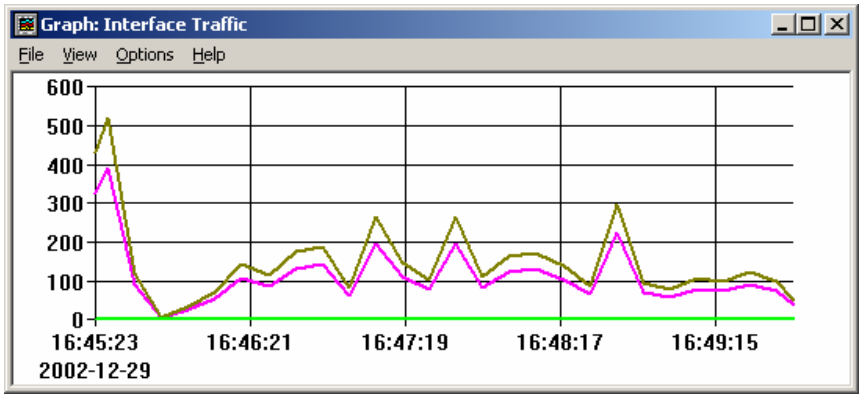


图 8　接口流量统计图

26.4.2 SNMP 访问列表关联控制实例

锐捷产品可以设置访问列表关联的方式，只要访问列表中允许的 NMS 才可以利用 SNMP 对 Agent 进行监控和管理，限制 NMS 对网络设备的访问，提高 SNMP 的安全性。

在全局配置模式下：

```
Ruijie(config)# access-list 1 permit 192.168.12.181
Ruijie(config)# snmp-server community public RO 1
```

通过如上配置，只有 IP 地址为 192.168.12.181 的主机才能利用 SNMP 对网络设备进行监控和管理了。

26.4.3 SNMPv3 相关配置实例

以下的配置允许 SNMPv3 的管理者采用认证＋加密模式通过用户名 v3user 对 MIB-2(1.3.6.1.2.1)节点下的管理变量进行设置和查看。采用的认证模式为 MD5，使用的认证密码为 MD5-Auth，采用 DES 加密，加密密钥为 DES-Priv。同时允许向 192.168.65.199 以 SNMPv3 格式发送 Trap。发送 Trap 使用的用户名为 v3user，采用认证＋加密模式发送，采用的认证模式为 MD5，使用的认证密码为 MD5-Auth，采用 DES 加密，加密密钥为 DES-Priv。

```
Ruijie(config)# snmp-server view v3userview 1.3.6.1.2.1 include
Ruijie (config)# snmp-server group v3usergroup v3 priv read v3userview write v3userview
Ruijie (config)# snmp-server user v3user v3usergroup v3 auth md5 md5-auth priv des56 des-priv
Ruijie (config)# snmp-server host 192.168.65.199 traps version 3 priv v3user
```

27 RMON 配置

27.1 概述

RMON（Remote Monitoring，远程监视）是 IETF(Internet Engineering Task Force，Internet 工程专门小组)标准的监控规范，这个规范可以让各种网络监控器和控制台系统之间交换网络监控数据。RMON 在网络节点上放置探测器，网络管理平台决定这些探测器汇报哪些信息，如被监视的统计信息，收集历史信息所使用的时间段等等。例如交换机和路由器等网络设备，在网络上相当一个网络节点，通过 RMON 功能，可以监视当前所处节点位置的信息。

RMON 的发展经历了三个阶段，第一阶段是以太网远程监视；第二阶段增加了令牌环的功能，称为令牌环远程监视模块；第三阶段被称为 RMON2，从而使 RMON 功能发展到协议监视的更高层次。

第一阶段的 RMON（下称 RMON1）包含九个组，所有的组都是可选择性（而非强制性）的，但有些组的使用必须有其他组的支持。

交换机实现其中的第 1，2，3，9 组的内容：统计组、历史组、警告组、事件组。

27.1.1 统计组

统计组是 RMON 中的第 1 组，统计组统计被监控的每个子网的基本统计信息。目前只能对网络设备的以太网接口进行监控、统计。该组包含一个以太网统计表，统计的内容包括丢弃的数据包、广播数据包、CRC 错误、大小块、冲突等。

27.1.2 历史组

历史组(History)是 RMON 中的第 2 组，历史组定期地收集网络统计信息，并记录下来以便日后处理。它包含两个小组：

1. HistoryControl 组用来设置采样间隔时间、采样数据源等控制信息。

2. EthernetHistory 组为管理员提供有关网段流量、错误包、广播包、利用率以及碰撞次数等其他统计信息的历史数据。

27.1.3 警告组

警报组(Alarm)是 RMON 中的第 3 组，以指定的时间间隔监控一个特定的 MIB(Management Information Base，管理信息库)对象，当这个 MIB 对象的值超过一个设定的上限值或低于一个设定的下限值时，会触发警报。警报被当作事件来处理，处理事件的方式可以是记录日志或发送 SNMP Trap 的方式。

27.1.4 事件组

事件组(Event)是 RMON 中的第 9 组，决定由于警报而产生事件时，处理行为是产生一个日志记录表项还是一个 SNMP Trap。

27.2 RMON 配置任务列表

27.2.1 配置统计组

您可以使用如下命令添加一个统计表项。

命令	作用
Ruijie(config-if)# **rmon collection stats** *index* [**owner** *ownername*]	添加一个统计项
Ruijie(config-if)# **no rmon collection stats** *index*	删除一个统计项

注意：

锐捷产品当前版本只支持以太网接口的统计。索引值应该是一个 1-65535 之间的整数，目前可最多同时配置 100 条统计项。

27.2.2 配置历史控制组

你可以使用如下命令添加一条历史控制表项：

命令	作用
Ruijie(config-if)# **rmon collection history** *index* [**owner** *ownername*] [**buckets** *bucket-number*] [**interval** *seconds*]	添加一个历史控制表项
Ruijie(config-if)# **no rmon collection history** *index*	删除一个历史控制表项

注意：

锐捷产品当前版本只支持以太网的纪录。索引值应该在 1-65535 之间，最多可以配置 10 条控制表项。

Bucket-number：控制项指定了采用的数据源、时间间隔。每个采样区间，都进行一次采样。采样的结果保存下来，Bucket-number 指定了保存采样的最大数目，当采样纪录达到最大值时，则新纪录覆盖最早的纪录。Bucket-number 取值范围是 1-65535，默认值是 10。

Interval：采样的时间间隔。默认值是 1800 秒，取值在 1-3600 之间。

27.2.3 配置警告组和事件组

你可以使用如下命令配置警告表：

命令	作用
Ruijie(config)# **rmon alarm** *number variable interval* {**absolute** | **delta**} **rising-threshold** *value* [*event-number*] **falling-threshold** *value* [*event-number*] [**owner** *ownername*]	添加一个历史控制表项
Ruijie(config)# **rmon event** *number* [**log**] [**trap** *community*] [**description** *description-string*]	添加一个事件组表项
Ruijie(config)# **no rmon alarm** *number*	删除一个警告组
Ruijie(config)# **no rmon event** *number*	删除一个事件组

number：警告表(事件表)的索引，范围 1-65535。

variable：警告表监控的变量。变量必须是整数类型。

interval：采样的时间间隔。范围<1-4294967295>

关键字 Absolute 表示拿每次采样得到的值和上限、下限比较，关键字 Delta 表示利用和上次采样的差值和上限、下限比较。

value 定义了上限、下限的值。

event-number：当超过了上限或者下限的时候，触发事件组索引为 Event-number 的事件。

关键字 Log 表示事件触发的动作是：纪录事件

关键字 Trap 表示事件触发后的动作是：发送 Trap 消息到管理站。

community：发送 Trap 时的认证名。

description-string：事件的描述

27.2.4 显示 RMON 状态

命令	作用
Ruijie(config)**# show rmon alarm**	显示警告组
Ruijie(config)**# show rmon event**	显示事件组
Ruijie(config)**# show rmon history**	显示历史组
Ruijie(config)**# show rmon statistics**	显示统计组

27.3 RMON 配置实例

27.3.1 配置统计组实例

如果您希望统计以太端口 3，使用如下命令：

```
Ruijie(config)# interface gigabitEthernet 0/3
Ruijie(config-if)# rmon collection stats 1 owner zhangsan
```

27.3.2 配置历史组实例

如果您希望每隔 10 分钟统计以太网端口 3 的历史信息，使用如下命令：

```
Ruijie(config)# interface gigabitEthernet 0/3
Ruijie(config-if)# rmon collection history 1 owner zhangsan
interval 600
```

27.3.3 配置警告组和事件组实例

如果您希望配置对一个可统计的 MIB 变量的报警功能。下面的例子说明了对 MIB-II 中 IfEntry 表中实例 ifInNUcastPkts.6(端口 6 上收到的非单播帧的个数，实例的标识符为 1.3.6.1.2.1.2.2.1.12.6)设置报警功能。具体功能为：交换机每隔 30 秒检查端口 6 上收到的非单播帧的个数的变化，如果收到的非单播帧的个数比上次检查时(30 秒前)增加了 20 个或 20 个以上，或者比上次只增加 10 个或 10 以下，则警报被触发，同时警报将触发事件 1 进行相应的操作(记录到日志中，并发送认证名为”rmon”的 Trap，事件的描述为“ifInNUcastPkts is too much”)。警报和事件表项的拥有者均为 zhangsan。

```
Ruijie(config)# rmon alarm 10 1.3.6.1.2.1.2.2.1.12.6 30 delta
rising-threshold 20 1 falling-threshold 10 1 owner zhangsan

Ruijie(config)# rmon event 1 log trap rmon description "ifIn
NUcastPkts is too much " owner zhangsan
```

27.3.4 显示 rmon 状态实例

27.3.4.1 show rmon alarm

```
Ruijie# show rmon alarm
Alarm : 1
Interval : 1
Variable : 1.3.6.1.2.1.4.2.0
Sample type : absolute
Last value : 64
Startup alarm : 3
Rising threshold : 10
Falling threshold : 22
Rising event : 0
Falling event : 0
Owner : zhangsan
```

27.3.4.2 show rmon event

```
Ruijie# show rmon event
Event : 1
Description : firstevent
Event type : log-and-trap
Community : public
Last time sent : 0d:0h:0m:0s
Owner : zhangsan
Log : 1
Log time : 0d:0h:37m:47s
Log description : ipttl
Log : 2
Log time : 0d:0h:38m:56s
Log description : ipttl
```

27.3.4.3 show rmon history

```
Ruijie# show rmon history
Entry : 1
Data source : Gi1/1
Buckets requested : 65535
Buckets granted : 10
Interval : 1
Owner : zhangsan
```

```
Sample : 198
Interval start : 0d:0h:15m:0s
DropEvents : 0
Octets : 67988
Pkts : 726
BroadcastPkts : 502
MulticastPkts : 189
CRCAlignErrors : 0
UndersizePkts : 0
OversizePkts : 0
Fragments : 0
Jabbers : 0
Collisions : 0
Utilization : 0
```

27.3.4.4 show rmon statistics

```
Ruijie#  show rmon statistics
Statistics : 1
Data source : Gi1/1
DropEvents : 0
Octets : 1884085
Pkts : 3096
BroadcastPkts : 161
MulticastPkts : 97
CRCAlignErrors : 0
UndersizePkts : 0
OversizePkts : 1200
Fragments : 0
Jabbers : 0
Collisions : 0
Pkts64Octets : 128
Pkts65to127Octets : 336
Pkts128to255Octets : 229
Pkts256to511Octets : 3
Pkts512to1023Octets : 0
Pkts1024to1518Octets : 1200
Owner : zhangsan
```

注意:

在 S3760 系列交换机中，**show rmon statistics** 命令中关于报文长度的统计，报文包括接收到的报文和发送的报文。

28 VRF 配置

28.1 VRF 概述

VRF(Virtual Routing Forwarding)，可以通俗的理解为一个独立的转发表；一台设备可以配置多个 VRF，则该设备同时维护多个转发表。各个转发表之间相互独立，互不影响，从而达到不同用户之间数据的隔离，一台设备充当多台设备使用的效果。

28.2 VRF 配置任务

- 创建 VRF
- 接口使能 VRF 功能
- 配置 VRF 路由

28.2.1 创建 VRF

命令	作用
Ruijie(config)# **ip vrf** *vrf-name*	创建 VRF
Ruijie(config)# **no ip vrf** *vrf-name*	删除 VRF

vrf-name 不要超过 64 个字符

28.2.2 接口使能 VRF 功能

命令	作用
Ruijie(config-if)# **ip vrf forwarding** *vrf-name*	接口使能 VRF
Ruijie(config-if)# **no ip vrf forwarding** *vrf-name*	接口关闭 VRF

默认情况下，接口不属于任何 VRF，即属于全局路由

注意：

接口使能 VRF 之后，原接口上的地址配置将失效；通常在接口上先使能 VRF，再配置 IP 地址

28.2.3 配置 VRP 路由

命令	作用
Ruijie(config)**# ip route vrf** *vrf-name* **network** *mask* **interface nexthop**	添加路由
Ruijie(config)**# no ip route vrf** *vrf-name* **network** *mask*	删除路由

28.3 VRF 调试

查看 VRF 中的路由表，使用如下命令

命令	作用
Ruijie**# show ip route vrf** *vrf-name*	显示指定 VRF 中的路由

命令详细语法请参考《配置 VRF 命令》

清除 VRF 中的路由表，使用如下命令

命令	作用
Ruijie**# clear ip route vrf** *vrf-name* *	清除指定 VRF 中的路由

命令详细语法请参考《配置 VRF 命令》

查看系统中的 VRF，使用如下命令

命令	作用
Ruijie**# show ip vrf** *vrf-name*	显示指定 VRF 信息

命令详细语法请参考《配置 VRF 命令》

29 RIP 路由协议配置

29.1 RIP 简介

RIP (Routing Information Protocol) 路由协议是一种相对古老，在小型以及同介质网络中得到了广泛应用的路由协议。RIP 采用距离向量算法，是一种距离向量协议。RIPv1 在 RFC 1058 文档中定义，RIPv2 在 RFC 2453 文档中定义，我司 RGOS 软件同时支持这两个版本。

RIP 使用 UDP 报文交换路由信息，UDP 端口号为 520。通常情况下 RIPv1 报文为广播报文；而 RIPv2 报文为组播报文，组播地址为 224.0.0.9。RIP 每隔 30 秒向外发送一次更新报文。如果设备经过 180 秒没有收到来自对端的路由更新报文则将所有来自此设备的路由信息标志为不可达，路由进入不可达状态后，120 秒内仍未收到更新报文就将这些路由从路由表中删除。

RIP 使用跳数来衡量到达目的地的距离，称为路由量度。在 RIP 中，设备到与它直接相连网络的跳数为 0；通过一个设备可达的网络的跳数为 1 ，其余依此类推；不可达网络的跳数为 16。

运行 RIP 路由协议的设备，可以从邻居学到缺省路由，也可以自己产生缺省路由。当满足以下条件之一，我司产品就可以通过 **default-information originate** 命令引入缺省路由，并通告给邻居设备：

- 配置了 **ip default-network**
- 其它路由协议学到的缺省路由或配置了静态缺省路由

RIP 将向指定的网络接口发送更新报文，如果接口的网络没有与 RIP 路由进程关联，该接口就不会通告任何更新报文。RIP 有 RIPv1 和 RIPv2 两个版本，RIPv2 支持明文认证、MD5 密文认证和支持可变长子网掩码。

我司 RIP 采用水平分割（Split Horizon）等手段防止形成环路路由。

29.2 RIP 配置任务列表

RIP 配置任务列表如下：

- 创建 RIP 路由进程（必须）
- RIP 报文单播配置（可选）
- 水平分割配置（可选）
- RIP 版本定义（可选）
- 配置路由汇聚功能（可选）
- RIP 时钟调整（可选）
- RIP 路由源地址验证配置（可选）

- RIP 接口状态控制（可选）
- RIP 接口通告缺省路由（可选）
- RIP VRF 配置 (可选)

关于以下主题的配置，请参见《IP 路由"协议无关"特性配置》章节。

- 路由信息过滤
- 路由重分布
- 缺省路由分发配置

29.2.1 创建 RIP 路由进程

设备要运行 RIP 路由协议，首先需要创建 RIP 路由进程，并定义与 RIP 路由进程关联的网络。

要创建 RIP 路由进程，在全局配置模式中执行以下命令：

命令	作用
Ruijie(config)# **router rip**	创建 RIP 路由进程
Ruijie(config-router)# **network** *network-number wildcard*	定义关联网络

用户可以同时配置 *network-number* 和 *wildcard* 参数，使得在该地址范围内的接口地址网段参与 RIP 的运行。

如果未配置 *wildcard* 参数，RGOS 将默认按照有类地址范围来处理，使得在该有类地址范围内的接口地址网段参与 RIP 运行。

说明：

network 命令定义的关联网络有两层意思：

1）RIP 对外通告关联网络的路由信息；

2）RIP 只通过关联网络的接口通告和接收路由更新信息。

29.2.2 RIP 报文单播配置

RIP 通常为广播或组播协议，如果 RIP 路由信息需要通过非广播网传输，则需要配置设备，以便支持 RIP 利用单播通告路由信息更新报文。

要配置 RIP 报文更新单播通告，在 RIP 路由进程配置模式中执行以下命令：

命令	作用
Ruijie(conf-router)# **neighbor** *ip-address*	配置 RIP 报文单播通告

配合该命令的使用，您还可以控制接口是否允许通告 RIP 路由更新报文，限制一个接口通告广播式的路由更新报文，需要在路由进程配置模式中配置 **passive-interface** 命令。关于路由信息通告限制的相关描述，请参见《协议无关配置》中的“路由过滤”章节。

说明：

在配置 FR、X.25 时，如果地址映射时指定了 Broadcast 关键字，则无需配置 **neighbor**。**Neighbor** 命令的作用更多体现在减少广播报文和路由过滤上。

29.2.3 水平分割配置

多台设备连接在 IP 广播类型网络上，又运行距离向量路由协议时，就有必要采用水平分割的机制以避免路由环路的形成。水平分割可以防止设备将某些路由信息从学习到这些路由信息的接口通告出去，这种行为优化了多个设备之间的路由信息交换。

然而对于非广播多路访问网络（如帧中继、X.25 网络），水平分割可能造成部分设备学习不到全部的路由信息。在这种情况下，可能需要关闭水平分割。如果一个接口配置了次 IP 地址，也需要注意水平分割的问题。

要配置关闭或打开水平分割，在接口配置模式中执行以下命令：

命令	作用
Ruijie(config-if)# **no ip split-horizon**	关闭水平分割
Ruijie(config-if)# **ip split-horizon**	打开水平分割

所有接口的默认行为设置为开启水平分割。

29.2.4 定义 RIP 版本

我司产品支持 RIP 版本 1 和版本 2，RIPv2 可以支持认证、密钥管理、路由汇聚、CIDR 和 VLSMs。其中密钥管理、VLSMs 描述请参见《协议无关配置》。

缺省情况下，我司产品可以接收 RIPv1 和 RIPv2 的数据包，但是只发送 RIPv1 的数据包。您可以通过配置，只接收和发送 RIPv1 的数据包，也可以只接收和发送 RIPv2 的数据包。

要配置软件只接收和发送指定版本的数据包，在路由进程配置模式中执行以下命令：

命令	作用
Ruijie(config-router)# **version {1 | 2}**	定义 RIP 版本

以上命令使软件缺省情况下只接收和发送指定版本的数据包，如果需要可以更改每个接口的缺省行为。

要配置接口只发送哪个版本的数据包，在接口配置模式中执行以下命令：

命令	作用
Ruijie(config-if)# **ip rip send version 1**	指定只发送 RIPv1 数据包
Ruijie(config-if)# **ip rip send version 2**	指定只发送 RIPv2 数据包
Ruijie(config-if)# **ip rip send version 1 2**	指定只发送 RIPv1 和 RIPv2 数据包

要配置接口只接收哪个版本的数据包，在接口配置模式中执行以下命令：

命令	作用
Ruijie(config-if)# **ip rip receive version 1**	指定只接收 RIPv1 数据包
Ruijie(config-if)# **ip rip receive version 2**	指定只接收 RIPv2 数据包
Ruijie(config-if)# **ip rip receive version 1 2**	指定只接收 RIPv1 和 RIPv2 数据包

29.2.5 配置路由汇聚功能

RIP 路由自动汇聚，就是当子网路由穿越有类网络边界时，将自动汇聚成有类网络路由。RIPv2 缺省情况下将进行路由自动汇聚，RIPv1 不支持该功能。

RIPv2 路由自动汇聚的功能，提高了网络的伸缩性和有效性。如果有汇聚路由存在，在路由表中将看不到包含在汇聚路由内的子路由，这样可以大大缩小路由表的规模。

通告汇聚路由会比通告单独的每条路由更有效率，主要有以下因素：

- 当查找 RIP 数据库时，汇聚路由会得到优先处理；
- 当查找 RIP 数据库时，任何子路由将被忽略，减少了处理时间。

有时可能希望学到具体的子网路由，而不愿意只看到汇聚后的网络路由，这时需要关闭路由自动汇聚功能。

要配置路由自动汇聚，在 RIP 路由进程模式中执行以下命令：

命令	作用
Ruijie(config-router)# **no auto-summary**	关闭路由自动汇聚

Ruijie(config-router)# **auto-summary**	打开路由自动汇聚

还可以配置接口级汇聚，在某个接口下配置路由汇聚到指定的有类子网范围，在接口模式下执行以下命令：

命令	作用
Ruijie(config-if)# **ip summary-address rip** *ip-address ip-network-mask*	配置接口级路由汇聚
Ruijie(config-if)# **no ip summary-address rip** *ip-address ip-network-mask*	取消接口级路由汇聚

29.2.6 RIP 认证配置

RIPv1 不支持认证，如果设备配置 RIPv2 路由协议，可以在相应的接口配置认证。

我司产品 RIPv2 支持两种认证方式：明文认证和 MD5 认证。缺省的认证方式为明文认证。

如果要使用明文认证，用户可以直接使用 **ip rip authentication text-password** 方式配置明文认证字符串，也可以通过关联密钥串获取明文认证字符串，后者优先级高于前者。

如果要使用 MD5 认证，必须通过关联密钥串进行 MD5 认证。

如果采用明文认证，但未配置明文认证字符串，或者未配置关联密钥串，或者关联了密钥串但密钥串实际未配置，此时并不会有认证行为发生。同样，如果采用 MD5 认证，但未配置关联密钥串，或者关联了密钥串但密钥串实际未配置，也不会有认证行为发生。

要配置 RIP 认证，在接口配置模式中执行以下命令：

命令	作用
Ruijie(config-if)# **ip rip authentication mode {text | md5}**	配置接口 RIP 认证。 **text**：明文认证。 **md5**：MD5 认证。
Ruijie(config-if)# **ip rip authentication text-password** *password-string*	配置明文认证字符串，长度为 1～16 字节。
Ruijie(config-if)# **ip rip authentication key-chain** *key-chain-name*	配置使用密钥串进行认证。

29.2.7 RIP 时钟调整

RIP 提供了时钟调整的功能，您可以根据网络的具体情况进行时钟调整，使 RIP

路由协议能够运行得更好。可以对以下时钟进行调整：

路由更新时间：以秒计，定义了设备发送路由更新报文的周期；

路由无效时间：以秒计，定义了路由表中路由因没有更新而变为无效的时间；

路由清除时间：以秒计，该时间过后，该路由将被清除出路由表；

通过调整以上时钟，可能会加快路由协议的收敛时间以及故障恢复时间。要调整 RIP 时钟，在 RIP 路由进程配置模式中执行以下命令：

命令	作用
Ruijie(config-router)# **timers basic** *update invalid flush*	调整 RIP 时钟

缺省情况下，更新时间为 30 秒，无效时间为 180 秒，清除时间为 120 秒。

说明：

连接在同一网络上的设备，RIP 时钟值一定要一致。

29.2.8 RIP 路由源地址验证配置

缺省情况下，RIP 会对接收的路由更新报文源地址进行验证，如果源地址无效，RIP 就会丢弃该报文。判断源地址是否有效，就是判断源 IP 地址是否与接口 IP 地址在相同的网络内。如果是无编号 IP 地址接口，将不会进行有效性验证。

要配置路由源地址验证，在 RIP 路由进程配置模式中执行以下命令：

命令	作用
Ruijie(config-router)# **no validate-update-source**	关闭源地址验证
Ruijie(config-router)# **validate-update-source**	启动源地址验证

29.2.9 RIP 接口状态控制

在某些情况下，需要对 RIP 工作进行灵活配置，如仅希望设备学习 RIP 路由，并不进行 RIP 路由通告，此时可以配置被动接口；或希望单独配置某个接口的状态，则可以使用命令对指定接口的 RIP 报文的发送或接收进行控制。

要配置某接口为被动模式，在 RIP 路由进程配置模式中执行以下命令：

命令	作用
Ruijie(config-router)# **passive-interface** {**default** | *interface-type interface-num*}	配置被动接口

Ruijie(config-router)**#no passive-interface** {**default** | *interface-type interface-num*}	取消被动接口

说明：

被动接口接收到 RIP 请求后，不进行响应；但在收到非 RIP（如路由诊断程序等）请求后，会进行响应，因为这些请求程序希望了解所有设备的路由情况。

要禁止或允许某接口接收 RIP 报文，在接口配置模式中执行以下命令

命令	作用
Ruijie(config-if)**# no ip rip receive enable**	禁止接口接收 RIP 报文
Ruijie(config-if)**# ip rip receive enable**	允许接口接收 RIP 报文

要禁止或允许某接口接收 RIP 报文，在接口配置模式中执行以下命令

命令	作用
Ruijie(config-if)**# no ip rip send enable**	禁止接口发送 RIP 报文
Ruijie(config-if)**# ip rip send enable**	允许接口发送 RIP 报文

29.2.10 RIP 接口通告缺省路由

如果需要在某个指定接口的更新报文中产生一条默认路由（0.0.0.0/0）,可以在接口配置模式下执行以下命令：

命令	作用
Ruijie(config-if)**# ip rip default-information originate** [**metric** *metric-value*]	通告默认路由，也通告其它路由
Ruijie(config-if)**# no ip rip default-information**	取消接口通告默认路由

如果需要在某个指定接口的更新报文中产生一条默认路由（0.0.0.0/0）,并且该接口只通告这条默认路由，不通告其它 RIP 路由，可以在接口配置模式下执行以下命令：

命令	作用
Ruijie(config-if)**# ip rip default-information only** [**metric** *metric-value*]	通告默认路由，不通告其它路由
Ruijie(config-if)**# no ip rip default-information**	取消接口通告默认路由

说明：

如果同时配置了接口下的 **ip rip default-information** 和 RIP 进程的 **default-information originate**，则只通告接口配置的缺省路由。

29.2.11 RIP VRF 配置

RIP 支持 VRF，在 RIP 进程中可创建多个 RIP 实例，分别管理对应的 VRF。缺省情况下 RIP 进程只有一个实例，用来管理全局路由表。当 VRF 创建后，如需使用 RIP 管理 VRF 路由表，可以通过创建新的 RIP 实例管理该 VRF 路由表。

使用 **address-family** 命令使路由设备进入地址族配置子模式（提示为：(config-router-af)#）。当第一次指定子模式关联的 VRF 时，RIP 会创建对应该 VRF 的 RIP 实例。在该子模式下，可以配置相应的 VRF 的 RIP 实例。该模式下的 RIP 配置方法与全局路由中的 RIP 配置方法完全相同。

要离开地址族配置子模式并返回路由配置模式，使用 **exit-address-family** 或者 **exit** 命令。

要配置管理 VRF 的 RIP 实例，在 RIP 路由进程配置模式中执行以下命令：

命令	作用
Ruijie(config-router)# **address-family ipv4 vrf** *vrf-name*	创建管理与名为 *vrf-name* 对应的 VRF 的 RIP 实例
Ruijie(config-router)# **no address-family ipv4 vrf** *vrf-name*	删除与名为 *vrf-name* 的 VRF 对应的 RIP 实例

29.3 RIP 配置范例

本章提供了 4 个 RIP 配置范例：

- RIP 水平分割配置例子
- RIP 认证配置例子
- RIP 报文单播配置例子
- RIP VRF 配置例子

29.3.1 RIP 水平分割配置例子

- **配置要求**

有五台设备，其中 RouterA、RouterD、RouterE 三台设备通过以太网连接，RouterA、

RouterB、RouterC 三台设备通过帧中继连接。IP 地址分配及设备连接图见图 1，其中 RouterD 配置了次地址。

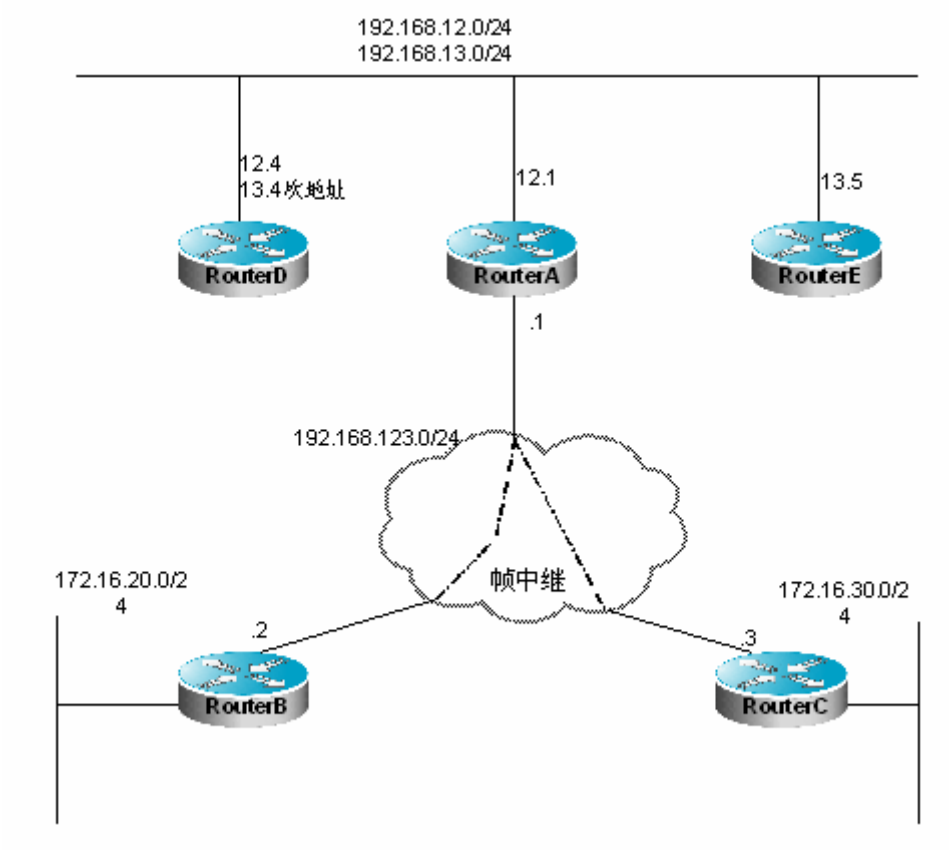


图 1　RIP 水平分割配置例子

通过配置设备需要达到以下要求：

1） 所有设备运行 RIP 路由协议；

2） RouterB、RouterC 能够相互学到对方通告的网段路由；3）RouterE 能够学到 192.168.12.0/24 的网段路由。

- **设备具体配置**

在该范例配置中，为了达到配置要求，RouterA 和 RouterD 一定要关闭水平分割功能。否则 RouterA 将不会将 RouterB 通告的路由通告给 RouterC，RouterD 也不会将 192.168.12.0 网段通告给 RouterE。以下为每台设备的具体配置。

设备 A 的配置：

```
# 配置以太网端口
interface FastEthernet0/0
ip address 192.168.12.1 255.255.255.0

# 配置广域网端口
interface Serial1/0
ip address 192.168.123.1 255.255.255.0
encapsulation frame-relay
```

```
no ip split-horizon

# 配置 RIP 路由协议
router rip
version 2
network 192.168.12.0
network 192.168.123.0
```

设备 B 的配置：

```
# 配置以太网端口
interface FastEthernet0/0
ip address 172.16.20.1 255.255.255.0

# 配置广域网端口
interface Serial1/0
ip address 192.168.123.2 255.255.255.0
encapsulation frame-relay

# 配置 RIP 路由协议
router rip
version 2
network 172.16.0.0
network 192.168.123.0
no auto-summary
```

设备 C 的配置

```
# 配置以太网端口
interface FastEthernet0/0
ip address 172.16.30.1 255.255.255.0

# 配置广域网端口
interface Serial1/0
ip address 192.168.123.3 255.255.255.0
encapsulation frame-relay

# 配置 RIP 路由协议
router rip
version 2
network 172.16.0.0
network 192.168.123.0
no auto-summary
```

设备 D 的配置

```
# 配置以太网端口
interface FastEthernet0/0
ip address 192.168.12.4 255.255.255.0
ip address 192.168.13.4 255.255.255.0 secondary
no ip split-horizon
```

```
# 配置 RIP 路由协议
router rip
version 2
network 192.168.12.0
network 192.168.13.0
```

设备 E 的配置

```
# 配置以太网端口
interface FastEthernet0/0
ip address 192.168.13.5 255.255.255.0

# 配置 RIP 路由协议
router rip
version 2
network 192.168.13.0
```

29.3.2 RIP 认证配置例子

- **配置要求**

两台设备通过以太网互连，运行 RIP 路由协议，采用 MD5 认证方式。设备之间连接图以及 IP 地址分配见 图 2。

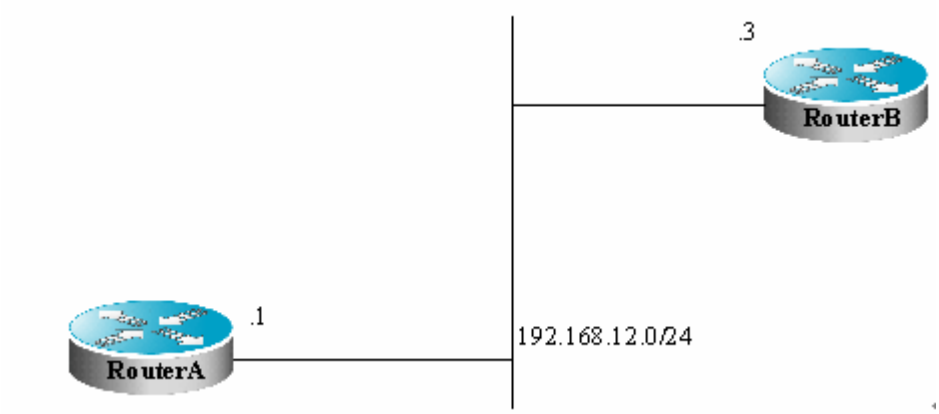


图 2　RIP 认证配置例子

要求 Router A 发送 RIP 包的认证密钥为 Keya，可接收认证密钥为 Keya、Keyb 的 RIP 包；Router B 发送 RIP 包的认证密钥为 Keyb，可接收认证密钥为 Keya、Keyb 的 RIP 包。

- **设备具体配置**

设备 A 的配置：

```
# 密钥串配置
key chain ripkey
key 1
key-string keya
```

```
accept-lifetime 00:00:00 Dec 3 2000 infinite
send-lifetime 00:00:00 Dec 4 2000 infinite
key 2
key-string keyb
accept-lifetime 00:00:00 Dec 3 2000 infinite
send-lifetime 00:00:00 Dec 4 2000 infinite

# 配置以太网端口
interface FastEthernet0/0
ip address 192.168.12.1 255.255.255.0
ip rip authentication mode md5
ip rip authentication key-chain ripkey

# 配置 RIP 路由协议
router rip
version 2
network 192.168.12.0
```

设备 B 的配置:

```
# 密钥串配置
key chain ripkey
key 1
key-string keyb
accept-lifetime 00:00:00 Dec 3 2000 infinite
send-lifetime 00:00:00 Dec 4 2000 00:00:00 Dec 5 2000
key 2
key-string keya
accept-lifetime 00:00:00 Dec 3 2000 infinite
send-lifetime 00:00:00 Dec 4 2000 infinite

# 配置以太网端口
interface FastEthernet0/0
ip address 192.168.12.2 255.255.255.0
ip rip authentication mode md5
ip rip authentication key-chain ripkey

# 配置 RIP 路由协议
router rip
version 2
network 192.168.12.0
```

29.3.3 RIP 报文单播配置例子

- **配置要求**

三台设备全部连接在局域网上，并且全部运行 RIP 路由协议。设备 IP 地址分配和设备连接图见图 3。

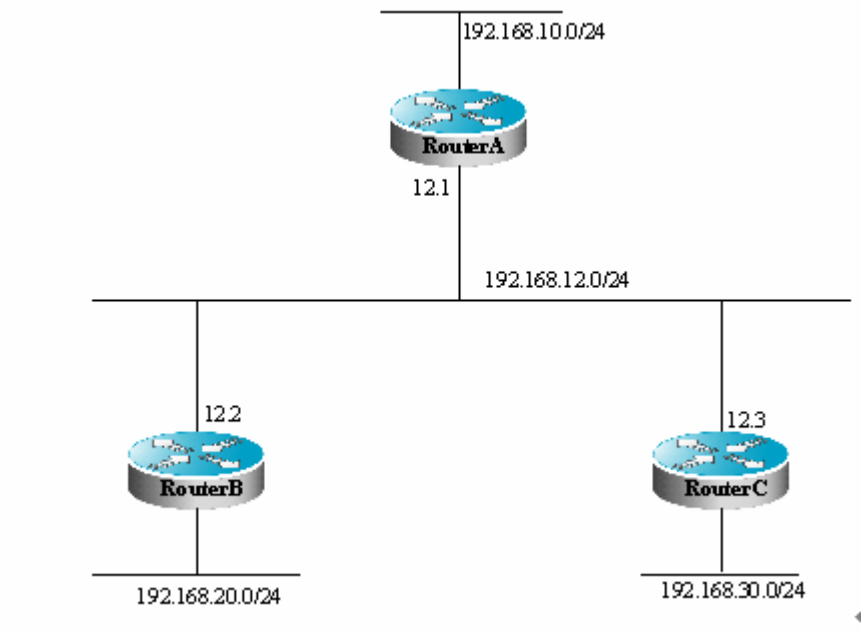


图 3　RIP 报文单播配置例子

通过 RIP 报文单播配置，要求实现以下目标：

1） Router A 可以学到 Router C 通告的路由；

2） Router C 学不到 Router A 通告的路由。

- **设备具体配置**

为了实现以上配置要求，需要在设备 A 配置 RIP 报文单播。

设备 A 的配置：

```
# 配置以太网端口
interface FastEthernet0/0
ip address 192.168.12.1 255.255.255.0

# 配置回环端口
interface Loopback0
ip address 192.168.10.1 255.255.255.0

# 配置 RIP 路由协议
router rip
version 2
network 192.168.12.0
network 192.168.10.0
passive-interface FastEthernet0/0
neighbor 192.168.12.2
```

设备 B 的配置：

```
# 配置以太网端口
interface FastEthernet0/0
```

```
ip address 192.168.12.2 255.255.255.0

# 配置回环端口
interface Loopback0
ip address 192.168.20.1 255.255.255.0

# 配置 RIP 路由协议
router rip
version 2
network 192.168.12.0
network 192.168.20.0
```

设备 C 的配置：

```
# 配置以太网端口
interface FastEthernet0/0
ip address 192.168.12.3 255.255.255.0

# 配置回环端口
interface Loopback0
ip address 192.168.30.1 255.255.255.0

# 配置 RIP 路由协议
router rip
version 2
network 192.168.12.0
network 192.168.30.0
```

29.3.4 RIP VRF 配置例子

- **配置要求**

两台设备通过以太网互连，运行 RIP 路由协议。设备之间连接图以及 IP 地址分配见 图 4。

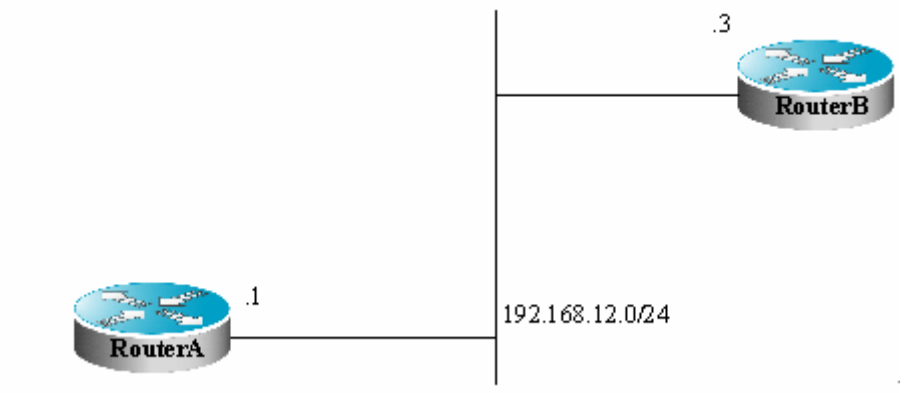


图 4　RIP VRF 配置例子

通过 RIP，在 Router A 的名为 redvpn 的 VRF 和 Router B 的名为 bluevpn 的 VRF

之间交互路由信息

- **设备具体配置**

设备 A 的配置：

```
# 创建 VRF
ip vrf redvpn

# 接口绑定 VRF，并配置端口地址。
interface FastEthernet 1/0
ip vrf forwarding redvpn
ip address 192.168.12.1 255.255.255.0

# 配置 RIP 路由协议，创建 RIP 实例
router rip
address-family ipv4 vrf redvpn
network 192.168.12.0
exit-address-family
```

设备 B 的配置:

```
# 创建 VRF
ip vrf  bluevpn

# 接口绑定 VRF，并配置端口地址。
interface FastEthernet 1/0
ip vrf forwarding bluevpn
ip address 192.168.12.3 255.255.255.0

# 配置 RIP 路由协议，创建 RIP 实例
router rip
address-family ipv4 vrf bluevpn
network 192.168.12.0
exit-address-family
```

30 OSPF 路由协议配置

30.1 OSPF 简介

OSPF（Open Shortest Path First）为 IETF OSPF 工作组开发的一种基于链路状态的内部网关路由协议。OSPF 是专为 IP 开发的路由协议，直接运行在 IP 层上面，协议号为 89，采用组播方式进行 OSPF 包交换，组播地址为 224.0.0.5（全部 OSPF 设备）和 224.0.0.6（指定设备）。

链路状态算法是一种与哈夫曼向量算法（距离向量算法）完全不同的算法，应用哈夫曼向量算法的传统路由协议为 RIP，而 OSPF 路由协议是链路状态算法的典型实现。与 RIP 路由协议对比，OSPF 除了算法上的不同，还引入了路由更新认证、VLSMs(可变长子网掩码)、路由汇聚等新概念。即使 RIPv2 做了很大的改善，可以支持路由更新认证、可变长子网掩码等特性，但是 RIP 协议还是存在两个致命弱点：1）收敛速度慢；2）网络规模受限制，最大跳数不超过 16 跳。OSPF 的出现克服了 RIP 的弱点，使得 IGP 协议也可以胜任中大型、较复杂的网络环境。

OSPF 路由协议利用链路状态算法建立和计算到每个目标网络的最短路径，该算法本身较复杂，以下简单地、概括性地描述了链路状态算法工作的总体过程：

- 初始化阶段，设备将产生链路状态通告，该链路状态通告包含了该设备全部链路状态；

- 所有设备通过组播的方式交换链路状态信息，每台设备接收到链路状态更新报文时，将拷贝一份到本地数据库，然后再传播给其它设备；

- 当每台设备都有一份完整的链路状态数据库时，设备应用 Dijkstra 算法针对所有目标网络计算最短路径树，结果内容包括：目标网络、下一跳地址、花费，是 IP 路由表的关键部分。

如果没有链路花费、网络增删变化，OSPF 将会十分安静，如果网络发生了任何变化，OSPF 通过链路状态进行通告，但只通告变化的链路状态，变化涉及到的设备将重新运行 Dijkstra 算法，生成新的最短路径树。

一组运行 OSPF 路由协议的设备，组成了 OSPF 路由域的自治域系统。一个自治域系统是指由一个组织机构控制管理的所有设备，自治域系统内部只运行一种 IGP 路由协议，自治域系统之间通常采用 BGP 路由协议进行路由信息交换。不同的自治域系统可以选择相同的 IGP 路由协议，如果要连接到互联网，每个自治域系统都需要向相关组织申请自治域系统编号。

当 OSPF 路由域规模较大时，一般采用分层结构，即将 OSPF 路由域分割成几个区域（AREA），区域之间通过一个骨干区域互联，每个非骨干区域都需要直接与骨干区域连接。

在 OSPF 路由域中，根据设备的部署位置，有三种设备角色：

1) 区域内部设备，该设备的所有接口网络都属于一个区域；

2) 区域边界设备，也称为 ABR（Area Border Routers）,该设备的接口网络至少属于两个区域，其中一个必须为骨干区域；

3) 自治域边界设备，也称为 ASBR（Autonomous System Boundary Routers），是 OSPF 路由域与外部路由域进行路由交换的必经之路。

我司产品对 OSPF 的实现完全遵循了 RFC 2328 中定义的 OSPF v2，以下列出了我司产品实现的 OSPF 的主要特性：

1) 支持 OSPF 多进程，最多可以支持 64 个 OSPF 进程同时运行；

2) 支持 VRF；可以基于不同 VRF 运行 OSPF；

3) 残域——完全支持残域的定义；

4) 路由重分布——实现了与静态路由、直连路由以及 RIP、BGP 等动态路由协议之间的路由信息重分布；

5) 认证——支持邻居之间明文或者 MD5 的认证；

6) 虚拟链路——支持虚拟链路；

7) 可变长子网掩码 VLSMs 的支持；

8) 区域的划分；

9) NSSA（Not So Stubby Area），RFC 3101 定义；

10) Graceful Restart，RFC 3623 定义

30.2 OSPF 配置任务列表

OSPF 的配置需要在各设备（包括内部设备、区域边界设备和自治系统边界设备等）之间相互协作。在未作任何配置的情况下，设备的各参数使用缺省值，此时，发送和接收报文都无须进行验证，接口也不属于任何一个自治系统的分区。在改变缺省参数的过程中，请务必保证各设备之间的配置相互一致。

为了配置 OSPF，需要完成的工作如下所列。其中，创建 OSPF 路由进程是必需的，其他选项可选，但也可能为特定的应用所必需。以下为 OSPF 路由协议的配置步骤：

- 创建 OSPF 路由进程(必须）
- 配置 OSPF 接口参数（可选）
- 配置 OSPF 以适应不同物理网络（可选）
- 配置 OSPF 区域参数（可选）
- 配置 OSPF NSSA 区域（可选）
- 配置 OSPF 路由汇聚（可选）
- 创建虚拟链接（可选）
- 产生缺省路由（可选）
- 用 Loopback 地址做路由标识符（可选）

- 更改 OSPF 缺省管理距离（可选）
- 配置路由计算计时器（可选）
- 状态更新信息调整时间 （可选）
- 配置 OSPF 接口度量 （可选）
- 配置接口接收数据库描述报文时是否校验 MTU 值（可选）
- 配置禁止接口发送 OSPF 报文（可选）
- 配置 OSPF 网管功能（可选）
- 配置 OSPF BFD （可选）

关于以下主题的配置，请参见《协议无关配置》章节。

- 路由信息过滤
- 路由重分布

以下为 OSPF 的缺省配置：

网络接口	接口代价:不预设接口代价 LSA 重传间隔:5 秒 LSA 发送延迟:1 秒 hello 报文发送间隔:10 秒(对于非广播网络为 30 秒) 邻接设备失效的时间: hello 报文发送间隔的四倍 优先级：1 认证类型：0 (无认证) 认证密码：无
区间	认证类型：0(无认证) 进入 Stub 或 NSSA 区域的汇聚路由的缺省代价: 1 区间路由汇聚范围：未定义 存根区间(STUB): 未定义 不完全存根区间(NSSA): 未定义
虚拟连接	未定义虚拟连接。 有关虚拟连接参数的缺省值: LSA 重传间隔:5 秒 LSA 发送延迟:1 秒 hello 报文发送间隔:10 秒 邻接设备失效的时间: hello 报文发送间隔的四倍 认证类型：无认证 认证密码：无
自动代价计算	打开； 缺省的自动代价参考是 100 Mbps；
缺省路由产生	关闭； 如果打开则缺省使用的 metric 是 1，类型是 type-2
缺省 metric (Default metric)	重分发其他路由协议所使用的缺省 metric；

管理距离	区间内路由信息：110 区间间路由信息：110 外部路由信息：110
数据库过滤	关闭，所有接口都可以接收状态更新信息 LSA。
邻居变化日志记录	打开
邻居 (neighbor)	无
邻居数据库过滤	关闭，输出的 LSA 发送到所有邻居；
区间网络范围 (network area)	无
设备 ID	未定义,缺省 ospf 协议没有运行
外部路由汇聚 (summary-address)	未定义
状态更新信息调整时间	240 秒
最短路径优先算法定时器	收到拓扑改变信息到下一次开始调用 SPF 算法计算的延迟时间: 5 秒. 两次计算至少间隔的时间: 10 秒.
计算外部路由时采用的最优路径的规则	采用 RFC1583 中的规则
内存不足情况下 OSPF 行为	进入 OVERFLOW 状态 GR restarter：关闭 GR helper：打开
OSPFv2 GR 行为	GR helper：打开
OSPFv2 MIB 绑定	进程号最小的 OSPFv2 进程上
OSPFv2 TRAP 发送	关闭

30.2.1 创建 OSPF 路由进程

创建 OSPF 路由进程，并定义与该 OSPF 路由进程关联的 IP 地址范围，以及该范围 IP 地址所属的 OSPF 区域。OSPF 路由进程只在属于该 IP 地址范围的接口发送、接收 OSPF 报文,并且对外通告该接口的链路状态。目前我们支持 64 个 OSPF 路由进程。

要创建 OSPF 路由进程，可以按照如下步骤进行 ：

命令	含义
Ruijie **# configure terminal**	进入全局配置模式。
Ruijie (config)**# ip routing**	启用路由功能(如果为关闭的话)

Ruijie (config)# **router ospf** *process_id* [**vrf** *vrf-name*]	打开 OSPF，进入 OSPF 配置模式
Ruijie (config-router)# **network** *address wildcard-mask* **area** *area-id*	定义属于一个区间的地址范围；
Ruijie (config-router)# **end**	退回到特权模式。
Ruijie # **show ip protocols**	显示当前运行的路由协议。
Ruijie # **write**	保存配置。

说明：

我司产品可选参数 **vrf** vrf-name，用于指定 OSPF 所属 vrf，创建 OSPF 进程时若若未指定此 VRF 参数，则 OSPF 进程属于默认 VRF；**Network** 命令中 32 个“比特通配符”与掩码的取值相反，取“1”代表不比较该比特位，“0”代表比较该比特位。不过用掩码的方式来定义，我司产品也会自动翻译成比特通配符。只有接口地址与 network 命令定义的 IP 地址范围相匹配，该接口才属于指定的区域。当接口地址同时与多个 OSPF 进程 **network** 命令定义的 IP 地址范围相匹配时，按照最优匹配方式，确定接口参与的 OSPF 进程。

使用命令 **no router ospf** *process-id* 关闭 OSPF 协议。以下为打开 OSPF 协议的示例：

```
Ruijie(config)# router ospf 1
Ruijie(config-router)# network 192.168.0.0 0.0.0.255 area 0
Ruijie(config-router)# end
```

30.2.2 配置 OSPF 接口参数

OSPF 允许用户更改某些特定的接口参数，用户可以根据实际应用的需要设置这些参数。应该注意的是，一些参数的设置必须保证与该接口相邻接的设备的相应参数一致，这些参数通过 ip ospf hello-interval, ip ospf dead-interval, ip ospf authentication，ip ospf authentication-key 和 ip ospf message-digest-key 五个接口参数进行设置，当使用这些命令时应该注意邻居设备也有同样的配置。

要配置 OSPF 接口参数，在接口配置模式中进行执行以下命令：

命令	含义
Ruijie # **configure terminal**	进入全局配置模式。
Ruijie (config)# **ip routing**	启用路由功能(如果为关闭的话)
Ruijie (config)# **interface** *interface-id*	进入接口配置模式。

Ruijie (config-if)# **ip ospf cost** *cost-value*	（可选）定义接口费用，
Ruijie (config-if)# **ip ospf retransmit-interval** *seconds*	（可选）设置链路状态重传间隔；
Ruijie (config-if)# **ip ospf transmit-delay** *seconds*	（可选）设置链路状态更新报文传输过程的估计时间；
Ruijie (config-if)# **ip ospf hello-interval** *seconds*	（可选）设置 hello 报文发送间隔，对于整个网络的节点，该值要相同；
Ruijie (config-if)# **ip ospf dead-interval** *seconds*	（可选）设置相邻设备失效间隔，对于整个网络的节点，该值必须相同；
Ruijie (config-if)# **ip ospf priority** *number*	（可选）优先级，用于选举指派设备（DR）和备份指派设备（BDR）
Ruijie (config-if)# **ip ospf authentication** [**message-digest** | **null**]	（可选）设置接口的认证方式
Ruijie (config-if)# **ip ospf authentication-key** *key*	（可选）配置接口文本认证的密码，
Ruijie (config-if)# **ip ospf message-digest-key** *keyid* **md5** *key*	（可选）配置接口的 MD5 加密认证的密码
Ruijie (config-if)# **ip ospf database-filter all out**	（可选）阻止接口泛洪链路状态更新报文；缺省情况下，OSPF 将接收的 LSA 信息从属于同一区间的所有接口上泛洪出去，除了接收该 LSA 信息的接口；
Ruijie (config-if)# **end**	退回到特权模式。
Ruijie # **show ip ospf** [*process-id*] **interface** [*interface-id*]	显示当前运行的路由协议。
Ruijie # **write**	（可选）保存配置。

使用以上命令的 no 模式，可以取消原来的设置或者恢复缺省值。

30.2.3 配置 OSPF 以适应不同物理网络

根据不同的媒介的传输性质，OSPF 将网络分为三种类型：

- 广播网络（以太网，令牌环，FDDI）
- 非广播网络（帧中继，X.25）
- 点到点网络（HDLC，PPP，SLIP）

其中非广播网络，根据 OSPF 的操作模式不同又分为两种子类型：

1. 非广播多路访问网络(NBMA)类型。NBMA 要求所有互联的设备必须能够直接

通讯，只有全网状的连接才能达到该要求，如果采用 SVC（比如 X.25）连接没问题，但是采用 PVC（如帧中继）组网将有一定难度。OSPF 在 NBMA 网络上操作与广播网络上类似，需要选举指定设备（Designated Router），并由指定设备通告 NBMA 网络的链路状态。

2. 点到多点网络类型。如果网络拓扑结构不是全网状的非广播网络，OSPF 需要将该接口网络类型设置成点到多点网络类型。在点到多点网络类型中，OSPF 将所有的设备之间的连接看作点到点的链路，所以没有指定设备的选举。

不管接口的缺省网络类型是什么，都可以设置为广播网络类型。比如可以将非广播多路访问网络（帧中继，X.25）设置成广播网络，这样在 OSPF 路由进程配置时，省去了要配置邻居的步骤。通过 X.25 map 和 Frame-relay map 命令，可以让 X.25 和帧中继网络具有广播的能力，这样 OSPF 可以把 X.25 和帧中继这样的网络看作广播网络。

点到多点网络的接口可以看作是有一个或多个邻居的标识的点到点接口，当 OSPF 配置成点到多点网络类型时，会生成多个主机路由。点到多点网络类型与 NBMA 网络类型相比有以下好处：

- 容易配置，不需要配置邻居，也没有指定设备的竞选；
- 代价小，不要求全网状拓扑。

要配置网络类型，在接口配置模式中执行以下命令：

命令	作用
Ruijie(config-if)**# ip ospf network {broadcast | non-broadcast | point-to-point | {point-to-multipoint [non-broadcast]} }**	配置 OSPF 网络类型

对应于不同的链路封装类型,缺省情况下的网络类型如下：

- 点到点网络类型

PPP、SLIP、帧中继点到点子接口、X.25 点到点子接口封装

- NBMA 网络类型(non-broadcast)

帧中继、X.25 封装（点到点子接口除外）

- 广播网络类型

以太网封装

- 缺省类型没有为点到多点网络类型

配置时需要注意两端网络类型必须一致，否则可能出现邻居 FULL，但是路由计算错误的异常情况；

30.2.3.1 配置点到多点广播网络

当设备通过 X.25、帧中继网络互联时，不是全网状拓扑结构或者不想进行指定设备的选举时，就可以将 OSPF 接口网络类型设置为点到多点类型，由于点到多点网络类型将链路看作多个点到点链路，因此将产生多个主机路由。另外由于在点到

多点网络中，所有的邻居花费都是一样的，如果要求每个邻居的花费不同，可以通过 neighbor 命令进行设置。

要设置点到多点网络类型，在接口配置模式中执行以下命令：

命令	作用
Ruijie(config-if)# **ip ospf network point-to-multipoint**	配置一个接口为点到多点广播网络类型
Ruijie(config-if)# **exit**	退出到全局配置模式
Ruijie(config)# **router ospf** *1*	进入路由进程配置模式
Ruijie(config-router)# **neighbor ip-address cost** *cost*	可选，指定邻居的花费

说明：

OSPF 点到多点类型网络虽然属于非广播网络，但是通过帧中继、X.25 映射手工配置或者自动学习，可以让非广播网络具有广播的能力。所以在配置点到多点网络类型时，可以不要指定邻居。

30.2.3.2 配置非广播网络

当 ospf 工作在非广播网络上时既可以配置成 NBMA 也可以配置成点到多点非广播类型。因为不具备广播能力所以无法动态的发现邻居，因此 ospf 工作于非广播网络上时必须手工为其配置邻居。

考虑到以下情况时，可以配置 NBMA 网络类型：

1. 当一个非广播类型网络具有全网状拓扑结构；

2. 可以将一个广播类型的网络设置成 NBMA 网络类型，这样可以减少广播报文的产生，节约网络带宽，也可以在一定的程度上避免随便的接收和发布路由。配置 NBMA 网络必须要指定邻居，由于有指定设备的选择，你可能需要明确哪台设备作为指定设备，这就需要配置优先级了，优先值越大就越有可能成为指定设备。

要配置 NBMA 网络类型，在接口配置模式中执行以下命令:

命令	作用
Ruijie (config-if)# **ip ospf network non-broadcast**	指定该接口的网络类型为 NBMA 类型
Ruijie (config-if)# **exit**	退出到全局配置模式
Ruijie (config)# **router ospf** *1*	进入路由进程配置模式

Ruijie(config-router)# **neighbor** *ip-address* [*priority number*] [*poll-interval seconds*]	指定邻居,并且指定它的优先级和 hello 的 轮询间隔

当在一个非广播网络中，不能保证任意两台设备之间都是直接可达的话，更好的解决方法是将 ospf 的网络类型设置为点到多点非广播网络类型。

无论在点到多点广播或者非广播网络中，由于所有的邻居花费都是一样的，使用的花费都是使用 ip ospf cost 所配置的值.而实际上可能到每个邻居的带宽是不同的,因此花费也应该不同。因此可以通过 neighbor 命令为每个邻居指定所要的花费,这一点只适合点到多点类型（广播或者非广播）的接口。

在一个非广播网络中,如果要将接口配置成点到多点类型，在接口配置模式中执行以下命令：

命令	作用
Ruijie (config-if)# **ip ospf network point-to-multipoint non-broadcast**	指定该接口的网络类型为点到多点非广播 类型
Ruijie (config-if)# **exit**	退出到全局配置模式
Ruijie (config)# **router ospf** *1*	进入路由进程配置模式
Ruijie(config-router)# **neighbor** *ip-address* [*cost number*]	指定邻居,并且指定到该邻居的花费

注意步骤 4，如果没有邻居指定花费，那么将使用接口配置模式下的 ip ospf cost 命令所参考的代价。

30.2.3.3 配置广播网络类型

OSPF 广播类型网络，需要选举指定设备（DR,Designated Router）和备份指定设备（BDR，Backup Designated Router），由指定设备对外通告该网络的链路状态。所有的设备之间都保持邻居关系，但是所有设备只与指定设备和备份指定设备之间保持邻接关系，也就是说每台设备只与指定设备和备份指定设备交换链路状态数据包，然后由指定设备通告给所有的设备，从而每台设备能够保持一致的链路状态数据库。

你可以通过 OSPF 优先级参数设置来控制指定设备的选举结果。但是参数设置并不能马上产生作用，影响当前的指定设备，只有在新一轮的选举中，设置的参数才会起作用。进行新一轮指定设备选举的唯一条件是：OSPF 邻居在一定时间内没有接收到指定设备的 HELLO 报文，认为指定设备宕机了。

要配置广播网络类型，在接口配置模式中执行以下命令:

命令	作用

Ruijie (config-if)**#ip ospf network broadcast**	指定该接口的类型为广播网络类型
Ruijie (config-if)**#ip ospf priority** *priority*	可选，指定该接口的优先级

30.2.4 配置 OSPF 区域参数

当你需要配置区域认证、残域、缺省汇聚路由花费时，就需要通过配置区域命令来实现。

配置区域认证是为了防止学到非认证、无效路由，以及避免通告有效路由到非认证设备，在广播类型网络中，区域认证还可以避免非认证设备成为指定设备的可能性，保证了路由系统的稳定性和抗入侵性。

当一个区域为 OSPF 路由域的叶区域，即该区域不作为过渡区域，也不会向 OSPF 路由域注入外部路由时，可以考虑将该区域配置为残域（Stub Area）。残域设备只能学到三种路由：1）残域内部路由；2）其它区域路由；3）残域边界设备通告的缺省路由。由于没有大量的外部路由，所以残域设备的路由表将很小，可以节约设备的资源，因此残域的设备可以为中低端设备。为了进一步减少发送到 Stub 区域中的链路状态广播（LSA）的数量可以将区域配置为全残域(配置 no-summary 选项)，全残域设备就只能学到两种路由：1）残域内部路由；2）残域边界设备通告的缺省路由。全残域的配置，使得 OSPF 占用设备资源最小化，提高了网络传输效率。

如果残域设备可以学到多条缺省路由，就需要对缺省路由的花费进行设置(通过 area default-cost 配置)，这样就可以控制残域设备优先使用指定的缺省路由。

配置 STUB 区域要注意以下几点：

- 骨干区域不能配置成 STUB 区域，不能将 STUB 区域作为虚拟连接的传输区域。
- 如果想将一个区域配置成 STUB 区域，则该区域中的所有设备必须都要配置该属性。
- Stub 区域内不能存在 ASBR，即自治系统外部的路由不能在本区域内传播。

配置 OSPF 区域参数，在路由进程配置模式下执行以下命令：

命令	作用
Ruijie (config-router)**#area** *area-id* **authentication**	区域认证方式设置为明文认证
Ruijie (config-router)**#area** *area-id* **authentication message-digest**	区域认证方式设置为MD5认证

Ruijie (config-router)**#area** *area-id* **stub** [**no-summary**]	将区域配置成残域, **no-summary:** 将该区域配置成全残域，阻止Stub区间上的ABR发送summary-LSAs信息进入stub区间
Ruijie (config-router)**#area** *area-id* **default-cost** *cost*	配置发送到STUB区域的缺省路由的花费

说明:

认证的配置还需要在接口进行认证参数的配置，参见本章“配置 OSPF 接口参数”部分。残域的配置需要在该区域所有设备上进行配置；如果要配置成全残域，在残域基础配置上，只要在残域边界设备上配置全残域参数就行了，其它设备不要更改配置。

30.2.5 配置 OSPF NSSA

NSSA(Not-So-Stubby Area)是 OSPF STUB 区域的一个扩展,NSSA 也通过阻止第 5 类 LSA(AS-external-LSA)向 NSSA 的泛洪来降低设备的资源的消耗,但是和 STUB 不同的是 NSSA 可以注入一定数量的自治系统外部的路由信息到 OSPF 的路由选择域.

通过重分发，NSSA 允许输入类型 7 的自治系统外部路由到 NSSA 区域,这些类型 7 的外部 LSA 在 NSSA 的区域边界设备上将被转换成类型 5 的 LSA 并且泛洪到整个自治系统，在转换过程中可以实现对外部路由的汇聚和过滤。

配置 NSSA 时要注意以下几点：

- 骨干区域不能配置成 NSSA，不能将 NSSA 作为虚拟连接的传输区域。
- 如果想将一个区域配置成 NSSA，所有连接到 NSSA 区域的设备必须使用 area nssa 命令将该区域配置成 NSSA 属性。

要配置区域成为一个 NSSA 区域，在路由进程配置模式中执行以下命令：

命令	作用
Ruijie (config-router)**# area** *area-id* **nssa** [**no-redistribution**] [**no-summary**] [**default-information-originate**[**metric** *metric*][**metric-type** [**1** | **2**]]]	（可选）定义一个NSSA区间；
Ruijie (config-router)**# area** *area-id* **default-cost** *cost*	配置发送到NSSA区域的缺省路由的花费

参数 **default-information-originate** 用来产生默认的 Type-7 LSA，该选项在 nssa 的 ABR 和 ASBR 上有些差别，在 ABR 上无论路由表中是否存在缺省路由，都会产生 Type-7 LSA 缺省路由，在 ASBR 上(同时也不是 ABR)当路由表中存在缺省路由，才会产生 Type-7 LSA 缺省路由。

参数 no-redistribution 在 ASBR 上使得 OSPF 通过 redistribute 命令引入的其它外部路由不发布到 NSSA 区。该选项通常用于 NSSA 的设备既是 ASBR 又是 ABR 的时候，它可以阻止外部路由信息进入 nssa。

为了进一步减少发送到 NSSA 区域中的链路状态广播（LSA）的数量，可以在 ABR 上配置 no-summary 属性，禁止 ABR 向 NSSA 区域内发送 summary LSAs（Type-3 LSA）。

另外 area default-cost 用在连接该 NSSA 区域的边界设备 ABR 上。该命令配置区域边界设备发送到 NSSA 区域的缺省路由的花费值。缺省情况下，发送到 NSSA 区域缺省路由的花费值为 1。

30.2.6 配置 OSPF 路由汇聚

30.2.6.1 配置区间路由汇聚

区域边界设备(Area Border Routers)至少有两个接口属于不同的区域，而且其中一个区域一定为骨干区域。区域边界设备在 OSPF 路由域中起着枢纽作用，可以将一个区域的路由通告到其它区域，如果该区域的路由网络地址是连续的，则区域边界设备可以只通告一个汇聚路由到其它区域。区域之间的路由汇聚大大缩减了路由表的规模，提高了网络工作效率。

要配置区域之间路由汇聚，在路由进程配置模式中执行以下命令：

命令	作用
Ruijie (config-router)# **area** *area-id* **range** *ip-address mask* [**advertise** | **not-advertise**] [**cost** *cost*]	配置区间路由汇聚

说明：

如果配置了路由汇聚，落在该范围的详细路由将不再被区域边界设备通告到其它区域。

30.2.6.2 配置外部路由汇聚

当路由从其它路由进程重新分布，注入 OSPF 路由进程时，每条路由均分别单独

以一个外部链路状态的方式通告给 OSPF 设备。如果注入的路由是一个连续的地址空间，自治域边界设备也可以只通告一个汇聚路由，从而大大减小路由表的规模。

要配置外部路由汇聚，在路由进程配置模式中执行以下命令：

命令	作用
Ruijie (config-router)# **summary-address** *ip-address mask* {**not-advertise | tag** *tag-id*]	配置外部汇聚路由

30.2.6.3 配置汇聚路由表项添加到核心路由表的控制

在路由汇聚时，汇聚后的范围有可能超出路由表中实际的网络范围，如果数据发往汇聚范围内不存在的网络，将有可能发生路由环路或加重路由设备的处理负担。为阻止这种情况的发生，需要在 ABR 或 ASBR 添加一条 discard 路由到路由表。

要允许或禁止 discard 路由添加到路由表，在路由进程配置模式中执行以下命令：

命令	作用
Ruijie (config-router)# **discard-route {internal|external}**	允许discard路由添加到路由表
Ruijie (config-router)# **no discard-route {internal|external}**	禁止discard路由添加到路由表

缺省情况下，允许 discard 路由添加到路由表。

30.2.7 创建虚拟链接

在 OSPF 路由域中，非骨干区域之间的 OSPF 路由更新是通过骨干区域来交换完成的,所有的区域都必须与骨干域连接，如果骨干域断接，就需要配置虚拟链接将骨干域接续起来，否则网络通讯将出现问题。如果由于网络拓扑结构的限制可能无法保证物理上连通，也可以通过创建虚连接来满足这一要求。

虚拟链接需要在两个区域边界设备（ABR）之间创建，两个 ABR 共同所属的区域成为过渡区域。残域（Stub Area）和 NSSA 区域是不能作为过渡区域的。虚连接可以看成是在两台 ABR 之间通过传输区域建立的一条逻辑上的连接通道,它的两端必须是 ABR，而且必须在两端同时配置方可生效。虚连接由对端设备的 router-id 号来标识。为虚连接两端提供一条非骨干区域内部路由的区域称为传输区域（Transit Area），其区域号也必须在配置时指明。

虚连接要在传输区域内的路由计算出来后(到达对方设备的路由算出来)才会被激活，可以看成是一条点到点的连接，在这个连接上，和物理接口一样可以配置接口的多数参数，如 hello-interval,dead-interval 等。

“逻辑通道”是指两台 ABR 之间的多台运行 OSPF 的设备，只是起到一个转发报文的作用（由于协议报文的目的地址不是这些设备，所以这些报文对于他们而言是透

明的，只是当作普通的 IP 报文来转发），两台 ABR 之间直接传递路由信息。这里的路由信息是指由 ABR 生成的 type3 的 LSA，区域内的设备同步方式没有因此改变。

要建立虚拟链接，在路由进程配置模式中执行以下命令：

命令	作用
Ruijie (config-router)# **area** *area-id* **virtual-link** *router-id* [[**hello-interval** *seconds*]| [**retransmit-interval** *seconds*] |[**transmit-delay** *seconds*]|[**dead-interval** *seconds*]| [**authentication** [**message-digest | null**] |[[**authentication-key** *key* | **message-digest-key** *keyid* **md5** *key*]]]	创建一个虚拟连接

需要注意的是：如果自治系统被划分成一个以上的区域，则必须有一个区域是骨干区域，并且保证其它区域与骨干区域直接相连或逻辑上相连，且骨干区域自身也必须是连通的。

说明：

router-id 为 OSPF 邻居设备标识符，如果不确定 router-id 的值，请用 **show ip ospf** 命令进行确认或通过 **show ip ospf neighbor** 命令查看邻居的 router-id。如何手工配置 router-id，请参考本章“用 Loopback 地址做路由标识符”部分内容。

30.2.8 产生缺省路由

一个自治域系统边界设备(ASBR)，可以强迫其产生一条缺省路由注入到 OSPF 路由域中。如果一个设备强制产生缺省路由，该设备就自动成为 ASBR。但是 ASBR 不会自动产生缺省路由的。

要强制 ASBR 产生缺省路由，在路由进程配置模式下执行以下命令：

命令	作用
Ruijie (config-router)# **default-information originate** [**always**] [**metric** *metric-value*] [**metric-type** *type-value*] [**route-map** *map-name*]	配置产生缺省路由

说明：

当配置残域时，区域边界设备将自动产生缺省路由，然后通告到残域中的所有设备。

30.2.9 用 Loopback 地址做路由标识符

OSPF 路由进程总是用最大的接口 IP 地址作为设备标识符，如果该接口关闭或者该 IP 地址不存在了,OSPF 路由进程必须重新计算路由标识符并发送所有的路由信息给邻居。

如果配置了 Loopback（本地环路地址），则路由进程会选择 Loopback 接口的 IP 地址作为设备标识符。如果有多个 Loopback 接口，则选择 IP 地址最大的地址作为设备标识符。由于 Loopback 地址会永久存在，增强了路由表的稳定性。

要配置 Loopback 地址，在全局配置模式下，执行以下命令：

命令	作用
Ruijie (config)# **interface loopback** *1*	创建Loopback接口
Ruijie (config-if)# **ip address** *ip-address mask*	配置Loopback IP地址

说明：

若 OSPF 路由进程已经选举了普通接口的 IP 地址为路由标识符时，此时配置 loopback 口不会导致 OSPF 进程重新选举标识符。

30.2.10 更改 OSPF 缺省管理距离

路由管理距离，代表着该路由来源的可信度。管理距离是一个 0~255 之间的数值，管理距离值越大，说明该路由来源可信度越低。

我司产品的 OSPF 有三大类路由，管理距离缺省值均为 110：区域内部，区域之间，外部。属于一个区域的路由称为区域内部路由，到另一个区域的路由称为区域之间路由，到其它路由域（通过重新分布学到）的路由称为外部路由。

要更改 OSPF 管理距离，在路由进程配置模式下执行以下命令：

命令	作用
Ruijie(config-router)# **distance** {*distance* **| ospf** { **intra-area** *distance* **| inter-area** *distance* **| external** *distance* }}	更改OSPF管理距离值

30.2.11 配置路由计算计时器

在 OSPF 路由进程接收到网络拓扑变化的通知后，会延迟一段时间，然后运行 SPF 进行路由运算。该延时是可以配置的，另外还可以配置两次 SPF 运算之间最短的时间间隔。

要配置 OSPF 路由计算的计时器，在路由进程配置模式下执行以下命令：

命令	作用
Ruijie (config-router)# **timers throttle spf** *spf-delay spf-holdtime spf-max-waittime*	配置路由计算机时器

说明：

spf-delay 表示从拓扑发生变化到 SPF 开始计算，至少需要延迟的时间。第一次触发 SPF 计算到第二次触发 SPF 计算的最小时间间隔为 spf-holdtime， 此后，连续触发 SPF 计算的时间间隔至少为上一次时间间隔的两倍，当时间间隔达到 spf-max-waittime 之后，将不再增加。如果两次 SPF 计算的时间间隔已经超过了要求的最小值，那么 SPF 计算时间间隔将重新从 spf-holdtime 开始计算。

在正常情况，只是链路偶尔动荡时，减少 spf-delay 和 spf-holdtime 值，可以加快 OSPF 收敛速度；spf-max-waittime 设置为较大值，可以防止链路连续动荡导致的 OSPF 消耗大量 CPU 的情况。

举例配置值为（timers throttle spf 1000 5,000 100,000）：若拓扑连续变化，则 SPF 计算时间点分别为(SPF 计算时间间隔以二进制指数退避算法递增，但最大不超过 max-wait-time)：1s，6s，16s，36s，76s，156s，256s，256+100，…

如果只需要配置 OSPF 路由计算的延迟时间和保持时间，可以在路由进程配置模式下执行以下命令：

命令	作用
Ruijie (config-router)# **timers spf** *spf-delay spf-holdtime*	配置路由计算机时器，此命令以秒为单位

注意：

timers spf 与 timers throttle spf 两个配置命令会互相覆盖，后配置的命令生效，当两个命令都未配置时，此时系统的缺省行为是 timers throttle spf 的缺省值。

timers throttle spf 命令功能已经包含了 timers spf 命令的功能，并且更加强大；建议用户使用 timer throttle spf 命令。

30.2.12 状态更新信息调整时间

每个 LSA 都有自己的更新和老化的时间(LSA age)，如果对每个 LSA 都独立进行 LSA 的更新和老化计算，将消耗 CPU 大量的资源。为了有效利用 CPU 资源，将设备内 LSA 进行统一的更新和老化操作。

更新老化操作的缺省设置为 4 分钟，可以通过本命令修改该缺省值。该参数并不需要经常调整。最佳的时间是与路由设备有多少 LSA 需要计算成反比的。比如，数据库中有 10000 条 LSA，降低步调间隔将使你受益；如果只有 40~100 条，则调整到 10~20 分钟可能会更好。

在路由进程配置模式下执行以下命令：

命令	含义
Ruijie **# configure terminal**	进入全局配置模式
Ruijie (config)**# router ospf** *1*	打开 OSPF，进入 OSPF 配置模式
Ruijie (config-router)**# timers lsa-group-pacing** *seconds*	（可选）改变步调时间
Ruijie (config-router)**# end**	退回到特权模式。
Ruijie **# show running-config**	验证设置内容是否正确
Ruijie **# write**	（可选）保存配置。

使用 **no timers lsa-group-pacing** 命令将取消设置使其恢复使用缺省值。

30.2.13 配置 OSPF 接口度量

OSPF 利用度量（Cost）计算目的路径，Cost 最小者即为最短路径。缺省的路由度量是基于网络带宽的。在配置 OSPF 设备时还可根据实际情况，如链路带宽、时延或经济上的费用，设置链路度量大小。度量越小，则该链路被选为路由的可能性越大。如果发生路由汇聚，则使用汇聚的所有链路的度量最大值作为整个汇聚信息的度量。

路由选择的配置有两个部分。首先可以设置带宽产生度量的参考值，该值与接口带宽值一起计算生成缺省度量。其次，可以通过接口配置命令 ip ospf cost 设置每个接口各自的度量值，此时缺省度量对该接口就不产生作用了。比如，缺省的参考值为 100Mbps，一个以太网接口带宽为 10Mbps，则该接口缺省的度量为 100/10 + 0.5 ≈ 10。

在协议内部是这样选择接口代价的，用户设置的接口的代价优先级最高，如果用户设置了接口的代价，则选取该值作为接口代价，如果没有设置而自动代价功能是打

开的则将根据自动代价计算出来的值作为接口的代价，如果自动代价功能也关闭了则将使用缺省值 10 作为接口的代价。

配置过程如下：

命令	含义
Ruijie **# configure terminal**	进入全局配置模式。
Ruijie (config)**# router ospf** *1*	打开 OSPF，进入 OSPF 配置模式
Ruijie(config-router)**# auto-cost reference-bandwidth** *ref-bw*	（可选）根据接口带宽设置缺省度量，度量值根据 ref-bw 确定；
Ruijie (config-router)**# end**	退回到特权模式。
Ruijie **# show ip protocols**	显示当前运行的路由协议。
Ruijie **# write**	（可选）保存配置。

使用命令 **no auto-cost** 与 **no ip ospf cost** 取消设置内容。

30.2.14 配置接口接收数据库描述报文时是否校验 MTU 值

OSPF 在收到数据库描述报文时会校验邻居的接口的 MTU 和自己接口的 MTU 是否相同，如果收到的数据库描述报文中指示的接口的 MTU 大于接收接口的 MTU,那么邻接关系将不能被建立,此时可以通过关闭 MTU 的校验来解决.要关闭某个接口的 MTU 校验可以在接口模式下执行如下命令：

命令	含义
Ruijie (config-if)**# ip ospf mtu-ignore**	关闭该接口在收到数据库描述报文时对 MTU 进行校验

缺省情况下，接口的 MTU 校验功能是打开的.

30.2.15 配置禁止接口发送 OSPF 报文

为了防止网络中的其他设备动态的学习到本设备的路由信息，可以将本设备的指定网络接口设为被动接口，通过使用 **passive-interface** 命令来禁止在此接口上发送 OSPF 报文。

在特权模式下，用户可以按如下步骤进行被动口的配置：

命令	含义
Ruijie **# configure terminal**	进入全局配置模式。

Ruijie (config)# **router ospf** *1*	进入路由协议配置模式
Ruijie (config-router)# **passive-interface** *interface-name*	(可选)将指定的网络接口设为被动。
Ruijie (config-router)# **passive-interface default**	(可选)设置所有的网络接口为被动
Ruijie (config-router)# **end**	退回到特权模式。
Ruijie# **write**	保存配置

缺省情况下，允许所有接口收发 OSPF 报文。如果要让网络接口重新发送路由信息，则可以使用 **no passive-interface** *interface-id* 命令进行设置。如果使用参数 default 则是对所有的接口进行设置。

30.2.16 配置 OSPF 容量保护

内存不足情况下，允许 OSPF 进入 OVERFLOW 状态。OVERFLOW 状态是指在该状态下，OSPF 协议将触发如下行为：

- 对于学习到的 LSA：对于表示区内区间拓扑的 LSA，需要接收；对于外部 LSA，如果 LSA 所表示的路由目的地址为已经学习到的某条非缺省路由的精细路由，则接收；其他情况，LSA 不接收。
- 对于自身成的外部 LSA：清除除表示缺省路由外的其他外部 LSA。
- 由于路由学习和通告的不完整性，可能会引起网络出现路由环路的现象，为了减少出现这种现象的产生，OSPF 会生成一条指向 NULL 口的缺省路由，该路由在 OVERFLOW 状态下将一直存在。

设置内存不足时 OSPF 进入 OVERFLOW 状态的配置步骤如下：

	命令	作用
Step 1	Ruijie(config)**#router ospf** *process-id*	进入 OSPF 配置模式
Step 2	Ruijie(config-router)**#overflow memory-lack**	设置内存不足时 OSPF 进入 OVERFLOW 状态

说明

缺省情况下，内存不足将自动进入 OVERFLOW 状态，如果用户不希望进入 OVERFLOW 状态，可以使用 **no overflow memory-lack** 命令关闭该功能。

注意

在进入 OVERFLOW 状态后，目前 OSPF 协议只支持使用 **clear ip ospf process** 命令，或者关闭再重新启动 OSPF 协议的方式退出 OVERFLOW 状态。

30.2.17 配置 OSPF 网管功能

30.2.17.1 配置 OSPFv2 MIB 绑定

由于 OSPFv2 MIB 本身没有 OSPFv2 进程信息，所以用户通过 SNMP 操作 OSPFv2 进程，只能操作唯一的进程。缺省情况下，OSPFv2 MIB 绑定在进程号最小的 OSPFv2 进程上，用户的操作都对该进程生效。

如果用户希望能够通过 SNMP 操作指定的 OSPFv2 进程，可以通过手工配置的方式将 OSPFv2 MIB 绑定该进程上。

在路由进程配置模式下执行以下命令：

命令	作用
Ruijie (config-router)**#enable mib-binding**	配置将OSPFv2 MIB绑定到指定的OSPFv2进程上

30.2.17.2 配置 OSPFv2 TRAP 功能

OSPFv2 协议定义了多种类型的 TRAP 信息，用于报告 OSPFv2 协议内部发生的各类事件。OSPFv2 TRAP 信息的发送不受 OSPFv2 进程绑定 MIB 的限制，允许不同进程同时打开 TRAP 开关。

在全局配置模式下，按照如下步骤执行：

命令	含义
Ruijie **# configure terminal**	进入全局配置模式。
Ruijie (config)**# snmp-server host** *host-ip* **version** *version-no string* [**ospf**]	配置用于接收 TRAP 的 snmp-server； host-ip 指 server 对应的地址；version-no 指 server 对应的 snmp 版本；string，是 snmp 的通信认证码，通常为 public；可选参数 ospf，指此 snmp-server 接收 OSPF TRAP（默认情况 server 接收所有类型的 TRAP）
Ruijie (config)**#snmp-server enable traps ospf**	打开 OSPF TRAP 发送开关
Ruijie (config)**#router ospf** *process_id* [**vrf** *vrf-name*]	打开 OSPF，进入 OSPF 配置模式

Ruijie (config-router)# **enable traps** [**error** [**ifauthfailure | ifconfigerror | ifrxbadpacket | virtifauthfailure | virtifconfigerror | virtifrxbadpacket**] | **lsa** [**lsdbapproachoverflow | lsdboverflow | maxagelsa | originatelsa**] | **retransmit** [**iftxretransmit | virtiftxretransmit**] | **state-change** [**ifstatechange | nbrstatechange | virtifstatechange | virtnbrstatechange**]]	打开指定的 OSPF TRAP 开关
Ruijie (config)# **end**	退回到特权模式。
Ruijie# **write**	保存配置

30.3 监视和维护 OSPF

可以显示 OSPF 的路由表，缓存，数据库等数据。下表列出了部分具体可以显示的内容供参考。

命令	含义
Ruijie# **show ip ospf** [*process-id*]	显示相应进程 OSPF 协议的一般信息，未指定进程号则显示所有进程。
Ruijie# **show ip ospf** [*process-id area-id*] **database** [**adv-router** *ip-address* | {**asbr-summary** | **external** | **network** | **nssa-external | opaque-area | opaque-as | opaque-link | router** | **summary**} [*link-state-id*] [{**adv-router** *ip-address* | **self-originate**}] | **database-summary** | **max-age** | **self-originate**]	OSPF 数据库的信息 可以看指定进程每种 LSA 类型的信息。
Ruijie# **show ip ospf** [*process-id*] **border-routers**	查看指定进程到达 ABR 与 ASBR 路由信息
Ruijie# **show ip ospf interface** [*interface-name*]	查看参与 ospf 路由的接口信息

Ruijie**# show ip ospf** [*process-id*] **neighbor**[*interface-name*] [*neighbor-id*] [**detail**]	接口相邻设备的信息 interface-name ： 与该 neighbor 相连的本地接口。neighbor-id: neighbor 的设备 ID
Ruijie**# show ip ospf** [*process-id*] **virtual-links** *ip-address*	查看指定进程虚拟连接信息
Ruijie**# show ip ospf** [*process-id*] **route** [**count**]	显示 OSPF 路由表路由

具体的命令解释，请参见 OSPF 命令参考中对各条命令的解释。常用的监视和维护命令有以下几个：

1）显示 OSPF 邻居状态

使用 **show ip ospf** [*process-id*] **neighbor** 显示 OSPF 进程的所有邻居信息，包括邻居的状态、角色、设备标识、IP 地址、BFD 状态等。

```
Ruijie # show ip ospf neighbor
OSPF process 1:
Neighbor ID      Pri   State               BFD State  Dead
Time    Address         Interface
10.10.10.50      1   Full/DR             Up
00:00:38      10.10.10.50   eth0/0
OSPF process 100:
Neighbor ID      Pri   State                  BFD State
Dead Time   Address        Interface
10.10.11.50       1   Full/Backup       Down       00:00:31
   10.10.11.50  eth0/1
Ruijie # show ip ospf 1 neighbor
OSPF process 1:
Neighbor ID      Pri   State                  BFD State
Dead Time   Address           Interface
10.10.10.50      1   Full/DR                Up
00:00:38     10.10.10.50    eth0/0
Ruijie # show ip ospf 100 neighbor
OSPF process 100:
Neighbor ID      Pri   State                  BFD State
Dead Time   Address           Interface
10.10.11.50       1    Full/Backup         Down
00:00:31     10.10.11.50  etheth0/1
```

2）显示 OSPF 接口状态

以下显示表明 F0/1 端口属于 OSPF 的区域 0，设备标识符为 172.16.120.1，网络类型为“BROADCAST”—广播类型。要特别注意 Area 、Netwrok Type 、Hello、

Dead 等参数，如果这些参数与邻居不一致，将不会建立邻居关系。

```
Ruijie#sh ip ospf interface fastEthernet 1/0
FastEthernet 1/0 is up, line protocol is up
Internet Address 192.168.1.1/24, Ifindex: 2 Area 0.0.0.0, MTU
1500
Matching network config: 192.168.1.0/24
Process ID 1, Router ID 192.168.1.1, Network Type BROADCAST,
Cost: 1
Transmit Delay is 1 sec, State DR, Priority 1
Designated Router (ID) 192.168.1.1, Interface Address
192.168.1.1
Backup Designated Router (ID) 192.168.1.2, Interface Address
192.168.1.2
Timer intervals configured, Hello 10, Dead 40, Wait 40,
Retransmit 5
Hello due in 00:00:04
Neighbor Count is 1, Adjacent neighbor count is 1
Crypt Sequence Number is 30
Hello received 972 sent 990, DD received 3 sent 4
LS-Req received 1 sent 1, LS-Upd received 10 sent 26
LS-Ack received 25 sent 7, Discarded 0
```

3）显示 OSPF 路由进程信息

以下命令可以显示路由标识符，设备类型,区域的信息,区域汇聚的信息等。

```
Ruijie# show ip ospf
Routing Process "ospf 1" with ID 1.1.1.1
Process uptime is 4 minutes
Process bound to VRF default
Conforms to RFC2328, and RFC1583Compatibility flag is enabled
Supports only single TOS(TOS0) routes
Supports opaque LSA
This router is an ASBR (injecting external routing information)
SPF schedule delay 5 secs, Hold time between two SPFs 10 secs
LsaGroupPacing: 240 secs
Number of incomming current DD exchange neighbors 0/5
Number of outgoing current DD exchange neighbors 0/5
Number of external LSA 4. Checksum 0x0278E0
Number of opaque AS LSA 0. Checksum 0x000000
Number of non-default external LSA 4
External LSA database is unlimited.
Number of LSA originated 6
Number of LSA received 2
Log Neighbor Adjency Changes : Enabled
Number of areas attached to this router: 1
Area 0 (BACKBONE)
```

```
Number of interfaces in this area is 1(1)
Number of fully adjacent neighbors in this area is 1
Area has no authentication
SPF algorithm last executed 00:01:26.640 ago
SPF algorithm executed 4 times
Number of LSA 3. Checksum 0x0204bf

Routing Process "ospf 20" with ID 2.2.2.2
Process uptime is 4 minutes
Process bound to VRF default
Conforms to RFC2328, and RFC1583Compatibility flag is enabled
Supports only single TOS(TOS0) routes
Supports opaque LSA
SPF schedule delay 5 secs, Hold time between two SPFs 10 secs
LsaGroupPacing: 240 secs
Number of incomming current DD exchange neighbors 0/5
Number of outgoing current DD exchange neighbors 0/5
Number of external LSA 0. Checksum 0x000000
Number of opaque AS LSA 0. Checksum 0x000000
Number of non-default external LSA 0
External LSA database is unlimited.
Number of LSA originated 0
Number of LSA received 0
Log Neighbor Adjency Changes : Enabled
Number of areas attached to this router: 0
```

30.3.1 配置 OSPF GR

GR 是 Graceful Restart（优雅重启），主要是为了实现在协议的重新启动过程中数据转发不间断。目前我司支持在高端设备的主备切换过程中启动 GR 功能，以保证关键业务不中断。

30.3.1.1 OSPF GR 的工作机制

OSPF GR 实现标准

RFC3623：Graceful OSPF Restart

RFC3623 工作机制

RFC3623 是 IETF 为 OSPF 制定的标准 GR 协议。 RFC 定义了执行 Graceful Restart 时需要满足的条件、操作方法及注意事项；RFC3623 中定义了 GR 有两个重要的原则：网络拓扑要保持稳定；重启协议路由器可以在重启过程中保持转发表。

OSPF GR 的执行不是一个单独的个体过程。从功能上分为 GR Restart 能力和 GR Help 能力，具备 GR Restart 能力的设备能够自主的执行优雅重启，而具备 GR Help 能力的设备可以接收 Grace-LSA，并协助邻居执行优雅重启。

把具备 GR Restart 能力且正在进行 GR 的设备称为 GR Restarter，把具备 GR Help 能力且正在辅助 GR Restarter 进行 GR 的设备称为 GR Helper。GR 的过程是 Restarter 发送一个 Grace LSA 的通告开始的。邻居设备在收到 Grace LSA 后进入 Helper 模式，辅助 Restarter 重建邻接关系，同时向外保持与 Restarter 的邻接关系，使网络中的其他设备无法感知网络的变化，从而保证数据不间断转发。

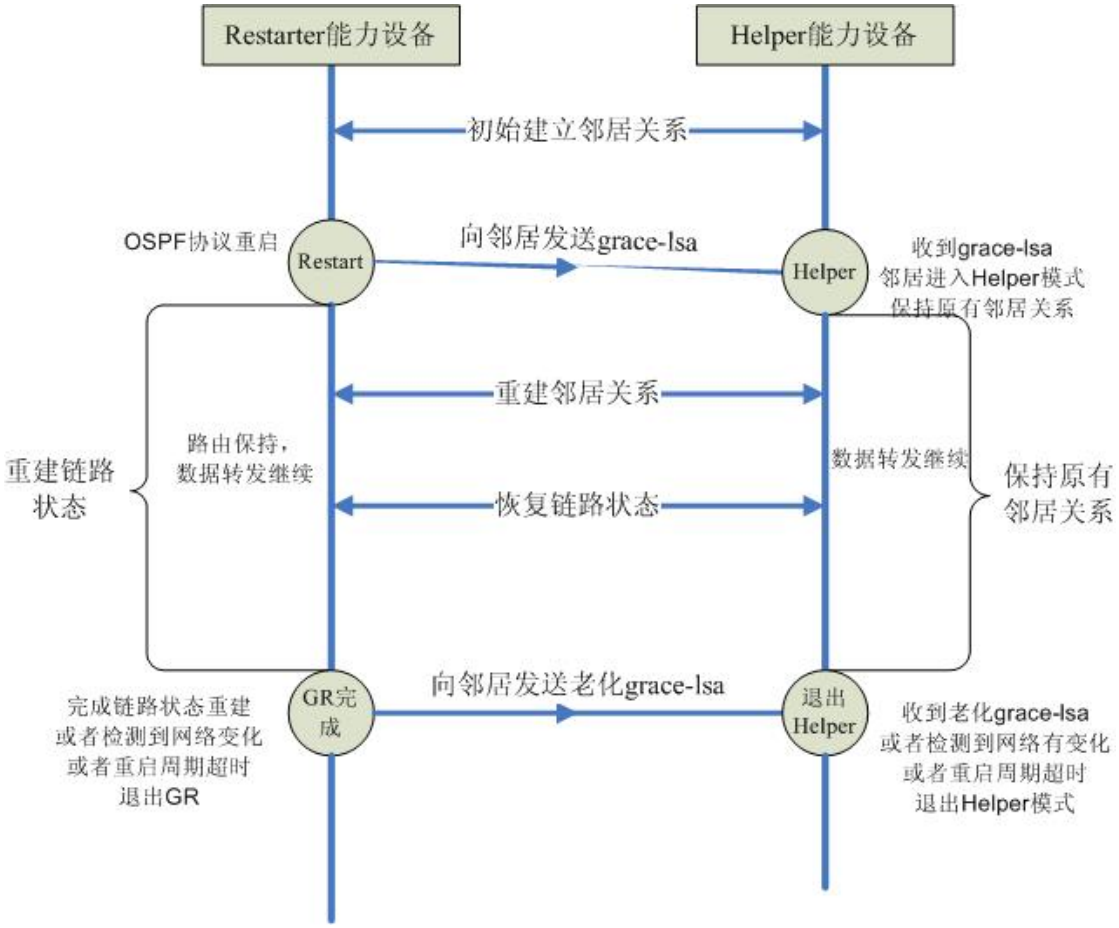


图 1　OSPF GR 执行过程图

上图简要描述了整个 OSPF GR 的执行流程。优雅重启周期即为“重建链路状态”的最长时间。在完成链路重建或者优雅重启周期超时时，Restarter 将退出 GR 的执行。

30.3.1.2　配置 OSPF GR helper

默认情况下 GR Helper 无需配置，它默认为使能的。软件提供了禁止 GR Helper 能力和配置 Helper 检测网络变化的机制。下面举例说明如何禁止 GR Helper 能力和重新使能，同时说明如何修改 Helper 检测网络变化的机制：

命令	作用

Step 1	Ruijie **# configure terminal**	进入全局配置模式
Step 2	Ruijie (config)**# router ospf** *1*	打开 OSPF，进入 OSPF 配置模式
Step 3	Ruijie (config-router)**# graceful-restart helper disable**	配置禁止 OSPF 的 GR Helper 能力，禁止对邻居执行 GR 辅助。
Step 4	Ruijie (config-router)**# no graceful-restart helper disable**	重新开启 OSPF 的 GR Helper 能力，恢复为默认行为。
Step 5	Ruijie (config-router)**# graceful-restart helper {strict-lsa-checking | internal-lsa-checking}**	配置 OSPF 的 GR Helper 启动通过检查 LSA 的变化判断网络是否发生变化，如果网络发生变化，将退出 GR Helper，默认在 GR Helper 期间不检查网络是否发生变化 **strict-lsa-checking**: 通过检查 types 1-5,7 的 LSA 的变化进行判断 **internal-lsa-checking**: 通过检查 types 1-3 的 LSA 的变化进行判断
Step 6	Ruijie (config-router)**# end**	退回到特权模式。
Step 7	Ruijie **# show running-config**	验证设置内容是否正确
Step 8	Ruijie **# write**	（可选）保存配置。

在网络规模较大的情况下，不建议用户启动 LSA 检测选项，因为局部网络的变化会触动 GR 结束，从而导致整网的收敛降低。

30.4 OSPF 配置范例

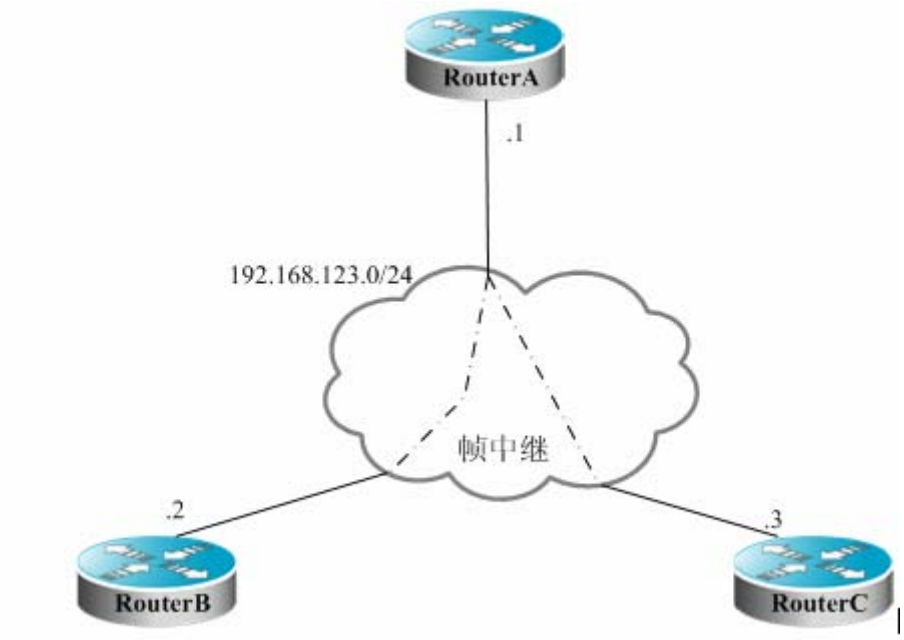
本章节提供了 8 个 OSPF 配置范例：

- OSPF NBMA 网络类型配置例子
- OSPF 点到多点网络类型配置例子
- OSPF 认证配置例子
- OSPF 路由汇聚配置例子
- OSPF ABR、ASBR 配置例子
- OSPF 残域配置例子
- OSPF 虚拟链接配置例子

30.4.1 OSPF NBMA 网络类型配置例子

- **配置要求**

三台设备要通过帧中继网络实现网状全连接，每台设备只有一条帧中继线路，线路带宽和 PVC 速率全部一样。IP 地址分配和设备连接图见图 1。



OSPF NBMA 网络类型配置例子

要求：

（1） 设备 A、B、C 之间配置 NBMA 类型网络；

（2） 设备 A 为指定设备，设备 B 为备份指定设备；

（3） 全部网络都在一个区域内；

（4） 加快拓扑收敛。

- 设备具体配置

由于OSPF 没有特殊配置，将采用组播方式自动发现邻居，如果接口配置了 NBMA 网络类型，该接口就不会发 OSPF 组播报文，因此需要指定邻居的 IP 地址。可以通过设置较短的 SPF 计算等待时间来加快拓扑收敛。

设备 A 的配置：

#配置广域网端口

```
interface Serial 1/0
ip address 192.168.123.1 255.255.255.0
encapsulation frame-relay
ip ospf network non-broadcast
ip ospf priority 10
```

#配置 OSPF 路由协议，到设备 B 的花费更小

```
router ospf 1
network 192.168.123.0 0.0.0.255 area 0
neighbor 192.168.123.2  priority 5
neighbor 192.168.123.3
timers spf 500 1000 10000
```

设备 B 的配置：

#配置广域网端口

```
interface Serial 1/0
ip address 192.168.123.2 255.255.255.0
encapsulation frame-relay
ip ospf network non-broadcast
ip ospf priority 5
```

#配置 OSPF 路由协议

```
router ospf 1
network 192.168.123.0 0.0.0.255 area 0
neighbor 192.168.123.1 priority 10
neighbor 192.168.123.3
timers spf 500 1000 10000
```

设备 C 的配置：

#配置广域网端口

```
interface Serial 1/0
ip address 192.168.123.3 255.255.255.0
encapsulation frame-relay
ip ospf network non-broadcast
```

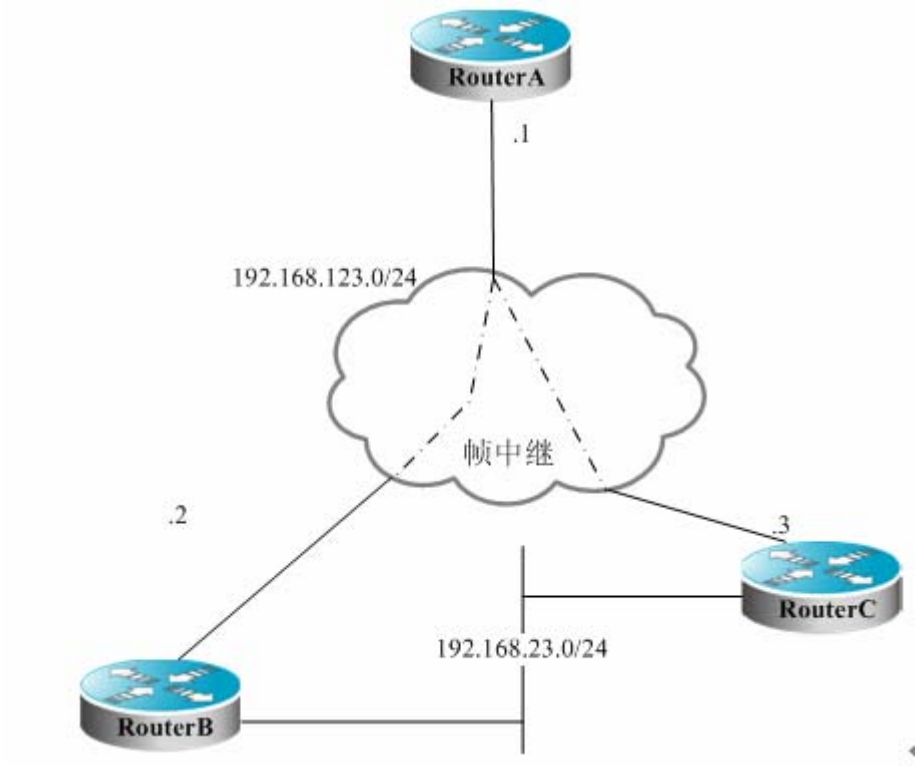
#配置 OSPF 路由协议

```
router ospf 1
network 192.168.123.0 0.0.0.255 area 0
neighbor 192.168.123.1 10
neighbor 192.168.123.2 5
timers spf 500 1000 10000
```

30.4.2 OSPF 点到多点广播网络类型配置例子

- 配置要求

三台设备要通过帧中继网络实现互联，每台设备只有一条帧中继线路，线路带宽和 PVC 速率全部一样。IP 地址分配和设备连接图见图 2。



OSPF 点到多点网络类型配置例子

要求：1）设备 A 与 B、C 之间配置点到多点类型网络。

- 设备具体配置

如果接口配置了点到多点网络类型，点到多点网络类型没有竞选指定设备的过程，OSPF 操作与点到点网络类型的行为很类似。

设备 A 的配置：

#配置以太网端口

```
interface FastEthernet 0/0
ip address 192.168.12.1 255.255.255.0
```

#配置广域网端口

```
interface Serial 1/0
ip address 192.168.123.1 255.255.255.0
encapsulation frame-relay
ip ospf network point-to-multipoint
```

#配置 OSPF 路由协议

```
router ospf 1
network 192.168.23.0 0.0.0.255 area 0
network 192.168.123.0 0.0.0.255 area 0
```

设备 B 的配置：

#配置以太网端口

```
interface FastEthernet 0/0
ip address 192.168.23.2 255.255.255.0
```

#配置广域网端口

```
interface Serial 1/0
ip address 192.168.123.2 255.255.255.0
encapsulation frame-relay
ip ospf network point-to-multipoint
```

#配置 OSPF 路由协议

```
router ospf 1
network 192.168.23.0 0.0.0.255 area 0
network 192.168.123.0 0.0.0.255 area 0
```

设备 C 的配置：

#配置以太网端口

```
interface FastEthernet 0/0
ip address 192.168.23.3 255.255.255.0
```

#配置广域网端口

```
interface Serial 1/0
ip address 192.168.123.3 255.255.255.0
encapsulation frame-relay
ip ospf network point-to-multipoint
```

#配置 OSPF 路由协议

```
router ospf 1
network 192.168.23.0 0.0.0.255 area 0
network 192.168.123.0 0.0.0.255 area 0
```

在上图的配置中假设还有一个需求是:

设备 A 到 192.168.23.0/24 目标网络将优先选择设备 B；要达到优先选路的要求，必须在配置邻居时设置该邻居的花费.

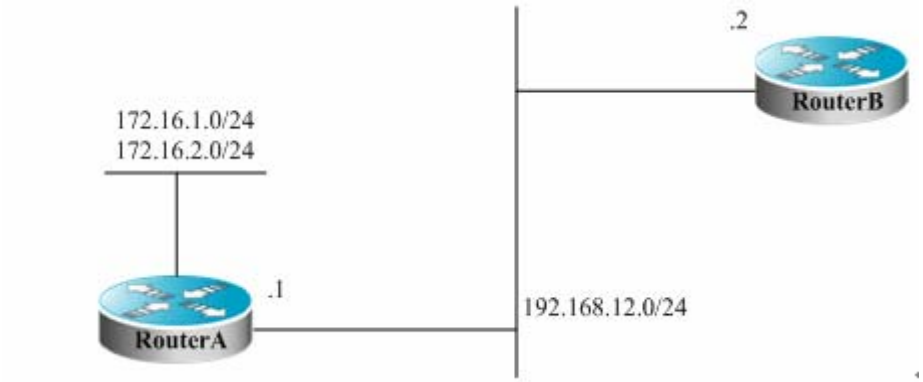
可以在设备 A 中配置如下命令:

```
router ospf 1
neighbor 192.168.123.2 cost 100
neighbor 192.168.123.3 cost 200
```

30.4.3 OSPF 认证配置例子

配置要求

两台设备通过以太网互连，运行 OSPF 路由协议，采用 MD5 认证方式。设备之间连接图以及 IP 地址分配见图 3。



OSPF 认证配置例子

- 设备具体配置

OSPF 的认证配置有两个地方：

1）路由配置模式中，指定区域的认证方式 ；

2）在接口中配置认证的方式和密钥。

如果区域认证方式和接口认证方式都配置了则以接口的认证方式为准

设备 A 的配置：

#配置以太网端口

```
interface FastEthernet0/0
ip address 192.168.12.1 255.255.255.0
ip ospf message-digest-key 1 md5 hello
```

#配置 OSPF 路由协议

```
router ospf 1
network 192.168.12.0 0.0.0.255 area 0
area 0 authentication message-digest
```

设备 B 的配置：

#配置以太网端口

```
interface FastEthernet0/0
ip address 192.168.12.2 255.255.255.0
ip ospf message-digest-key 1 md5 hello
```

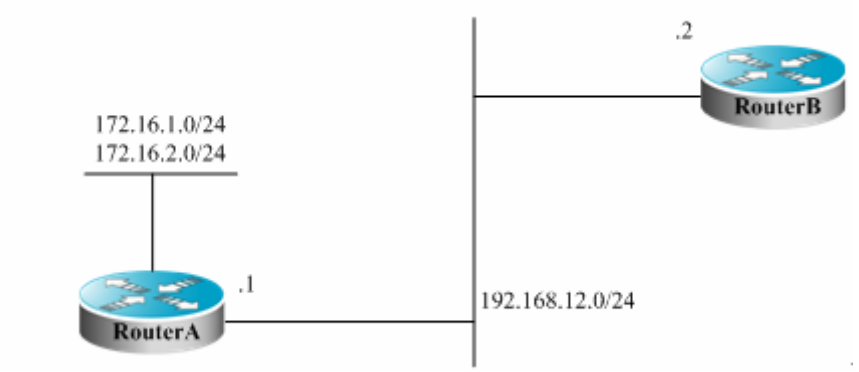
#配置 OSPF 路由协议

```
router ospf 1
network 192.168.12.0 0.0.0.255 area 0
area 0 authentication message-digest
```

30.4.4 OSPF 路由汇聚配置例子

- 配置要求

两台设备通过以太网连接， IP 地址分配和设备连接图见图 4。



OSPF 路由汇聚配置例子

要求：

1）两台设备均运行 OSPF 路由协议，网络 192.168.12.0/24 属于区域 0，网络 172.16.1.0/24 和 172.16.2.0/24 属于区域 10；

2）通过配置 Router A，使得路由 A 只通告 172.16.0.0/22 路由，而不通告 172.16.1.0/24、172.16.2.0/24 的路由。

3）通过配置 Router A，使得设备 A 通告的汇聚路由的度量值为 20，并且不向路由表添加 discard 路由。

- 设备具体配置

需要在 Router A 上配置 OSPF 区域路由汇聚，注意区域路由汇聚只能在区域边界设备上配置。

设备 A 的配置：

#配置以太网端口

```
interface FastEthernet0/0
ip address 192.168.12.1 255.255.255.0
```

#配置以太网卡上的 2 个端口

```
interface FastEthernet1/0
ip address 172.16.1.1 255.255.255.0
interface FastEthernet1/1
ip address 172.16.2.1 255.255.255.0
```

#配置 OSPF 路由协议

```
router ospf 1
network 192.168.12.0 0.0.0.255 area 0
```

```
network 172.16.1.0 0.0.0.255 area 10
network 172.16.2.0 0.0.0.255 area 10
no discard-route internal
area 10 range 172.16.0.0 255.255.252.0 cost 20
```

设备 B 的配置：

#配置以太网端口

```
interface FastEthernet0/0
ip address 192.168.12.2 255.255.255.0
```

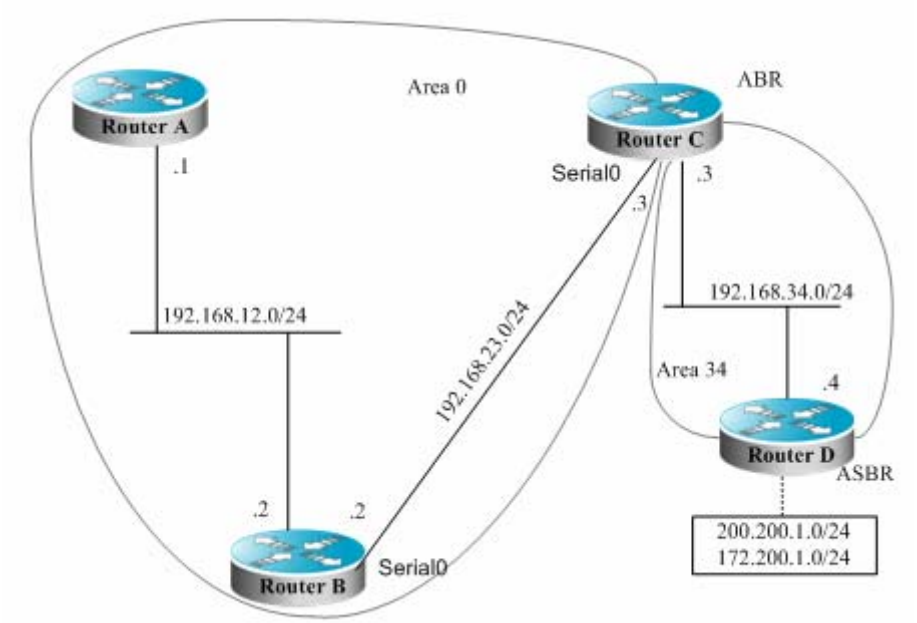
#配置 OSPF 路由协议

```
router ospf 1
network 192.168.12.0 0.0.0.255 area 0
```

30.4.5 OSPF ABR、ASBR 配置例子

- 配置要求

有四台设备组成了一个 OSPF 路由域，网络 192.168.12.0/24、192.168.23.0/24 属于区域 0，网络 192.168.34.0/24 属于区域 34。具体 IP 地址分配和设备连接图见图 5。



OSPF ABR、ASBR 配置例子

如图所示，设备 A、B 为区域内部设备，设备 C 为区域边界设备，设备 D 为自治域边界设备。200.200.1.0/24、172.200.1.0/24 为 OSPF 路由域外的网络。通过配置各设备，使得所有 OSPF 设备学到外部路由,要求外部路由携带“34”标记以及类型为外部路由 I 型。

- 设备具体配置

OSPF 重分布其它来源的路由时，缺省类型为 II 型，而且不携带任何标记。

设备 A 的配置：

#配置以太网端口

```
interface FastEthernet0/0
ip address 192.168.12.1 255.255.255.0
```

#配置 OSPF 路由协议

```
router ospf 1
network 192.168.12.0 0.0.0.255 area 0
```

设备 B 的配置：

#配置以太网端口

```
interface FastEthernet0/0
ip address 192.168.12.2 255.255.255.0
```

#配置广域网端口

```
interface Serial 1/0
ip address 192.168.23.2 255.255.255.0
```

#配置 OSPF 路由协议

```
router ospf 1
network 192.168.12.0 0.0.0.255 area 0
network 192.168.23.0 0.0.0.255 area 0
```

设备 C 的配置：

#配置以太网端口

```
interface FastEthernet 0/0
ip address 192.168.34.3 255.255.255.0
```

#配置广域网端口

```
interface Serial 1/0
ip address 192.168.23.3 255.255.255.0
```

#配置 OSPF 路由协议

```
router ospf 1
network 192.168.23.0 0.0.0.255 area 0
network 192.168.34.0 0.0.0.255 area 34
```

设备 D 的配置：

#配置以太网端口

```
interface FastEthernet 0/0
ip address 192.168.34.4 255.255.255.0
```

#配置以太网卡上的端口

```
interface FastEthernet 1/0
ip address 200.200.1.1 255.255.255.0
interface FastEthernet 1/1
ip address 172.200.1.1 255.255.255.0
```

#配置 OSPF 路由协议，重分布 RIP 路由

```
router ospf 1
network 192.168.34.0 0.0.0.255 area 34
redistribute rip metric-type 1 subnets tag 34
```

#配置 RIP 路由协议

```
router rip
network 200.200.1.0
network 172.200.0.0
```

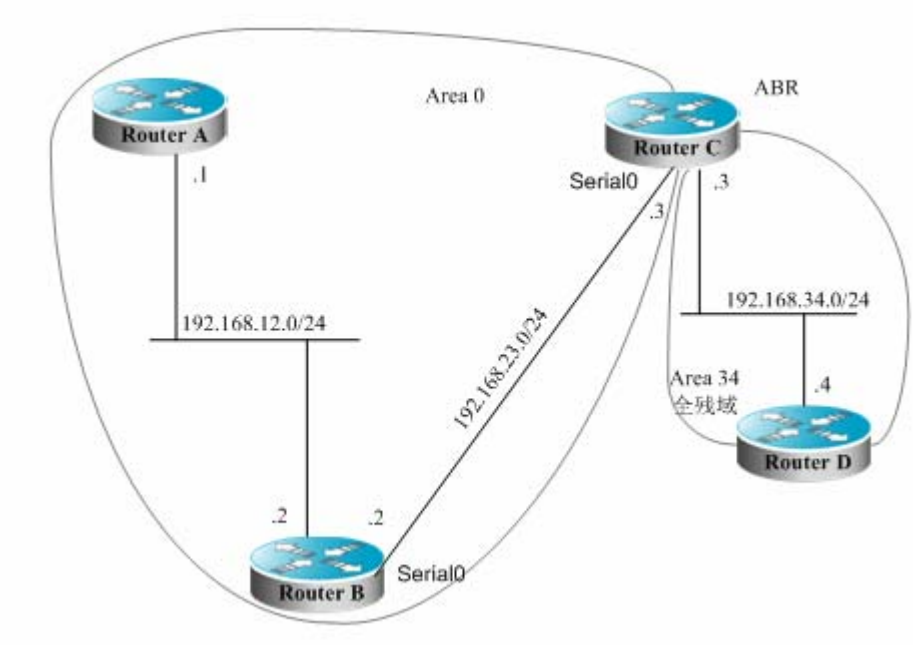
在设备 B 上看到 ospf 生成的路由如下，注意外部路由类型变为“E1”了。

```
O E1 200.200.1.0/24 [110/85] via 192.168.23.3, 00:00:33,
Serial1/0
O IA 192.168.34.0/24 [110/65] via 192.168.23.3, 00:00:33,
Serial1/0
O E1   172.200.1.0 [110/85] via 192.168.23.3, 00:00:33,
Serial1/0
```

30.4.6 OSPF 残域的配置例子

- 配置要求

有四台设备组成了一个 OSPF 路由域，网络 192.168.12.0/24、192.168.23.0/24 属于区域 0，网络 192.168.34.0/24 属于区域 34。具体 IP 地址分配和设备连接图见图 6。



OSPF 残域配置例子

要求经过配置 RouterD 的路由表中将只能看到 OSPF 缺省路由和本区域网络路由。

- 设备具体配置

只有在全残域中的设备，路由表才能达到最简化，没有到外部和区域间的路由。残域的配置需要在该区域的所有路由上配置。为了在设备 D 上可以看到区域间路由，设备 C 还通告了一个 192.168.30.0/24 的网络。

设备 A 的配置：

#配置以太网端口

```
interface FastEthernet0/0
ip address 192.168.12.1 255.255.255.0
```

#配置 OSPF 路由协议

```
router ospf 1
network 192.168.12.0 0.0.0.255 area 0
```

设备 B 的配置：

#配置以太网端口

```
interface FastEthernet0/0
ip address 192.168.12.2 255.255.255.0
```

#配置广域网端口

```
interface Serial1/0
ip address 192.168.23.2 255.255.255.0
```

#配置 OSPF 路由协议

```
router ospf 1
network 192.168.12.0 0.0.0.255 area 0
network 192.168.23.0 0.0.0.255 area 0
```

设备 C 的配置：

#配置以太网端口

```
interface FastEthernet0/0
ip address 192.168.34.3 255.255.255.0
```

#配置广域网端口

```
interface Serial1/0
ip address 192.168.23.3 255.255.255.0
```

#增加一个网络

```
interface Dialer10
ip address 192.168.30.1 255.255.255.0
```

配置 OSPF 路由协议

```
router ospf 1
network 192.168.23.0 0.0.0.255 area 0
network 192.168.34.0 0.0.0.255 area 34
network 192.168.30.0 0.0.0.255 area 34
area 34 stub no-summary
```

设备 D 的配置：

#配置以太网端口

```
interface FastEthernet0/0
ip address 192.168.34.4 255.255.255.0
```

#配置 OSPF 路由协议

```
router ospf 1
network 192.168.34.0 0.0.0.255 area 34
area 34 stub
```

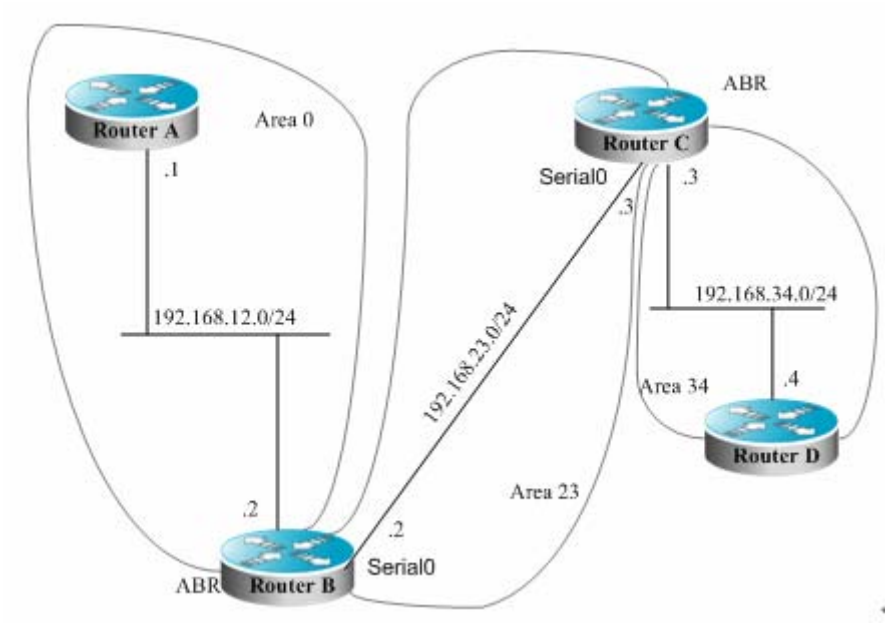
设备 D 上看到的 ospf 生成 的路由如下：

```
O 192.168.30.0/24 [110/1786] via 192.168.34.3, 00:00:03,
FastEthernet0/0
O*IA 0.0.0.0/0 [110/2] via 192.168.34.3, 00:00:03,
FastEthernet0/0
```

30.4.7 OSPF 虚拟链接配置例子

- 配置要求

有四台设备组成了一个 OSPF 路由域，网络 192.168.12.0/24 属于区域 0，网络 192.168.23.0/24 属于区域 23，网络 192.168.34.0/24 属于区域 34。具体 IP 地址分配和设备连接图见图 7。



OSPF 虚拟链接配置例子

通过配置使得设备 D 能够学到 192.168.12.0/24、192.168.23.0/24 的路由。

- 设备具体配置

OSPF 路由域由多个区域组成时，每个区域必须域骨干域（area 0）直接连接，如果没有直接连接，则必须创建虚拟链接，使得在逻辑上跟骨干域是直接相连的，否则区域间将不能连通。虚拟链接必须在 ABR 上进行配置。

设备 A 的配置：

#配置以太网端口

```
interface FastEthernet0/0
ip address 192.168.12.1 255.255.255.0
```

#配置 OSPF 路由协议

```
router ospf 1
network 192.168.12.0 0.0.0.255 area 0
```

设备 B 的配置：

#配置以太网端口

```
interface FastEthernet0/0
ip address 192.168.12.2 255.255.255.0
```

#配置广域网端口

```
interface Serial1/0
```

```
ip address 192.168.23.2 255.255.255.0
```

#增加回环 IP 地址作为 OSPF 设备标识符

```
interface Loopback2
ip address 2.2.2.2 255.255.255.0
```

#配置 OSPF 路由协议

```
router ospf 1
network 192.168.12.0 0.0.0.255 area 0
network 192.168.23.0 0.0.0.255 area 23
area 23 virtual-link 3.3.3.3
```

设备 C 的配置：

#配置以太网端口

```
interface FastEthernet0/0
ip address 192.168.34.3 255.255.255.0
```

#配置广域网端口

```
interface Serial1/0
ip address 192.168.23.3 255.255.255.0
```

#增加回环 IP 地址作为 OSPF 设备标识符

```
interface Loopback2
ip address 3.3.3.3 255.255.255.0
```

#配置 OSPF 路由协议

```
router ospf 1
network 192.168.23.0 0.0.0.255 area 23
network 192.168.34.0 0.0.0.255 area 34
area 23 virtual-link 2.2.2.2
```

设备 D 的配置：

#配置以太网端口

```
interface FastEthernet0/0
ip address 192.168.34.4 255.255.255.0
```

#配置 OSPF 路由协议

```
router ospf 1
network 192.168.34.0 0.0.0.255 area 34
```

设备 D 上看到的 ospf 生成的路由如下：

```
O IA 192.168.12.0/24 [110/66] via 192.168.34.3, 00:00:10,
FastEthernet0/0
O IA 192.168.23.0/24 [110/65] via 192.168.34.3, 00:00:25,
FastEthernet0/0
```

31 协议无关配置

31.1 配置 IP 路由

31.1.1 静态路由配置

静态路由就是手工配置的路由，使得数据包能够按照预定的路径传送到指定的目标网络。当不能通过动态路由协议学到一些目标网络的路由时，配置静态路由就会显得十分重要。通常可以给没有确切路由的数据包静态配置缺省路由。

要配置静态路由，在全局配置模式中执行以下命令：

命令	作用
Ruijie(config)**# ip route** [**vrf** *vrf_name*] *network mask* {*ip-address* | *interface-type interface-number* [*ip-address*]} [*distance*] [**tag** *tag*] [**permanent**] [**weight** *weight*] [**track** *object-number*]	配置静态路由
Ruijie(config)**# no ip route** *network mask*	删除静态路由
Ruijie(config)**# ip static route-limit** *number*	指定静态路由最大值
Ruijie(config)**# no ip static route-limit**	恢复静态路由默认最大值

静态路由的配置例子，请参见本章节的“动态路由覆盖静态路由例子” 。

如果没有执行删除动作，我司产品将永久保留静态路由。但是您可以用动态路由协议学到得更好路由来替代静态路由，更好的路由是指管理距离更小的距离，包括静态路由在内所有的路由都携带管理距离的参数。以下为我司产品各种来源路由的管理距离表：

路由来源	缺省管理距离值
直连网络	0
静态路由	1
OSPF 路由	110
ISIS 路由	115
RIP 路由	120
不可达路由	255

说明：

对于静态路由，如果需要由 RIP、OSPF 等动态路由协议通告，需要对动态路由协议配置重分布静态路由。

当一个接口属于“down”状态时，所有指向该接口的路由将全部从路由表中消失。另外，当我司产品找不到静态路由下一跳地址的转发路由时，该静态路由也会从路由表中消失。

指定 VRF 的静态路由添加到对应的 VRF 中，如果同时指定了出口，但是出口所属的 VRF 和指定 VRF 不符合，将不会添加成功，如果没有指定 VRF，则默认添加到全局路由表中。

静态路由的默认权重为 1，可以使用 **show ip route weight** 命令查看非默认权重的静态路由。权重参数 weight 用于实现 WCMP 功能，当存在负载均衡路由可到达某地址时，交换机会根据各条路由的权重值分配数据流量，weight 较大的路由会分担较多的数据报文，较小者会分担较少的数据报文。路由器的 WCMP 限制一般为 32，而交换机则由于各款芯片所能支持的权重不同，所以其 WCMP 限制数量与型号有关；具体型号的路由权重值，请参考产品规格书。

当负载均衡路由的权重之和大于 WCMP 限制时，超出限制的路由不会生效。例如，如果某设备上 WCMP 限制为 8，那么下面两条静态路由配置中，只有一条会生效：

```
Ruijie(config)#ip route 10.0.0.0 255.0.0.0 172.0.1.2 weight 6
Ruijie(config)#ip route 10.0.0.0 255.0.0.0 172.0.1.4 weight 6
Ruijie(config)#show ip route 10.0.0.0

Routing entry for 10.0.0.0/8
  Distance 1, metric 0
  Routing Descriptor Blocks:
  *172.0.1.2, generated by "static"
Ruijie(config)#show ip route weight

------------[distance/metric/weight]-----------
S    10.0.0.0/8 [1/0/6] via 172.0.1.2
```

静态路由默认最大值 1000，如果配置的静态路由数已经超过了指定的上限，则不会自动删除，但是添加不进来。

若要查看 IP 路由配置情况，可以使用 **show ip route** 命令查看 IP 路由表，可参见 RPI 命令参考手册。

31.1.2 缺省路由配置

不是所有的设备都有一张完整的全网路由表，为了使每台设备能够处理所有包的路由转发，通常的做法是功能强大的网络核心设备具有完整的路由表，其余的设备将

缺省路由指向核心设备。缺省路由可以通过动态路由协议进行传播，也可以在每台设备上进行手工配置。

缺省路由的产生有两种方法：1）手工配置缺省静态路由，具体配置参见上节“静态路由配置”；2）手工配置缺省网络。

多数的内部网关路由协议，都有一个将缺省路由传播到整个路由域的机制。要传播缺省路由的设备必须具有缺省路由。本节缺省路由的传播只适合 RIP 路由协议，RIP 向 RIP 路由域通告的缺省路由永远为“0.0.0.0”网络。OSPF 路由协议如何产生和传播缺省路由请参见“OSPF 路由协议配置指南”相关章节。

要产生缺省路由，在全局命令配置模式中执行以下命令：

命令	作用
Ruijie(config)# **ip default-network** network	配置缺省网络
Ruijie(config)# **no ip default-network** network	删除缺省网络

说明：

通过 **default-network** 产生缺省路由只要满足以下条件即可，即该缺省网络不是直连接口网络，但在路由表中可到达。

同样的条件下，RIP 也可以传播缺省路由，但是 RIP 传播缺省路由还有另外一个办法，那就是配置缺省静态路由，或者通过其它路由协议学到了 0.0.0.0/0 的路由。

当设备有缺省路由时，不管是动态路由协议学习到的还是手工配置产生的，当执行 **show ip route 命令**时，路由表中的“gateway of last resort”会显示最后网关的信息。一个路由表可能会有多条网络路由为候选缺省路由，但只有最好的缺省路由才能成为“gateway of last resort”。

31.1.3 等价路由条数配置

如果需要负载均衡的功能，可以通过设置等价路由的条数来控制，等价路由即通过不同路径到达同一目标地址的路由。当只有一条等价路由时，同一目标地址只能配置一条路由，则取消了负载均衡功能。

要配置等价路由条数，在全局配置模式中执行以下命令，该命令的 no 形式为恢复默认等价路由数。

该命令的配置对 ipv4 和 ipv6 同样生效，也就是说，配置了该命令，到达某一 ipv4 目的地的最大等价路径条目数是该配置值，到达某一 ipv6 目的地的最大等价路径条目数也是该配置值。

命令	作用

maximum-paths [*number*]	配置等价路由条数，不同的产品系列可配置的最大等价路径条目数不同。路由器均为 32 条，交换机请参考界面提示。

31.2 配置路由图

路由图(**route-map**)是与具体路由协议无关的一种过滤策略集，提供给路由协议和策略路由使用。在路由协议中，用于路由信息的过滤和修改；在策略路由中，用于控制报文转发。

定义路由图，在全局配置模式中执行以下命令：

命令	作用
Ruijie(config)**# route-map** *route-map-name* [**permit** | **deny**] *sequence*	定义路由图 *sequence* ：0-65535
Ruijie(config)**# no route-map** *route-map-name* {[**permit** | **deny**] *sequence*}	删除路由图

一个路由图规则配置中，可以执行一个或多个的 **match** 命令和一个或多个的 **set** 命令。如果没有 **match** 命令，则匹配所有；如果没有 **set** 命令，则不做任何操作。

要定义规则的匹配条件，在路由图配置模式中执行以下命令：

命令	作用
Ruijie(config-route-map)**# match community** {*standard-list-number* | *expanded-list-number* | *community-list-name*}	匹配 BGP 路由的团体属性
Ruijie(config-route-map)**# match interface** *interface-type interface-number*	匹配路由的下一跳接口
Ruijie(config-route-map)**# match ip address** *Access-list-number* [*…access-list-number*]	匹配访问列表中的地址
Ruijie(config-route-map)**# match ip next-hop** *access-list-number* [*…access-list-number*]	匹配访问列表中的下一跳地址
Ruijie(config-route-map)**# match ip route-source** *access-list-number* [*…access-list-number*]	匹配访问列表中的路由源地址
Ruijie(config-route-map)**# match ipv6 address** { *access-list-name* | **prefix-list** *prefix-list-name* }	匹配 IPv6 访问列表或前缀列表

Ruijie(config-route-map)# **match ipv6 next-hop** { *access-list-name* | **prefix-list** *prefix-list-name* }	匹配访问列表或前缀列表的下一跳地址
Ruijie(config-route-map)# **match ipv6 route-source** { *access-list-name* | **prefix-list** *prefix-list-name* }	匹配访问列表或前缀列表的路由源地址
Ruijie(config-route-map)# **match metric** *Metric*	匹配路由的量度值 *Metric* ：0-4294967295
Ruijie(config-route-map)# **match origin** {**egp** | **igp** | **incomplete**}	匹配路由来源的类型
Ruijie(config-route-map)# **match route-type** {**local** | **internal** | **external** [**level-1** | **level-2**]}	匹配路由的类型
Ruijie(config-route-map)# **match tag** *tag*	匹配路由的标记值 *tag* ：0-4294967295

要定义匹配后的操作，在路由图配置模式中执行以下命令：

命令	作用
Ruijie(config-route-map)# **set aggregator as** *as-num ip_addr*	设置路由的聚合者的 AS 属性值
Ruijie(config-route-map)# **set as-path prepend** *as-number*	设置 AS_PATH 属性值
Ruijie(config-route-map)# **set comm-list** *community-list-number* | *community-list-name* **delete**	设置删除 COMMUNITY_LIST 中的所有的 community 属性值
Ruijie(config-route-map)# **set community** {*community-number*[*community-number …*] **additive** | **none**}	设置 COMMUNITY 属性值
Ruijie(config-route-map)# **set dampening** *half-life reuse suppress max-suppress-time*	设置路由的路由振荡参数
Ruijie(config-route-map)# **set extcommunity** {**rt** *extend-community-value* | **soo** *extend-community-value*}	设置扩展团体属性值
Ruijie(config-route-map)# **set interface** *interface-type interface-number*	设置报文转发接口
Ruijie(config-route-map)# **set ip default next-hop** *ip-address*	设置默认下一跳 IP 地址
Ruijie(config-route-map)# **set ip next-hop** *ip-address*	设置下一跳 IP 地址

Ruijie(config-route-map)# **set ip next-hop verify-availability** *ip-address* **track** *track-object-num*	设置确认下一跳 IP 地址的可达性
Ruijie(config-route-map)# **set level** {**stub-area** | **backbone** | **level-1** | **level-1-2** | **level-2**}	设置输入路由的区域
Ruijie(config-route-map)# **set local-preference** *number*	设置 LOCAL_PREFERENCE 值
Ruijie(config-route-map)# **set metric** *metric*	设置重分布路由的量度值
Ruijie(config-route-map)# **set metric** [+ *metric-value* | **-** *metric-value* | *metric-value*]	设置重分布路由的类型
Ruijie(config-route-map)# **set metric-type** {**type-1** | **type-2 | external** | **internal**}	设置重分布路由的类型
Ruijie(config-route-map)# **set next-hop** *next-hop*	设置重分布路由的下一跳 *next-hop*：下一跳的 IP 地址
Ruijie(config-route-map)# **set origin** {**egp** | **igp** | **incomplete**}	设置路由来源属性
Ruijie(config-route-map)#**set originator-id** *ip-addr*	设置路由源发属性
Ruijie(config-route-map)# **set tag** *tag*	设置重分布路由的标记值
Ruijie(config-route-map)# **set weight** *number*	设置 BGP 路由权重值

路由图的 **match** 命令与 **set** 命令对路由图不同应用的支持情况不同，为了方便用户了解 **match** 命令、**set** 命令是否适用于当前应用，我们将在如下情况下给予用户提示信息：

- 配置关联 **route-map** 的命令时，检查 **route-map** 所配置的 **match** 命令与 **set** 命令在当前所关联应用的适用情况，存在不适用情况时给予用户提示信息。
- 配置 **route-map**、**match** 命令或者 **set** 命令时，检查该 **route-map** 关联的所有应用对于 **route-map** 配置的所有 **match** 命令与 **set** 命令的适用情况，存在不适用情况时给予用户提示信息。

命令不适用时给予用户提示信息的具体例子可以参看下文“路由图应用”中所举的例子。

注意：

目前策略路由(PBR)关联路由图的应用还不支持上述信息提示功能。

31.3 路由重分布

31.3.1 路由重分布配置

为了支持本设备能够运行多个路由协议进程，我司产品提供了路由信息从一个路由进程重分布到另外一个路由进程的功能。比如您可以将 OSPF 路由域的路由重新分布后通告到 RIP 路由域中，也可以将 RIP 路由域的路由重新分布后通告到 OSPF 路由域中。路由的相互重分布可以在所有的 IP 路由协议之间进行。

在路由重分布中，经常通过路由图（route maps）的应用，对两个路由域之间的路由相互分布进行有条件的控制。

要把路由从一个路由域分布到另一个路由域，并且进行控制路由重分布，在路由进程配置模式中执行以下命令：

命令	作用
Ruijie(config-router)# **redistribute** *protocol [process-id]* [**metric** *metric*] [**metric-type** *metric-type*] [**match internal** | **external** *type* | **nssa-external** *type*] [**tag** *tag*] [**route-map** *route-map-name*] [**subnets**]	进行重分布路由 *protocol*（协议类型）：bgp、connected、isis、rip、static
Ruijie(config-router)# **default-metric** *metric*	给所有重分布路由设置缺省量度值 *metric*

路由重分布很容易造成路由环路，在使用时应该谨慎。

说明：

OSPF 路由进程中配置路由重分布时，缺省情况下，赋予重分布路由的量度值为 20，类型为 Type-2，该类型路由属于 OSPF 最不可信的路由。

31.3.2 缺省路由分发配置

为了通告缺省路由，路由协议需要将缺省路由引入进程，或者强制生成一条缺省路由。

要对缺省路由进行分发设置，在路由进程配置模式中执行以下命令：

命令	作用

Ruijie(config-router)# **default-information originate** [**always**] [**metric** *metric*] [**metric-type** *type*] [**route-map** *map-name*]	将缺省路由引入路由协议进程并进行通告。 **always(可选)**：无论本地路由表是否存在缺省路由，都引入一条缺省路由。 **metric(可选)**：对引入的缺省路由的 metric 值进行设置 **metric-type(可选)**：设置 OSPF 引入的缺省路由类型 **route-map(可选)**：对引入缺省路由进行过滤和设置
Ruijie(config-router)# **no default-information originate** [**always**] [**metric** *metric*] [**metric-type** *type*] [**route-map** *map-name*]	取消将缺省路由引入路由协议进程并进行通告。

31.3.3 路由过滤配置

路由过滤就是对进出站路由进行控制，使得设备只学到必要、可预知的路由，对外只向可信任的设备通告必要的、可预知的路由。路由的泄漏和混乱，是会影响网络运行的，因此特别是电信运营商和金融业务网络，是十分有必要配置路由过滤的。

31.3.3.1 控制路由更新通告

为了防止本地网络上的其它路由设备学到不必要的路由信息，可以通过控制路由更新通告来遏制指定路由的更新。

要遏制路由更新通告，在路由进程配置模式中执行以下命令:

命令	作用
Ruijie(config-router)# **distribute-list** {[*access-list-number* | *access-list-name*] **prefix** *prefix-list-name*} **out** [*interface-type interface-number*]	根据访问列表规则，允许或拒绝某些路由被分发出去。 **prefix**：本关键字表明用前缀列表来过滤路由；前缀列表需要另外通过 ip prefix-list 配置。
Ruijie(config-router)# **no distribute-list** {[*access-list-number* | *access-list-name*] | **prefix** *prefix-list-name* } **out** [*interface-type interface-number* | *protocol*]	取消遏制路由更新通告

说明:

配置 OSPF 时，不能指定接口，而且该特性只适用于 OSPF 路由域的外部路由。

31.3.3.2 控制路由更新处理

为了避免处理进站路由更新报文的某些指定路由，可以配置该特性。该特性不适用于 OSPF 路由协议。

要控制路由更新处理，在路由进程配置模式中执行以下命令：

命令	作用
Ruijie(config-router)# **distribute-list** {[*access-list-number* | *access-list-name*] | **prefix** *prefix-list-name* [**gateway** *prefix-list-name*] | **gateway** *prefix-list-name*} **in** [*interface-type interface-number*]	根据访问列表规则，允许或拒绝接收分发进来的指定的路由。 **prefix**：本关键字表明用前缀列表来过滤路由；前缀列表需要另外通过 ip prefix-list 配置。 **gateway**：使用前缀列表根据路由的源进行对分发进来的路由进行过滤。
Ruijie(config-router)# **no distribute-list** {[*access-list-number* | *name*] | **prefix** *prefix-list-name* [**gateway** *prefix-list-name*] | **gateway** *prefix-list-name* } **in** [*interface-type interface-number*]	取消控制路由更新处理

31.4 配置范例

31.4.1 路由图应用

路由图的配置十分灵活，可以应用在路由重分布和策略路由的配置上。不管如何应用路由图，其配置原理是一样的，只是适用的命令集不同而已。即使同样应用在路由重分布上，不同的路由协议应用路由图，其能够使用的命令也不尽相同。当路由图应用不支持所配置的路由图规则时，会看到相应的提示信息。

以下的配置例子，是 OSPF 路由协议重分布 RIP 路由，要求只重分布跳数为 4 的 RIP 路由，在 OSPF 路由域中，该路由的类型为外部路由 type-1，初始量度值为 40，路由标记值设为 40。

配置 OSPF

```
Ruijie(config)# router ospf 1
Ruijie(config-router)# redistribute rip subnets route-map redrip
```

```
Ruijie(config-router)# network 192.168.12.0 0.0.0.255 area 0
```

配置访问控制列表

```
Ruijie(config)# access-list 20 permit 200.168.23.0
```

配置路由图

```
Ruijie(config)# route-map redrip permit 10
Ruijie(config-route-map)# match metric 4
Ruijie(config-route-map)# set metric 40
Ruijie(config-route-map)# set metric-type type-1
Ruijie(config-route-map)# set tag 40
```

以下配置例子，是 RIP 路由协议重分布 OSPF 路由，要求只重分布标记为 10 的 OSPF 路由，初始量度值设为 10。

配置 RIP

```
Ruijie(config)# router rip
Ruijie(config-router)# version 2
Ruijie(config-router)# redistribute ospf 1 route-map redospf
Ruijie(config-router)# network 200.168.23.0
```

配置路由图

```
Ruijie(config)# route-map redospf permit 10
Ruijie(config-route-map)# match tag 10
Ruijie(config-route-map)# set metric 10
```

以下的配置例子，是 OSPF 路由协议重分布 RIP 路由，由于路由图配置了该应用不支持的规则，所以在配置重分布关联该路由图时，会看到信息提示该应用不支持相应的规则：

配置路由图

```
Ruijie(config)# route-map redrip permit 10
Ruijie(config-route-map)# match length 1 3
Ruijie(config-route-map)# match route-type external
Ruijie(config-route-map)# set level backbone
```

配置 OSPF

```
Ruijie(config)# router ospf 1
Ruijie(config-router)# redistribute rip subnets route-map
redrip

% ospf redistribute rip not support match length
% ospf redistribute rip not support match route-type
% ospf redistribute rip not support set level backbone
```

31.4.2 静态路由重分布

- **配置要求**

一台设备，通过 RIP 与其它设备交换路由信息，另外还有 3 条静态路由，要求 RIP 重新分布静态路由，并且只允许通告 172.16.1.0/24、192.168.1.0/24 两条路由。

- **路由设备具体配置**

这也是在实际应用中比较常见的路由过滤配置例子，通过配置分布列表来实现。另外注意以下配置中没有指定重分布路由的量度值，因为此处重分布的路由为静态路由，RIP 会自动分配量度值。RIP 配置中的版本指定与关闭自动汇总是必须的，因为访问列表中允许的路由是 172.16.1.0/24 的路由，RIP 要对外通告该路由，首先得支持无类路由的存在，而且在通告时不能被汇总成 172.16.0.0/16 网络。

```
# 配置静态路由
Ruijie(config)# ip route 172.16.1.0 255.255.255.0 172.200.1.2
Ruijie(config)# ip route 192.168.1.0 255.255.255.0 172.200.1.2
Ruijie(config)# ip route 192.168.2.0 255.255.255.0 172.200.1.4

# 配置 RIP
Ruijie(config)# router rip
Ruijie(config-router)# version 2
Ruijie(config-router)# redistribute static
Ruijie(config-router)# network 192.168.34.0
Ruijie(config-router)# distribute-list EXT_ACL out static
Ruijie(config-router)# no auto-summary

# 配置扩展 ACL
Ruijie(config)# ip access-list extended EXT_ACL
Ruijie(config-ext-nacl)#10 permit ip 192.168.1.0 0.0.0.255
any
Ruijie(config-ext-nacl)#10 permit ip 172.16.1.0 0.0.0.255 any
```

31.4.3 动态路由协议重分布

- **配置要求**

有四台设备，设备连接图见图 1。路由设备 A 属于 OSPF 路由域，路由设备 C 属于 RIP 路由域，路由设备 D 属于 BGP 路由域，路由设备 B 连接三个路由域。路由设备 A 通告 192.168.10.0/24、192.168.100.1/32 两条路由，路由设备 C 通告 192.168.3.0/24、192.168.30.0/24 两条路由，路由设备 D 通告 192.168.4.0/24、192.168.40.0/24 两条。

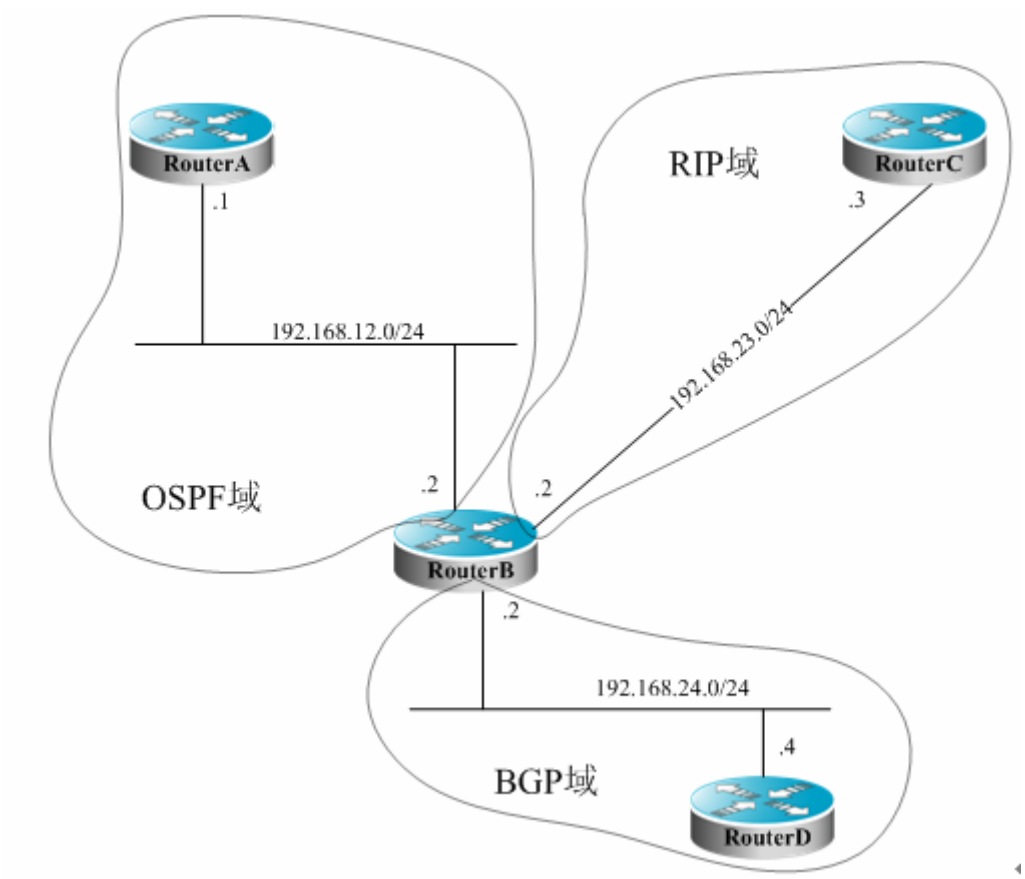


图 1　动态路由协议重分布

在设备 Router B 上，OSPF 重分布 RIP 路由域中的路由，路由类型为 Type-1，并重分布 BGP 路由域中携带团体属性 11:11 的 BGP 路由；RIP 重分布 OSPF 路由域中的 192.168.10.0/24 一条路由，而且量度值设为 3，并且通告一条默认路由到 RIP 域。

- **路由设备具体配置**

路由协议之间进行重分布路由，简单的路由过滤可以由分布列表来控制，但是要对不同的路由设置不同的属性，分布列表是做不到的，需要配置路由图进行控制。路由图比分布列表提供了更多的控制功能，但是配置也相对复杂，一般情况下可以不用路由图的尽量不用，使得设备配置简单化。该例子的配置应用路由图对 BGP 路由的团体属性进行匹配。

路由设备 A 的配置：

```
# 配置网络接口
Ruijie(config)# interface gigabitEthernet 0/0
Ruijie(config-if)# ip address 192.168.10.1 255.255.255.0
Ruijie(config)# interface loopback 1
Ruijie(config-if)# ip address 192.168.100.1 255.255.255.0
Ruijie(config-if)# no ip directed-broadcast
Ruijie(config)# interface gigabitEthernet 0/1
Ruijie(config-if)# ip address 192.168.12.55 255.255.255.0
```

```
# 配置 OSPF
Ruijie(config)# router ospf 12
Ruijie(config-router)# network 192.168.10.0 0.0.0.255 area 0
Ruijie(config-router)# network 192.168.12.0 0.0.0.255 area 0
Ruijie(config-router)# network 192.168.100.0 0.0.0.255 area 0
```

路由设备 B 的配置：

```
# 配置网络接口
Ruijie(config)# interface gigabitEthernet 0/0
Ruijie(config-if)# ip address 192.168.12.5 255.255.255.0
Ruijie(config)# interface Serial 1/0
Ruijie(config-if)# ip address 192.168.23.2 255.255.255.0

# 配置 OSPF，设置重分布路由的类型
Ruijie(config)# router ospf 12
Ruijie(config-router)# redistribute rip metric 100 metric-type
1 subnets
Ruijie(config-router)# redistribute bgp route-map ospfrm
subnets
Ruijie(config-router)# network 192.168.12.0 0.0.0.255 area 0

# 配置 RIP，分布列表过滤重分布路由
Ruijie(config)# router rip
Ruijie(config-router)# redistribute ospf 12 metric 2
Ruijie(config-router)# network 192.168.23.0
Ruijie(config-router)# distribute-list 10 out ospf
Ruijie(config-router)# default-information originate always
Ruijie(config-router)# no auto-summary

# 配置 BGP
Ruijie(config)# router bgp 2
Ruijie(config-router)# neighbor 192.168.24.4 remote-as 4
Ruijie(config-router)# address-family ipv4
Ruijie(config-router-af)# neighbor 192.168.24.4 activate
Ruijie(config-router-af)# neighbor 192.168.24.4
 send-community

# 配置路由图
Ruijie(config)# route-map ospfrm
Ruijie(config-route-map)# match community cl_110

# 定义访问列表
Ruijie(config)# access-list 10 permit 192.168.10.0

# 定义 community 列表
Ruijie(config)# ip community-list standard cl_110 permit 11:11
```

路有器 C 的配置：

```
# 配置网络接口
```

```
Ruijie(config)# interface gigabitEthernet 0/0
Ruijie(config-if)# ip address 192.168.30.1 255.255.255.0
Ruijie(config)# interface gigabitEthernet 0/1
Ruijie(config-if)# ip address 192.168.3.1 255.255.255.0
Ruijie(config)# interface Serial 1/0
Ruijie(config-if)# ip address 192.168.23.3 255.255.255.0
```

配置 RIP

```
Ruijie(config)# router rip
Ruijie(config-router)# network 192.168.23.0
Ruijie(config-router)# network 192.168.3.0
Ruijie(config-router)# network 192.168.30.0
```

路有器 D 的配置：

配置网络接口

```
Ruijie(config)# interface gigabitEthernet 0/0
Ruijie(config-if)# ip address 192.168.40.1 255.255.255.0
Ruijie(config)# interface gigabitEthernet 0/1
Ruijie(config-if)# ip address 192.168.4.1 255.255.255.0
Ruijie(config)# interface gigabitEthernet 1/0
Ruijie(config-if)# ip address 192.168.24.4 255.255.255.0
```

配置 BGP

```
Ruijie(config)# router bgp 4
Ruijie(config-router)# neighbor 192.168.24.2 remote-as 2
Ruijie(config-router)# redistribute connected route-map bgprm
Ruijie(config-router)# address-family ipv4
Ruijie(config-router-af)# neighbor 192.168.24.2 activate
Ruijie(config-router-af)# neighbor 192.168.24.2
 send-community
```

配置路由图

```
Ruijie(config)# route-map bgprm
Ruijie(config-route-map)# set community 22:22
```

路由设备 A 看到的 OSPF 路由:

```
O E1 192.168.30.0/24 [110/101] via 192.168.12.5, 00:04:07,
FastEthernet0/1
O E1 192.168.3.0/24 [110/101] via 192.168.12.5, 00:04:07,
FastEthernet0/1
```

路由设备 C 看到的 RIP 路由：

```
R   0.0.0.0/0 [120/1] via 200.168.23.2, 00:00:00, Serial1/0
R   192.168.10.0/24 [120/2] via 200.168.23.2, 00:00:00,
Serial1/0

!
```

31.5 交换机快转 ECMP/WCMP 策略配置

在交换机中硬件转存在 ECMP/WCMP 路由时，同样也存在负载均衡策略。当该路由存在多个下一跳时，硬件能够依据我们设定的策略来选择其中一个下一跳。交换机会可以根据我们的设置选取报文的不同字段作为关键字，将这些关键字作为输入通过 hash，选取相应的一跳。通过选取合适的报文特征字段和 hash 算法可以让报文的出口流量更均衡。

说明：

S3760 系列交换机中如果两个路由的下一跳完全一样（所有下一跳都一致，包括 weight），则两个路由共享下一跳。这样可以节约硬件资源，但在删除路由或者更新路由的时候，可能由于硬件资源不足导致删除路由失败或者更新路由失败，从而删除掉整条路由。您可以通过查看提示信息了解到这种情况的发生。提示信息格式如下所示：

*Aug 24 11:20:39: %7: Warning: Because hardware resource is not enough, route:
*Aug 24 11:20:39: %7: 191.168.200.120/32 191.168.200.120
*Aug 24 11:20:39: %7: were deleted

31.5.1 Hash 关键字

S3760 的 hash 关键字有：

- MAC SA [5:0]
- MAC DA [5:0]
- SIP [22:16] [6:0]
- DIP [22:16] [6:0]

S3760 的负载均衡策略是不可配置的。

31.5.2 Hash 算法的选择

S3760 的 hash 算法不可选择。

31.5.3 配置命令

命令	作用

Ruijie(config)# **aggregateport load-balance** { **src-dst-mac** | **src-dst-ip** }	设置 ECMP/WCMP 的负载均衡方式，选择使用的算法： **src-dst-ip**：根据源 IP 与目的 IP 进行流量分配。不同的源 IP——目的 IP 对的流量通过不同的下一跳，同一源 IP——目的 IP 对通过相同的下一跳。 **src-dst-mac**：根据源 MAC 与目的 MAC 进行流量分配。不同的源 MAC——目的 MAC 对的流量通过不同的下一跳，同一源 MAC——目的 MAC 对通过相同的下一跳。

31.6 配置举例

下面配置将配置 hash 算法设置为 ip：

```
Ruijie(config)# aggregateport load-balance src-dst-ip
```

32 策略路由配置

策略路由提供了一种比基于目的 IP 地址进行路由转发更加灵活的数据包路由转发机制。接口应用了策略路由，设备将通过路由图决定如何处理从这个接口上收到的数据报文。

应用策略路由，必须要指定策略路由使用的路由图，并且要创建路由图。一个路由图由很多条策略组成，每个策略都定义了 1 个或多个的匹配规则和对应操作。一个接口应用策略路由后，将对该接口接收到的所有包进行检查，不符合路由图任何策略的数据包将按照通常的路由转发进行处理，符合路由图中某个策略的数据包就按照该策略中定义的操作进行处理。路由图的配置详见协议无关命令配置指导。

配置一个策略路由分为以下几个步骤:

1. 定义路由图，一个路由图可以由好多策略组成，策略按序号大小排列，只要符合了前面策略，就退出路由图的执行；

要定义重分布路由图，在全局配置模式中执行以下命令：

命令	作用
Ruijie(config)# **route-map** *route-map-name* [**permit** | **deny**] *sequence*	定义路由图
Ruijie(config)# **no route-map** *route-map-name* {[**permit** | **deny**] *sequence*}	删除路由图

2. 定义路由图每个策略的匹配规则或条件；

要定义策略的匹配规则，在路由图配置模式中执行以下命令：

命令	作用
Ruijie(config-route-map)# **match ip address** *access-list-number*	匹配访问列表中的地址
Ruijie(config-route-map)# **match length** *min max*	匹配报文的长度

3. 定义满足匹配规则后，设备的操作；

要定义匹配规则后的操作，在路由图配置模式中执行以下命令：

命令	作用
Ruijie(config-route-map)# **set ip default next-hop** *ip-address*[*weight*][*ip-address*[*weight*]]	为路由表中没有明确路由的数据分组指定下一跳 IP 地址

Ruijie(config-route-map)**# set ip next-hop** *ip-address* [*weight*][*ip-address*[*weight*]]	设置数据包的下一跳 IP 地址

4. 在指定接口中应用路由图。

要在接口上应用策略路由,在接口模式下执行以下命令:

命令	作用
Ruijie(config-if)**# ip policy route-map** *name*	在接口上使用指定的 route-map 进行过滤
Ruijie(config-if)**# no ip policy route-map**	在接口上取消应用的 route-map

例如:

在 **fastethernet** 0/1 口上配置策略路由,使得所有进入的报文都转发到下一跳为 192.168.5.5 的设备

```
Ruijie(config)# access-list 1 permit any
Ruijie(config)# route-map name
Ruijie(config-route-map)# match ip address 1
Ruijie(config-route-map)# set ip next-hop 192.168.5.5
Ruijie(config-route-map)# exit
Ruijie(config-route-map)# int fastethernet 0/1
Ruijie(config-if)# ip policy route-map name
```

在策略路由中配置负载均衡或者冗余备份模式，全局模式下使用命令：

命令	作用
Ruijie(config)**# ip policy {load-balance | Redundance}**	设置策略路由的转发是冗余备份还是负载均衡
Ruijie(config)**# no ip policy**	恢复策略路由的负载分担模式

策略路由执行负载均衡时，WCMP(Weighted Cost Multiple Path)最多支持 4 个下一跳，ECMP(Equal Cost Multiple Path)最多支持 32 个下一跳。

配置默认策略路由时，WCMP 最多支持 4 个下一跳，ECMP 最多支持 32 个下一跳。

route-map 配置相关命令详见《配置协议无关命令》

交换机上支持的命令：

1. **[no] ip policy route-map**
2. **match ip address**

3. **set ip next-hop**

4. **set ip default next-hop**

注意:

1. 我司产品上一个接口最多只能配置一个路由图，在同一个接口上多次配置路由图会相互覆盖，并且在策略路由中只应用子路由图(route-map sequence)中最先配置的一个 ACL。所以在使用策略路由时，每个子路由图建议只配置一个 ACL；

2. 如果配置的子路由图中只有 nexthop 而没有配置 ACL，则等价于所有报文都匹配；如果子路由图中只有 ACL 而没有 nexthop 则匹配的报文普通转发；如果子路由图中即没有 ACL 也没有 nexthop,则等价所有报文普通转发。

3. 策略路由只支持配置 ACL 号,不支持 ACL 名字配置，如果配置了 ACL 号但是该 ACL 不存在，等价所有报文都匹配，如果配置了 ACL 号但是其中没有任何 ACE，则跳过该子路由图，从下一个子路由图的 ACL 开始匹配；

4. 交换机上，如果 IP 报文匹配 deny 类型的 ACE，那么对该报文执行普通转发；如果 IP 报文匹配 deny any any 的 ACE（每一个 ACL 的最后都有一个默认的 deny any any 的 ACE），那么跳到下一个路由子图重新开始匹配；路由器上，如果 IP 报文匹配 deny 类型的 ACE 且该路由子图还有其它的 ACL，那么就跳到该路由子图的下一个 ACL 继续匹配，如果该路由子图没有其它的 ACL 了则跳到下一个路由子图的 ACL 进行匹配；依此进行直到匹配 permit 类型的 ACE，然后执行路由子图对应的 set 规则；路由图的所有 ACL 都不匹配则执行普通转发。

5. 如果用户希望发往本机的 IP 报文不使用策略路由，则用户需要在 PBR 规则中在 ACL 前面手工添加”deny 设备 IP 地址”的 ACE。

6. 工作在冗余备份模式下时，第一个解析的 nexthop 生效，如果所有的 nexthop 都没有解析，则设置 drop 行为，如果第一个 nexthop 先不解析后来打通了，则也会生效。

33 IPv6 配置

33.1 IPv6 相关知识

随着 Internet 的迅速增长以及 IPv4 地址空间的逐渐耗尽，IPv4 的局限性就越来越明显。对新一代互联网络协议（Internet Protocol Next Generation - IPng）的研究和实践已经成为热点，Internet 工程任务工作小组(IETF)的 IPng 工作组确定了 IPng 的协议规范，并称之为"IP 版本 6"（IPv6），该协议的规范在 RFC2460 中有详细的描述。

IPv6 的主要特点：

- 更大的地址空间

地址长度由 IPv4 的 32 位扩展到 128 位，即 Ipv6 有 2^128-1 个地址，IPv6 采用分级地址模式，支持从 Internet 核心主干网到企业内部子网等多级子网地址分配方式。

- 简化了报头格式

新 IPv6 报文头的设计原则是力图将报文头开销降到最低，因此将一些非关键性字段和可选字段从报文头中移出，放到扩展的报文头中，虽然 IPv6 地址长度是 IPv4 的四倍，但包头仅为 IPv4 的两倍。改进的 IPv6 报文头在设备转发时拥有更高的效率，例如 IPv6 报文头中没有校验和，IPv6 设备在转发中不需要去处理分片(分片由发起者完成)。

- 高效的层次寻址及路由结构

IPv6 采用聚合机制，定义非常灵活的层次寻址及路由结构，同一层次上的多个网络在上层设备中表示为一个统一的网络前缀，这样可以显著减少设备必须维护的路由表项，这也大大降低了设备的选路和存储开销。

- 简单的管理：即插即用

通过实现一系列的自动发现和自动配置功能，简化网络节点的管理和维护。比如邻接节点发现（Neighbor Discovery）、最大传输单元发现（MTU Discovery）、路由器通告（Router Advertisement）、路由器请求（Router Solicitation）、节点自动配置（Auto-configuration）等技术就为即插即用提供了相关的服务。特别要提到的是 IPv6 支持全状态和无状态两种地址配置方式，在 IPv4 中，动态主机配置协议 DHCP 实现了主机 IP 地址及其相关配置的自动设置，IPv6 承继 IPv4 的这种自动配置服务，并将其称为全状态自动配置(Stateful Autoconfiguration)。除了全状态自动配置，IPv6 还采用了一种被称为无状态自动配置（Stateless Autoconfiguration）的自动配置服务。在无状态自动配置过程中，主机自动获得链路本地地址、本地设备的地址前缀以及其它一些相关的配置信息。

- 安全性

IPSec 是 IPv4 的一个可选扩展协议，但是在 IPv6 中它是 IPv6 的一个组成部分，

用于提供 IPv6 的安全性。目前，IPv6 实现了认证头（Authentication Header，AH）和封装安全载荷（Encapsulated Security Payload，ESP）两种机制。前者实现数据的完整性及对 IP 包来源的认证，保证分组确实来自源地址所标记的节点；后者提供数据加密功能，实现端到端的加密。

- 更好的 QoS 支持

IPv6 包头的新字段定义了数据流如何识别和处理。IPv6 包头中的流标识（Flow Label）字段用于识别数据流身份，利用该字段，IPv6 允许用户对通信质量提出要求。设备可以根据该字段标识出同属于某一特定数据流的所有包，并按需对这些包提供特定的处理。

- 用于邻居节点交互的新协议

IPv6 的邻居发现协议（Neighbor Discovery Protocol）使用一系列 IPv6 控制信息报文（ICMPv6）来实现相邻节点（同一链路上的节点）的交互管理。邻居发现协议以及高效的组播和单播邻居发现报文替代了以往基于广播的地址解析协议 ARP、ICMPv4 路由器发现等报文。

- 可扩展性

IPv6 特性具有很强的可扩展性，新特性可以添加在 IPv6 包头之后的扩展包头中。不象 IPv4，包头最多只能支持 40 字节的的可选项，IPv6 扩展包头的大小仅受到整个 IPv6 包最大字节数的限制。

目前已经实现的 IPv6 支持下列特性：

- IPv6 协议
- IPv6 地址格式
- IPv6 地址类型
- ICMPv6
- IPv6 邻居发现
- 路径 MTU 发现
- ICMPv6 重定向
- 地址冲突检测
- IPv6 无状态自动配置
- IPv6 地址配置
- IPv6 路由转发，支持静态路由配置
- 配置 IPv6 协议的各种参数
- 诊断工具 Ping IPv6

33.1.1 IPv6 地址格式

IPv6 地址的基本表达方式是 X : X : X : X : X : X : X : X，其中 X 是一个 4 位十六进制整数(16 位)。每一个数字包含 4 个比特，每个整数包含 4 个十六进制数字，每个地址包括 8 个整数，一共 128 位。下面是一些合法的 IPv6 地址：

2001:ABCD:1234:5678:AAAA:BBBB:1200:2100
800 : 0 : 0 :0 : 0 : 0 : 0 : 1
1080 : 0 : 0 : 0 : 8 : 800 : 200C : 417A

这些整数是十六进制整数，其中 A 到 F 表示的是 10 到 15。地址中的每个整数都必须表示出来，但起始的 0 可以不必表示。某些 IPv6 地址中可能包含一长串的 0 (就像上面的第二和第三个例子一样)。当出现这种情况时，允许用“：：”来表示这一长串的 0。即地址 800 : 0 : 0 : 0 : 0 : 0 : 0 : 1 可以被表示为：800 :: 1

这两个冒号表示该地址可以扩展到一个完整的 128 位地址。在这种方法中，只有当 16 位组全部为 0 时才会被两个冒号取代，且两个冒号在地址中只能出现一次。

在 IPv4 和 IPv6 的混合环境中还有一种混合的表示方法。IPv6 地址中的最低 32 位可以用于表示 IPv4 地址，该地址可以按照一种混合方式表达，即 X : X : X : X : X : X : d . d . d . d，其中 X 表示一个 1 6 位整数，而 d 表示一个 8 位的十进制整数。例如，地址 0 : 0 : 0 : 0 : 0 : 0 : 192 .168 . 20 : 1 就是一个合法的 IPv6 地址。使用简写的表达方式后，该地址也可以表示为：: : 192 .168 . 20 . 1

由于 IPv6 地址被分成两个部分：子网前缀和接口标识符，因此可以按照类似 CIDR 地址的方式被表示为一个带额外数值的地址，其中该数值指出了地址中有多少位是代表网络部分(网络前缀)，即 IPv6 节点地址中指出了前缀长度，该长度与 IPv6 地址间以斜杠区分，例如：12AB::CD30:0:0:0:0/60，这个地址中用于选路的前缀长度为 60 位。

33.1.2 IPv6 地址类型

RFC2373 中定义了三种 IPv6 地址类型：

- 单播(Unicast)：一个单接口的标识符。送往一个单播地址的包将被传送至该地址标识的接口上。
- 泛播(Anycast)：一组接口的标识符。送往一个泛播地址的包将被传送至该地址标识的接口之一(根据选路协议选择“最近”的一个)。
- 组播(Multicast)：一组接口(一般属于不同节点)的标识符。送往一个组播地址的包将被传送至加入该组播地址的所有接口上。

注意：

在 IPv6 中已经没有定义广播地址。

下面逐一介绍这几类地址：

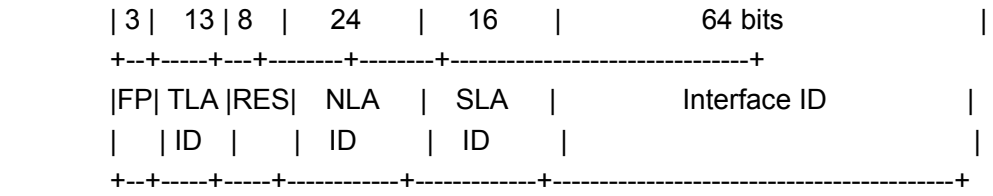
33.1.2.1 单播地址（Unicast Addresses）

IPv6 单播地址包括下面几种类型：

- 可聚集全球地址
- 链路本地地址
- 站点本地地址
- 嵌有 IPv4 地址的 IPv6 地址

1. 可聚集全球地址

可聚集全球单播地址的格式如下：



图中包括下列字段：

- FP 字段(Format Prefix)：

IPv6 地址中的格式前缀，占 3 个比特，用来标识该地址在 IPv6 地址空间中属于哪类地址。该字段为’0 0 1’，表示这是可聚集全球单播地址。

- TLA ID 字段(Top-Level Aggregation Identifier)：

顶级聚集标识符，包含最高级地址选路信息。指的是网络互连中最大的选路信息。占 1 3 比特，可得到最大 8192 个不同的顶级路由。

- RES 字段(Reserved for future use)：

保留字段，占 8 个比特，将来可能会用于扩展顶级或下一级聚集标识符字段。

- NLA ID 字段(Next-Level Aggregation Identifier)：

下一级聚集标识符，占 24 个比特。该标识符被一些机构用于控制顶级聚集以安排地址空间。换句话说，这些机构(比如大型 ISP)能按照他们自己的寻址分级结构来将此 24 位字段分开用。比如一个大型 ISP 可以用 2 位分割成 4 个内部的顶级路由，其余的 22 位地址空间分配给其他实体(如规模较小的本地 ISP)。这些实体如果得到足够的地址空间，可将分配给它们的空间用同样的方法再子分。

- SLA ID 字段(Site-Level Aggregation Identifier)：

站点级聚集标识符，被一些机构用来安排内部的网络结构。每个机构可以用与 IPv4 同样的方法来创建自己内部的分级网络结构。若 16 位字段全部用作平面地址空间，则最多可有 65535 个不同子网。如果用前 8 位作该组织内较高级的选路，那么允

许 255 个大型的子网，每个大型子网可再分为多达 255 个小的子网。

- 接口标识符字段(Interface Identifier)：

64 位长，包含 IEEE EUI-64 接口标识符的 64 位值。

2. 链路本地地址

链路本地地址的格式如下：

```
|   10     |
|  bits    |            54  bits            |               64  bits               |
+----------+--------------------------------+--------------------------------------+
|1111111010|                0               |           interface ID               |
+----------+--------------------------------+--------------------------------------+
```

链路本地地址用于单个网络链路上给主机编号。前缀的前 10 位标识的地址即链路本地地址。设备永远不会转发源地址或者目的地址带有链路本地地址的报文。该地址的中间 54 位置成 0。后 64 位表示接口标识符，地址空间的这部分允许单个网络连接多达（2 的 64 次方减 1）个主机。

3. 站点本地地址

站点本地地址的格式如下：

```
|   10     |
|  bits    |    38  bits    |   16  bits   |              64  bits                |
+----------+----------------+--------------+--------------------------------------+
|1111111011|       0        | subnet ID    |              interface ID            |
+----------+----------------+--------------+--------------------------------------+
```

站点本地地址可以用在站点内传送数据，设备不会将源地址或者目的地址带有站点本地地址的报文转发到 internet 上，即这样的包路只能在站点内转发，而不能把包转发到站点外去。站点本地地址的 10 位前缀与链路本地地址的 10 位前缀略有区别，中间 38 位是 0，站点本地地址的子网标识符为 16 位，而后 64 位也是表示接口标识符，通常是 IEEE 的 EUI-64 地址。

4. 嵌有 IPv4 地址的 IPv6 地址

RFC2373 中还定义了 2 类嵌有 IPv4 地址的特殊 IPv6 地址。

IPv4 兼容的 IPv6 地址格式（IPv4-compatible IPv6 address）

```
|                        80  bits                        | 16 |        32  bits          |
+--------------------------------------------------------+----+--------------------------+
|0000.....................................................0000|0000|     IPv4 address     |
+--------------------------------------------------------+----+--------------------------+
```

IPv4 映射的 IPv6 地址格式（IPv4-mapped IPv6 address）

| 80 bits | 16 | 32 bits |
+----------+----+----------+

|0000.....0000|ffff| IPv4 address |
+----------+----+----------+

IPv4 兼容的 IPv6 地址主要是用在自动隧道上，这类节点即支持 IPv4 也支持 IPv6，IPv4 兼容的 IPv6 地址通过 IPv4 设备以隧道方式传送 IPv6 报文。而 IPv4 映射的 IPv6 地址则被 IP6 节点用于访问只支持 IPv4 的节点，例如当一个 IPv4/IPv6 主机的 IPv6 应用程序请求解析一个主机名字(该主机只支持 IPv4)时，那么名字服务器内部将动态生成 IPv4 映射的 IPv6 地址返回给 IPv6 应用程序。

33.1.2.2 组播地址（Multicast Addresses）

IPv6 组播的地址格式如下：

| 8 | 4 | 4 | 112 bits |
+----------+----+----+----------+

|11111111|flgs|scop| group ID |
+----------+----+----+----------+

地址格式中的第 1 个字节为全“1”代表是一个组播地址。

- 标志字段：

由 4 个比特位组成。目前只指定了第 4 位，该位用来表示该地址是由 Internet 编号机构指定的知名的组播地址，还是特定场合使用的临时组播地址。如果该标志位为“0”，表示该地址为知名组播地址；如果该位为“1”，表示该地址为临时地址。其他 3 个标志位保留将来用。

- 范围字段：

由 4 个比特位组成，用来表示组播的范围。即组播组是包括本地节点、本地链路、本地站点、还是包括 IPv6 全球地址空间中任何位置的节点。

- 组标识符字段：

长 112 位，用于标识组播组。根据组播地址是临时的还是知名的以及地址的范围，同一个组播标识符可以表示不同的组。

IPv6 的组播地址是以 FF00::/8 为前缀的这类地址。一个 IPv6 的组播地址通常标识一系列不同节点的接口。当一个报文发送到一个组播地址上时，那么该报文将分发到标识有该组播地址的每个节点的接口上。一个节点(主机或者设备)必须加入下列的组播：

- 本地链路所有节点组播地址 FF02::1
- 被请求节点的组播地址， 前缀为 FF02:0:0:0:0:1:FF00:0000/104

如果是设备那么还必须加入本地链路所有设备的组播地址 FF02::2。

被请求节点的组播地址是对应于 IPv6 单播(unicast)和泛播(anycast)地址的，IPv6 节点必须为配置的每个单播地址和泛播地址加入其相应的被请求节点的组播地址。被请求节点的组播地址的前缀为 FF02:0:0:0:0:1:FF00:0000/104，另外 24 位由单播地址或者泛播地址的低 24 比特组成，例如对应于单播地址 FE80::2AA:FF:FE21:1234 的被请求节点的组播地址是 FF02::1:FF21:1234 ，

被请求节点组播地址通常用于邻居请求(NS)报文中，被请求节点组播地址的格式如下：

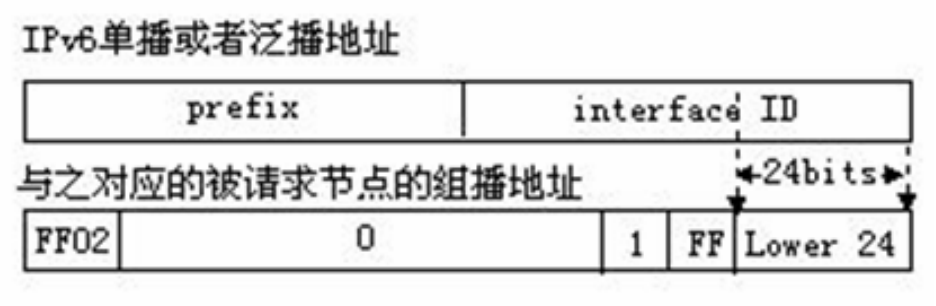


图 1

33.1.2.3 泛播地址（Anycast Addresses）

泛播地址与组播地址类似，同样是多个节点共享一个泛播地址，不同的是只有一个节点期待接收给泛播地址的数据包而组播地址成员的所有节点均期待着接收发给该地址的所有包。泛播地址被分配在正常的 IPv6 单播地址空间，因此泛播地址在形式上与单播地址无法区分开，一个泛播地址的每个成员，必须显式地加以配置，以便识别是泛播地址。

注意:

泛播地址只能分配给设备，不能分配给主机，并且泛播地址不能作为报文的源地址。

在 RFC2373 中预定义了一个泛播地址，称之为子网路由器的泛播地址。下图显示了子网路由器的泛播地址格式，这类地址由子网前缀后面跟着一系列的 0(作为接口标识符)组成。

其中子网前缀标识了一个指定的链路(子网)，送给子网路由器泛播地址的报文将被分发到在该子网上的一个设备。子网路由器的泛播地址通常是被用于一个节点上的应用程序需要和远程子网的一个设备通信而使用。

图 2

33.1.3 IPv6 包头结构

IPv6 包头格式如下图：

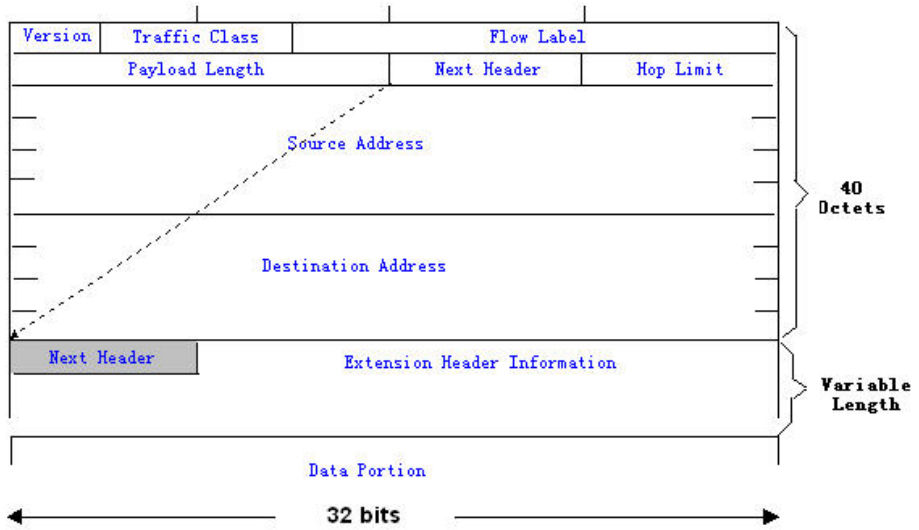


图 3

在 IPv4 中，所有包头以 4 字节为单位。在 IP v 6 中，包头以 8 字节为单位，包头的总长度是 40 字节。IPv6 包头定义了以下字段：

- 版本(Version)：

长度为 4 位，对于 I P v 6 该字段必须为 6。

- 类别(Traffic Class)：

长度为 8 位，指明为该包提供了某种服务，相当于 IPv4 中的“TOS”。

- 流标签(Flow Label)：

长度为 20 位，用于标识属于同一业务流的包，一个节点可以同时作为多个业务流的发送源，流标签和源节点地址唯一标识了一个业务流。

- 净荷长度(Payload Length)：

长度为 16 位，其中包括包净荷的字节长度，同时也包含了各个 IPv6 扩展选项的长度(如果存在)，换句话说就是包含了除 IPv6 头本身外的 IPv6 包的长度。

- 下一个头(Next Header)：

这个字段指出了 IPv6 头后所跟的头字段中的协议类型。与 IPv4 协议字段类似，下一个头字段可以用来指出高层是 TCP 还是 UDP，它也可以用来指明 IPv6 扩展头的存在。

- 跳数(Hop Limit)：

长度为 8 位。每当设备对包进行一次转发之后，这个字段就会被减 1，如果该字段达到 0，这个包就将被丢弃。它与 IPv4 包头中的生存期字段类似。

- 源地址(Source Address)：

长度为 128 位，指出了 IPv6 包的发送方地址。

- 目的地址(Destination Address)：

长度为 128 位，指出了 IPv6 包的接收方地址。

IPv6 的扩展头，目前 IPv6 定义了下列的扩展头：

- 逐跳选项头(Hop-by-Hop Options)：

此扩展头必须紧随在 IPv6 头之后，它包含包所经过的路径上的每个节点都必须检查的选项数据。

- 选路头（Routing （Type 0））：

此扩展头指明包在到达目的地途中将经过哪些节点，它包含包沿途经过的各节点的地址列表。IPv6 头的最初目的地址是路由头的一系列地址中的第一个地址，而不是包的最终目的地址。此地址对应的节点接收到该包之后，对 IPv6 头和选路头进行处理，并把包发送到选路头列表中的第二个地址，如此继续，直到包到达其最终目的地。

- 分片头（Fragment）：

此扩展头用于源节点对长度超出源节点和目的节点路径 MTU 的包进行分片。

- 目的地选项头（Destination Options）：

此扩展头代替了 IPv4 选项字段，目前唯一定义的目的地选项是在需要时把选项填充为 6 4 位（8 字节）的整数倍，此扩展头可以用来携带由目的地节点检查的信息。

- 上层扩展头(Upper-layer header)：

指明了上层传输数据的协议，如 TCP(6)、UDP(17)。

此外还有身份验证头(Authentication)和封装安全性净荷(Encapsulating Security Payload)的扩展头，这将放到 IPSec 章节描述，目前我们实现的 IPv6 还未支持 IPSec。

33.1.4 IPv6 的路径 MTU 发现

和 IPv4 的路径 MTU 发现类似，IPv6 的路径 MTU 发现允许一台主机动态的发现

并调整发送数据路径上的 MTU 的大小。另外，当主机要发送的数据包的大小如果比发送数据路径上的 MTU 大时，那么将由主机自己负责分片。这种由主机分片的行为使得设备无需处理分片从而节省了 IPv6 设备的资源，同时也提高了 IPv6 网络的效率。

注意:

在 IPv4 最小的链路 MTU 是 68 字节，这意味着在每条发送数据的路径上的链路必须至少支持 68 字节大小的链路 MTU。在 IPv6 最小的链路 MTU 是 1280 字节，对于 IPv6 的链路强烈推荐使用 1500 的链路 MTU。

33.1.5 IPv6 邻居发现

IPv6 的邻居发现处理是利用 ICMPv6 的报文和被请求邻居组播地址来获得同一链路上的邻居的链路层地址，并且验证邻居的可达性，维持邻居的状态。下面分别简单描述一下这几类报文。

33.1.5.1 邻居请求报文（Neighbor Solicitation）

当一个结点要与另外一个结点通信时，那么该结点必须获取对方的链路层地址，此时就要向该结点发送邻居请求(NS)报文,报文的目的地址是对应于目的结点的 IPv6 地址的被请求多播地址，发送的 NS 报文同时也包含了自身的链路层地址。当对应的结点收到该 NS 报文后发回一个响应的报文称之为邻居公告报文(NA)，其目的地址是 NS 的源地址，内容为被请求的结点的链路层的地址。当源结点收到该应答报文后就可以和目的结点进行通讯了。

下图是邻居请求的过程：

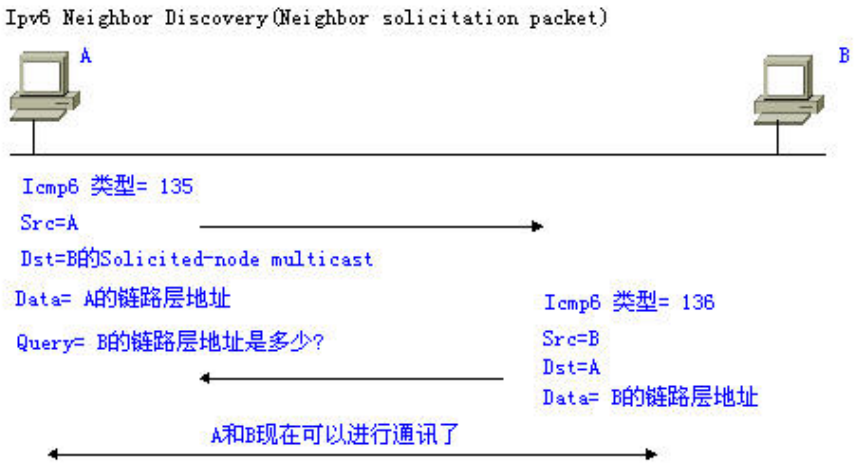


图 4

邻居请求报文同时也可被用来检测邻居的可达性(对已经存在的邻居)，此时邻居请求报文的目的地址是该邻居的单播地址。

当一个结点的链路层地址发生变化时，邻居公告报文将主动被发送，此时邻居公告报文的目的地址为本地链路所有结点的地址。

当一个邻居被认为可到达的时间到期时，将要启动 NUD(Neighbor Unreachability Detection),邻居不可达检测只有发生在要向该邻居发送单播报文的时候，对于发送多播报文是不启动 NUD 的。

另外在无状态地址自动配置中邻居请求报文也被用来检测地址的唯一性，即地址冲突检测，此时报文的源地址是未指定地址(: :)。

33.1.5.2 路由器公告报文(Router Advertisement)

路由器公告报文(RA)在设备上是定期被发往链路本地所有结点的。

路由器公告报文发送如下图：

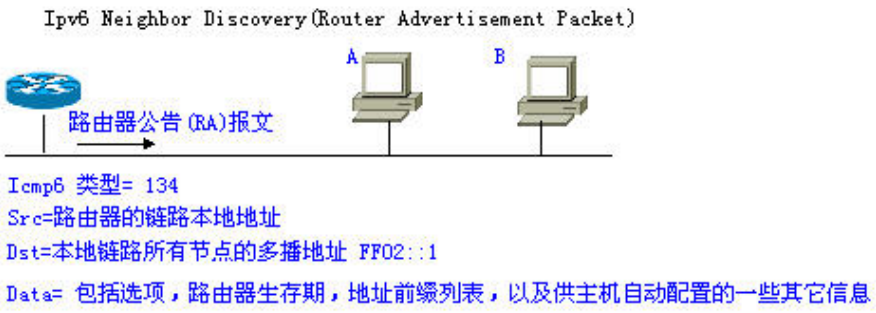


图 5

路由器公告报文中通常包含如下内容：

- 一个或者多个 IPv6 地址前缀用来提供给主机进行地址自动配置。
- IPv6 地址前缀的有效期。
- 主机自动配置使用的方式(有状态还是无状态)。
- 作为缺省设备的信息(即决定本设备是否要作为缺省设备，如果是那么还宣布自己充当缺省设备的时间)。
- 提供给主机配置的一些其它信息如跳数限制、MTU、邻居请求重传间隔时间等。

路由器公告报文同时也用来应答主机发出的路由器请求(RS)报文，路由器请求报文允许主机一旦启动后可以立即获得自动配置的信息而无需等待设备发出的路由器公告报文(RA)。当主机刚启动时如果没有单播地址，那么主机发出的路由器请求报文将使用未指定地址(0:0:0:0:0:0:0:0)作为请求报文的源地址，否则使用已有的单播地址作为源地址，路由器请求报文使用本地链路所有设备组播地址(FF02::2)作为

目的地址。作为应答路由器请求(RS)报文的路由器公告(RA)报文将使用请求报文的源地址作为目的地址(如果源地址是未指定地址那么将使用本地链路所有结点组播地址 FF02::1)。

在路由器公告报文中下列参数是可以被配置的：

Ra-interval 路由器公告报文的发送间隔。

Ra-lifetime 路由器生存期，即设备是否充当本地链路的缺省路由器以及充当该角色的时间。

Prefix 本地链路的 IPv6 地址前缀，可用来供主机进行自动配置，包括前缀的其它参数配置 。

Rs-initerval 邻居请求报文重传的时间间隔。

Reachabletime 检测到邻居可到达事件后认为邻居是可到达的所维持的时间。

以上这些参数在 IPv6 接口属性中进行配置。

注意：

1) 缺省在该接口上不主动发送路由器公告报文，如果想允许路由器公告报文的发送可以在接口配置模式下使用命令 **no ipv6 nd suppress-ra** 。
2) 为了能够使节点无状态地址自动配置能够正常工作，路由器公告报文中公告的前缀的长度必须为 64 比特。

33.2 IPv6 的配置

下面将分别介绍 IPv6 的各个功能模块的配置。

33.2.1 配置 IPv6 的地址

本节的任务描述了如何在一个接口上配置 IPv6 地址。缺省没有配置 IPv6 地址。

注意：

一旦 IPv6 的接口创建并且该接口的链路是 UP 状态的，那么系统将为该接口自动生成链路本地地址。目前 IPv6 不支持配置泛播地址(Anycast)。

配置 IPv6 地址过程如下：

命令	含义

configure terminal	进入全局配置模式。
interface *interface-id*	进入接口配置模式。
ipv6 enable	打开接口的 IPv6 协议。如果没有执行这条命令，给接口配置 IPv6 地址时会自动打开 IPv6 协议。
ipv6 address *ipv6-prefix/prefix-length* [**eui-64**]	为该接口配置 IPv6 的单播地址，Eui-64 关键字表明生成的 ipv6 地址由配置的地址前缀和 64 比特的接口 ID 标识符组成。 注意：无论是否使用 eui-64 关键字，在删除地址时都必须输入完整的地址形式(前缀+接口 ID/前缀长度)。 如果在接口上配置 IPv6 地址，那么接口的 IPv6 协议就会自动打开，即使使用 **no ipv6 enable** 也不能关闭 IPv6 协议。
end	退回到特权模式。
show ipv6 interface *interface-id*	查看 IPv6 接口的相关信息
copy running-config startup-config	保存配置。

使用 **no ipv6 address** *ipv6-prefix/prefix-length* [**eui-64**]命令删除已配置的地址，下面是配置 IPv6 地址的例子：

```
Ruijie(config)# interface vlan 1
Ruijie(config-if)# ipv6 enable
Ruijie(config-if)# ipv6 address fec0:0:0:1::1/64
Ruijie(config-if)# end
Ruijie(config-if)# show ipv6 interface vlan 1
Interface vlan 1 is Up, ifindex: 2001
address(es):
Mac Address: 00:00:00:00:00:01
INET6: fe80::200:ff:fe00:1 , subnet is fe80::/64
INET6: fec0:0:0:1::1 , subnet is fec0:0:0:1::/64
Joined group address(es):
ff01:1::1
ff02:1::1
ff02:1::2
ff02:1::1:ff00:1
MTU is 1500 bytes
ICMP error messages limited to one every 100 milliseconds
ICMP redirects are enabled
ND DAD is enabled, number of DAD attempts: 1
ND reachable time is 30000 milliseconds
```

```
ND advertised reachable time is 0 milliseconds
ND retransmit interval is 1000 milliseconds
ND advertised retransmit interval is 0 milliseconds
ND router advertisements are sent every 200 seconds<160--240>
ND router advertisements live for 1800 seconds
```

33.2.2 配置 ICMPv6 的重定向功能

本节描述如何配置接口的 ICMPv6 的重定向功能，缺省情况下在接口上是打开 IPv6 重定向功能的。当设备在转发包的过程中遇到下列情况同时发生时必须向报文的发起者发送重定向报文：

- 报文的目的地址不是一个多播地址；
- 报文的目的地址不是设备本身；
- 设备为该报文决定的下一跳的输出接口和收到该报文的接口相同，即下一跳和发起者处在同一条链路上；
- 报文的源地址标识的节点是本地设备上的一个邻居，即在设备上的邻居表中存在该邻居。

注意：

只有设备可以产生重定向报文，主机不能产生，并且设备收到重定向报文不会为其更新路由表。

下面是配置一个接口打开重定向功能的过程：

命令	含义
configure terminal	进入全局配置模式。
interface vlan *1*	进入 SVI 1 接口配置模式
ipv6 redirects	打开该接口的 IPv6 重定向功能
end	退回到特权模式。
show ipv6 interface vlan *1*	查看接口配置的相关信息。
copy running-config startup-config	保存配置。

使用 **no ipv6 redirects** 命令关闭重定向的功能，下面是配置重定向功能打开的例子：

```
Ruijie(config)# interface vlan 1
Ruijie (config-if)# ipv6 redirects
Ruijie (config-if)# end
```

```
Ruijie # show ipv6 interface vlan 1
Interface vlan 1 is Up, ifindex: 2001
address(es):
Mac Address: 00:d0:f8:00:00:01
INET6: fe80::2d0:f8ff:fe00:1 , subnet is fe80::/64
INET6: fec0:0:0:1::1 , subnet is fec0:0:0:1::/64
Joined group address(es):
ff01:1::1
ff02:1::1
ff02:1::2
ff02:1::1:ff00:1
MTU is 1500 bytes
ICMP error messages limited to one every 100 milliseconds
ICMP redirects are enabled
ND DAD is enabled, number of DAD attempts: 1
ND reachable time is 30000 milliseconds
ND advertised reachable time is 0 milliseconds
ND retransmit interval is 1000 milliseconds
ND advertised retransmit interval is 0 milliseconds
ND router advertisements are sent every 200 seconds<160--240>
ND router advertisements live for 1800 seconds
```

33.2.3 配置静态邻居

本节描述如何配置一个静态邻居，缺省情况下没有配置静态邻居，通常邻居是通过邻居发现协议(NDP)来动态学习并且维护其状态的，同时也允许手工为其配置静态的邻居。

下面是配置一个静态邻居的过程：

命令	含义
configure terminal	进入全局配置模式。
ipv6 neighbor *ipv6-address interface-id hardware-address*	使用该命令在接口上配置一个静态的邻居。
end	退回到特权模式。
show ipv6 neighbors	查看邻居表。
copy running-config startup-config	保存配置。

使用 **no ipv6 neighbor** 命令允许删除指定的邻居，下面是在 SVI 1 上配置一个静态邻居的例子：

```
Ruijie(config)# ipv6 neighbor fec0:0:0:1::100 vlan 1
```

```
00d0.f811.1234
Ruijie (config)# end
Ruijie# show ipv6 neighbors verbose fec0:0:0:1::100
IPv6 Address       Linklayer Addr   Interface
fec0:0:0:1::100    00d0.f811.1234   vlan 1
State: REACH/H Age: - asked: 0
```

33.2.4 配置地址冲突检测

本节主要描述如何配置地址冲突检测的次数，地址冲突检测是在所有的单播地址被正式赋予接口前所要进行的工作，即用来检测地址的唯一性。无论是手工配置的、无状态自动配置的还是有状态自动配置的地址都要进行地址冲突检测。但是有下面 2 种情况不需要进行地址冲突检测：

- 管理上禁止地址冲突检测，即将用于地址冲突检测时发送的邻居请求报文数设置为 0
- 被明确的配置为泛播地址的是不能应用于地址冲突检测的

另外如果接口的地址冲突检测功能没有关闭的话，接口每次从 Donw 状态变为 Up 状态时都会为配置的地址重新启动地址冲突检测过程。

下面是配置用于地址冲突检测时所发送的邻居请求报文的数量的过程：

命令	含义
configure terminal	进入全局配置模式。
interface vlan *1*	进入 SVI 1 的配置模式。
ipv6 nd dad attempts *attempts*	用于地址冲突检测时所发送的邻居请求报文的数量。当配置为 0 时表示禁止在该接口上启用地址冲突检测功能
end	退到特权模式。
show ipv6 interface vlan *1*	查看 SVI 1 上的 IPv6 的信息
copy running-config startup-config	保存配置。

使用 **no ipv6 nd dad attempts** 命令将恢复缺省值，下面是配置 SVI1 上地址冲突检测发送邻居请求(NS)报文次数的例子：

```
Ruijie(config)# interface vlan 1
Ruijie(config-if)# ipv6 nd dad attempts 3
Ruijie(config-if)# end
Ruijie# show ipv6 interface vlan 1
Interface vlan 1 is Up, ifindex: 2001
address(es):
Mac Address: 00:d0:f8:00:00:01
INET6: fe80::2d0:f8ff:fe00:1 , subnet is fe80::/64
```

```
INET6: fec0:0:0:1::1 , subnet is fec0:0:0:1::/64
Joined group address(es):
ff01:1::1
ff02:1::1
ff02:1::2
ff02:1::1:ff00:1
MTU is 1500 bytes
ICMP error messages limited to one every 100 milliseconds
ICMP redirects are enabled
ND DAD is enabled, number of DAD attempts: 3
ND reachable time is 30000 milliseconds
ND advertised reachable time is 0 milliseconds
ND retransmit interval is 1000 milliseconds
ND advertised retransmit interval is 0 milliseconds
ND router advertisements are sent every 200 seconds<160--240>
ND router advertisements live for 1800 seconds
```

33.2.5 配置路由器的其它接口参数

设备的接口中关于 IPv6 的配置参数主要分为 2 部分，一类是用来控制设备本身行为的，一类是用来控制设备发送的路由器公告报文的内容以决定收到该路由器公告报文的主机进行什么样的动作。

下面逐一介绍这些命令：

命令	含义
configure terminal	进入全局配置模式。
interface *interface-id*	进入接口配置模式。
ipv6 enable	启用 IPv6 功能
ipv6 nd ns-interval *milliseconds*	（可选）定义了邻居请求报文的重传间隔
ipv6 nd reachable-time *milliseconds*	（可选）定义了邻居被认为可到达的时间。注意：根据 RFC4861 规定，邻居的实际可达时间要在配置时间基础上进行一定的随机浮动，范围在配置时间的 0.5 倍到 1.5 倍之间。
ipv6 nd prefix *ipv6-prefix/prefix-length* | **default** [[*valid-lifetime preferred-lifetime*] | [**at** *valid-date preferred-date*] | **infinite** | **no-advertise**]]	（可选）设置路由器公告(RA)报文中所要公告的地址前缀

ipv6 nd ra-lifetime *seconds*	（可选）设置路由器公告(RA)报文中的路由器生存期时间，即充当缺省设备的时间，当设置为 0 表示不充当其所直连的网络的缺省设备
ipv6 nd ra-interval *seconds*	（可选）设置设备定期发送路由器公告(RA)报文的间隔
ipv6 nd managed-config-flag	（可选）设置路由器公告(RA)报文中的“managed address configuration”标志位，决定了收到该路由器公告的主机是否要使用全状态自动配置来获取地址
ipv6 nd other-config-flag	（可选）设置路由器公告报文中的“other stateful configuration”标志位，决定了收到该路由器公告的主机是否要使用全状态的自动配置来获取除地址之外的信息
ipv6 nd suppress-ra	（可选）设置是否在该接口上阻止路由器公告（RA）报文发送
end	退回到特权模式。
show ipv6 interface [*interface-id*] [**ra-info**]	显示接口的 ipv6 信息或者该接口发送 RA 的信息。
copy running-config startup-config	（可选）保存配置。

使用以上命令的 **no** 命令可以将其恢复缺省值，具体命令的使用指南可以参考“配置 IPv6 命令”。

33.3 IPv6 的监控与维护

主要是提供相关的命令用来显示 IPv6 协议内部的一些信息，比如显示接口的 IPv6 信息、邻居表和路由表等信息。

命令	含义
show ipv6 interface [*interface-id*] [**ra-info**]	显示接口上关于 IPv6 的信息
show ipv6 neighbors [**verbose**] [*interface-id*] [*ipv6-address*]	显示邻居的信息
show ipv6 route [**static**] [**local**] [**connected**]	显示 IPv6 路由表的信息

1) 查看一个接口上关于 IPv6 的信息

```
Ruijie# show ipv6 interface
interface vlan 1 is Down, ifindex: 2001
```

```
address(es):
Mac Address: 00:d0:f8:00:00:01
INET6: fe80::2d0:f8ff:fe00:1 , subnet is fe80::/64
INET6: fec0:1:1:1::1 , subnet is fec0:1:1:1::/64
Joined group address(es):
ff01:1::1
ff02:1::1
ff02:1::2
ff02:1::1:ff00:1
MTU is 1500 bytes
ICMP error messages limited to one every 100 milliseconds
ICMP redirects are enabled
ND DAD is enabled, number of DAD attempts: 1
ND reachable time is 30000 milliseconds
ND advertised reachable time is 0 milliseconds
ND retransmit interval is 1000 milliseconds
ND advertised retransmit interval is 0 milliseconds
ND router advertisements are sent every 200 seconds<160--240>
ND router advertisements live for 1800 seconds
```

2) 查看一个接口上关于设备在该接口上要发送的路由器公告报文的信息

```
Ruijie# show ipv6 interface ra-info
vlan 1: DOWN
RA timer is stopped
waits: 0, initcount: 3
statistics: RA(out/in/inconsistent): 4/0/0, RS(input): 0
Link-layer address: 00:00:00:00:00:01
Physical MTU: 1500
ND router advertisements live for 1800 seconds
ND router advertisements are sent every 200 seconds<160--240>
Flags: !M!O, Adv MTU: 1500
ND advertised reachable time is 0 milliseconds
ND advertised retransmit time is 0 milliseconds
ND advertised CurHopLimit is 64
Prefixes: (total: 1)
fec0:1:1:1::/64(Def, Auto, vltime: 2592000, pltime: 604800,
flags: LA)
```

3) 查看 IPv6 的邻居表信息

```
Ruijie# show ipv6 neighbors
IPv6 Address              Linklayer Addr  Interface
fe80::200:ff:fe00:1       0000.0000.0001 vlan 1
State: REACH/H Age: - asked: 0
fec0:1:1:1::1             0000.0000.0001 vlan 1
State: REACH/H Age: - asked: 0
```

34 IPv6 隧道配置

34.1 概述

IPv6 的根本目的是继承和取代 IPv4，但从 IPv4 到 IPv6 的演进是一个逐渐的过程。因此在 IPv6 完全取代 IPv4 之前，不可避免地，这两种协议要有一个共存时期。在这个过渡阶段的初期，IPv4 网络仍然是主要的网络，IPv6 网络类似孤立于 IPv4 网络中的小岛。过渡的问题可以分成两大类：

1) 被孤立的 IPv6 网络之间透过 IPv4 网络互相通信的问题；

2) IPv6 的网络与 IPv4 网络之间通信的问题；

本文讨论的隧道（Tunnel）技术，就是解决问题 1 的，解决问题 2 的方案是 NAT-PT（网络地址转换-协议转换），不在本文讨论范围内。

IPv6 隧道是将 IPv6 报文封装在 IPv4 报文中，这样 IPv6 协议包就可以穿越 IPv4 网络进行通信。因此被孤立的 IPv6 网络之间可以通过 IPv6 的隧道技术利用现有的 IPv4 网络互相通信而无需对现有的 IPv4 网络做任何修改和升级。IPv6 隧道可以配置在边界路由器之间也可以配置在边界路由器和主机之间，但是隧道两端的节点都必须既支持 IPv4 协议栈又支持 IPv6 协议栈。目前，我公司支持下列几种隧道技术:

隧道类型	参考标准
Manually Config Tunnel	RFC2893
automatic 6to4 Tunnel	RFC3056
Intra-Site Automatic Tunnel Addressing Protocol(ISATAP)	draft-ietf-ngtrans-isatap-22

注意:

通过 IPv6 隧道技术将被孤立的 IPv6 网络互联起来并不是最终的 IPv6 的网络架构，而只是一种过渡的技术。

使用隧道技术的模型如下图：

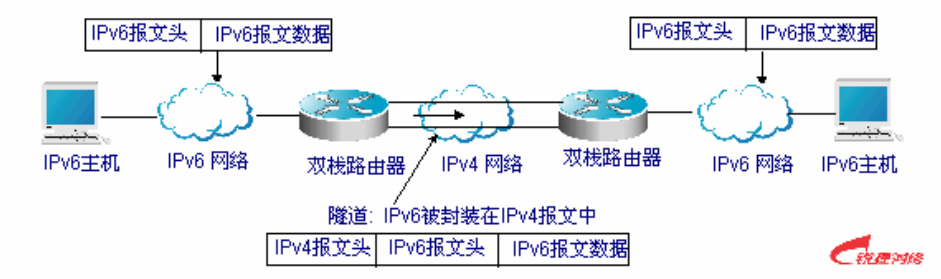


图 1

下面分别介绍各隧道的特点。

34.1.1 手工配置隧道(IPv6 Manually Configured Tunnel)

一个手工配置隧道类似于在两个 IPv6 域之间通过 IPv4 的主干网络建立了一条永久链路。适合用在两台边界路由器或者边界路由器和主机之间对安全性要求较高并且比较固定的连接上。

在隧道接口上，IPv6 地址需要手工配置，并且隧道的源 IPv4 地址(Tunnel Source)和目的 IPv4 地址(Tunnel Destination)必须手工配置。隧道两端的节点必须支持 IPv6 和 IPv4 协议栈。手工配置隧道在实际应用中总是成对配置的，即在两台边缘设备上同时配置，可以将其看作是一种点对点的隧道。

34.1.2 6to4 自动隧道(Automatic 6to4 Tunnel)

6to4 自动隧道技术允许将被孤立的 IPv6 网络透过 IPv4 网络互联。它和手工配置隧道的主要区别是手工配置隧道是点对点的隧道，而 6to4 隧道是点对多点的隧道。

6to4 隧道将 IPv4 网络视为 Nonbroadcast Multi-access（NBMA，非广播多路访问）链路，因此 6to4 的设备不需要成对的配置，嵌入在 IPv6 地址的 IPv4 地址将用来寻找自动隧道的另一端。6to4 隧道可以看做是点到多点的隧道。6to4 自动隧道可以被配置在一个被孤立的 IPv6 网络的边界路由器上，对于每个报文它将自动建立隧道到达另一个 IPv6 网络的边界路由器。隧道的目的地址就是另一端的 IPv6 网络的边界路由器的 IPv4 地址，该 IPv4 地址将从该报文的目的 IPv6 地址中提取，其 IPv6 地址是以前缀 2002::/16 开头的，形式如下:

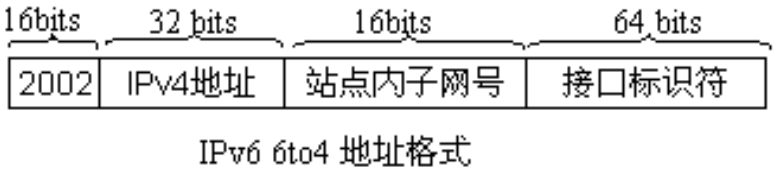


图 2

6to4 地址是用于 6to4 自动构造隧道技术的地址，其内嵌的 IPv4 地址通常是站点边界路由器出口的全局 IPv4 地址，在自动隧道建立时将使用该地址作为隧道报文封装的 IPv4 目的地址。6ot4 隧道两端的设备同样必须都支持 IPv6 和 IPv4 协议栈。6to4 隧道通常是配置在边界路由器之间。

例如：6to4 站点边界路由器出口的全局 IPv4 地址是 211.1.1.1（用十六进制数表达为 D301:0101），站点内的某子网号为 1，接口标识符为 2e0:ddff:fee0:e0e1，那么其对应的 6to4 地址可以表示为：

2002: D301:0101:1: 2e0:ddff:fee0:e0e1

注意:

6to4 地址内嵌的 IPv4 地址不能为私有的 IPv4 地址（即 10.0.0.0/8，172.16.0.0/12，192.168.0.0/16 网段的地址）而必须是全局的 IPv4 地址。

6to4 隧道常用的应用模型：

- 简单应用模型

6to4 隧道最简单、最常用的应用是用来多个 IPv6 站点之间的互联，每个站点至少必须有一个连接通向一个它们共享的 IPv4 网络。这个 IPv4 网络可以是 Internet 网络也可以是某个组织团体内部的主干网。关键是每个站点必须要一个全局唯一的 IPv4 地址，6to4 隧道将使用该地址构照一个 6to4/48 的 IPv6 前缀：2002:IPV4 地址/48。

- 混合应用模型

在上面所描述的应用的基础上，通过在纯 IPv6 网络的边缘提供 6to4 中继设备，实现其它 6to4 网络接入纯 IPv6 网络中，实现该功能的设备我们称之为 6to4 中继路由器(6to4 Relay Router)。

34.1.3 ISATAP 自动隧道(ISATAP Tunnel)

站内自动隧道寻址协议(ISATAP)是一种站点内部的 IPv6 体系架构将 IPv4 网络视为一个非广播型多路访问（NBMA）链路层的 IPv6 隧道技术，即将 IPv4 网络当作 IPv6 的虚拟链路层。

ISATAP 主要是用在当一个站点内部的纯 IPv6 网络还不能用，但是又要在站点内部传输 IPv6 报文的情况，例如站点内部有少数测试用的 IPv6 主机要互相通讯。使用 ISATAP 隧道允许站点内部同一虚拟链路上的 IPv4/IPv6 双栈主机互相通讯。

在 ISATAP 站点上，ISATAP 设备提供标准的路由器公告报文，从而允许站点内部的 ISATAP 主机进行自动配置；同时 ISATAP 设备也执行站点内的 ISATAP 主机和站点外的 IPv6 主机转发报文的功能。

ISATAP 使用的 IPv6 地址前缀可以是任何合法的 IPv6 单点传播的 64 位前缀，包括全球地址前缀、链路本地前缀和站点本地前缀等，IPv4 地址被置于 IPv6 地址最后的 32 比特上，从而允许自动建立隧道。

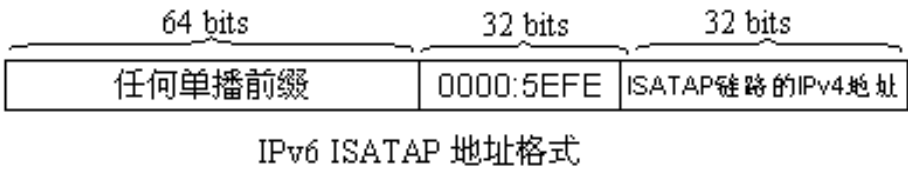
ISATAP 很容易与其他过渡技术结合起来使用，尤其是在和 6to4 隧道技术相结合使用时，可以使内部网的双栈主机非常容易地接入 IPv6 主干网。

- ISATAP 接口标识符

ISATAP 使用的单播地址的形式是 64 比特的 IPv6 前缀加上 64 比特的接口标识符。64 比特的接口标识符是由修正的 EUI-64 地址格式生成的，其中接口标识符的前 32 比特的值为 0000:5EFE，这就意味着这是一个 ISATAP 的接口标识符。

- ISATAP 的地址结构

ISATAP 地址是指接口标识符中包含 ISATAP 接口标识符的单播地址，下图显示了 ISATAP 的地址结构：



IPv6 ISATAP 地址格式

图 3

从上图中可以看到接口标识符中包含了 IPv4 的地址，该地址就是双栈主机的 IPv4 地址，在自动建立自动隧道时将被使用。

例如：IPv6 的前缀是 2001::/64，嵌入的 IPv4 的地址是 192.168.1.1，在 ISATAP 地址中，IPv4 地址用十六进制数表达为 C0A8:0101，因此其对应的 ISATAP 地址为：

2001::0000:5EFE:C0A8:0101

34.2 IPv6 隧道的配置

34.2.1 配置手工 IPv6 隧道

本节解释如何配置手工隧道。

要配置手工隧道，在隧道接口上要配置一个 IPv6 地址并且要手工配置隧道的源端和目的端的 IPv4 地址。在配置隧道两端的主机或者设备必须支持双栈(IPv6 协议栈和 IPv4 协议栈)。

注意:

不能在设备上配置隧道的源（Tunnel Source）和目的（Tunnel Destination）地址相同的手工隧道。

简要步骤

```
config terminal
interface tunnel tunnel-num
tunnel mode ipv6ip
ipv6 enable
tunnel source {ip-address | type num}
```

```
tunnel destination ip-address
end
```

详细步骤

命令	含义
configure terminal	进入全局配置模式。
interface tunnel *tunnel-num*	指定隧道接口号创建隧道接口，并进入接口配置模式。
tunnel mode ipv6ip	指定隧道的类型为手工配置隧道。
ipv6 enable	使能该接口的 IPv6 功能，您也可以通过配置 IPv6 地址直接启动该接口的 IPv6 功能。
tunnel source {*ip-address* | *type num*}	指定隧道的 IPv4 源地址或者引用的源接口号。注意如果指定了接口，那么接口上必须已经配置了 IPv4 的地址。
tunnel destination *ip-address*	指定隧道的目的地址。
end	退到特权模式。
copy running-config startup-config	保存配置。

参考“验证 IPv6 隧道的配置和监控”一节来检查隧道的工作情况。

34.2.2 配置 6to4 隧道

本节介绍如何配置 6to4 隧道。

6to4 隧道的目的地址是由从 6to4 IPv6 地址中提取的IPv4 地址决定的，6to4 隧道两端的设备必须支持双栈，即支持IPv4 和IPv6 协议栈。

注意:

在一台设备上只支持配置一个 6to4 隧道。6to4 隧道使用的封装源地址（IPv4 地址）必须是全局可路由的地址，否则 6to4 隧道将不能正常工作。

简要步骤

```
config terminal
interface tunnel tunnel-num
tunnel mode ipv6ip 6to4
ipv6 enable
```

```
tunnel source {ip-address | type num}
exit
ipv6 route 2002::/16 tunnel tunnel-number
end
```

详细步骤

命令	含义
configure terminal	进入全局配置模式。
interface tunnel *tunnel-num*	指定隧道接口号创建隧道接口，并进入接口配置模式。
tunnel mode ipv6ip 6to4	指定隧道的类型为 6to4 隧道。
ipv6 enable	使能该接口的 IPv6 功能，您也可以通过配置 IPv6 地址直接启动该接口的 IPv6 功能。
tunnel source {*ip-address* | *type num*}	指定隧道的封装源地址或者引用的源接口号。 注意： 被引用的接口上必须已经配置了 IPv4 的地址。 使用的 IPv4 地址必须是全局可路由的地址。
Exit	退回全局配置模式。
ipv6 route *2002::/16* **tunnel** *tunnel-number*	为 IPv6 6to4 前缀 2002::/16 配置一条静态的路由并关联输出接口到该隧道接口上(即前面步骤 2 中指定的隧道接口)。
End	退回特权模式。
copy running-config startup-config	保存配置。

参考“验证 IPv6 隧道的配置和监控”一节来检查隧道的工作情况。

34.2.3 配置 ISATAP 隧道

本节介绍如何配置 ISATAP 设备。

在ISATAP隧道接口上，ISATAP IPv6 地址的配置以及前缀的公告配置和普通IPv6接口的配置是一样的，但是为ISATAP隧道接口配置的地址必须使用修正的EUI-64地址，因为IPv6 地址中的接口标识符的最后 32 位是由隧道源地址(Tunnel Source)引用的接口的IPv4 地址)构成的。ISATAP地址格式参见上面的章节。

注意:

设备上允许同时配置多个 ISATAP 隧道，但是每个 ISATAP 隧道的 tunnel source

必须是不同的，否则收到 ISATAP 隧道报文时无法区分是属于哪个 ISATAP 隧道。

简要步骤

```
config terminal
interface tunnel tunnel-num
tunnel mode ipv6ip isatap
ipv6 address ipv6-prefix/prefix-length eui-64
tunnel source interface-type num
no ipv6 nd suppress-ra
end
```

详细步骤

命令	含义
configure terminal	进入全局配置模式。
interface tunnel *tunnel-num*	指定隧道接口号创建隧道接口，并进入接口配置模式。
tunnel mode ipv6ip isatap	指定隧道的类型为 ISATAP 隧道。
ipv6 address *ipv6-prefix*/*prefix-length* [**eui-64**]	配置 IPv6 ISATAP 地址,注意指定使用 eui-64 关键字，这样将自动生成 ISATAP 的地址，在 ISATAP 接口上配置的地址必须为 ISATAP 的地址。
tunnel source *type num*	指定隧道引用的源接口号,被引用的接口上必须已经配置了 IPv4 的地址。
no ipv6 nd suppress-ra	缺省情况下是禁止在接口上发送路由器公告报文的,使用该命令打开该功能从而允许 ISATAP 主机进行自动配置。
end	退回特权模式。
copy running-config startup-config	保存配置。

参考 “验证 IPv6 隧道的配置和监控”一节来检查隧道的工作情况。

34.3 验证 IPv6 隧道的配置和监控

本节介绍如何验证 IPv6 隧道的配置以及实际运行情况：

简要步骤

```
enable
show interface tunnel number
```

```
show ipv6 interface tunnel mumber
ping protocol destination
show ip route
show ipv6 route
```

详细步骤

命令	含义
enable	进入特权配置模式。
show interface tunnel *tunnel-num*	查看指定 tunnel 接口的信息.
show ipv6 interface tunnel *tunnel-num*	查看 tunnel 接口的 IPv6 信息
ping *protocol destination*	检查网络的基本连通性
show ip route	查看 IPv4 路由器表
show ipv6 route	查看 IPv6 路由表

1. 查看 Tunnel 接口的信息

```
Ruijie# show interface tunnel 1
Tunnel 1 is up, line protocol is Up
Hardware is Tunnel, Encapsulation TUNNEL
Tunnel source 192.168.5.215 , destination 192.168.5.204
Tunnel protocol/transport IPv6/IP
Tunnel TTL is 9
Tunnel source do conformance check set
Tunnel source do ingress filter set
Tunnel destination do safety check not set
Tunnel disable receive packet not set
```

2. 查看 Tunnel 接口的 IPv6 信息

```
Ruijie# show ipv6 interface tunnel 1
interface Tunnel 1 is Up, ifindex: 6354
address(es):
Mac Address: N/A
INET6: fe80::3d9a:1601 , subnet is fe80::/64
Joined group address(es):
ff02::2
ff01::1
ff02::1
ff02::1:ff9a:1601
INET6: 3ffe:4:0:1::1 , subnet is 3ffe:4:0:1::/64
Joined group address(es):
ff02::2
ff01::1
```

```
ff02::1
ff02::1:ff00:1
MTU is 1480 bytes
ICMP error messages limited to one every 100 milliseconds
ICMP redirects are enabled
ND DAD is enabled, number of DAD attempts: 1
ND reachable time is 30000 milliseconds
ND advertised reachable time is 0 milliseconds
ND retransmit interval is 1000 milliseconds
ND advertised retransmit interval is 0 milliseconds
ND router advertisements are sent every 200 seconds<160--240>
ND router advertisements live for 1800 seconds
```

34.4 IPv6 隧道配置实例

- 下面章节介绍 IPv6 隧道配置实例
- 配置手工隧道例子
- 配置 6to4 隧道例子
- 配置 ISATAP 隧道例子
- 配置 ISATAP 和 6to4 隧道的例子

34.4.1 配置手工 IPv6 隧道实例

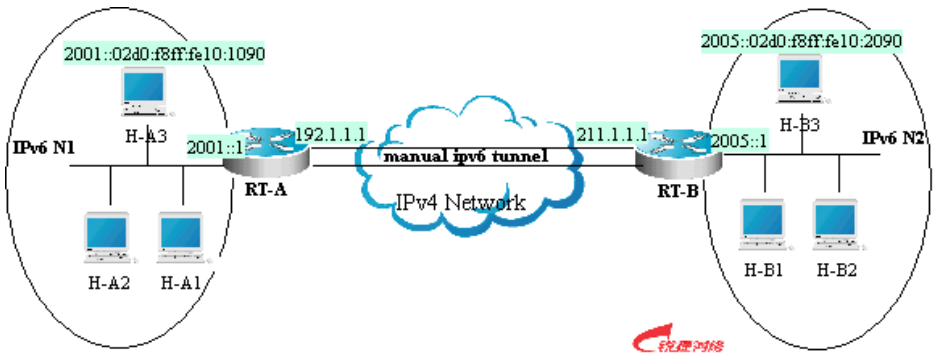


图 4

如上图，IPv6 网络 N1 和 N2 被 IPv4 网络隔离开了，现在通过配置手工隧道将这两个网络互联起来，如可以使 N1 中的 H-A3 主机可以访问 N2 中的 H-B3 主机。

图中 RT-A 和 RT-B 是支持 IPv4 和 IPv6 协议栈的设备，隧道的配置在 N1 和 N2 的边界路由器上(RT-A 和 RT-B)进行，注意手工隧道必须对称配置，既在 RT-A 和 RT-B 上都要配置。

和隧道相关的具体配置分别如下：

前提：假设 IPv4 的路由是连通的，下面不再列出关于 IPv4 的路由配置情况。

RT-A 的配置

```
# 连接 IPv4 网络的接口
interface FastEthernet 2/1
no switchport
ip address 192.1.1.1 255.255.255.0

# 连接 IPv6 网络的接口
interface FastEthernet 2/2
no switchport
ipv6 address 2001::1/64
no ipv6 nd suppress-ra  (可选)

# 配置手工隧道接口
interface Tunnel 1
tunnel mode ipv6ip
ipv6 enable
tunnel source FastEthernet 2/1
tunnel destination 211.1.1.1

# 配置进隧道的路由
ipv6 route 2005::/64 tunnel 1
```

RT-B 的配置

```
# 连接 IPv4 网络的接口
interface FastEthernet 2/1
no switchport
ip address 211.1.1.1 255.255.255.0

# 连接 IPv6 网络的接口
interface FastEthernet 2/2
no switchport
ipv6 address 2005::1/64
no ipv6 nd suppress-ra  (可选)

# 配置手工隧道接口
interface Tunnel 1
tunnel mode ipv6ip
ipv6 enable
tunnel source FastEthernet 2/1
tunnel destination 192.1.1.1

# 配置进隧道的路由
ipv6 route 2001::/64 tunnel 1
```

34.4.2 配置 6to4 隧道实例

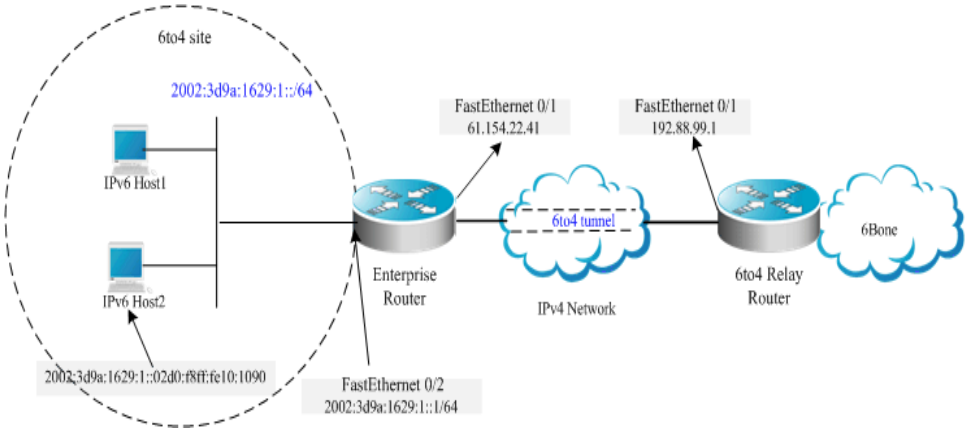


图 5

如上图是一个 IPv6 网络(6to4 站点)使用 6to4 隧道通过 6to4 中继路由器接入 IPv6 主干网(6bone)的一个例子。

在前面已经介绍了 6to4 隧道技术主要用在将孤立的 IPv6 网络互联起来，并且可以通过 6to4 中继路由器方便的接入 IPv6 主干网络。6to4 隧道是自动隧道，嵌入在 IPv6 地址的 IPv4 地址将用来寻找自动隧道的另一端，因此 6to4 隧道无须配置隧道的目的端，同时 6to4 隧道的配置不像手工隧道要对称配置。

61.154.22.41 的十六进制格式为：3d9a:1629

192.88.99.1 的十六进制格式为：c058:6301

注意:

在边界路由器上配置 6to4 隧道时，必须使用全局可路由的 IPv4 地址。否则 6to4 隧道将不能正常工作。

下面是图中 2 个设备的配置（假设 IPv4 的路由是连通的，不考虑 IPv4 路由配置）：

Enterprise Router 的配置

```
# 连接 IPv4 网络的接口
interface FastEthernet 0/1
no switchport
ip address 61.154.22.41 255.255.255.128

# 连接 IPv6 网络的接口
interface FastEthernet 0/2
no switchport
ipv6 address 2002:3d9a:1629:1::1/64
no ipv6 nd suppress-ra
```

```
# 配置 6to4 隧道接口
interface Tunnel 1
tunnel mode ipv6ip 6to4
ipv6 enable
tunnel source FastEthernet 0/1

# 配置进隧道的路由
ipv6 route 2002::/16 Tunnel 1

# 配置到 6to4 中继路由器的路由,以便可以访问 6bone
ipv6 route ::/0 2002:c058:6301::1
```

ISP 6to4 Relay Router 的配置

```
# 连接 IPv4 网络的接口
interface FastEthernet 0/1
no switchport
ip address 192.88.99.1 255.255.255.0

# 配置 6to4 隧道接口
interface Tunnel 1
tunnel mode ipv6ip 6to4
ipv6 enable
tunnel source FastEthernet 0/1

# 配置进隧道的路由
ipv6 route 2002::/16 Tunnel 1
```

34.4.3 配置 ISATAP 隧道实例

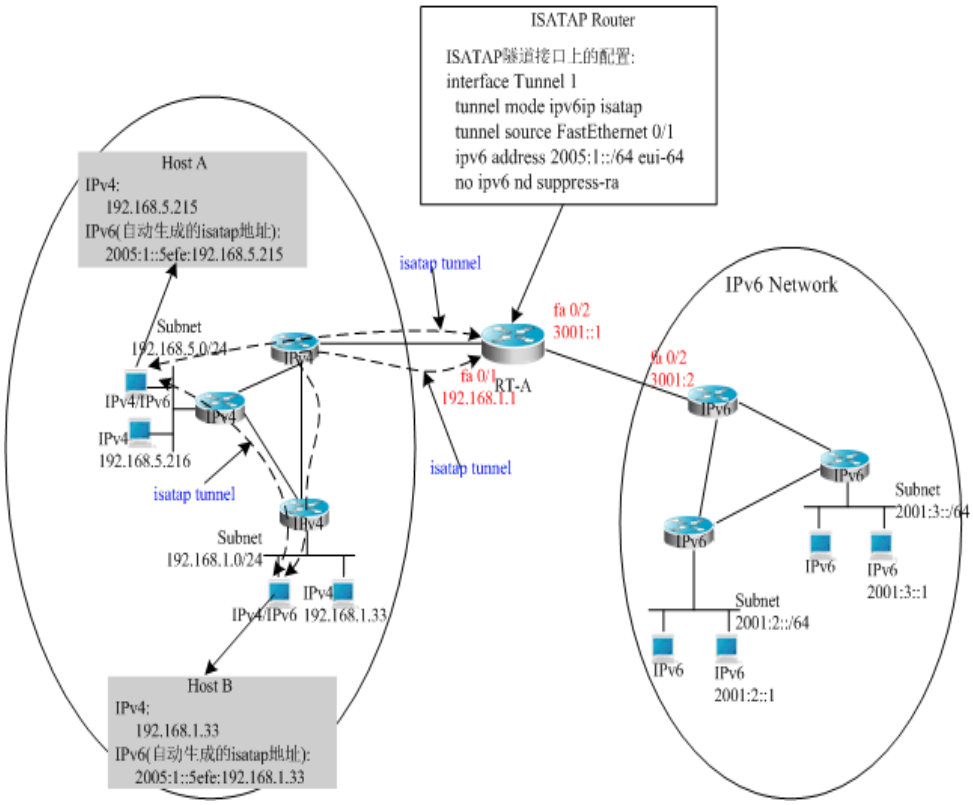


图 6

如上图，是一个使用 ISATAP 隧道的典型拓扑，ISATAP 隧道主要用于 IPv4 站点内部被隔离的 IPv4/IPv6 双栈主机之间进行通信，而 ISATAP 设备在 ISATAP 站点中的主要功能有 2 个：

- 接收站内 ISATAP 主机发来的路由器请求报文后应答路由器公告报文用来提供给站点内的 ISATAP 主机进行自动配置。
- 负责站点内的 ISATAP 主机和站点外的 IPv6 主机转发报文的功能

上图中当 Host A 和 Host B 发送路由器请求给 ISATAP Router，ISATAP Router 将应答路由器公告报文，主机收到该报文后进行自动配置同时也会生成各自的 ISATAP 地址。之后 Host A 和 Host B 的 IPv6 通信将通过 ISATAP 隧道进行。当 Host A 或者 Host B 要与站点外的 IPv6 主机通信时，那么 Host A 先将该报文通过 ISATAP 隧道发送到 ISATAP 路由器 RT-A 上，然后由 RT-A 转发到 IPv6 网络上。

上图中 ISATAP Router（RT-A）的配置如下：

```
# 连接 IPv4 网络的接口
interface FastEthernet 0/1
no switchport
ip address 192.168.1.1 255.255.255.0

# 配置 ISATAP 隧道接口
interface Tunnel 1
```

```
tunnel mode ipv6ip isatap
tunnel source FastEthernet 0/1
ipv6 address 2005:1::/64 eui-64
no ipv6 nd suppress-ra

# 连接 IPv6 网络的接口
interface FastEthernet  0/2
no switchport
ipv6 address 3001::1/64

# 配置到 IPv6 网络的路由
ipv6 route 2001::/64 3001::2
```

34.4.4 配置 ISATAP 和 6to4 隧道综合应用实例

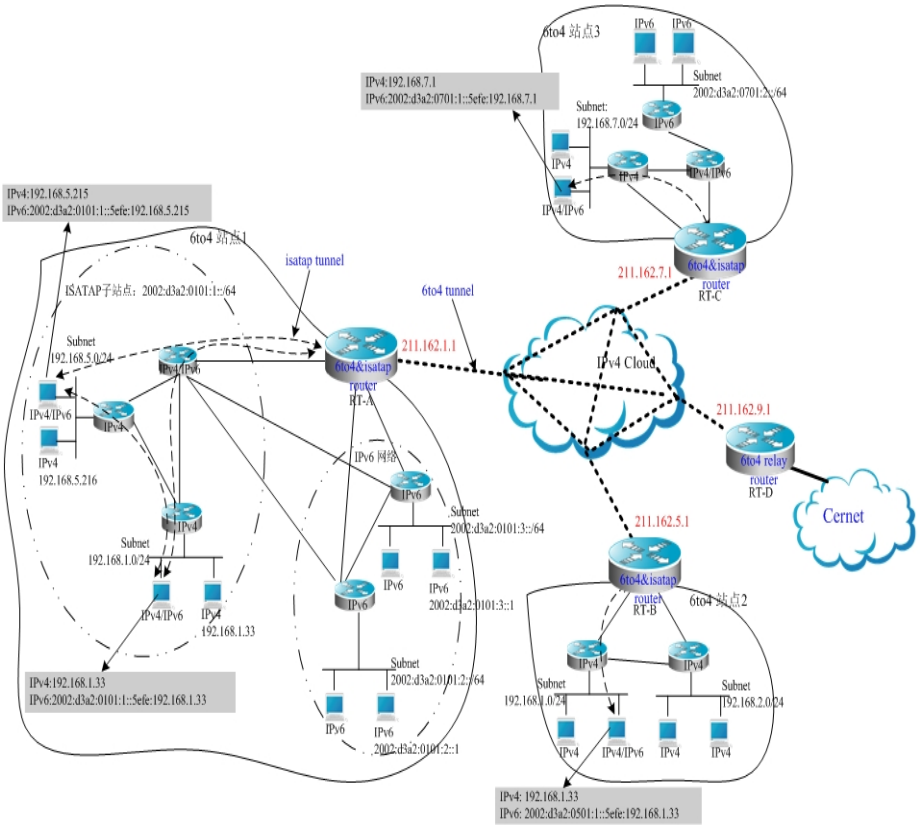


图 7

说明：

上图是一个 6to4 隧道和 ISATAP 隧道混合使用的例子。通过 6to4 隧道技术将各个 6to4 站点互联起来，并通过 **6to4 relay router** 将 6to4 站点接入 Cernet 网络，同时在 6to4 站点内部使用了 ISATAP 隧道技术，在站内被 IPv4 隔离的 IPv6 主机通过 ISATAP 隧道进行 IPv6 通信。

注意：

上图中所使用的全局 IP 地址包括 6to4 Relay 路由器的地址仅仅是为了举例方便，实际规划拓扑时应该使用真正的全局 IP 地址以及 6to4 Relay 的地址，目前有不少组织提供了免费公开的 6to4 Relay 路由器的地址。

下面分别介绍上图中 6to4 站点的边界路由器的配置，注意这里只列出相关的主要配置。

RT-A 的配置：

```
# 连接 Internet 网络的接口
interface GigabitEthernet 0/1
no switchport
ip address 211.162.1.1 255.255.255.0

# 连接站点内部的 IPv4 网络的接口
interface FastEthernet  0/1
no switchport
ip address 192.168.0.1 255.255.255.0

# 配置 ISATAP 隧道接口
interface Tunnel 1
tunnel mode ipv6ip isatap
tunnel source FastEthernet 0/1
ipv6 address 2002:d3a2:0101:1::/64 eui-64
no ipv6 nd suppress-ra

# 连接 IPv6 网络的接口 1
interface FastEthernet 0/2
no switchport
2002:d3a2:0101:10::1/64

# 连接 IPv6 网络的接口 2
interface FastEthernet  0/2
no switchport
2002:d3a2:0101:20::1/64

# 配置 6to4 隧道接口
interface Tunnel 2
tunnel mode ipv6ip 6to4
ipv6 enable
tunnel source GigabitEthernet 0/1

# 配置进 6to4 隧道的路由
ipv6 route 2002::/16 Tunnel 2
```

```
# 配置到 6to4 中继路由器 RT-D 的路由,以便可以访问 Cernet 网络
ipv6 route ::/0 2002:d3a2::0901::1
```

RT-B 的配置：

```
# 连接 Internet 网络的接口
interface GigabitEthernet 0/1
no switchport
ip address 211.162.5.1 255.255.255.0

# 连接站点内部的 IPv4 网络的接口 1
interface FastEthernet  0/1
no switchport
ip address 192.168.10.1 255.255.255.0

# 连接站点内部的 IPv4 网络的接口 2
interface FastEthernet  0/2
no switchport
ip address 192.168.20.1 255.255.255.0

# 配置 ISATAP 隧道接口
interface Tunnel 1
tunnel mode ipv6ip isatap
tunnel source FastEthernet 0/1
ipv6 address 2002:d3a2:0501:1::/64 eui-64
no ipv6 nd suppress-ra

# 配置 6to4 隧道接口
interface Tunnel 2
tunnel mode ipv6ip 6to4
ipv6 enable
tunnel source GigabitEthernet 0/1

# 配置进 6to4 隧道的路由
ipv6 route 2002::/16 Tunnel 2

# 配置到 6to4 中继路由器 RT-D 的路由,以便可以访问 Cernet 网络
ipv6 route ::/0 2002:d3a2::0901::1
```

RT-C 的配置：

```
# 连接 Internet 网络的接口
interface GigabitEthernet 0/1
no switchport
ip address 211.162.7.1 255.255.255.0

# 连接站点内部的 IPv4 网络的接口
interface FastEthernet 0/1
no switchport
ip address 192.168.0.1 255.255.255.0

# 配置 ISATAP 隧道接口
```

```
interface Tunnel 1
tunnel mode ipv6ip isatap
tunnel source FastEthernet 0/1
ipv6 address 2002:d3a2:0701:1::/64 eui-64
no ipv6 nd suppress-ra

# 连接 IPv6 网络的接口
interface FastEthernet 0/2
no switchport
2002:d3a2:0701:10::1/64

# 配置 6to4 隧道接口
interface Tunnel 2
tunnel mode ipv6ip 6to4
ipv6 enable
tunnel source GigabitEthernet 0/1

# 配置进 6to4 隧道的路由
ipv6 route 2002::/16 Tunnel 2

# 配置到 6to4 中继路由器 RT-D 的路由,以便可以访问 Cernet 网络
ipv6 route ::/0 2002:d3a2::0901::1
```

RT-D(6to4 Relay)的配置：

```
# 连接 Internet 网络的接口
interface GigabitEthernet 0/1
no switchport
ip address 211.162.9.1 255.255.255.0

# 连接 IPv6 网络的接口
interface FastEthernet 0/1
no switchport
2001::1/64
no ipv6 nd suppress-ra

# 配置 6to4 隧道接口
interface Tunnel 1
tunnel mode ipv6ip 6to4
ipv6 address 2002:d3a2::0901::1/64
tunnel source GigabitEthernet 0/1

# 配置进 6to4 隧道的路由
ipv6 route 2002::/16 Tunnel 1
```

35 OSPFv3 配置

OSPF 版本 2（RFC2328，OSPFv2）运行在 IPv4 下。在 RFC2740 中描述了 OSPF 版本 3（OSPFv3），其扩展 OSPFv2 协议，提供了对 IPv6 路由的支持。本文档简要描述了 OSPFv3 协议以及运行 OSPFv3 的配置。

注意：

在学习本文档之前，您必须先了解 OSPFv2 协议和相关配置。OSPFv3 协议扩展了 OSPFv2 协议，其协议内部主要运行机制和大部分配置仍然同 OSPFv2 保持一致。

35.1 OSPFv3 协议概述

OSPF 作为一种内部网关协议（Interior Gateway Protocol，IGP），运行在同一个自治域（AS）中的三层设备之间。

不同于矢量距离协议，OSPF 是一种链路状态协议。通过设备之间交换用以记录链路状态信息的各类型的 link-state advertisements(LSAs)，实现设备之间链路状态信息的同步，随后通过 Dijkstra algorithm 计算出 OSPF 路由表项。

OSPFv3 在 RFC2740 中描述，其支持 IPv6。本节描述了同 OSPFv2 相比，OSPFv3 在实现上的变化。

35.1.1 LSA 关联变化

正如上面描述，OSPF 是链路状态协议，其实现的基础是 LSA，通过 LSA，我们可以获知网络的拓扑和地址信息。同 IPv4 相比，IPv6 使用了 128 位的 IP 地址结构，使得对 LSA 的设计进行了相应的修改，先对各 LSA 类型描述如下：

- Router-LSAs (Type 1)

每台设备自己生成，描述了每台设备在指定区域内各链路的状态和到各链路的代价。同 OSPFv2 相比，OSPFv3 的 Router-LSAs 将只单纯表示链路的状态信息，而不再记录 router 所连网络地址信息，这些信息将通过新增的 LSA 类型获取。此外，在 OSPFv2 中，只允许为每台设备在每个区域生成一个 Router-LSA，而在 OSPFv3 中，允许生成多个 Router-LSAs，这样在进行 SPF 计算时，我们必须考虑这台设备生成的所有 Router-LSAs。Router-LSAs 和 Network-LSAs 共同描述了区域的链路拓扑图。

注意：

通过 Router-LSAs 上的标志位我们可以获知 router 是否是 area border

router(ABR，区域边界路由器)、 AS boundary routers (ASBR，自治系统边界路由器)或者是 virtual link(虚链路)的一端。

- Network-LSAs (Type 2)

Network-LSAs 只在广播网络或者 NBMA 网络中，由网络的 DR(指派路由器)生成，描述了在网络上指定区域内连接的所有 routers 信息。同 Router-LSAs 相同，Network-LSAs 也只将表示链路状态信息，而不再记录网络地址信息。Network-LSAs 和 Router-LSAs 共同描述了区域的链路拓扑图。

- Inter-Area-Prefix-LSAs (Type 3)

由区域的 ABR 为该区域生成，用来描述到达其他区域的网络信息。取代 OSPFv2 中 type 3 summary-LSAs，同 OSPFv2 相比，其使用了前缀结构来描述目的网络信息。

- Inter-Area-Router-LSAs (Type 4)

由区域的 ABR 为该区域生成，用来描述到达其他区域 ASBR 的路径信息，取代 OSPFv2 中 type 4 summary-LSAs。

- AS-external-LSAs (Type 5)

由 ASBR 生成该类型的 LSA，用来描述了到达 AS 外部的网络信息。通常这些网络信息是通过其他路由协议生成，同 OSPFv2 相比，其使用了前缀结构来描述目的网络信息。

- NSSA-LSA (Type 7)

同 type 5 AS-external-LSAs 的作用相同，所不同的是这是由 NSSA 区域的 ASBR 生成。

- Link-LSAs (Type 8)

在 OSPFv3 中，新增加的 LSA 类型，由每台设备为其所连接的每个链路生成，描述了该设备在当前链路上的链路本地地址和所有设置的 IPv6 地址前缀信息。

- Intra-Area-Prefix-LSAs (Type 9)

在 OSPFv3 中，新增加的 LSA 类型，其主要是为 Router-LSA 或者 Network-LSA 提供附加地址信息，所以其有两个作用：

1) 关联 network-LSA，记录 transit network 的前缀信息；
2) 关联 router-LSA，记录 router 在当前区域中，所有 Loopback 接口、点对点链路、点对多点链路、虚链路和 stub network 的前缀信息。

LSA 关联的其他主要变化：

- LSA 的泛洪范围变化

在 OSPFv2 中，LSA 的泛洪范围包括区域内部泛洪和 AS 内部泛洪。在 OSPFv3 中，还增加了链路本地泛洪范围，type 8 Link-LSAs 属于这一类型，即只能在链路本地范围内泛洪。

- 未知的 LSA 类型的处理

这是 OSPFv3 对 OSPFv2 的改进。

在 OSPFv2 中，在初始建立邻接关系过程中，需要进行数据库的同步。此时如果在数据库描述报文中存在无法识别的 LSA 类型，那么邻居关系将无法正常建立。如果在链路状态更新报文中，存在无法识别的 LSA 类型，那么将丢弃该类型的 LSA。

而在 OSPFv3 中，允许接收未知的 LSA 类型，通过在 LSA header 中记录的信息用来确定如何处理接收到的无法识别的 LSA 类型。

35.1.1.1 接口的设置

在 OSPFv3 中，基于接口配置的几点改变：

1. 接口要参与 OSPFv3 运行，必须在接口配置模式下明确启动。而在 OSPFv2 中，这是通过在 OSPF 路由配置模式下，通过 network 命令间接启动。
2. 如果配置接口参与 OSPFv3 运行，那么接口上所有的地址都将参与 IPv6 运行。而在 OSPFv2 中，所有地址都必须通过 **network** 命令启动。
3. 在运行 OSPFv3 的环境中，在同一个链路上能够允许运行多个 OSPF 实体，该链路所连的不同设备可以选择参与运行某一个 OSPF 实体。OSPFv2 并不支持该功能。

35.1.1.2 Router ID 的设置

每台运行 OSPFv3 进程的设备都必须使用一个 router ID 来标识，router ID 使用 IPv4 地址格式。

不同于 OSPFv2，OSPFv2 进程将自动获取 IPv4 地址作为 router ID，设备在启动 OSPFv3 进程后，用户必须使用 router-id 命令为 OSPFv3 进程配置 router ID，否则 OSPFv3 进程将无法启动。

35.1.1.3 认证机制的设置

OSPFv2 自身支持使用明文认证和基于 MD5 的密钥认证两种认证方式，OSPFv3 自身不提供认证，其将使用 IPSec 的认证机制。我们将在今后支持 IPSec 认证机制。

35.1.2 OSPFv3 基本配置

锐捷网络的 OSPFv3 协议有如下特点：

- 支持多实例 OSPF；
- 支持网络类型设置；
- 支持虚拟连接；

- 支持消极接口；
- 支持接口选择参与的 OSPF 实体；
- 支持存根区间(Stub 区域)；
- 支持路由重分发；
- 支持路由信息聚合；
- 支持定时器设置；

后续支持：

- 支持 NSSA 区域；
- 支持认证，OSPFv3 将使用 IPSec 认证机制；

缺省的 OSPFv3 配置：

路由器 ID		未定义
接口配置	接口类型	广播网络
	接口代价	未定义
	hello 报文发送间隔	10 秒
	邻接设备失效的时间	hello 报文发送间隔的四倍
	LSA 发送延迟	1 秒
	LSA 重传间隔	5 秒
	优先级	1
	数据库描述报文的 MTU 检查	开启
虚拟连接	虚拟连接	未定义
	hello 报文发送间隔	10 秒
	邻接设备失效的时间	hello 报文发送间隔的四倍
	LSA 发送延迟	1 秒
	LSA 重传间隔	5 秒
区域配置	区域	未定义
	stub 和 NSSA 区域的缺省路由代价	1
路由信息汇聚	区间路由汇聚	关闭
	外部路由汇聚	关闭
管理距离	区内路由	110

	区间路由	110
	外部路由	110
自动代价生成		打开 缺省的代价参考是 100 Mbps
状态更新信息调整时间		240 秒
最短路径优先算法定时器		收到拓扑改变信息到下一次开始调用 SPF 算法计算的延迟时间：5 秒； 两次计算至少间隔的时间：10 秒
路由重分发		关闭
路由信息过滤		关闭
被动接口		关闭

要运行 OSPFv3，在特权模式下，按照如下步骤进行：

命令	作用
configure terminal	进入全局配置模式。
ipv6 router ospf *process-id*	启动 OSPFv3 路由进程，进入 OSPFv3 配置模式。
router-id *router-id*	配置本设备运行 OSPFv3 时使用的 Router ID。
interface *interface-id*	进入接口配置模式。
ipv6 ospf *process-id* **area** *area-id* [**instance** *instance-id*]	在接口上启动 OSPFv3。 instance-id：设置参与 OSPFv3 的接口实体号，在连接同一个网络上的不同设备的接口可以选择参与不同的 OSPFv3 实体。
copy running-config startup-config	保存配置。

注意：

可以在接口配置模式下，先启动接口参与 OSPFv3，再配置 ospfv3 进程，配置完 ospfv3 进程后，接口将自动参与相应的进程

35.1.3 配置 OSPFv3 接口参数

在接口配置模式下，您可以根据实际应用的需要修改接口参数值。

要配置 OSPFv3 接口参数，在接口配置模式中进行执行以下命令：

命令	作用
ipv6 ospf *process-id* **area** *area-id* [**instance** *instance-id*]	设置接口参与 OSPFv3 路由进程。
ipv6 ospf network {**broadcast** |**non-broadcast** | **point-to-point** |**point-to-multipoint** [**non-broadcast**]} [**instance** *instance-id*]	设置接口的网络类型，缺省为广播网络类型。
ipv6 ospf cost *cost* [**instance** *instance-id*]	(可选)定义接口代价。
ipv6 ospf hello-interval *seconds* [**instance** *instance-id*]	(可选)设置接口上发送 Hello 报文的时间间隔。对于整个网络的所有节点，该值必须相同。
ipv6 ospf dead-interval *seconds* [**instance** *instance-id*]	(可选)设置接口上认为邻居失效的时间。 对于整个网络的所有节点，该值必须相同。
ipv6 ospf transmit-delay *seconds* [**instance** *instance-id*]	(可选)设置链路状态重传间隔。
ipv6 ospf retransmit-interval *seconds* [**instance** *instance-id*]	(可选)设置在接口上传送 LSA 时的延迟时间。
ipv6 ospf priority *number* [**instance** *instance-id*]	(可选)设置接口的优先级，用于选举指派路由器（DR）和备份指派路由器（BDR）。

使用以上命令的 **no** 模式，可以使配置的内容失效。

注意:

您可以根据实际需要修改接口的参数设置，但必须注意一些参数的设置必须同邻居保持一致，否则将无法建立邻居关系。这些参数包括：**instance**、**hello-interval**、**dead-interval**。

35.1.4 配置 OSPFv3 区域参数

OSPF 协议使用一种“分层结构”的思想，允许将网络分割成由一个“主干”连接的一组相互独立的部分，这些相互独立的部分被称为“区域(Area)”，“主干”的部分称为“主干区域”，其始终使用数值 0(或者 0.0.0.0)来表示。

使用这种分层结构，允许每台设备只保持自己所在区域的链路状态数据库，区域内的拓扑对区域外是不可见的。这样，使得每台设备的链路状态数据库都可以保持合理的大小，路由计算的时间、报文数量都不会过大。

在 OSPF 中，根据实际需要定义了两种类型的特殊区域：

- stub 区域

我们称之为存根区域。

如果一个区域处于整个网络的末梢部分，可以将该区域设计为 stub 区域。

如果区域设计为 stub 区域，其将学习不到 AS 外部路由信息(type 5 LSAs)，而在实际应用中，外部路由信息在链路状态数据库中又占有很大比重，所以 stub 区域内部设备将只学习到很少的路由信息，从而降低了运行 OSPF 协议所需要占用的系统资源。

stub 区域内部的设备如果要到达 AS 外部，可以通过 stub 区域的区域边界路由器发布的缺省路由信息(type 3 LSA)生成的缺省路由表项，从而到达 AS 外部。

- NSSA 区域 (Not-So-Stubby Area)

我们称之为不完全存根区域。

NSSA 区域是从 stub 区域扩展而来，也通过阻止向 NSSA 内部的设备泛洪 type 5 LSAs 来降低设备的资源的消耗。但是和 stub 区域不同，允许通过其他方式向 NSSA 注入一定数量的 AS 外部路由信息，即通过 type 7 LSAs 的方式注入到 NSSA 区域中。

目前锐捷还未实现 OSPFv3 的 NSSA 区功能；

要配置 OSPFv3 区域参数，在 OSPFv3 配置模式中进行执行以下命令：

命令	作用
area *area-id* **stub** [**no-summary**]	配置存根区域。 no-summary：将该区域配置完全存根区域，阻止 stub 区间上的区域边界路由器发送 type 3 信息进入 stub 区内。
area *area-id* **default-cost** *cost*	配置发送到 stub 区域的缺省路由的代价。

使用以上命令的 **no** 模式，可以使配置的内容失效。

注意：

配置为 stub 区后可以配置 default-cost 参数，此时若将该区改回普通区，default-cost 配置将会自动被删除。

35.1.5 配置 OSPFv3 虚拟连接

在 OSPF 中，所有区域必须连接到主干区域才能保证与其他区域的通信。如果某些区域无法直接与主干区域连接时，就需要通过虚拟连接同主干区域连接。

要建立虚拟连接，在 OSPFv3 配置模式下执行：

命令	作用
area *area-id* **virtual-link** *router-id* [**hello-interval** *seconds*] [**dead-interval** *seconds*] [**transmit-delay** *seconds*] [**retransmit-interval** *seconds*] [**instance** *instance-id*]	配置虚链路。

使用该命令的 **no** 模式，可以使配置的内容失效。

注意:

1. 不允许在 stub 区域和 NSSA 区域内创建虚拟连接；
2. 虚拟连接可以看成是一个特殊的接口，所以同普通接口的配置相同，必须保证虚拟连接两端配置的 **instance**、**hello-interval**、**dead-interval** 配置一致。

35.1.6 配置 OSPFv3 路由信息汇聚

如果没有路由汇聚，网络中的每台设备都要维持到每个网络的路由信息。使用汇聚后，就能将一些信息整合到一起，减轻了三层设备内部与网络带宽两方面的负担。随着网络规模的增大，路由汇聚的重要性也逐渐增大。

锐捷公司三层设备支持两种路由汇聚配置：区间汇聚与外部路由汇聚。

35.1.6.1 配置区间汇聚

OSPF 的 ABR 需要负责将一个区域内的路由通告到其它区域，如果该区域的路由地址是连续的，则 ABR 可以将这些路由信息聚合起来，通告出去。

要配置区间汇聚，在 OSPFv3 配置模式下执行：

命令	作用
area *area-id* **range** *ipv6-prefix*/*prefix-length* [**advertise**	**not-advertise**]	配置区间汇聚。

使用 **no area** *area-id* **range** *ipv6-prefix* /*prefix-length* 删除配置的区间汇聚。

35.1.7 配置 OSPFv3 接口度量的带宽参考值

OSPF 协议的度量是基于接口的带宽值，根据接口的带宽计算出接口的代价值。

例：接口带宽参考值为 100 Mbps，网络接口的带宽值为 10Mbps，则该网络接口自动计算生成的接口代价值为 100/10 = 10。

目前本公司的产品网络接口带宽缺省值为 100 Mbps。

要更改 OSPFv3 接口带宽参考值，在 OSPFv3 配置模式下执行以下命令：

命令	作用
auto-cost [**reference-bandwidth** *ref-bw*]	配置接口度量的带宽参考值。

注意：

您可以通过在接口配置模式下执行命令 **ipv6 ospf cost** *cost-value* 为指定接口设置代价，其优先选择高于根据度量参考值计算出来的代价值。

35.1.8 配置 OSPFv3 缺省路由

在 OSPF 协议中，提供多种方式生成缺省路由。

参看配置 OSPFv3 区域参数，在 Stub 区域将自动生成由 Type 3 的 LSA 表示的缺省路由。

此外，您还可以通过配置，产生一条由 Type5 的 LSA 表示的缺省路由，发布到整个 OSPF 自治系统内。在 OSPFv3 配置模式下，执行如下命令：

命令	作用
default-information originate [**always**] [**metric** *metric-value*] [**metric-type** *type-value*] [**route-map** *map-name*]	配置产生缺省路由。

使用 **no default-information originate** 命令，可以取消生成的缺省路由。

注意：

1) 该命令不允许在 Stub 区域内的设备上配置；

2) 一旦配置该命令，设备也将自动变成 ASBR。

35.1.9 配置 OSPFv3 定时器

OSPF 协议属于链路状态协议，当链路状态发生变化时，OSPF 进程将会触发 SPF 计算，您可以根据设计情况，如用如下命令配置进行 SPF 计算的延时时间，以及两次 SPF 计算的时间间隔。

在 OSPFv3 配置模式下，执行以下命令：

命令	作用
timers spf *delay holdtime*	配置进行 SPF 计算的延时时间，以及两次 SPF 计算的时间间隔。

35.1.10 配置 OSPFv3 路由重分发

路由信息重分发可以将一个路由协议的路由信息重新分发到另一个路由协议。

要配置 OSPFv3 路由重分发，在 OSPFv3 配置模式下执行：

命令	作用
redistribute *protocol* [**metric** *metric-value*] [**metric-type** *type-value*] [**route-map** *map-tag*] [**match** [**internal** | [**external** | **nssa-external** [1|2]]]]	重新分发其他协议的路由信息，同时可以设置重新分发的条件。 目前 OSPFv3 支持 static、connect、rip 与 ospf 路由重分发。 重分布 ospf 路由时，允许配置 match 参数，表示只重分布指定子类型的 ospf 路由。
default-metric *number*	配置重分发信息的缺省度量值。

使用 **no redistribute** *protocol* 模式，可以取消路由信息重分发。

注意:

目前锐捷产品暂不支持 tag 参数的应用。

35.1.11 配置 OSPFv3 被动接口

为了防止网络中的其他三层设备动态的学习到本设备的路由信息，可以在路由协议配置模式下，设定网络接口为被动接口。

对于 OSPFv3 协议，如果将网络接口配置为被动网络接口，则该网络接口将不收发任何 OSPF 报文。

要配置 OSPFv3 被动接口，在 OSPFv3 配置模式下执行：

命令	作用
passive-interface {**default** | *interface-type interface-number* }	配置被动接口。

使用 **no passive-interface** {*interface-id* | **default**}命令，可以取消被动接口配置。

35.1.12 OSPFv3 的调试和监控

OSPFv3 进程支持了丰富的调试命令和监控命令。

35.1.12.1 OSPFv3 的调试命令

在特权配置模式下，使用如下命令启动 OSPFv3 进程的调试命令：

命令	作用
debug ipv6 ospf events	显示出 OSPFv3 事件信息。
debug ipv6 ospf ifsm	显示出接口状态机事件和变迁
debug ipv6 ospf lsa	显示出 OSPFv3 lsa 相关信息。
debug ipv6 ospf nfsm	显示出邻居状态机事件和变迁。
debug ipv6 ospf nsm	ospf 与 NSM 模块相关信息。
debug ipv6 ospf packet	显示 ospf 报文信息。
debug ipv6 ospf route	显示 OSPF 路由计算及添加信息。

使用上述 **undebug** 命令关闭上述打开的 **debug** 命令。

注意：

debug 命令更多的是提供给技术人员。

使用 **debug** 命令后，将会一定程度上影响系统的性能，所以请使用完 **debug** 命令后，及时使用 **undebug** 命令关闭系统性能。

35.1.12.2 OSPFv3 的监控命令

在特权配置模式下，使用如下命令启动 OSPFv3 进程的监控命令：

命令	作用
show ipv6 ospf	显示 OSPFv3 进程信息
show ipv6 ospf [*process-id*] **database** [*lsa-type* [**adv-router** *router-id*]]	显示 OSPFv3 进程数据库信息。
show ipv6 ospf interface [*interface-type interface-number*]	显示 OSPFv3 进程接口信息。

show ipv6 ospf [*process- id*] **neighbor** [*interface-type interface-number* [**detail**]| *neighbor-id* |**detail**]	显示 OSPFv3 进程邻居信息。
show ipv6 ospf [*process-id*] **route**	显示 OSPFv3 路由信息。
show ipv6 ospf [*process-id*] **topology** [**area** *area-id*]	显示 OSPFv3 每个区拓扑信息。
show ipv6 ospf [*process-id*] **virtual-links**	显示 OSPFv3 进程虚链路信息。

36 IPv4 组播路由配置

36.1 概述

IPV4 组播是一种允许一个组播流向多个接收者转发的的网络技术。组播源发出特定组的组播数据流，而只有加入该组播组的主机才能接收到数据包。组播可以大大的节省网络带宽，因为无论有多少个接收主机，在整个网络的任何一条链路上只传送单一的数据包。

组播使用 IANA 规定的 D 类网络地址。D 类地址的最高位为 1110。因此，组播地址范围为 224.0.0.0 到 239.255.255.255。不过其中并不是所有地址都可以被用户使用，范围为 224.0.0.1 到 224.0.0.255 的组地址被保留为协议使用或其他地址。比如 224.0.0.1 表示所有组播主机地址，224.0.0.2 表示所有组播设备地址。

组播数据报文为 UDP 报文，只是一种尽力保证的服务，不提供类似于 TCP 的可靠传输和差错控制。

构成组播的应用需要发送方和接收方。发送方发送的组播数据流必须包含组地址，这个组地址用以区分不同的组播流，而接收方必须事先加入相关组，才能接收到到该组的数据报文。

组成员的关系是动态的，主机可以随时加入或者离开某个组。如果需要，一台主机可以同时加入多个组播组。因此，组的活动状态和组成员的个数可随着时间而发生变化。

设备通过运行组播路由协议（例如 PIM-DM，PIM-SM 等）来维护转发组播报文的路由表，通过 IGMP 协议来维护在直连网段上组成员的状态。主机通过发送 IGMP 相当的协议报文，来决定加入或者离开相应在的组播组。

IPV4 组播技术适合于“一对多”的多媒体应用。

36.1.1 组播路由实现

在设备软件中，组播路由包括如下协议：

- IGMP：运行于组播设备和主机之间，跟踪学习组成员的关系。
- PIM-DM：密集模式组播路由协议，运行在组播设备之间，通过建立组播路由表来实现组播转发。
- PIM-SM：稀疏模式组播路由协议，运行在组播设备之间，通过建立组播路由表来实现组播转发。

下图显示了在 IPV4 组播环境中应用到的组播协议的作用位置：

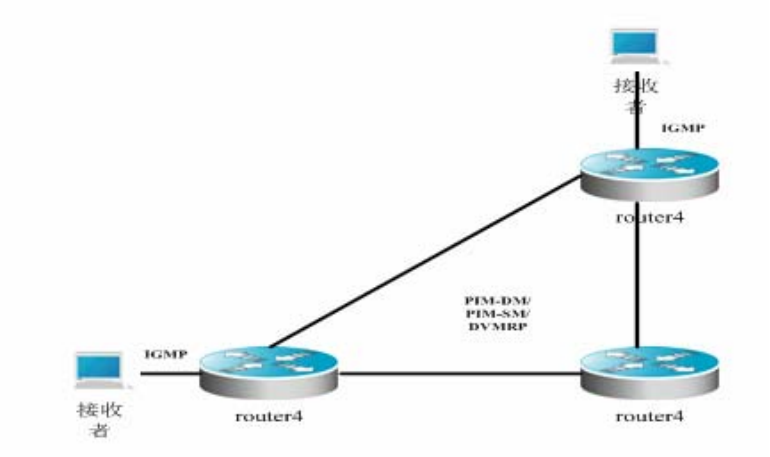


图 1　IPV4 组播环境中应用到的组播协议图

36.1.2 IGMP 概述

要实现 IPV4 组播，组播主机、设备必须能支持 IGMP 操作。该协议是主机用来通知设备与他们直连网段的组播成员关系，以决定对组播流的转发。利用从 IGMP 获得的信息，设备维护一张组播成员表，该表是基于每个接口和每个组的。当一个接口至少有一个主机是该组的组成员时，组播成员表就被创建。

目前 IGMP 支持 IGMPv1、IGMPv2、IGMPv3，其中 IGMPv2 基本是在 IGMPv1 的基础上，通过增加离开报文以使主机可以主动要求离开组播组。IGMP 协议的行为可以分成两个部分：主机行为和设备行为。

36.1.2.1　IGMPV1

在版本 1 中，只有两种报文类型：

- Membership query
- Membership report

主机发送 report 报文表示加入，设备定期发送 query 报文，确认一个组至少有一个主机存在，当一个组的所有主机都不存在时，设备将该组删除。

36.1.2.2　IGMPV2

在版本 2 中，规定了四种报文类型：

1) Membership query
2) Version 1 membership report
3) Version 2 membership report
4) Leave group

版本 2 与版本 1 基本类似，但在主机的离开机制上做了改进。版本 2 支持主机发送 leave 报文通知主机离开特定组，设备再发送 Query 确认该组内是否仍有主机存在。这样加入和离开的效率得到了提高。

在运行 IGMP 协议的组播网络中，会有一个被选为查询者的组播设备负责发送 IGMP 查询报文。这个查询者是通过选举产生的。起初所有设备都处于查询者状态。当设备收到的查询报文时，会将报文的源 IP 地址进行比较，IGMPv1 以 IP 地址高者为查询者，IGMPv2/IGMPv3 以 IP 地址低者作为查询者。另外，如果接收到不同版本的 IGMP 查询报文，以低版本的查询报文作为查询者。

当查询者设备失效时，会重新进行查询者的选举过程。非查询者设备维护其他查询者当前的间隔定时器。每次设备收到成员关系查询报文时，都重置此定时器。如果定时器超时，则此设备认为自己是查询者，开始发送查询报文，新一轮查询者设备选举又开始了。

查询者设备必须定期发送成员关系查询请求，查询者设备维护一个查询间隔定时器。当发送成员关系查询报文时，重置此定时器。间隔定时器超时时，查询者设备发送相应的查询报文。

在设备第一次出现即新增一个设备时，将发送一系列一般查询报文，以查看哪些接口上的主机需要接收哪些组，达到快速收敛的目的。设备发送的普通查询报文次数基于设备配置的启动查询计数值。初始普通查询报文间的间隔由启动查询间隔值定义。

查询者设备收到离开报文时，必须发送特定组的成员关系查询报文以查看该主机是否是该接口上最后一个离开组的。设备在停止为那个组转发数据报文之前发送多次这种特定组查询报文，发送次数等于最后的成员查询次数。设备发送多个特定组成员关系查询以确保那个组内没有成员了。每隔最后成员查询间隔秒数发送一个这样的查询，以隔开查询。如果设备都收不到主机的应答时，设备停止在相应接口上为组播流转发组播数据。

36.1.2.3 IGMPV3

在 IGMPv1 和 v2 的应用中存在以下几个缺陷:

- 缺乏有效控制组播源的手段。
- 由于不知道组播源的位置，组播路径的建立比较困难。
- 发现一个唯一的组播地址十分困难，可能出现多个的组播组使用同一个组播地址的情况。

在 IGMPv1/v2 的基础上，IGMP V3 提供了过滤组播功能。在 IGMP v1/v2 中，主机只根据组地址来决定加入某个组并从任何一个源接收发给该组地址的组播流。而使用 IGMP V3 的主机不仅可以通告该主机所希望加入的组播组，同时还通告该主机所希望接收的组播源的地址。主机可以通过一个包括列表或一个排除列表来指明希望从哪些源能接收组播流。同时 IGMP v3 带来的另外一个好处是节省带宽，避免不需要的、非法的组播数据流占用网络带宽，这尤其在多个组播源共用一个组播地址的网络环境中表现明显。IGMP v1 和 IGMP v2 也可以实现某种意义上的“源地

址过滤”，但它是在组播流接收端上做的。如下图所示：组播源有 S1 和 S2，发送同一组播地址 G 的数据流。S1、S2 的组播流会向所有接收 G 的主机发送，而如果主机 A 只想接收 S1 的，为避免 S2 数据流的干扰，只能用相应的客户端软件进行终端上的过滤。

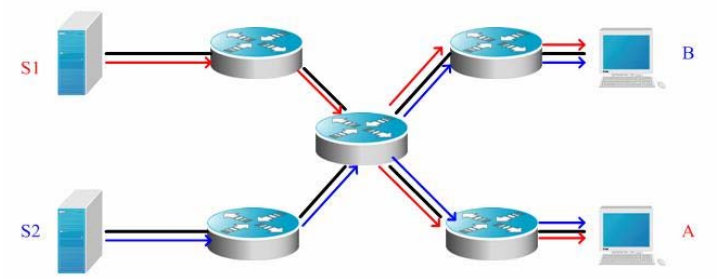


图 2 不进行源过滤的组播路由转发

而如果网络内的设备支持 IGMP v3，主机 A 只想接收 S1 来的数据流，发送 join G include S1 的 IGMPv3 报文；主机 B 只想接收 S2 来的数据流，发送 join G include S2 的 IGMPv3 报文，于是数据流的转发就如下所示。节省了一部分带宽。

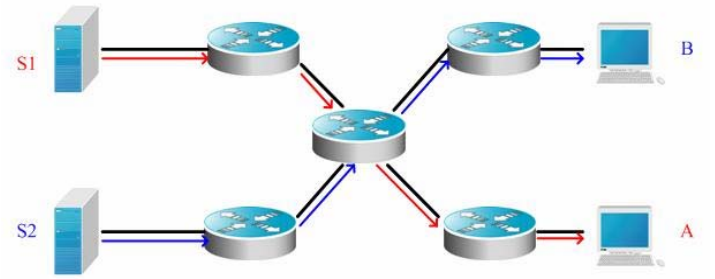


图 3 进行源过滤的组播路由转发

同版本 2 对比，版本 3 的规定了以下两种报文类型：

- Membership Query
- Membership Report

其中 Membership Query 分为三种：

- General Query：用于查询接口下所有组播成员信息；
- Group-Specific Query：用于查询接口下指定组的成员信息；
- Group-and-Source-Specific Query：该类型为 IGMPv3 中新增加的，用于查询接口下是否有成员需要接收指定源列表中的源所发出的特定组的组播流。

同 IGMPV2 中的 Membership Report 不同， IGMPv3 中发出的 Membership Report 报文中可以包含多个组的信息。IGMPv3 的 report 报文的目的地址为 224.0.0.22,它可以携带一个或者多个的组记录，在每个组记录中，包含了组地址和源地址列表信息。组记录的类型如下：

- IS_IN：表示组播组与组播源列表之间的过滤模式为 INCLUDE，即只接收从指定组播源列表发往该组播组的组播数据。如果此时的指定组播源列表为空，则表示离开该组播组，相当于 IGMPv2 的 leave 报文。

- IS_EX：表示组播组与组播源列表之间的过滤模式为 EXCLUDE，即只接收从指定组播源列表之外的组播源发往该组播组的组播数据。
- TO_IN：表示组播组与组播源列表之间的过滤模式将由 EXCLUDE 转变为 INCLUDE。
- TO_EX：表示组播组与组播源列表之间的过滤模式将由 INCLUDE 转变为 EXCLUDE。
- ALLOW：表示增加从某些组播源接收组播数据。如果当前的对应关系为 INCLUDE，则向现有组播源列表中添加这些组播源；如果当前的对应关系为 EXCLUDE，则从现有组播源列表中删除这些组播源。
- BLOCK：表示不再希望从某些组播源接收组播数据。如果当前的对应关系为 INCLUDE，则从现有组播源列表中删除这些组播源；如果当前的对应关系为 EXCLUDE，则向现有组播源列表中添加这些组播源。

基于兼容性考虑，IGMPv3 也能够识别 IGMPv1/IGMPv2 的协议报文。

36.1.3 PIM-SM 概述

PIM(Protocol Independent Multicast)由 IDMR（域间组播路由）工作组设计，顾名思义，PIM 不依赖于某一特定单播路由协议，它可利用各种单播路由协议建立的单播路由表完成 RPF 检查功能，而不是维护一个分离的组播路由表实现组播转发。由于 PIM 无需收发组播路由更新，所以与其它组播协议相比，PIM 开销降低了许多。PIM 的设计出发点是在 Internet 范围内同时支持 SPT 和共享树，并使两者之间灵活转换，因而集中了它们的优点提高了组播效率。PIM 定义了两种模式：稠密模式(Dense-Mode)和稀疏模式（Sparse-Mode）。

PIM-SM(Protocol Independent Multicast Sparse Mode)是一种稀疏模式的组播路由协议。在 PIM-SM 域中，运行 PIM-SM 协议的设备周期性地发送 Hello 消息，用以发现邻接的PIM-SM 设备，并且负责在多路访问网络中进行指派设备DR的选举。这里，DR 负责为其直连组成员朝着组播分发树根节点方向发送“加入/剪枝”消息，或是将直连组播源的数据发向组播分发树。

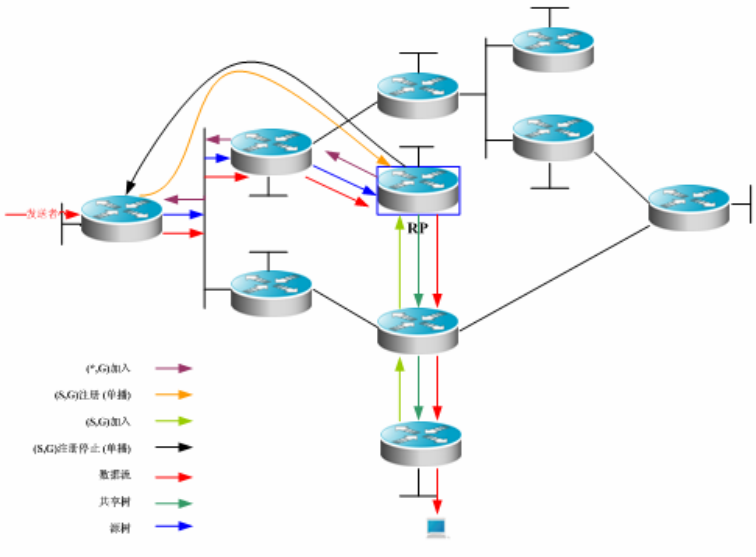


图 5　pim-sm 显式的加入机制

PIM-SM 通过建立组播分发树来进行组播数据包的转发。组播分发树分为两种：以组 G 的 RP 为根的共享树(Shared Tree)和以组播源为根的最短路径树(Shortest Path Tree)。PIM-SM 通过显式地加入/剪枝机制来完成组播分发树的建立与维护。如上图所示：当接收端 DR 收到一个发自接收端的 report 报文，它就会向着组 G 的 RP 方向逐跳组播发出一个(*.G)加入信息用以加入共享树；源主机发送组播数据时，源的 DR 收到组播数据时，将其封装在注册消息内，并单播至 RP，RP 再将源的解封装数据包沿着共享树转发到各个组成员；RP 朝着源方向第一跳设备发送（S，G）加入信息，用以加入此源的最短路径树，这样源的数据包将沿着其最短路径树不加封装地发送到 RP；当第一个组播数据沿此树到达时，RP 向源的 DR 发送注册-停止消息，以使 DR 停止注册封装过程。此后，这个源的组播数据不再注册封装，而是先沿着源的最短路径树发送到 RP，再由 RP 将其沿着共享树转发各个组成员。当不再需要组播数据时，接收端 DR 向着组 G 的 RP 逐跳组播剪枝消息用以剪枝共享树。

PIM-SM 中还涉及到根节点 RP 的选择机制。PIM-SM 域内配置了一个或多个候选自举设备(Candidate-BSR).应用一定的规则从中选出自举设备(BSR)。PIM-SM 域中还配置了候选 RP 设备(Candidate-RP),这些候选 RP 将包含了它们的地址及可以服务的组播组等信息的包单播至 BSR 设备。BSR 定期生成包括一系统候选 RP 以及相应的组地址的”自举”消息。”自举”消息在整个域中逐跳发送。设备接收并保存这些“自举”消息。若 DR 从直连主机收到了某组的成员关系报告后，如果它没有这个组的路由项，DR 将使用一个 hash 算法将组地址映射至一个可以为该组服务的候选 RP。然后 DR 将向 RP 方向逐跳组播”加入/剪枝”消息。若 DR 从直连主机收到组播数据包，DR 将使用 hash 算法将组地址映射至一个可以为该组服务的候选 RP，然后 DR 将组播数据封装在注册消息中单播到 RP。

PIM-SM 与基于“扩散/剪枝”模型的 PIM-DM 根本差别在于 PIM-SM 是基于显式加入模型，即接收者向 RP 发送加入消息，而设备只在已加入某个组播组输出接口上转发那个组播组的数据包。PIM-SM 采用共享树进行组播数据包转发。每一个组有一个汇合点（Rendezvous Point: RP），组播源沿最短路径向 RP 发送数据，再由 RP 沿最短路径将数据发送到各个接收端。这一点类似于 CBT，但 PIM-SM 不使

用核的概念。PIM-SM 主要优势之一是它不局限于通过共享树接收组播信息，还提供从共享树向 SPT 转换的机制。尽管从共享树向 SPT 转换减少了网络延迟以及在 RP 上可能出现的阻塞，但这种转换耗费了相当的设备资源，所以它适用于有多对组播数据源和网络组数目较少的环境。

PIM-SM 使用共享树和 SPT 来分发组播帧。此时，假设其他设备都不想收到这些组播，除非有明确的加入声明。当一个主机加入了一个组，与该主机相连的设备就要用 PIM 加入消息通知根，即 RP。这一加入消息通过设备依次传递，建立了一个共享树的结构。于是，RP 记录这一传递路径，同时也记录从发送组播源的第一跳设备（DR）发来的注册消息，根据这两个信息，完善共享树。枝叶信息的更新是通过定期的查询信息实现。在使用共享树时，组播源首先发送组播到 RP，这样才能保证所有的接收者能收到。

PIMv2 BSR 是一种分发 group-to-RP 消息到所有设备的方法。它消除了为每个设备设置 RP 的需要。BSR 使用 hop-by-hop 扩散 BSR 消息来分发映射信息。首先，BSR 是从设备中选举出来的。选举方法类似于 STP 中选举 root-bridge 的过程，使用一个优先级值，每个 BSR 设备检查 BSR 消息，并只转发优先级比它自己高的或相等（IP 地址比它高）的 BSR 消息。被选中的 BSR 发送 BSR 消息到 all-PIM-routers multicast group (224.0.0.13) ，TTL 为 1。邻居的 PIMv2 设备收到后，再将它组播出来，并再设置 TTL 为 1。这样，BSR 消息一跳一跳地被所有设备收到。由于消息中包括 BSR 的 IP 地址，所以侯选的 BSR 能通过该消息获知那个设备为当前的 RP。侯选 RP 发送候选的 RP 公告以宣称在哪些地址范围内他们可以成为 RP，BSR 将他们保存在自己的本地候选 RP 缓存中，BSP 定期将本地候选 RP 发送通知所有的 PIM 设备。这些消息同样一跳一跳地到达各设备。

36.1.4 PIM-DM 概述

PIM-DM（Protocol Independent Multicast-Dense Mode）是一种密集模式的组播路由协议，适用于网络规模比较小、组播成员相对集中的情况。因为 PIM-DM 不依赖于任何特定的单播路由协议，所以被称作是协议无关的（Protocol Independent）组播路由协议。PIM-DM 在 RFC 3973 文档中定义。

PIM-DM 设备之间通过 Hello 消息来发现邻居。一旦 PIM-DM 设备启动，它就周期性地在每个配置了 PIM-DM 的接口上发送 Hello 消息。Hello 消息有一个保持时间（Hello Hold Time）字段，这个时间参数定义了邻居等待下一个消息的最长时间。如果邻居在这个时间内没有接收到发送 Hello 消息的设备的下一个 Hello 消息，就宣布这个设备已经死亡。

PIM-DM 使用扩散与剪枝（flood and prune）来建立组播树。PIM-DM 假定当组播源开始发送组播数据报文时，网络中的所有系统都需要接收该报文，因此报文被转发给每一个系统。从设备上游接口接收到的报文都要经过 RPF（Reverse Path Forwarding，反向路径转发）检查，没有通过 RPF 检查的报文将被丢弃。对于通过了 RPF 检查的组播报文，设备根据报文的（S, G）对，即根据组播报文的源地址和组地址计算外出接口。如果计算出的外出接口不为空，则对该（S, G）对建立一个外出接口的表项，并且将该组播报文由外出接口转发；如果计算出的外出接口为空，则向 RPF 发送一个剪枝报文，通知上游邻居不要再向本接口转发来自该（S, G）的组播报文。上游接口接收到剪枝报文以后，把发送该剪枝报文的接口记为剪

枝状态（Pruned），并设置一个剪枝状态计时器。这样就建立了一棵以组播源为根的组播转发树。

PIM-DM 使用 Assert 机制来消除冗余路由。

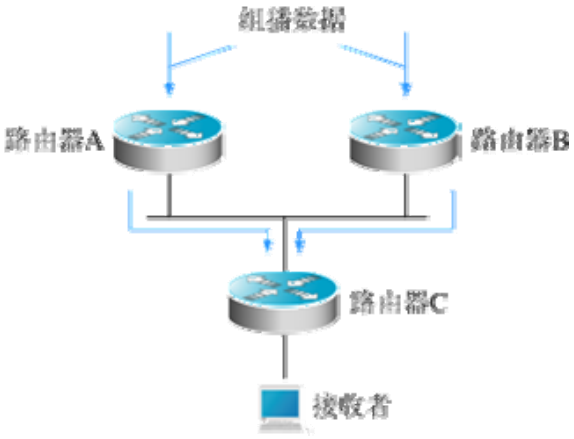


图 4 PIM-DM 的 Assert 机制

如图 1 所示，组播数据报文同时到达设备 A 和设备 B 时，设备 A 和设备 B 都向设备 C 转发，这时设备 C 就会收到同一份报文的两个拷贝，这是不允许的。因此必须使用某种机制，在设备 A 和设备 B 中间选择一个向设备 C 转发组播数据报文，而另一个则不向设备 C 转发，这就是 PIM-DM 中的 Assert 机制。

PIM-DM 使用状态更新消息（State Refresh Message）来更新网络的状态信息。与组播源直接相连的设备定期向下游设备发送状态更新消息，以通告网络的拓扑变化情况。收到状态更新消息的设备通过修改消息中的某些字段把本机的拓扑状态信息也加入到消息中，然后发送给下游设备。到达叶子设备时，整个网络的状态信息从上到下都得到了更新。

PIM-DM 使用嫁接（Graft）机制来重新建立与上游设备的连接。如果处于剪枝状态的下游设备的网络拓扑状态发生了变化，需要接收来自某个（S, G）对的组播数据报文，可以向上游设备发送嫁接消息。上游设备收到这条嫁接消息以后，回应一条嫁接确认（Graft-Ack）消息，并重新向该设备接口转发组播数据报文。

36.2 基本的组播路由配置

配置基本的组播路由配置如下：

- 启动组播路由转发
- 配置组播路由的 RPF 检查

36.2.1 启动组播路由转发

启动组播路由转发功能后，组播数据报文及协议报文才能够被组播相关协议接收处理。在全局配置态下输入以下命令启动组播路由转发：

命令	目的

Ruijie(config)# **ip multicast-routing**	启动组播路由转发。
Ruijie(config)# **no ip multicast-routing**	关闭组播路由转发。

注意：

在锐捷的产品 S3760 上，组播路由转发功能和 IGMP SNOOPING 互斥。

36.2.2 配置组播路由的 RPF 检查模式

配置设备的组播路由 RPF 检查模式。

在全局配置态下输入以下命令配置组播路由的检查模式，默认为 SVI 模式：

命令	目的
Ruijie(config)# **ip multicast-rpf** *rpf_mode*	配置 RPF 检查模式。 rpf_mode： routed-port，上游接口为路由口时进行检查。 Svi，上游接口为 SVI 接口时进行检查。

说明：

S3760 设备在默认情况下支持对 SVI 口进行 RPF 检查，不支持对路由口进行检查。通过配置组播路由的 RPF 检查模式，可配置对路由口进行 RPF 检查。该配置命令仅在 S3760 设备上支持。

36.3 配置组播路由特性

高级的 IPV4 组播核心功能配置包括：

- 配置 TTL 阈值（可选）
- 限制能够加入到 IPV4 组播路由表中的条目数量（可选）
- 设置特定 IP 组范围的 IPV4 组播边界（可选）
- 配置静态 IPV4 组播播路由（可选）
- IPV4 组播路由的监控和维护（可选）

36.3.1 配置 TTL 阈值

如果要对接口通过的数据报文的 ttl 进行限制，可以通过配置 TTL 阈值。

在接口配置模式下，使用 **ip multicast ttl-threshold** 配置接口上允许通过的组播报文 TTL 的阈值，使用 **no ip multicast ttl-threshold** 恢复缺省值，缺省值为 1。当 TTL 阈值配置为 0 时，表示相应接口将不作为数据流的出口。

命令	目的
Ruijie(config-if) **#ip multicast ttl-threshold** *ttl-value*	配置接口的 TTL 阈值。 ttl-value：<0-255>

36.3.2 限制能够加入到 IPV4 组播路由表中的条目数量

在全局配置模式下，使用 **ip multicast route-limit** *limit* [*threshold*] 限制能够加入到组播路由表中的条目数量，使用 **no** 命令恢复缺省值。组播路由条目的缺省值为 1024。

命令	目的
Ruijie(config) # **ip multicast route-limit** *limit* [*threshold*]	限制能够加入到组播路由表中的条目数量。 limit：能够加入到组播路由中的数量 1~2147483647。缺省值为 1024。 threshold：（可选）触发产生警示消息的组播路由数量。缺省值为 2147483647. 注意：不同型号设备的硬件资源所限，超过硬件表项的组播报文将需要软件转发，这将占用设备的 cpu 资源，导致系统性能下降。

36.3.3 设置特定 IP 组范围的 IPV4 组播边界

在接口配置模式下，使用 **ip multicast boundary** *access-list* 配置接口成为特定 IP 组范围的 IPV4 组播边界，使用 **no** 命令恢复缺省值。

命令	目的
Ruijie(config-if) # **ip multicast boundary** *access-list* { | **in** | **out** |}	设置特定 IP 组范围的 IPV4 组播边界。支持 IP 标准的数字 acl 或者命名指定特定的 IP 组范围 注意：这条命令关联的 ACL 只支持对目的 IP 地址的匹配过程，不支持对组地址和源地址的匹配。

该命令可以对和该 IP 组范围相关的 IGMP 、PIM-SM、PIM-DM 协议报文进行过滤，组播数据流不会从该组播边界接口流入和流出。

36.3.4 配置 IPV4 组播静态路由

组播静态路由允许组播转发路径不同于单播路径。组播报文转发时都会进行 RPF 检查：报文的实际接收接口是期望接收的接口（该接口就是到达发送方的单播路由下一跳接口）。如果单播的拓扑和组播的拓扑一致，这样的检查是合理的。但是，在某些情况下，还是希望单播的路径和组播的路径有所不同。

通过配置组播静态路由，能够使设备根据配置信息进行 RPF 检查，而不是单播路由表。因此，组播报文使用隧道，单播报文不走隧道。组播静态路由只存在本地，并不会宣告出去或者进行路由转发。

在全局配置态下，使用如下命令来配置组播静态路由。

命令	目的
Ruijie(config) # **ip mroute** *source-address mask* [**bgp | isis | ospf | static**] { *v4rpf-address* |*interface-type interface-number*} [*distance*]	配置组播静态路由。并可以指定这些路由的协议类型。Distance：<1-255>

说明：

如果要指定静态组播路由的出口而非下一跳 ip，那么该出口必须为点对点类型的。

36.3.5 配置组播流二层流向控制

如果需要控制端口的数据流流向，可以使用该命令。可以控制组播流在二层的流向。对于某一条组播流可以配置多条命令，也就是说配置多个被允许转发的端口。一旦为某组播流配置了流向控制，该组播流只可能由这些已配置的端口转发出去。其他未被允许的端口将不允许转发组播流。

命令	目的
Ruijie(config) # **ip multicast static** *source-address group-address interface-type interface-number*	控制二层端口的数据流流向。配置的静态出口必须为二层端口。

该命令只控制组播流在端口上的转发行为，不直接影响组播协议对协议报文的处理。但是由于组播协议（如：PIM-DM 或 PIM-SM）的某些特性是依赖组播数据流驱动的，所以组播路由协议的行为仍然可能被影响。

36.3.6 组播路由的监控和维护

在特权配置模式下，使用如下命令来查看 v4 组播转发表信息

命令	目的
Ruijie# **show ip mroute** [*v4group-address*] [*v4 source-address*] [**dense**|**sparse**][**summary**|**coun t**]	查看 v4 组播转发表信息

在特权配置模式下，使用如下命令来删除 v4 组播转发表

命令	目的
Ruijie# **clear ip mroute** {***** | *v4group-address* [*v4source -address*]}	删除 v4 组播转发表

在特权配置模式下，使用如下命令来复位 v4 组播转发表统计信息

命令	目的
Ruijie# **clear ip mroute statistics** {***** | *v4 group-address* [*v4source -address*]}	复位 v4 组播转发表统计信息

在特权配置态下，使用如下命令来显示特定 v4 源地址的 RPF 信息

命令	目的
Ruijie# **show ip rpf** *v4source-address*	显示特定 v4 源地址的 RPF 信息

在特权配置态下，使用如下命令来显示 v4 组播接口的信息

命令	目的
Ruijie# **show ip mvif** [*interface-type interface-number*]	显示 v4 组播接口的信息

在特权配置态下，使用如下命令来查看组播核心的运行过程

命令	目的
Ruijie# **debug nsm mcast all**	查看组播核心的运行过程

在特权配置态下，使用如下命令来查看 v4 组播核心对协议模块进行通信过程

命令	目的
Ruijie# **debug nsm mcast fib-msg**	查看 v4 组播核心对协议模块进行通信过程

在特权配置态下，使用如下命令来查看 v4 组播核心关于接口运行过程

命令	目的

Ruijie# **debug nsm mcast vif**	查看 v4 组播核心关于接口运行过程

在特权配置态下，使用如下命令来查看 v4 组播核心关于接口和表现统计信息的处理过程

命令	目的
Ruijie# **debug nsm mcast stats**	查看 v4 组播核心关于接口和表现统计信息的处理过程

36.4 IGMP 配置任务列表

IGMP 配置任务包括以下各项，但只有一些项配置是必选的，其它任务可以根据网络的具体需要决定是否要执行。注意：以下命令必须在三层接口下配置。

- 配置开启 IGMP 服务 （必选）
- 配置 IGMP 的版本（必选）
- 配置最后成员查询间隔（可选）
- 配置最后成员查询次数（可选）
- 配置普通成员查询间隔（可选）
- 配置最大响应间隔（可选）
- 配置其他查询者计时器间隔（可选）
- 配置组播组访问控制 （可选）
- 配置立即离开组 （可选）
- 配置主机组 join-group(可选)
- 配置静态组 static-group(可选)
- 配置 IGMP 状态数量限制 （可选）
- 配置 IGMP PROXY-SERVICE（可选）
- 配置 IGMP MROUTE-PROXY（可选）
- 启动 IGMP SSM-MAP（可选）
- 配置 IGMP SSM-MAP STATIC
- 清除 IGMP 缓存中通过响应消息得来的动态组成员信息（可选）
- 清除 IGMP 缓存中特定接口上的所有信息（可选）
- 显示直连子网中组成员的状况（可选）
- 显示 IGMP 接口的配置信息（可选）
- 显示 IGMP SSM-MAP 的配置信息（可选）
- 显示 IGMP 调试开关的打开情况（可选）

- 打开 IGMP 调试信息开关（可选）

36.4.1 缺省 IGMP 配置

IGMP 版本	所有接口支持版本 2
查询响应时间	10 秒
查询间隔时间	125 秒
组播组访问控制	允许所有组
其它查询计时器间隔	255 秒
健壮性变量	2
最后成员查询间隔	1s
最后成员查询计数	2
IGMP 状态	关闭

36.4.2 启动 IGMP 协议

要开启 IGMP，请在接口模式用以下命令：

命令	功能
Ruijie(config-if) **# ip pim {sparse-mode| dense-mode}**	启动组播路由协议，也启动 IGMP
Ruijie(config-if) **# no ip pim {sparse-mode|dense-mode}**	关闭组播路由协议，也关闭 IGMP

说明：

接口上启用组播路由路由转发和组播路由协议会同时启动 IGMP 功能。

同一台设备上只能运行一种模式的组播路由协议。

36.4.3 配置 IGMP 的版本

要配置 IGMP 的版本，请在接口模式用以下命令：

命令	功能
Ruijie(config-if) **# ip igmp version {1|2|3}**	配置 IGMP 的版本，缺省值为 2

Ruijie(config-if) **# no ip igmp version**	恢复 IGMP 的版本为缺省值

36.4.4 配置最后成员查询间隔

在收到一个组离开报文后，查询者设备要发送特定组成员查询，确认该组内是否依然存在成员。在最后成员查询间隔时间内，若没有收到任何报告，则设备认为离开的主机为该组的最后一个组成员，将删除该组的信息。该时间缺省值为 10(单位 0.1s)。

在接口模式下，用以下命令进行配置:

命令	功能
Ruijie(config-if) **# ip igmp last-member-query-interval** *interval*	配置最后成员查询间隔 *interval* 范围为：<1-255> 单位：0.1s
Ruijie(config-if) **# no ip igmp last-member-query-interval**	恢复最后成员查询间隔为缺省值

36.4.5 配置最后成员查询次数

为防止查询设备发送的特定组成员查询报文丢失，需要多发几次以保证可靠性，因此需要配置最后成员查询次数值大于 1.

请在接口模式用以下命令：

命令	功能
Ruijie(config-if) **# ip igmp last-member-query-count** *count*	配置最后成员查询次数 默认为围为 2-7，缺省值为 2
Ruijie(config-if) **# no ip igmp last-member-query-count**	恢复最后成员查询次数为缺省值

36.4.6 配置普通成员查询间隔

查询者设备每经过一个普通组成员查询间隔时间，就定期发送普通组成员查询报文，以确认当前的组成员关系。普通组成员查询报文发送的目的地址是 all-hosts 组播地址：224.0.0.1，同时 TTL 为 1。该时间缺省值为 125 秒。

请在接口模式用以下命令：

命令	功能
Ruijie(config-if) **# ip igmp query-interval** *seconds*	配置普通成员查询间隔,缺省值为 125s 可选范围为：1~18000

Ruijie(config-if) # **no ip igmp query-interval**	恢复普通成员查询间隔为缺省值

36.4.7 配置最大响应间隔

查询者设备发送的组成员关系查询报文中要求的最大响应时间。减小该时间，可以使设备快速获知组成员的变化，然而导致网络中扩散的成员报告数量相应增加。网络管理员可以在这两个因素中权衡，确定合适的时间取值。缺省时，该时间为 10 秒。另外，该时间的配置有一个需要注意的地方，即它要比查询间隔时间短。

在接口模式下，用以下命令进行配置：

命令	功能
Ruijie(config-if)# **ip igmp query-max-response-time** *seconds*	配置最大响应间隔，缺省值为 10s 可选<1-25>，单位：s
Ruijie(config-if)# **no ip igmp query-max-response-time**	恢复最大响应间隔为缺省值

36.4.8 配置其它查询者存活的计时器间隔

当该定时器超时后，设备认为网络上已经不存在其它查询者了。配置其它查询者存活的计时器时间间隔，可以调整查询者设备之间选举的时间。在查询者设备经常更换的环境下可以考虑缩短其值以提高反应速度。

请在接口模式使用以下命令：

命令	功能
Ruijie(config-if)# **ip igmp query-timeout** *seconds*	配置其它查询者存活的计时器间隔，范围为：60～300，缺省值为 255s
Ruijie(config-if)# **no ip igmp query-timeout**	恢复其它查询者存活的计时器间隔为缺省值

36.4.9 配置组播组访问控制

缺省情况下，一个接口上的主机可以加入任何组播组。当管理员希望限制主机允许加入的组播组范围时，可以使用本特性。通过配置一个标准 IP 访问列表，设置允许或禁止的组播组地址范围，并应用到特定接口上。

请使用以下命令：

命令	功能
Ruijie **# config terminal**	进入全局配置模式

Ruijie (config) # **access-list** *access-list-num* **permit** *A.B.C.D A.B.C.D*	定义一个访问控制列表
Ruijie (config)# **interface** *interface-id*	进入接口配置模式
Ruijie (config-if) # **ip igmp access-group** *access-list-name*	配置组地址落在控制访问列表为 *access-list-name* 的地址范围内的组播组可以进入该接口
Ruijie (config-if) # **no ip igmp access-group**	取消访问控制，允许所有组进入该接口

36.4.10 配置立即离开组

在 IGMP version2 中，使用该命令以减少离开组的延迟时间。当主机发出离开报文后，就要立即离开，不需要查询者设备发出特定组查询。该命令使用在一个接口只有一个接收主机的情况下。

请使用以下命令：

命令	功能
Ruijie **# config terminal**	进入全局配置模式
Ruijie(config)**# access-list** *access-list-num* **permit** *A.B.C.D A.B.C.D*	设置成员组列表的地址范围.
Ruijie (config)**# interface** *interface-id*	进入接口配置模式
Ruijie(config-if)**# ip igmp immediate-leave group-list** *access-list-name*	设置快速离开成员组列表为 *access-list-name* 的组
Ruijie (config-if) # **exit**	进入特权模式

36.4.11 配置主机组 join-group

该命令将交换机相关接口配置成具有主机行为并需要加入相应组播组，这样子交换机就主动可以学习到相应的组信息，在需要将接口添加一个组成员时使用此配置。使用该命令的 no 选项取消交换机加入该组播组。

命令	功能
Ruijie **# config terminal**	进入全局配置模式
Ruijie (config)**# interface** *interface-id*	进入接口配置模式
Ruijie(config-if)**# ip igmp join-group** *group-address*	配置接口上加入主机组
Ruijie (config-if) # **exit**	进入特权模式

使用命令 **no ip igmp join-group** *group-address* 将离开相应组播组。

36.4.12 配置静态组 static-group

该命令将交换机相关接口直接添加一个组记录，当需要将接口添加一个组成员记录时使用此配置。使用该命令的 no 选项取消交换机添加该静态组播组。

命令	功能
Ruijie **# config terminal**	进入全局配置模式
Ruijie (config)**# interface** *interface-id*	进入接口配置模式
Ruijie(config-if)**# ip igmp static-group** *group-address*	配置接口上加入静态组
Ruijie (config-if) **# exit**	进入特权模式

使用命令 **no ip igmp static-group** *group-address* 将接口上加入的静态组删除。

36.4.13 配置 IGMP 组成员数量限制

使用该命令全局配置对 IGMP 组记录数量进行限制，成员报文在超出限制的部分将不会进行 IGMP 缓存，不被转发。

使用该命令的可以对每个接口进行配置，接口和全局可以独立的配置，如果超出了接口或者全局配置的限制数量，成员报文将被忽略掉。请在使用以下命令：

命令	功能
Ruijie(config）**# ip igmp limt** *number*	全局配置 IGMP 状态数量限制 范围与具体设备有关，缺省值 65536
Ruijie(config-if) **# ip igmp limit** *number*	接口上配置 IGMP 状态数量限制 范围与具体设备有关，缺省值为 1024

36.4.14 配置 IGMP PROXY-SERVICE

该命令用来启动所有下联 mroute-proxy 接口的服务功能。在接口上配置该命令，该接口就成为相应 mroute-proxy 的上联口，它将关联所拥有的下联口，并维护下联口通告上来的组信息。

当前对这个命令的可配置最大个数做了限制，最多只允许配置 32 个。每个 proxy-service 下面最多可以关联 255 个下联口。当接口接收到查询报文后，proxy-service 接口将根据自身维护的成员信息，作出相应的应答。proxy-service 接口自身维护的成员信息是从配置了 mroute-proxy 的接口中收集过来的。因此，配置了 proxy-service，相当于此接口只执行主机行为，而不执行路由器行为。请在接口模式用以下命令：

命令	功能
Ruijie(config-if）**# ip igmp proxy-service**	接口上配置为 proxy-service 状态

36.4.15 配置 IGMP MROUTE - PROXY

该命令配置完，接口就可以向其相应的上联口转发报文。当相应的上联口配置为 proxy-service 接口后，该接口将可以转发其成员所发过来的各种 igmp 协议报文。

请在接口模式用以下命令：

命令	功能
Ruijie(config-if）**# ip igmp mroute-proxy** *interfacename*	*Interfname* 为相应的上联口名称

36.4.16 启动 IGMP SSM-MAP

当这个命令被配置，动态学习到的组信息将被强制添加相关联的源记录信息。该命令一般与命令 **ip igmp ssm-map static** 配合使用。

请在全局配置模式使用以下命令：

命令	功能
Ruijie(config）**# ip igmp ssm-map enable**	启动 ssm-map 功能

36.4.17 配置 IGMP SSM-MAP STATIC

该命令与 **ip igmp ssm-map enable** 配合使用，配置完该命令后，对接收到的 v3 以下的报文都将映射上相应的源记录中。

请在全局配置模式用以下命令：

命令	功能
Ruijie(config）**# ip igmp ssm-map static** *11 192.168.2.2*	将符合 acl 11 的所有组映射上源地址 192.168.2.2

36.4.18 清除 IGMP 缓存中通过响应消息得来的动态组成员信息

要清除 IGMP 缓存中通过响应消息得来的动态组成员信息，请在特权模式用以下命令：

命令	功能

Ruijie**# clear ip igmp group**	清除 IGMP 缓存中通过响应消息得来的所有动态组成员的信息，不带参数代表清除所有 igmp group 的信息

36.4.19 清除 IGMP 缓存中特定接口上的所有信息

要清除 IGMP 缓存中特定接口上的所有信息，请在特权模式用以下命令：

命令	功能
Ruijie**# clear ip igmp interface** *interface-type*	清除 IGMP 缓存中接口的信息

36.4.20 显示直连子网中组成员的状况

要显示直连子网中组成员的状况，请在特权模式用以下命令：

命令	功能
Ruijie**# show ip igmp groups**	显示直连子网中所有组的状况
Ruijie**# show ip igmp groups detail**	显示直连子网中所有组的详细信息
Ruijie**# show ip igmp groups** *A.B.C.D*	显示直连子网中指定组的状况
Ruijie**# show ip igmp groups** *A.B.C.D* **detail**	显示直连子网中指定组的详细信息
Ruijie**# show ip igmp interface** *interface-type*	显示直连子网中指定接口的igmp配置信息
Ruijie**# show ip igmp groups** *interface-type* **detail**	显示直连子网中指定接口的所有组的详细信息
Ruijie**# show ip igmp groups** *interface-type A.B.C.D*	显示直连子网中指定接口的特定组的信息
Ruijie**# show ip igmp groups** *interface-type A.B.C.D* **detail**	显示直连子网中指定接口的特定组的详细信息

36.4.21 显示 IGMP 接口的配置信息

要显示 IGMP 接口的配置信息，请在特权模式下用以下命令：

命令	功能
Ruijie**# show ip igmp interface** [*interface-type interface-number*]	显示 IGMP 接口的配置信息
Ruijie**# show ip igmp interface**	显示 IGMP 所有接口的配置信息

36.4.22 显示 IGMP SSM-MAP 的配置信息

要显示 IGMP SSM-MAP 的配置信息，请在特权模式下用以下命令：

命令	功能
Ruijie# **show ip igmp ssm-mapping**	显示 IGMP SSM-MAP 的配置信息
Ruijie# **show ip igmp ssm-mapping** *233.3.3.3*	显示 IGMP SSM-MAP 对组 233.3.3.3 的映射信息

36.4.23 显示 IGMP 调试开关的打开情况

要显示 IGMP 调试开关的打开情况，请在特权模式用以下命令：

命令	功能
Ruijie# **show debugging**	显示 IGMP 调试开关的打开情况

36.4.24 打开 IGMP 调试信息开关，观察 IGMP 的行为

要打开 IGMP 调试信息开关，观察 IGMP 的行为，请在特权模式下用以下命令：

命令	功能
Ruijie# **debug ip igmp all**	打开 IGMP 所有调试信息开关
Ruijie# **debug ip igmp decode**	打开 IGMP 调试解码信息开关
Ruijie# **debug ip igmp encode**	打开 IGMP 调试编码信息开关
Ruijie# **debug ip igmp events**	打开 IGMP 调试事件信息开关
Ruijie# **debug ip igmp fsm**	打开 IGMP 调试有限状态机信息开关
Ruijie# **debug ip igmp tib**	打开 IGMP 调试树信息开关
Ruijie# **debug ip igmp warning**	打开 IGMP 调试警告开关

36.5 PIM-SM 配置任务列表

PIM-SM 配置任务包括以下各项，其中只有第一项必选的，其他项可根据网络的具体情况决定是否要配置。

- 启动 PIM-SM（必选）
- 配置 Hello 消息发送间隔（可选）
- 配置 PIM-SM 邻居过滤（可选）

- 配置指定设备 DR 的优先权值（可选）
- 配置静态 RP（可选）
- 配置设备候选 BSR 状态（可选）
- 配置忽略 RP-SET 中的 RP 的优先级（可选）
- 配置候选 RP（可选）
- RP 注册报文可达性检测（可选）
- 配置 RP 对注册报文的地址过滤（可选）
- 配置注册报文发送的速度限制（可选）
- 配置注册报文的校验和的计算方式为 cisco 的计算方式（可选）
- 配置注册报文的源地址（可选）
- 配置注册报文抑制时间（可选）
- 配置 RP KAT 定时器的时间值（可选）
- 配置 Join/Prune 报文的发送间隔（可选）
- 最后一跳设备由从共享树切换到最短路径树（可选）
- 配置使用 dense-mode 的 mib（可选）
- 配置特定源组播（可选）
- 查看 PIM-SM 状态信息（可选）
- 删除 PIM-SM 内部信息(可选)

36.5.1 启动 PIM-SM

PIM-SM 必须在各个接口上分别启动。设备在接口上启动了 PIM-SM 以后，才可以与其他设备进行 PIM-SM 控制消息的交互，维持和更新组播路由表，并进行组播报文的转发。

要在接口上配置 PIM-SM，请在接口模式下执行以下命令：

命令	作用
Ruijie(config-if）**# ip pim sparse-mode**	在接口上启动 PIM-SM 协议
Ruijie(config-if）**# no ip pim sparse-mode**	在接口上关闭 PIM-SM 协议

注意：

必须在全局配置模式下启动了组播路由之后，在接口上启动 PIM-SM 才会起作用。

配置该命令的时候，如果会出现“Failed to enable PIM-SM on <接口名>, resource temporarily unavailable, please try again”，请再次尝试配置该命令。

配置该命令的时候，如果会出现“PIM-SM Configure failed! VIF limit exceeded in NSM!!! ”，表示当前组播接口配置的数量已经达到设备可配置的组播接口上限。如果仍然需要在该接口下开启 PIM-SM 应用，请删除一些不必要的 PIM-SM 或 PIM-DM 接口。

不建议在同一台交换机/路由器上的不同接口配置不同的 v4 组播路由协议。

36.5.2 配置 Hello 消息发送间隔

接口启动了 PIM-SM 以后，会周期性地向相邻设备接口发送 Hello 消息。接口向相邻设备接口发送 Hello 消息的时间间隔可以根据相连网络的实际情况加以修改。

要配置 Hello 消息发送间隔，请在接口模式下执行以下命令：

命令	作用
Ruijie(config-if）**# ip pim query-interval** *interval-seconds*	将接口的 Hello 消息发送间隔设置为 *interval-seconds*，范围为<1-65535>，单位为秒
Ruijie(config-if）**# no ip pim query-interval**	将接口的 Hello 消息发送间隔还原为默认值

缺省情况下，接口上的 Hello 消息发送间隔是 30 秒。

说明：

每当 Hello 消息发送间隔被更新时，Hello 消息保持时间也会按照以下规则随之更新：Hello 消息保持时间被更新为 Hello 消息发送间隔的 3.5 倍；如果 Hello 消息发送间隔 * 3.5 > 65535，则 Hello 消息保持时间被更新为 65535。

36.5.3 配置 PIM-SM 邻居过滤

可以在接口上设置邻居过滤功能，以提高网络的安全性。如果设置了邻居过滤，只要某个邻居被过滤访问列表拒绝，则 PIM-SM 不会与该邻居建立邻接关系，并且会删除与该邻居已经建立的邻接关系。

要配置 PIM 邻居过滤功能，请在接口模式下执行以下命令：

命令	作用
Ruijie(config-if）**# ip pim neighbor-filter** *access-list*	在当前接口上启动 PIM 邻居过滤功能。
Ruijie(config-if）**# no ip pim neighbor-filter** *access-list*	在当前接口上关闭 PIM 邻居过滤功能。

缺省情况下，接口上的 PIM 邻居过滤功能是关闭的。

说明：

ip pim neighbor-filter 命令说明：

只有符合 ACL 过滤条件的邻居地址才能够作为当前接口的 PIM 邻居，被 ACL 过滤的邻居地址将无法作为当前接口的 PIM 邻居。

36.5.4 配置指定设备 DR 的优先权值

使用此命令设置指定设备的优先权值。权值越高，优先权越大。

请在接口模式下执行以下命令：

命令	作用
Ruijie(config-if）**# ip pim dr-priority** *priority-value*	配置优先权值，*priority-value* 范围为<0-4294967294>
Ruijie(config-if）**# no ip pim dr-priority** *priority-value*	恢复优先权值为缺省值，缺省值为 1。

36.5.5 配置静态 RP

在网络规模比较小的情况下，可以使用配置静态 RP 来使用 PIM-SM，要求 PIM-SM 域内的所有设备静态 RP 配置必须一致，保证 PIM-SM 的组播路由没有歧义性。

请在全局配置模式下使用如下命令：

命令	作用
Ruijie(config）**# ip pim rp-address** *rp-address* [*access-list*]	配置本设备上的静态 RP
Ruijie(config）**# no ip pim rp-address** *rp-address* [*access-list*]	取消静态 RP 配置。

注意：

使用此命令需注意下面几点：

- 如果通过 BSR 机制和 RP 静态配置同时有效时，优先使用动态 RP。
- 使用控制列表静态配置 RP 地址可以配置多个的组播组（使用 ACL）或者所有的组播组（不使用 ACL），但是一个 RP 静态地址不能被多次配置使用。
- 如果有多个静态 RP 可以服务于同一个组，IP 地址高的 RP 将被优先采用。

- 只有在 ACL 中定义的允许过滤地址才是无效的组播组。缺省的过滤 0.0.0.0/0 将被认为是过滤所有的组播组 224/4。
- 配置完成后，静态的 RP 源地址将插入到基于组范围的静态 RP 组树结构中，每个组范围组播静态组维持一个静态 RP 组的链表结构，该链表按 IP 地址递减排序。当为一个组范围选择一个 RP，第一个元素，也就是 IP 地址最高的那个将首先被选上。
- 删除一个静态 RP 地址将向所有存在的组中删除这个地址，并从现有静态 RP 树结构中选择一个作为 RP 地址。

36.5.6 配置设备候选 BSR 状态

候选 BSR 的配置可以产生 PIM-SM 域内全局唯一的 BSR，由其进行域内 RP 的收集和分发，保证域内 RP 映射的唯一性。

请在全局模式下执行以下命令：

命令	作用
Ruijie(config）**# ip pim bsr-candidate** *interface-type interface-number* [*hash-mask-length*] [*priority-value*]	配置本设备为候选 BSR。通过 BSM 消息学习和竞争全局 BSR 角色。 *hash-mask-length* 默认是 10，范围为<0-32> *priority-value* 默认是 64，范围为<0-255>。
Ruijie(config）**# no ip pim bsr-candidate** *interface-type interface-number*	取消当前候选 BSR 的配置。

36.5.7 配置忽略 RP-SET 中的 RP 的优先级

当要对一个组播地址选择一个 RP 时，如果有多个 RP 都可以服务这个组播地址，此时在比较两个 RP 时要忽略掉 RP 的优先级，可以使用这条命令。如果不配置这条命令，那么在比较两个 RP 时，RP 的优先级会被考虑。

请在全局配置模式下使用如下命令：

命令	作用
Ruijie(config）**# ip pim ignore-rp-set-priority**	忽略 RP-SET 中 RP 的优先级
Ruijie(config）**# no ip pim ignore-rp-set-priority**	考虑 RP-SET 中 RP 的优先级

36.5.8 配置候选 RP

配置候选 RP 可以周期性地向 BSR 发送候选 RP 通告，这些候选 RP 通告所带的

信息会扩散到域内所有 PIM-SM 设备，保证 RP 映射的唯一。

请在全局配置模式下使用如下命令：

命令	作用
Ruijie(config）**# ip pim rp-candidate** *interface-type interface-number* [**priority** *priority-value*] [**interval** *interval-seconds*] [**group-list** *access-list*]	配置本设备上使用候选 RP， **priority** 缺省时，*priority-value* 默认是 192，*priority-value* 范围为<0-255> **interval** 缺省时，*interval-seconds* 默认是 60s，*interval-seconds* 范围为<1-16383> **group-list** 缺省时，*access-list* 默认是允许所有组播组即 224/4。
Ruijie(config）**# no ip pim rp-candidate** *interface-type interface-number*	取消候选 RP 配置。

说明：

若需要指定接口成为特定组范围的候选 RP，那么该命令可以带上 acl 选项。需要注意的是组范围的计算只基于 permit 的 ace，不会对 deny 的 ace 进行综合计算的。

注意：

其中 ace 的源范围将作为特定的组范围来进行匹配的

36.5.9 RP 注册报文可达性检测

从 DR 上要往 RP 发送注册报文，如果要检验 RP 是否可达，可以配置此命令。配置此命令之后，DR 发送注册报文之前将检验 RP 是否可达，即查询单播路由和静态组播路由表获知是否存在到达 RP 的路由，如果不存在到达 RP 的路由则不发送注册报文。

请在全局配置模式下使用如下命令：

命令	作用
Ruijie(config）**# ip pim register-rp-reachability**	对注册报文是否可达 RP 进行检测
Ruijie(config）**# no ip pim register-rp-reachability**	取消对可达性进行检测

36.5.10 配置 RP 对注册报文的地址过滤

在 RP 上，要对到达的注册报文中的源地址组地址对进行过滤，可以配置此命令。如果不配置这条命令，那么在 RP 上允许每一个到达的注册报文。如果配了这条命令，那么注册报文中源地址和组地址被 ACL 允许的才被继续处理，否则注册报文被过滤掉。

请在全局配置模式下使用如下命令：

命令	作用
Ruijie(config）**# ip pim accept-register list** *access-list*	配置注册报文的源地址组地址过滤
Ruijie(config）**# no ip pim accept-register**	取消注册报文的源地址组地址过滤

36.5.11 配置注册报文发送的速度限制

配置 DR 发送注册报文的速度，默认情况下不限制发送速度。它是对每个（S，G）状态的注册报文发送速度进行配置，并不是对整个系统的注册报文发送速度进行配置。

请在全局配置模式下使用如下命令：

命令	作用
Ruijie(config）**# ip pim register-rate-limit** *rate*	定义每秒可发送的最大注册报文数据包的数量，*rate* 取值为范围<1-65535>
Ruijie(config）**# no ip pim register-rate-limit**	取消配置，不限速。

36.5.12 配置注册报文的校验和的计算方式为 cisco 的计算方式

若要把注册报文中校验和的计算方式配置为 cisco 的计算方式，请配置这条命令。

不配置这条命令，那么注册报文中校验和的计算方式为协议规定的默认方式。

请在全局配置模式下使用如下命令：

命令	作用
Ruijie(config）**# ip pim cisco-register-checksum** [**group-list** *access-list*]	配置注册报文中的校验和的计算方式为 cisco 的计算方式。 **group-list** *access-list* 缺省时，对所有的组播地址，都应用这种配置。

Ruijie(config）**# no ip pim cisco-register-checksum** [**group-list** *access-list*]	取消配置注册报文中的校验和的计算方式为 cisco 的计算方式。 **group-list** *access-list* 缺省时，对所有的组播地址，都取消这种配置。

36.5.13 配置注册报文的源地址

如果要配置 DR 发送出来的注册报文的源地址，可以使用此命令。不配置此命令或是使用这条命令的 no 形式，注册报文源地址将会使用与组播源相连的 DR 接口地址。如果使用该命令的地址参数，配置的地址必须是单播路由可达的。如果使用该命令的接口参数，一般来说它是回环接口，但也可以是其它类型接口，这个接口的地址必须是被单播路由公告过的。

请在全局配置模式下使用如下命令：

命令	作用
Ruijie(config）**# ip pim register-source** { *local-address* | *Interface-type interface-number*}	配置注册报文中使用的源地址
Ruijie(config）**# no ip pim register-source**	用 RPF 的接口地址作为注册报文中的源地址

36.5.14 配置注册报文抑制时间

使用此命令配置注册报文抑制时间，在 DR 上配置此值将修改定义在 DR 上的注册报文抑制时间。如果 **ip pim rp-register-kat** 命令没有配置，在 RP 上定义此值将修改 RPkeepalive 的周期。

请在全局配置模式下使用如下命令：

命令	作用
Ruijie(config）**# ip pim register-suppression** *seconds*	配置注册报文抑制时间，*seconds* 取值范围为[11，21843]，单位秒
Ruijie(config）**# no ip pim register-suppression**	抑制时间为 60 秒

36.5.15 配置 RP KAT 定时器的时间值

如果要配置 RP 上注册报文所创建(S,G)状态的存活时间，可以使用此命令。

请在全局配置模式下使用如下命令：

命令	作用
Ruijie(config）**# ip pim rp-register-kat** *seconds*	配置 KAT 定时器时间值，*seconds* 取值范围为[1，65535]，单位为秒

Ruijie(config）**# no ip pim rp-register-kat**	使用 KAT 的缺省值

注意：

该定时器值最好要大于源 DR 设备上的注册抑制时间 * 3 + 注册探测时间，否则 RP 有可能在源 DR 再次发送注册报文之前把(S,G)状态超时，从而造成组播流短暂的断流。

36.5.16 配置 Join/Prune 报文的发送间隔

Join/Prune 报文的默认发送间隔是 60s，如果要修改这个发送间隔，可以配置这条命令。不配置这条命令，则 Join/Prune 的默认发送间隔为 60s。

在全局配置模式下，使用以下命令：

命令	作用
Ruijie(config）**# ip pim jp-timer** *interval-seconds*	将 Join/Prune 报文发送间隔设置为 interval-seconds，范围为<1-65535>，单位为秒
Ruijie(config）**# no ip pim jp-timer** [*interval-seconds*]	将 Join/Prune 报文发送间隔还原为默认值，即 60s

36.5.17 最后一跳设备由从共享树切换到最短路径树

允许最后一跳设备由从共享树切换到最短路径树。

当配置了这条命令，接收到第一个(S,G)报文，一个 PIM 的加入信息被触发，并构造一棵源树；如果 group-list 关键字被定义，所有的这个特定组会切换到源树；使用这条命令的 no 形式，该设备将重新转向共享树并向源发送出一个剪枝报文。

请在全局配置模式下使用如下命令：

命令	作用
Ruijie(config）**# ip pim spt-threshold** [**group-list** *access-list*]	如果 **group-list** 被定义了，则允许这个特定组的最后一跳设备由从共享树切换到最短路径树，如果 group-list 没被定义，则允许所有组播组。
Ruijie(config）**# no ip pim spt-threshold** [**group-list** *access-list*]	关闭该功能

36.5.18 配置使用 dense-mode 的 mib

当要使用 dense-mode 的 mib，配置这条命令；不配置这条命令时，是使用 sparse-mode 的 mib。

请在全局配置模式下使用如下命令：

命令	作用
Ruijie(config）**# ip pim mib dense-mode**	使用 dense-mode 的 mib。
Ruijie(config）**# no ip pim mib dense-mode**	使用 sparse-mode 的 mib

36.5.19 配置特定源组播

配置特定源组播可以直接从组播源接收组播数据报文，而不需要走 RP 树，请在全局配置模式下使用如下命令。

命令	作用
Ruijie(config）**# ip pim ssm** {**default** | **range** *access-list*}	配置特定源组播
Ruijie(config）**# no ip pim ssm**	取消配置特定源组播

36.5.20 查看 PIM-SM 状态信息

PIM-SM 提供 show 命令来监视和维护 PIM-SM。使用 show 命令可以查看 PIM-SM 的接口、组播组和组播路由表等信息。

命令	作用
Ruijie**# show debugging**	显示调试开关打开情况
Ruijie**#show ip pim sparse-mode bsr-router**	使用此命令显示 BSR 的详细信息。
Ruijie**#show ip pim sparse-mode interface** [i*nterface-type interface-number* [**detail**]]	显示接口的 PIM-SM 信息。
Ruijie**#show ip pim sparse-mode local-members** [*interface-type interface-number*]	显示 PIM-SM 的接口的本地 IGMP 信息
Ruijie**#show ip pim sparse-mode mroute** {*group_address* | *source_address* | }	显示 PIM-SM 组播路由表信息
Ruijie**#show ip pim sparse-mode neighbor** [**detail**]	显示 PIM-SM 邻居信息。
Ruijie**#show ip pim sparse-mode nexthop**	显示来自 NSM 的 PIM-SM 下一跳信息。

Ruijie**#show ip pim sparse-mode rp-hash** *group-address*	使用此命令显示出某个组地址 *group-address* 对应的 RP 的信息。
Ruijie**#show ip pim sparse-mode rp mapping**	使用此命令查看当前所有 RP 以及它们所服务的组的信息。

36.5.21 删除 PIM-SM 内部信息

提供以下命令来删除本机上的 PIM-SM 内部信息：

命令	作用
Ruijie**# clear ip mroute {* | *group_address* [*source_address*] }**	删除组播路由表项
Ruijie**# clear ip mroute statistics {* | *group_address* [*source_address*] }**	删除组播路由表项的统计信息
Ruijie**#clear ip pim sparse-mode bsr rp-set ***	删除 RP-SET

36.6 PIM-DM 配置任务列表

PIM-DM 配置任务包括以下各项，其中只有第一项是必选的，其他项可根据网络的具体情况决定是否要配置。

- 启动 PIM-DM（必选）
- 配置 Hello 消息发送间隔（可选）
- 配置 PIM 邻居过滤（可选）
- 配置 PIM 状态更新功能（可选）
- 配置 PIM 状态更新消息发送间隔（可选）
- 查看 PIM-DM 状态信息（可选）

36.6.1 启动 PIM-DM

PIM-DM 必须在各个接口上分别启动。设备在接口上启动了 PIM-DM 以后，才可以与其他设备进行 PIM-DM 控制消息的交互，维持和更新组播路由表，并进行组播报文的转发。

要在接口上配置 PIM-DM，请在接口模式下执行以下命令：

命令	作用
Ruijie(config-if）**#ip pim dense-mode**	在接口上启动 PIM-DM 协议
Ruijie(config-if）**#no ip pim dense-mode**	在接口上关闭 PIM-DM 协议

以下的例子演示了在接口 GabitEthernet 0/3 上配置 PIM 密集模式

```
Ruijie(config)# ip multicast-routing
Ruijie(config)# interface  gabitEthernet 0/3
Ruijie(config-if)# ip address 192.168.194.2 255.255.255.0
Ruijie(config-if)# ip pim dense-mode
```

注意：

必须在全局配置模式下启动了组播路由之后，在接口上启动 PIM-DM 才会起作用。

配置该命令的时候，如果出现“Failed to enable PIM-DM on <接口名>, resource temporarily unavailable, please try again”，请再次尝试配置该命令。

配置该命令的时候，如果出现“PIM-DM Configure failed! VIF limit exceeded in NSM!!! ”，表示当前组播接口配置的数量已经达到设备可配置的组播接口上限。如果仍然需要在该接口下开启 PIM-DM 应用，请删除一些不必要的 PIM-DM 或 PIM-SM 接口。

不建议在同一台交换机/路由器上的不同接口配置不同的 v4 组播路由协议。

36.6.2 配置 Hello 消息发送间隔

接口启动了 PIM-DM 以后，会周期性地向相邻设备接口发送 Hello 消息。接口向相邻设备接口发送 Hello 消息的时间间隔可以根据相连网络的实际情况加以修改。

要配置 Hello 消息发送间隔，请在接口模式下执行以下命令：

命令	作用
Ruijie(config-if）**#ip pim query-interval** *interval-seconds*	将接口的 Hello 消息发送间隔设置为 *interval-seconds*，单位为秒，范围为 <1-65535>
Ruijie(config-if）**#no ip pim query-interval**	将接口的 Hello 消息发送间隔还原为默认值

缺省情况下，接口上的 Hello 消息发送间隔是 30 秒。

说明：

每当 Hello 消息发送间隔被更新时，Hello 消息保持时间（Hello hold time）会自动更新为 Hello 消息发送间隔的 3.5 倍。如果 Hello 消息发送间隔 * 3.5 > 65535，则 Hello 消息保持时间被更新为 65535。

36.6.3 配置 PIM 邻居过滤

可以在接口上设置邻居过滤功能，以提高网络的安全性。如果设置了邻居过滤，只要某个邻居被过滤访问列表拒绝，则 PIM-DM 不会与该邻居建立邻接关系，或者会中止与该邻居已经建立的邻接关系。

要配置 PIM 邻居过滤功能，请在接口模式下执行以下命令：

命令	作用
Ruijie(config-if）**#ip pim neighbor-filter** *access-list*	在当前接口上启动 PIM 邻居过滤功能。
Ruijie(config-if）**#no ip pim neighbor-filter** *access-list*	在当前接口上关闭 PIM 邻居过滤功能。

缺省情况下，接口上的 PIM 邻居过滤功能是关闭的。

说明：

ip pim neighbor-filter 命令说明：

只有符合 ACL 过滤条件的邻居地址才能够作为当前接口的 PIM 邻居，被 ACL 过滤的邻居地址将无法作为当前接口的 PIM 邻居。

36.6.4 配置 PIM 状态更新功能

设备启动了 PIM-DM 以后，如果组播表项的 RPF 接口与组播源直接相连的，也就是说，有 PIM 接口是与组播源在同一个网段的，就会周期性地向下游设备发送状态更新消息，以更新整个网络的状态信息。可以在全局模式下禁止处理和转发 PIM-DM 状态更新消息。

要配置 PIM-DM 的状态更新功能，请在全局模式下执行以下命令：

命令	作用
Ruijie(config）**#ip pim state-refresh disable**	禁止处理和转发 PIM-DM 状态更新消息
Ruijie(config）**#no ip pim state-refresh disable**	允许处理和转发 PIM-DM 状态更新消息

缺省情况下，状态更新功能是打开的。

注意：

关闭状态刷新功能可能会导致已经收敛的 PIM-DM 组播转发树重新收敛，造成不

必要的带宽浪费和组播路由表振荡，所以一般情况下最好都不要关闭状态刷新功能。

36.6.5 配置 PIM 状态更新消息发送间隔

设备启动了 PIM-DM 以后，如果有接口是与组播源直接相连的，就会周期性地向下游设备发送状态更新消息，以更新整个网络的状态信息。可以根据设备所在网络的实际情况，对接口的 PIM 状态更新消息发送间隔加以修改。

要在接口上配置 PIM 状态更新消息发送间隔，请在接口模式下执行以下命令：

命令	作用
Ruijie(config-if）**#ip pim state-refresh origination-interval** *interval-seconds*	将当前接口的 PIM 状态更新消息发送间隔设为 *interval-seconds*，单位为秒，范围为<1-100>
Ruijie(config-if）**#no ip pim state-refresh origination-interval**	将接口上的 PIM 状态更新消息发送间隔恢复为默认值。

缺省情况下，接口上的 PIM 状态更新消息发送间隔为 60 秒。

说明：

只有与组播源直接相连的设备才会周期性地往下游接口发送 PIM 状态更新消息，所以如果设备不是与组播源直接相连的，那么在它的下游接口上配置 PIM 状态更新消息发送间隔是无效的。

36.6.6 查看 PIM-DM 状态信息

PIM-DM 提供 show 命令来监视和维护 PIM-DM。使用 **show** 命令可以查看 PIM-DM 的接口、组播组和组播路由表等信息。

命令	作用
Ruijie**#show ip pim dense-mode interface** [*interface-type interface-number*][**detail**]	显示接口的 PIM-DM 信息。
Ruijie**#show ip pim dense-mode neighbor** [*interface-type interface-number*]	显示 PIM-DM 邻居信息。
Ruijie**#show ip pim dense-mode nexthop**	显示 PIM-DM 的下一跳信息。
Ruijie**#show ip pim dense-mode mroute** [*A.B.C.D A.B.C.D*] [**summary**]	显示 PIM-DM 的路由表信息。

以上各命令的详细用法，请参考《PIM-DM 命令参考》。

下面给出使用这些命令的几个例子：

1. **show ip pim dense-mode interface detail** 命令:

```
Ruijie# show ip pim dense-mode interface detail
FastEthernet 0/45 (vif-id: 3):
Address 10.10.10.10
Hello period 30 seconds, Next Hello in 15 seconds
Over-ride interval 2500 milli-seconds
Propagation-delay 500 milli-seconds
Neighbors:
10.10.10.1
VLAN 4 (vif-id: 2):
Address 50.50.50.50
Hello period 30 seconds, Next Hello in 2 seconds
Over-ride interval 2500 milli-seconds
Propagation-delay 500 milli-seconds
Neighbors:
50.50.50.1
```

上面的例子说明接口 FastEthernet 0/45 的 IP 地址是 10.10.10.10，Hello 消息发送间隔是 30 秒，下一个 Hello 消息将于 15 秒后发送，邻居地址是 10.10.10.1。VLAN 4 的接口信息与 FastEthernet 0/45 类似。

2. **show ip pim dense-mode neighbor** 命令：

```
Ruijie# show ip pim dense-mode neighbor
Neighbor-Address Interface            Uptime/Expires    Ver
10.10.10.1      FastEthernet 0/45     00:19:29/00:01:21  v2
50.50.50.1      VLAN 4                00:22:09/00:01:39  v2
```

上面的例子说明设备有 2 个邻居。其中，邻居 10.10.10.1 与 FastEthernet 0/45 相连，已经存活了 19 分 29 秒，其邻居生存时间将于 1 分 21 秒后到期。邻居 50.50.50.1 与邻居 10.10.10.1 情况类似。

3. **show ip pim dense-mode nexthop** 命令：

```
Ruijie# show ip pim dense-mode nexthop
Destination  Nexthop   Nexthop      Nexthop       Metric Pref
               Num      Addr        Interface
1.1.1.111      1      50.50.50.1    VLAN 4          0     1
```

上面的例子说明到达组播源 1.1.1.111 的下一跳邻居地址是 50.50.50.1，出口是 VLAN4。

4. **show ip pim dense-mode mroute** 命令：

```
Ruijie# show ip pim dense-mode mroute
PIM-DM Multicast Routing Table
(1.1.1.111, 229.1.1.1)
MRT lifetime expires in 205 seconds
```

```
RPF Neighbor: 50.50.50.1, Nexthop: 50.50.50.1, VLAN 4
Upstream IF: VLAN 4
Upstream State: Pruned, PLT:200
Assert State: NoInfo
Downstream IF List:
FastEthernet 0/45:
Downstream State: NoInfo
Assert State: Loser, AT:170
```

上面的例子列出了(1.1.1.111, 229.1.1.1)的表项情况，其中 MRT 老化时间为 205 秒。RPF 邻居为 50.50.50.1，下一跳为 50.50.50.1，到达下一跳的出口为 VLAN4。表项的上游接口为 VLAN4，此时处于 Pruned 状态，表示表项没有下游转发出口。下游接口有 FastEthernet 0/45，处于 NoInfo 状态，并且接口的 Assert 状态处于 Loser，FastEthernet 0/45 不在转发出口中。

36.7 组播路由配置举例

36.7.1 PIM-DM 配置范例

36.7.1.1 配置要求

网络拓扑结构如图 7 所示。设备 1 与组播源处在同一个网络，设备 2 与接收者 A 处在同一个网络，设备 3 与接收者 B 处在同一个网络。假定设备与主机都正确连接，并且已经配置好 IP 地址和单播路由。

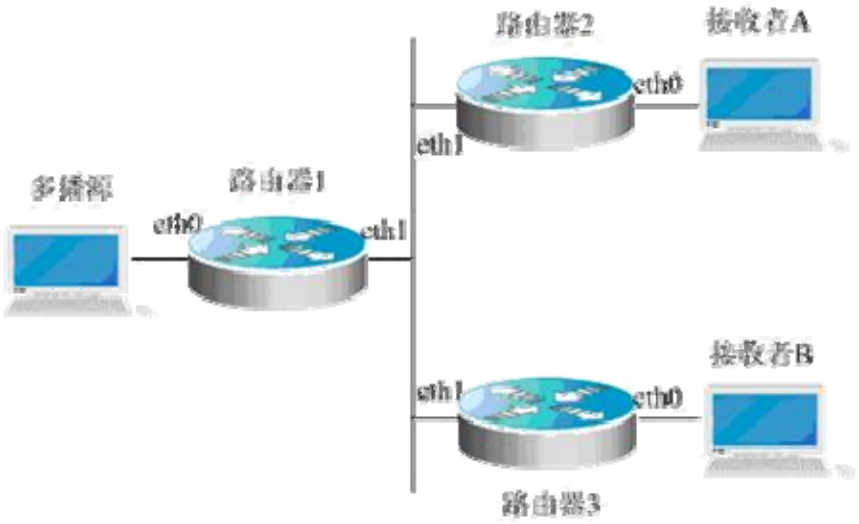


图 6　PIM-DM 配置范例拓扑结构图

36.7.1.2 设备配置

下面以设备 1 为例展示如何配置 PIM-DM，设备 2 和设备 3 的配置过程与设备 1 类似。

步骤 1：启动组播路由

```
Ruijie# configure terminal
Ruijie(config)# ip multicast-routing
```

步骤 2：在接口 eth0 上启动 PIM-DM

```
Ruijie(config)# interface eth 0
Ruijie(config-if)# ip pim dense-mode
Ruijie(config-if)# exit
```

步骤 3：在接口 eth1 上启动 PIM-DM，并返回特权用户模式。

```
Ruijie(config)# interface eth 1
Ruijie(config-if)# ip pim dense-mode
Ruijie(config-if)# end
```

设备 2 和设备 3 的配置与设备 1 类似，都是先启动组播路由，再分别在各个接口上启动 PIM-DM。

36.7.2 PIM-SM 配置范例

36.7.2.1 配置要求

网络拓扑结构如图 2 所示。R1 与组播源处于同一个网络，R2 将被设置为 RP，R3 与接收者 A 处于同一个网络，R4 与接收者 B 处于同一个网络。假定设备与主机都正确连接，并且已经在各接口配置了 IP 地址，在各设备启动了 IP 单播。

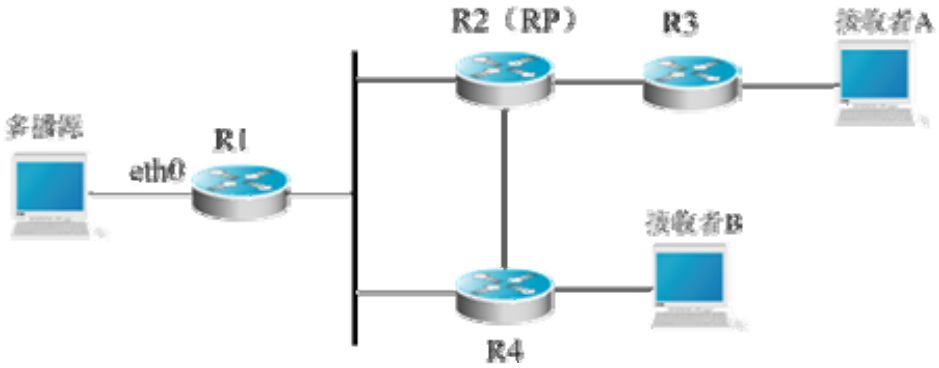


图 7　PIM-SM 配置范例拓扑结构图

36.7.2.2 设备配置

步骤 1：启动组播路由

以下在 R1 上启动 IPV4 组播路由，在 R2、R3、R4 上配置相似，配置过程略。

```
Ruijie# configure terminal
Ruijie(config)# ip multicast-routing
```

步骤 2：在接口上启动 PIM-SM

以下在 R1 的接口 eth0 上启动 PIM-SM，在 R1、R2、R3、R4 上各接口配置相似，配置过程略。

```
Ruijie(config)# interface eth 0
Ruijie(config-if)# ip pim sparse-mode
Ruijie(config-if)# end
```

步骤 3：配置候选 BSR 和候选 RP

将 R2 的 loopback1 配置为 C-BSR 和 C-RP

```
Ruijie(config)# interface loopback 1
Ruijie(config-if)# ip address 100.1.1.1 255.255.255.0
Ruijie(config-if)# ip pim sparse-mode
Ruijie(config-if)# exit
Ruijie(config)# ip pim bsr-candidate loopback 1
Ruijie(config)# ip pim rp-candidate loopback 1
Ruijie(config-if)# end
```

接收者加入组，组播源发送组播流之后，可以使用 PIM-SM 提供的 show 命令监控运行状态。

说明：

在启动 PIM-SM 的同时，IGMP 在各个接口上分别自动启动，不需要手动配置。

37 基于端口的流控制配置

37.1 风暴控制

37.1.1 概述

当 LAN 中存在过量的广播、多播或未知名单播包时，就会导致网络变慢和报文传输超时几率大大增加。这种情况我们称之为 LAN 风暴。协议栈的执行错误或对网络的错误配置都有可能导致风暴的产生。

我们可以分别针对广播、多播和未知名单播数据流进行风暴控制。当接口接收到的广播、多播或未知名单播包的速率超过所设定的阈值时，设备将只允许通过所设定阈值带宽的报文，超出阈值部分的报文将被丢弃，直到数据流恢复正常，从而避免过量的泛洪报文进入 LAN 中形成风暴。

37.1.2 配置风暴控制

在接口配置模式下，请使用如下命令配置风暴控制：

命令	作用
Ruijie(config-if)# **storm-control** {**broadcast** | **multicast** | **unicast**} [{ **level** *percent* | **pps** *packets* | *rate-bps*]	**broadcast** ：打开对广播风暴的控制功能； **multicast** ：打开对未知名多播风暴的控制功能； **unicast** ：打开对未知名单播风暴的控制功能； **level** *percent*： 以带宽的百分比进行设置如 20 表示端口速率限制为 20%带宽； **pps** *packet*：以报文为单位进行设置 即 packets per second，每秒允许通过的报文数； *Rate-bps*：以 bit 为单位进行设置，即 Kbits per second,每秒允许通过的千比特数。

您可以在接口配置模式下通过命令 **no storm-control broadcast ,no storm-control multicast , no storm-control unicast** 来关闭接口相应的风暴控制功能。

下面的例子打开 GigabitEthernet 0/1 上的多播风暴控制功能，并且设置允许的速率为 4M。

```
Ruijie# configure terminal
Ruijie(config)# interface GigabitEthernet 0/1
Ruijie(config-if)# storm-control multicast 4096
Ruijie(config-if)# end
```

注意：

1. 设备基于 level 的风暴控制，带宽基准值直接采用设置风暴控制的物理口支持的最大带宽值，不采用物理口实际工作时的带宽值进行换算。
2. 如果仅使能风暴控制功能，如配置 storm-control broadcast，这时风暴控制采用缺省设置，缺省设置值为端口带宽的百分之一。
3. S3760 系列交换机，同一台设备上只能支持设置同一种风暴控制模式的设置(level,pps,kbps)，并且风暴控制和入口速率限制功能互斥。比如，在端口 1 上采用 level 配置风暴控制，那么在端口 2 上配置基于 pps 的风暴将提示错误，在端口 3 上开启入口速率限制也将提示配置失败。如果 AP 某个成员口的风暴控制模式与同设备上的其他口的风暴控制模式不同，或者同设备上的其他口开启了入端速率限制。则该成员口退出 AP 时，其风暴控制的配制的配置将不生效。

37.1.3 查看风暴控制使能状态

查看接口的风暴控制使能状态：

命令	作用
Ruijie# **show storm-control** [*interface-id*]	显示风暴控制信息。

下面的例子为显示接口 Gi0/3 的风暴控制功能的使能状态：

```
Ruijie# show storm-control gigabitEthernet 0/3
Interface   Broadcast Control   Multicast Control   Unicast
Control action
GigabitEthernet 0/3 Disabled  Disabled   Disabled    none
```

您也可以一次查看所有接口的风暴控制功能的使能状态：

```
Ruijie# show storm-control
Interface   Broadcast Control   Multicast Control   Unicast
Control Action
--------- ----------------- ----------------- -------
GigabitEthernet 0/1   Disabled   Disabled   Disabled   none
GigabitEthernet 0/2   Disabled   Disabled   Disabled   none
GigabitEthernet 0/3   Disabled   Disabled   Disabled   none
GigabitEthernet 0/4   Disabled   Disabled   Disabled   none
GigabitEthernet 0/5   Disabled   Disabled   Disabled   none
GigabitEthernet 0/6   Disabled   Disabled   Disabled   none
GigabitEthernet 0/7   Disabled   Disabled   Disabled   none
GigabitEthernet 0/8   Disabled   Disabled   Disabled   none
```

```
GigabitEthernet 0/9   Disabled   Disabled   Disabled   none
GigabitEthernet 0/10  Disabled   Disabled   Disabled   none
GigabitEthernet 0/11  Disabled   Disabled   Disabled   none
GigabitEthernet 0/12  Disabled   Disabled   Disabled   none
GigabitEthernet 0/13  Disabled   Disabled   Disabled   none
GigabitEthernet 0/14  Disabled   Disabled   Disabled   none
GigabitEthernet 0/15  Disabled   Disabled   Disabled   none
GigabitEthernet 0/16  Disabled   Disabled   Disabled   none
GigabitEthernet 0/17  Disabled   Disabled   Disabled   none
GigabitEthernet 0/18  Disabled   Disabled   Disabled   none
GigabitEthernet 0/19  Disabled   Disabled   Disabled   none
GigabitEthernet 0/20  Disabled   Disabled   Disabled   none
GigabitEthernet 0/21  Disabled   Disabled   Disabled   none
GigabitEthernet 0/22  Disabled   Disabled   Disabled   none
GigabitEthernet 0/23  Disabled   Disabled   Disabled   none
GigabitEthernet 0/24  Disabled   Disabled   Disabled   none
```

37.2 Protected Port

37.2.1 概述

有些应用环境下，要求一台设备上的有些端口之间不能互相通讯。在这种环境下，这些端口之间的通讯，不管是单址帧，还是广播帧，以及多播帧，都不能在保护口之间进行转发。您可以通过将某些端口设置为保护口(Protected Port)来达到目的。

当您将某些端口设为保护口之后，保护口之间互相无法通讯，保护口与非保护口之间可以正常通讯。

保护口有两种模式，一种是阻断保护口之间的二层转发，但允许保护口之间进行 3 层路由，第二种是既阻断保护口之间的二层转发又阻断 3 层路由；在两种模式都支持的情况下，第一种模式将作为缺省配置模式。

当您将两个保护口设为一个 SPAN 端口对时，SPAN 的源端口发送或接收的帧将按照 SPAN 的设置发到 SPAN 目的端口。因此您最好不要将 SPAN 目的端口设为保护口(这样您还可以节约系统资源)。

设备支持将 Aggregated Port 设置为保护口，当您将一个 Aggregated Port 设置为保护口时，Aggregated Port 的所有成员口都被设置为保护口。

37.3 端口安全

37.3.1 概述

端口安全功能通过报文的源 MAC 地址或者源 MAC+源 IP 或者仅源 IP 来限定报文

是否可以进入交换机的端口，您可以静态设置特定的 MAC 地址，静态 IP+MAC 绑定或者仅 IP 绑定，或者动态学习限定个数的 MAC 地址来控制报文是否可以进入端口，使能端口安全功能的端口称为安全端口。只有源 MAC 地址为端口安全地址表中配置或者配置绑定的 IP+MAC 地址，或者配置的仅 IP 绑定地址，或者学习到的 MAC 地址的报文才可以进入交换机通信，其他报文将被丢弃。

为了增强安全性，您可以将 MAC 地址和 IP 地址绑定起来作为安全地址。当然您也可以只指定 MAC 地址而不绑定 IP 地址。

您可以使用下面几种方式加满端口上的安全地址：

- 通过使用接口配置模式下的命令来手工配置端口的所有安全地址。
- 让该端口自动学习地址，这些自动学习到的地址将变成该端口上的安全地址，直到达到最大个数。需要注意的是，自动学习的安全地址均不会绑定 IP 地址，如果在一个端口上，您已经配置了绑定 IP 地址的安全地址，则将不能再通过自动学习来增加安全地址。
- 手工配置一部分安全地址，剩下的部分让设备自己学习。

如果一个端口被配置为一个安全端口，当其安全地址的数目已经达到允许的最大个数后，如果该端口收到一个源地址不属于端口上的安全地址的包时，一个安全违例将产生。当安全违例将产生时，您可以设置下面几种针对违例的处理模式：

protect：当安全地址个数满后，安全端口将丢弃未知名地址（不是该端口的安全地址的任何一个）的包。该处理模式为缺省的对违例的处理模式。

restrict：当违例产生时，将发送一个 Trap 通知。

shutdown：当违例产生时，将关闭端口并发送一个 Trap 通知。

37.3.2 配置端口安全

37.3.2.1 端口安全的缺省配置

下表显示的端口安全的缺省配置：

内容	缺省设置
端口安全开关	所有端口均关闭端口安全功能
最大安全地址个数	128
安全地址	无
违例处理方式	保护(protect)

37.3.2.2 端口安全配置指导

配置端口安全时有如下一些限制：

- 一个安全端口不能是一个 Aggregate Port
- 一个安全端口不能是 SPAN 的目的端口
- 一个安全端口只能是一个 Access Port

802.1x 认证功能和端口安全功能互斥使能。802.1x 认证功能和端口安全功能都可以保证网络使用者的合法性，使能其一就可以达到控制端口接入的目的。

同时 IP+MAC 绑定和仅 IP 绑定的安全地址与 ACLs 共享系统的硬件资源，因此当您在某一个安全端口上应用了 ACLs，则该端口上所能设置的 IP+MAC 绑定和仅 IP 绑定的安全地址个数将会减少。

一个安全端口上的安全地址的格式必须保持一致，即一个端口上的安全地址要么全是绑定了 IP 地址的安全地址，要么都是不绑定 IP 地址的安全地址。如果一个安全端口同时包含这两种格式的安全地址，则不绑定 IP 地址的安全地址将失效(绑定 IP 地址的安全地址优先级更高)。

37.3.2.3 配置安全端口及违例处理方式

在接口配置模式下，请使用如下命令配置安全端口及违例处理方式：

命令	作用
Ruijie(config-if)# **switchport port-security**	打开该接口的端口安全功能
Ruijie(config-if)# **switchport port-security maximum** *value*	设置接口上安全地址的最大个数，范围是 1－1000，缺省值为 128。
Ruijie(config-if)# **switchport port-security violation{protect | restrict | shutdown}**	设置处理违例的方式： **protect**：保护端口，当安全地址个数满后，安全端口将丢弃未知名地址(不是该端口的安全地址中的任何一个)的包 **restrict**：当违例产生时，将发送一个 Trap 通知 **shutdown**：当违例产生时，将关闭端口并发送一个 Trap 通知。当端口因为违例而被关闭后，您可以在全局配置模式下使用命令 **errdisable recovery** 来将接口从错误状态中恢复过来。

在接口配置模式下，您可以使用命令 **no switchport port-security** 来关闭一个接口的端口安全功能。使用命令 **no switchport port-security maximum** 来恢复为缺省个数。使用命令 **no switchport port-security violation** 来将违例处理置为缺

省模式。

下面的例子说明了如何使能接口 gigabitethernet 0/3 上的端口安全功能，设置最大地址个数为 8，设置违例方式为 protect

```
Ruijie# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Ruijie(config)# interface gigabitethernet 0/3
Ruijie(config-if)# switchport mode access
Ruijie(config-if)# switchport port-security
Ruijie(config-if)# switchport port-security maximum 8
Ruijie(config-if)# switchport port-security violation protect
Ruijie(config-if)# end
```

说明：

1，如果安全端口上配置了仅 Ip 或者 Ip+Mac 绑定的的安全地址，当端口学习的 Mac 地址达到并超出设置的安全地址个数时，并不会产生违例事件。

2，只在首次端口违例产生才发出 trap 及 log 信息。

3，二层端口安全的违例产生到处理完成的生效时间大约在 1 秒内。

37.3.2.4 配置安全端口上的安全地址

在接口配置模式下，请使用如下命令为安全端口添加安全地址：

命令	作用
Ruijie(config-if)# **switchport port-security mac-address** *mac-address* [**ip-address** *ip-address*]	手工配置接口上的安全地址。**ip-address**(可选)：为这个安全地址绑定的 IP 地址。

在接口配置模式下，您可以使用命令 **no switchport port-security mac-address** *mac-address* 来删除该接口的安全地址。

下面的例子说明了如何为接口 gigabitethernet 0/3 配置一个安全地址：00d0.f800.073c，并为其绑定一个 IP 地址：192.168.12.202。

```
Ruijie# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Ruijie(config)# interface gigabitethernet 0/3
Ruijie(config-if)# switchport mode access
Ruijie(config-if)# switchport port-security
Ruijie(config-if)# switchport port-security mac-address
00d0.f800.073c ip-address 192.168.12.202
```

```
Ruijie(config-if)# end
```

说明：

S3760 的 IP+MAC 安全地址与仅 IP 安全地址在同一端口中不能共存。

S3760 在配置了仅 IP 安全地址模式情况下，能学习到 IP 地址匹配失败的报文的源 mac 地址，但 **show mac-address-table** 命令查看不到这部分 mac 地址。

37.3.2.5 配置安全地址的老化时间

您可以为一个接口上的所有安全地址配置老化时间。打开这个功能，您需要设置安全地址的最大个数，这样，您就可以让设备自动的增加和删除接口上的安全地址。

在接口配置模式下，请使用如下命令配置安全地址老化时间：

命令	作用
Ruijie(config-if)**#switchport port-security aging**{**static** | **time** *time* }	**static**：加上这个关键字，表示老化时间将同时应用于手工配置的安全地址和自动学习的地址，否则只应用于自动学习的地址。**time** ：表示这个端口上安全地址的老化时间，范围是 0－1440，单位是分钟。如果您设置为 0，则老化功能实际上被关闭。老化时间按照绝对的方式计时，也就是一个地址成为一个端口的安全地址后，经过 Time 指定的时间后，这个地址就将被自动删除。*Time* 的缺省值为 0。

您可以在接口配置模式下使用命令 **no switchport port-security aging time** 来关闭一个接口的安全地址老化功能，使用命令 **no switchport port-security aging static** 来使老化时间仅应用于动态学习到的安全地址。

下面的例子说明了如何配置一个接口 gigabitethernet 0/3 上的端口安全的老化时间，老化时间设置为 8 分钟，老化时间同时应用于静态配置的安全地址：

```
Ruijie# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Ruijie(config)# interface gigabitthernet 0/3
Ruijie(config-if)# switchport port-security aging time 8
Ruijie(config-if)# switchport port-security aging static
Ruijie(config-if)# end
```

37.3.3 查看端口安全信息

在特权模式下，您可以通过如下命令来查看端口安全的信息：

命令	作用
Ruijie**#show port-security interface** [*interface-id*]	查看接口的端口安全配置信息。
Ruijie**#show port-security address**	查看安全地址信息。
Ruijie**# show port-security address** [*interface-id*]	显示某个接口上的安全地址信息
Ruijie**# show port-security**	显示所有安全端口的统计信息，包括最大安全地址数，当前安全地址数以及违例处理方式等。

下面的例子显示了接口 Gigabitethernet 0/3 上的端口安全配置：

```
Ruijie# show port-security interface gigabitethernet 0/3
Interface : Gi0/3
Port Security: Enabled
Port status : down
Violation mode:Shutdown
Maximum MAC Addresses:8
Total MAC Addresses:0
Configured MAC Addresses:0
Aging time : 8 mins
SecureStatic address aging : Enabled
```

下面的例子显示了系统中的所有安全地址

```
Ruijie# show port-security address
Vlan Mac Address IP Address Type Port Remaining Age(mins)
---------------------------------------- ---------------
1  00d0.f800.073c   192.168.12.202    Configured   Gi0/3    8
1  00d0.f800.3cc9   192.168.12.5      Configured   Gi0/1    7
```

您也可以只显示一个接口上的安全地址，下面的例子显示了接口 gigabitstethernet 0/3 上的安全地址

```
Ruijie# show port-security address interface gigabitethernet
0/3
Vlan Mac Address IP Address Type Port Remaining Age(mins)
---- --------------- ---------- ------ -----------------
1    00d0.f800.073c  192.168.12.202 Configured Gi0/3    8
```

下面的例子显示的是安全端口的统计信息

```
Ruijie# show port-security
```

```
Secure Port MaxSecureAddr(count) CurrentAddr(count) Security
Action
----------  ------------------  -----------------  -------
Gi0/1       128                 1                  Restrict
Gi0/2       128                 0                  Restrict
Gi0/3       8                    1                  Protect
```

37.4 ARP-CHECK

37.4.1 概述

ARP 报文检查（ARP-CHECK）基于全局或者端口上的 MAC+IP 绑定安全功能，例如 DHCP Snooping，端口安全或者全局地址绑定等。通过丢弃非法用户的 ARP 报文来有效防止欺骗 ARP，防止非法信息点冒充网络关键设备的 IP（如服务器），造成网络通讯混乱。

ARP-CHECK 有三种模式：打开，关闭和自动模式，缺省为自动模式。

在打开模式下，无论端口上有没有安全配置都检查 ARP 报文。如果端口上没有合法用户，则来自这个端口的所有 arp 报文都将被丢弃。

在关闭模式下，不检查端口上的 ARP 报文。

在自动模式下，在端口上没有合法用户的情况下，不检查 ARP 报文；在有合法用户的情况下，检查 ARP 报文。

ARP 报文检查的限制：

1. 打开端口安全地址的 ARP 报文检查会使所有端口的绑定 IP 的安全地址最大数目减少一半。

2. 打开端口安全地址的 ARP 报文检查对已经存在的安全地址不生效。如果您要使以前设置的安全地址生效，可以重新关闭，再打开该端口的安全地址。端口 ARP 报文检查使用了策略管理模块，和其它策略管理模块共享硬件资源。如果硬件资源不足时，可能出现部分安全地址的 ARP 报文检查不生效的现象。

3. 当 MAC+IP 的安全地址表项比较多时，打开 ARP Check Cpu，对 CPU 性能影响比较大,会降低 CPU 效率。

注意:

S3760 交换机，由路由口送往 CPU 的 ARP 报文，在任何模式下都不会进行 ARP 报文检查。

37.4.2 配置 ARP-CHECK

从特权模式下开始配置 ARP-CHECK

命令	作用
Ruijie**# configure t**	进入配置模式
Ruijie(config)**#interface** *interface-id*	进入接口模式
Ruijie(config-if)**# arp-check**	打开 arp 报文检查
Ruijie(config-if)**# no arp-check**	关闭 arp 报文检查
Ruijie(config-if)**# arp-check auto**	还原成缺省配置

例如在端口上添加合法用户 mac 地址 00d0.f822.33ab，IP 地址为 192.168.2.5 时，端口上的 ARP 报文检查会自动启用。

```
Ruijie#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Ruijie(config)# interface fastEthernet 0/5
Ruijie(config-if)# switchport port-security
Ruijie(config-if)# switchport port-security mac-address
00d0.f822.33ab ip-address 192.168.2.5
```

ARP 报文检查自动启用了，如果你想关闭 ARP 报文检查。

```
Ruijie(config-if)# no arp-check
```

37.5 基于端口的流控制组合配置案例

37.5.1 拓扑图

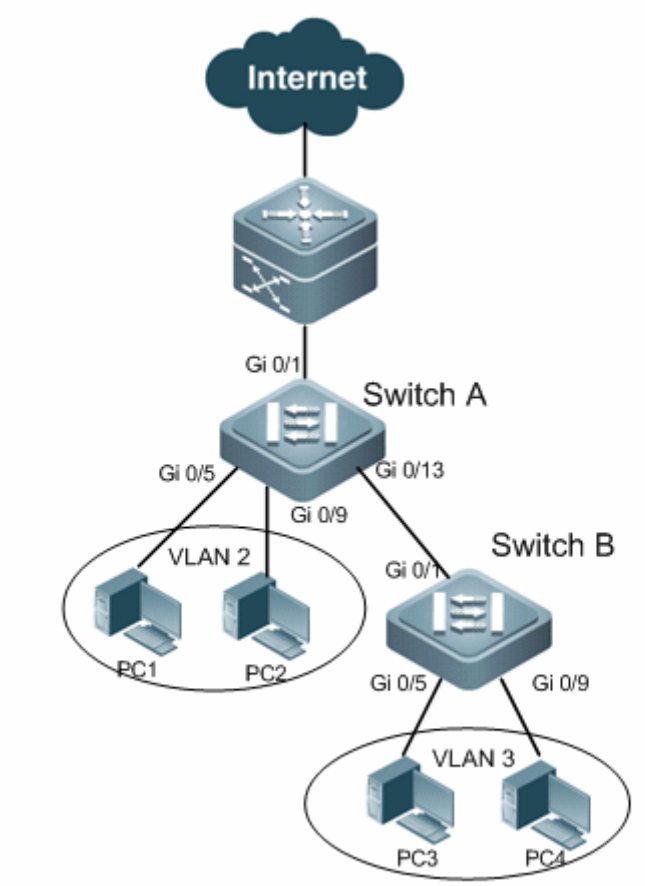


图 11 网络拓扑

37.5.2 应用需求

上图是典型企业网络的简化拓扑，组网需求如下：

1、 防止设备受到广播、多播、未知名单播报文攻击。

2、 只允许直连用户（本例指 Switch A 的直连用户）以指定的 IP/MAC 地址上网，源地址与指定 IP/MAC 不匹配的报文将被丢弃，防止源 IP,/源 MAC 欺骗。

3、 不允许接入用户（本例为 Switch B 的接入用户）之间进行二层报文通信，避免接入用户之间互相干扰（如 ARP 欺骗或 DOS 攻击等）

37.5.3 配置要点

- 配置要点

1. 在所有接入设备（本例为 Switch A、Switch B）的端口上开启风暴控制。
2. 在接入设备（本例为 Switch A）的端口（本例为 Gi 0/5 和 Gi 0/9）上配置端口安全功能可满足第二个需求。
3. 在接入设备（本例为 Switch B）上配置端口保护功能，可满足第三个需求。

- 注意事项

开启端口安全功能并配置 IP/MAC 表项后，ARP CHECK 功能将自动生效，根据设置的 IP/MAC 对 ARP 报文的源地址进行检查。

37.5.4 配置步骤

➢ **配置 Switch A**

第一步，创建交换机的 VLAN，并设置端口属性。

！创建 VLAN 2

```
Ruijie#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Ruijie(config)#vlan 2
Ruijie(config-vlan)#exit
```

！设置端口属性

```
Ruijie(config)#interface gigabitEthernet 0/5
Ruijie(config-if-GigabitEthernet 0/5)#switchport access vlan 2
Ruijie(config-if-GigabitEthernet 0/5)#exit
Ruijie(config)#interface gigabitEthernet 0/9
Ruijie(config-if-GigabitEthernet 0/9)#switchport access vlan 2
Ruijie(config-if-GigabitEthernet 0/9)#exit
Ruijie(config)#interface gigabitEthernet 0/1
Ruijie(config-if-GigabitEthernet 0/1)#switchport mode trunk
Ruijie(config-if-GigabitEthernet 0/1)#exit
Ruijie(config)#interface gigabitEthernet 0/13
Ruijie(config-if-GigabitEthernet 0/13)#switchport mode trunk
Ruijie(config-if-GigabitEthernet 0/13)#exit
```

第二步，在所有接入端口上开启风暴控制。

```
Ruijie(config)#interface range gigabitEthernet 0/1,0/5,0/9,0/13
Ruijie(config-if-range)#storm-control broadcast
Ruijie(config-if-range)#storm-control multicast
Ruijie(config-if-range)#storm-control unicast
Ruijie(config-if-range)#exit
```

第三步，在直连用例的端口上开启端口安全，并绑定 IP 和 MAC

！绑定接入用户：IP（1.1.1.1）/MAC（0000.0000.0001）

```
Ruijie(config)#interface gigabitEthernet 0/5
Ruijie(config-if-GigabitEthernet 0/5)#switchport port-security
```

```
Ruijie(config-if-GigabitEthernet 0/5)#switchport port-security
mac-address 0000.0000.0001 ip-address 1.1.1.1
Ruijie(config-if-GigabitEthernet 0/5)#exit
```

！绑定接入用户：IP（1.1.1.2）/MAC（0000.0000.0002）

```
Ruijie(config)#interface gigabitEthernet 0/9
Ruijie(config-if-GigabitEthernet 0/9)#switchport port-security
Ruijie(config-if-GigabitEthernet 0/9)#switchport port-security
mac-address 0000.0000.0002 ip-address 1.1.1.2
Ruijie(config-if-GigabitEthernet 0/9)#exit
```

- **配置 Switch B**

第一步，创建交换机的 VLAN，并设置端口属性

！创建 VLAN 3

```
Ruijie#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Ruijie(config)#vlan 3
Ruijie(config-vlan)#exit
Ruijie(config)#interface gigabitEthernet 0/5
Ruijie(config-if-GigabitEthernet 0/5)#switchport access vlan 3
Ruijie(config-if-GigabitEthernet 0/5)#exit
Ruijie(config)#interface gigabitEthernet 0/9
Ruijie(config-if-GigabitEthernet 0/9)#switchport access vlan 3
Ruijie(config-if-GigabitEthernet 0/9)#exit
Ruijie(config)#interface gigabitEthernet 0/1
Ruijie(config-if-GigabitEthernet 0/1)#switchport mode trunk
Ruijie(config-if-GigabitEthernet 0/1)#exit
```

第二步，在所有接入端口上开启风暴控制

```
Ruijie(config)#interface range gigabitEthernet 0/1,0/5,0/9
Ruijie(config-if-range)#storm-control broadcast
Ruijie(config-if-range)#storm-control multicast
Ruijie(config-if-range)#storm-control unicast
Ruijie(config-if-range)#exit
```

第三步，在接入端口上开启端口保护功能。

```
Ruijie(config)#interface gigabitEthernet 0/5
Ruijie(config-if-GigabitEthernet 0/5)#switchport protected
Ruijie(config-if-GigabitEthernet 0/5)#exit
Ruijie(config)#interface gigabitEthernet 0/9
Ruijie(config-if-GigabitEthernet 0/9)#switchport protected
Ruijie(config-if-GigabitEthernet 0/9)#exit
```

37.5.5 配置验证

第一步，查看 Switch A 的配置情况，关注点：是否在各端口上开启风暴控制、是

否在直连用户的端口上配置端口安全功能并静态绑定 IP+MAC 地址。

```
Ruijie#show running-config
vlan 2
!
interface GigabitEthernet 0/1
 switchport mode trunk
 storm-control broadcast
 storm-control multicast
 storm-control unicast
!
interface GigabitEthernet 0/5
 switchport access vlan 2
 switchport port-security mac-address 0000.0000.0001 ip-address
1.1.1.1
 switchport port-security
 storm-control broadcast
 storm-control multicast
 storm-control unicast
!
interface GigabitEthernet 0/9
 switchport access vlan 2
 switchport port-security mac-address 0000.0000.0002 ip-address
1.1.1.2
 switchport port-security
 storm-control broadcast
 storm-control multicast
 storm-control unicast
!
interface GigabitEthernet 0/13
 switchport mode trunk
 storm-control broadcast
 storm-control multicast
 storm-control unicast
```

第二步，查看 Switch B 的配置情况，关注点：是否在各端口开启风暴控制、是否在直连用户的端口上开启端口保护功能。

```
Ruijie#show running-config
vlan 3
!
interface GigabitEthernet 0/1
 switchport mode trunk
 storm-control broadcast
 storm-control multicast
 storm-control unicast
!
interface GigabitEthernet 0/5
```

```
 switchport access vlan 3
 switchport protected
 storm-control broadcast
 storm-control multicast
 storm-control unicast
!
interface GigabitEthernet 0/9
 switchport access vlan 3
 switchport protected
 storm-control broadcast
 storm-control multicast
 storm-control unicast
```

第三步，在 Switch A 查看端口安全地址绑定信息，以及 ARP CHECK 使能情况。

```
Ruijie#show port-security all
Vlan Port  Arp-Check  Mac Address IP Address  Type remaining Age
(mins)
---- -------- ---------- -------------- --------------
2    Gi0/5 Enabled  0000.0000.0001 1.1.1.1   Configured      -
2    Gi0/9 Enabled  0000.0000.0002 1.1.1.2   Configured      -
```

第四步，在 Switch B 查看端口 GigabitEthernet 0/5 的端口保护配置信息，其它端口的端口保护配置信息不再举例说明。

```
Ruijie#show interfaces gigabitEthernet 0/5 switchport
Interface   Switchport Mode    Access Native Protected VLAN lists
----------------------- ---------- --------- ------ ------
GigabitEthernet0/5 enabled ACCESS   3   1    Enabled   ALL
```

38 802.1X 配置

本章节描述基于 AAA 服务配置的相关内容。802.1x 用于控制用户对网络访问的认证，并对其提供授权与记帐功能。

本节包含如下内容：

- 概述
- 配置 802.1x
- 查看 802.1x 的配置及当前的统计值
- 配置 802.1x 其它注意事项

说明：

有关本节引用的 CLI 命令的详细使用信息及说明，请参照《配置 802.1X 命令》

38.1 概述

IEEE 802 LAN 中，用户只要能接到网络设备上，不需要经过认证和授权即可直接使用。这样，一个未经授权的用户，他可以没有任何阻碍地通过连接到局域网的设备进入网络。随着局域网技术的广泛应用，特别是在运营网络的出现，对网络的安全认证的需求已经提到了议事日程上。如何在以太网技术简单、廉价的基础上，提供用户对网络或设备访问合法性认证，已经成为业界关注的焦点。IEEE 802.1x 协议正是在这样的背景下提出的。

IEEE802.1x（Port-Based Network Access Control）是一个基于端口的网络存取控制标准，为 LAN 提供点对点式的安全接入。这是 IEEE 标准委员会针对以太网的安全缺陷而专门制定的标准，能够在利用 IEEE 802 LAN 优势的基础上，提供一种对连接到局域网设备的用户进行认证的手段。

IEEE 802.1x 标准定义了一种基于“客户端——服务器”（Client-Server）模式实现了限制未认证用户对网络的访问。客户端要访问网络必须先通过服务器的认证。

在客户端通过认证之前，只有 EAPOL 报文（Extensible Authentication Protocol over LAN）可以在网络上通行。在认证成功之后，正常的数据流便可在网络上通行。

应用 802.1x 锐捷的设备提供了 Authentication，Authorization，and Accounting 三种安全功能，简称 AAA。

- Authentication： 认证，用于判定用户是否可以获得访问权，限制非法用户

- Authorization： 授权，授权用户可以使用哪些服务，控制合法用户的权限
- Accounting： 计账，记录用户使用网络资源的情况，为收费提供依据

我们将从以下几个方面阐述 802.1x

- 设备的角色
- 认证的发起及认证过程中的报文交互
- 已认证用户及未认证用户的状态
- 典型应用的拓扑结构

38.1.1 设备的角色

IEEE802.1x 标准认证体系由恳请者、认证者、认证服务器三个角色构成，在实际应用中，三者分别对应为：工作站（Client）、设备(network access server，NAS)、Radius-Server。

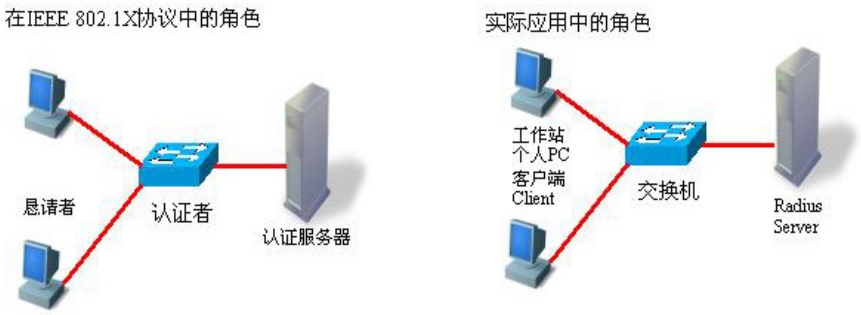


图 1

- 恳请者

恳请者是最终用户所扮演的角色，一般是个人 PC。它请求对网络服务的访问，并对认证者的请求报文进行应答。恳请者必须运行符合 IEEE 802.1x 客户端标准的软件，目前最典型的就是 WindowsXP 操作系统自带的 IEEE802.1x 客户端支持，另外，锐捷也已推出符合该客户端标准的 STAR Supplicant 软件。

- 认证者

认证者一般为交换机等接入设备。该设备的职责是根据客户端当前的认证状态控制其与网络的连接状态。在客户端与服务器之间，该设备扮演着中介者的角色：从客户端要求用户名，核实从服务器端的认证信息，并且转发给客户端。因此，设备除了扮演 IEEE802.1x 的认证者的角色，还扮演 RADIUS Client 角色，因此我们把设备称作 network access server(NAS)，它要负责把从客户端收到的回应封装到 RADIUS 格式的报文并转发给 RADIUS Server，同时它要把从 RADIUS Server 收到的信息解释出来并转发给客户端。

扮演认证者角色的设备有两种类型的端口：受控端口（controlled Port）和非受控端口（uncontrolled Port）。连接在受控端口的用户只有通过认证才能访问网络资源；

而连接在非受控端口的用户无须经过认证便可以直接访问网络资源。我们把用户连接在受控端口上，便可以实现对用户的控制；非受控端口主要是用来连接认证服务器，以便保证服务器与设备的正常通讯。

- 认证服务器

认证服务器通常为 RADIUS 服务器，认证过程中与认证者配合，为用户提供认证服务。认证服务器保存了用户名及密码，以及相应的授权信息，一台服务器可以对多台认证者提供认证服务，这样就可以实现对用户的集中管理。认证服务器还负责管理从认证者发来的记帐数据。锐捷网络科技公司实现 802.1x 的设备完全兼容标准的 Radius Server，如 MicroSoft win2000 Server 自带的 Radius Server 及 Linux 下的 Free Radius Server。

38.1.2 认证的发起及认证过程中的报文交互

恳请者和认证者之间通过 EAPOL 协议交换信息，而认证者和认证服务器通过 RADIUS 协议交换信息，通过这种转换完成认证过程。EAPOL 协议封装于 MAC 层之上，类型号为 0x888E。同时，标准为该协议申请了一个组播 MAC 地址 01-80-C2-00-00-03，用于初始认证过程中的报文传递。

下图是一次典型的认证过程中，三个角色设备的报文交互过程

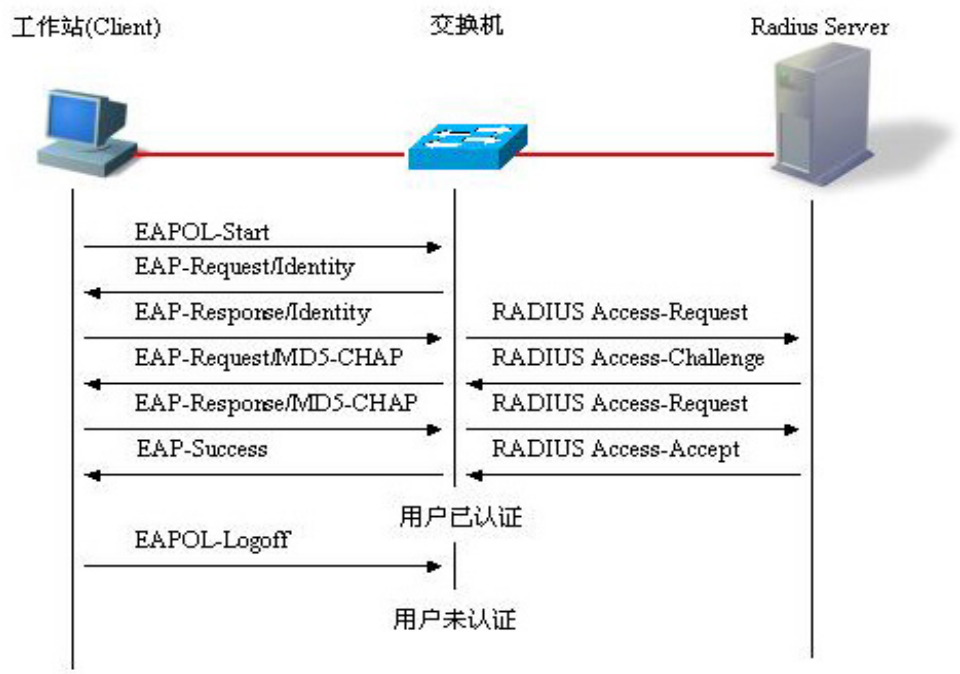


图 2

该过程是一个典型的由用户发起的认证过程（在一些特殊的情形下，设备也可能主动发出认证请求，过程与该图一致，只是少了用户主动发出请求这一步）。

38.1.3 已认证用户及未认证用户的状态

802.1x 中根据端口的认证状态来决定该端口上的用户是否允许访问网络，由于我们对 802.1X 进行扩展，是基于用户的，所以，我们是根据一个端口下的用户的认证状态来决定该用户是否允许访问网络资源。一个非受控端口下的所有用户均可使用网络资源，而一个受控端口下的用户只有处于已认证状态（Authorized）才能访问网络资源。一个用户刚发起认证时，状态处于未认证状态（unauthorized），这时它不能访问网络，在认证通过后，该用户的状态会变为已认证状态（authorized），此时该用户便可以使用网络资源。

如果工作站不支持 802.1x，而该机器连接在受控端口下，当设备请求该用户的用户名时，由于工作站不支持导致没对该请求做出响应。这就意味着该用户仍然处于未认证状态（unauthorized），不能访问网络资源。

相反地，如果工作站支持 802.1x，而所连的设备不支持 802.1x。用户发出的 EAPOL-START 帧无人响应，用户在发送一定数目的 EAPOL-START 帧仍未收到回应的情形下，将认为自己所连的端口是非受控端口，而直接使用网络资源。

在支持 802.1x 的设备下，所有端口的默认设置是非受控端口，我们可以把一个端口设置成受控端口，从而要求这个端口下的所有用户都要进行认证。

当用户通过了认证（设备收到了从 RADIUS Server 服务器发来的成功报文），该用户便转变成已认证状态(authorized)，该用户可以自由使用网络资源。如果用户认证失败以至仍然处于未认证状态，可以重新发起认证。如果设备与 RADIUS server 之间的通讯有故障，那么该用户仍然处于未认证状态（unauthorized），网络对该用户来说仍然是不可使用的。

当用户发出 EAPOL-LOGOFF 报文后，该用户的状态由已认证(authorized)转向未认证状态(unauthorized)。

当设备的某个端口变为 LINK-DOWN 状态，该端口上的所有用户均变为未认证(unauthorized)状态。

当设备重新启动，该设备上的所有用户均变为未认证状态(unauthorized)。

如果您要强制一个用户通过认证，可以通过添加静态 MAC 地址来实现。

38.1.4 典型应用的拓扑结构

A、带 802.1x 的设备作为接入层设备

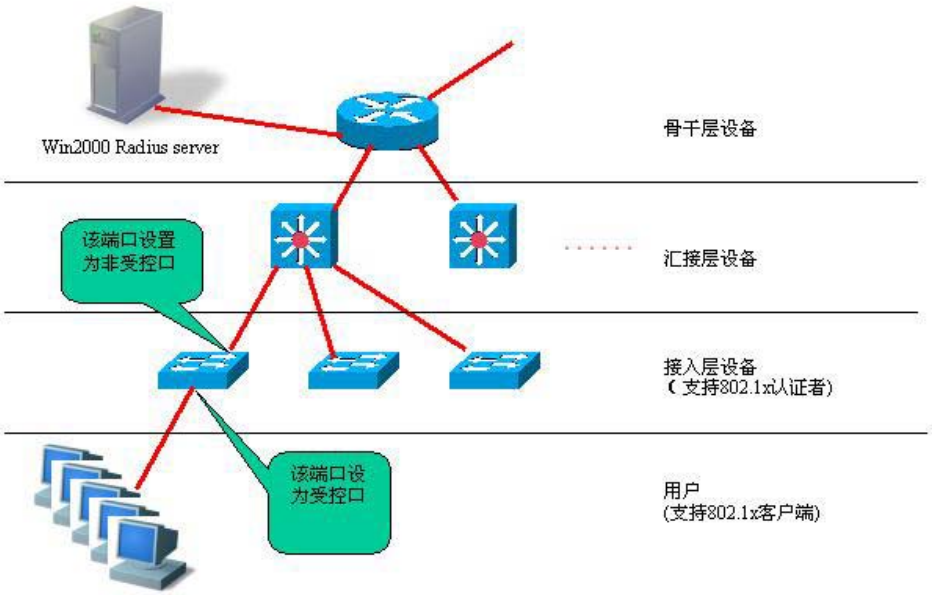


图 3

该方案的说明：

- 该方案的要求：

1. 用户支持 802.1x，即要装有 802.1x 的客户端软件（windowXp 自带，Star-supplicant 或其他符合 IEEE802.1x 标准的客户端软件）。

2. 接入层设备支持 IEEE 802.1x

3. 有一台（或多台）支持标准 RADIUS 的服务器作为认证服务器

- 该方案的配置要点：

1. 与 Radius Server 相连的口及上联口，配置成非受控口，以便设备能正常地与服务器进行通讯，以及使已认证用户能通过上联口访问网络资源

2. 与用户连接的端口要设置为受控口，以实现对接入用户的控制，用户必须通过认证才能访问网络资源。

- 该方案的特点：

1. 每台支持 802.1x 的设备所负责的客户端少，认证速度快。各台设备之间相互独立，设备的重启等操作不会影响到其它设备所连接的用户。

2. 用户的管理集中于 Radius Server 上，管理员不必考虑用户连接在哪台设备上，便于管理员的管理

3. 管理员可以通过网络管理接入层的设备

B、带 802.1x 的设备作为汇接层设备

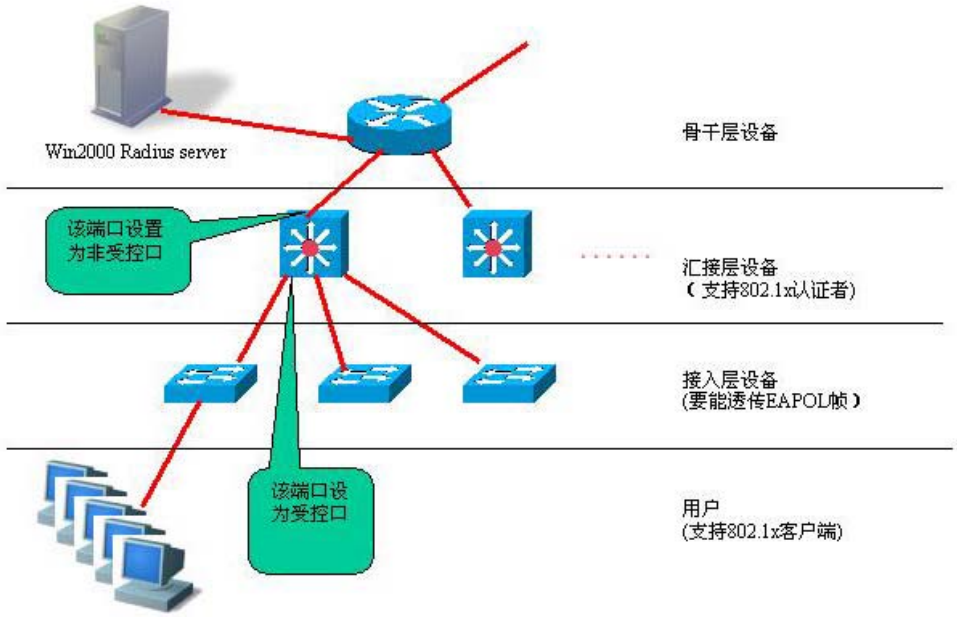


图 4

该方案的说明：

- 该方案的要求：

1. 用户支持 802.1x，即要装有 802.1x 的客户端软件（windowXp 自带，Star-supplicant 或其他符合 IEEE802.1x 标准的客户端软件）。

2. 接入层设备支持要能透传 IEEE 802.1x 帧(EAPOL)

3. 汇接层设备支持 802.1x（扮演认证者角色）

4. 有一台（或多台）支持标准 RADIUS 的服务器作为认证服务器

- 该方案的配置要点：

1. 与 Radius Server 相连的口及上联口，配置成非受控口，以便设备能正常地与服务器进行通讯，以及使已认证用户能通过上联口访问网络资源

2. 与接入层设备连接的端口要设置为受控口，以实现对接入用户的控制，用户必须通过认证才能访问网络资源。

- 该方案的特点：

1. 由于是汇接层设备，网络规模大，下接用户数多，对设备的要求高，若该层设备发生故障，将导致大量用户不能正常访问网络。

2. 用户的管理集中于 Radius Server 上，管理员无需考虑用户连接在哪台设备上，便于管理员的管理

3. 接入层设备可以使用较廉价的非网管型设备（只要支持 EAPOL 帧透传）

4. 管理员不能通过网络直接管理接入层设备

38.2 配置 802.1X

我们将通过以下章节说明如何配置 802.1x

- 802.1x 的默认配置
- 802.1x 的配置注意事项
- 配置设备与 RADIUS SERVER 之间的通讯
- 设置 802.1X 认证的开关
- 打开/关闭一个端口的认证
- 打开定时重认证
- 打开/关闭 过滤非锐捷 supplicant 功能的开关
- 改变 QUIET 时间
- 设置报文重传间隔
- 设置最大请求次数
- 设置最大重认证次数
- 设置 Server-timeout
- 配置设备主动发起 802.1x 认证
- 配置 802.1x 记帐
- 配置 IP 授权模式
- 发布广告信息
- 某端口下的可认证主机列表
- 授权
- 配置认证方式
- 配置备份认证服务器
- 对在线用户的配置及管理
- 实现用户与 IP 的捆绑
- 基于端口的流量记费
- 实现端口的 VLAN 自动跳转及控制开关
- 实现端口的 GUEST VLAN 功能
- 代理服务器屏蔽和拨号屏蔽
- 配置客户端在线探测
- 配置 EAPOL 帧带 TAG 的选项开关
- **错误！未找到引用源。**
- **错误！未找到引用源。**

- **错误！未找到引用源。**
- 配置MAC配置MAC旁路认证旁路认证
- 配置MAC旁路认证 超时时间
- 配置MAC配置MAC旁路认证违例旁路认证违例
- 配置失败 VLAN
- 配置失败VLAN配置失败VLAN尝试次数尝试次数

38.2.1 802.1x 的默认配置

下表列出 802.1x 的一些缺省值

内容	默认值
认证 Authentication	关闭 DISABLE
记帐 Accounting	关闭 DISABLE
认证服务器(Radius Server) *服务器 IP 地址(ServerIp) *认证 UDP 端口 *密码(Key)	*无缺省值 *1812 *无缺省值
记帐服务器(Accounting Server) *记帐服务器 IP 地址 *记帐 UDP 端口	*无缺省值 *1813
所有端口的类型	非受控端口（所有端口均无须认证便可直接通讯）
定时重认证 re-authentication	关闭
定时重认证周期 reauth_period	3600 秒
认证失败后允许再次认证的间隔	10 秒
重传时间间隔	3 秒
最大重传次数	3 次
客户端超时时间	3 秒，在该段时间内没有收到客户端的响应便认为这次通讯失败
服务器超时时间	5 秒，在该段时间内没有收到服务器的回应，便认为这次通讯失败
某端口下可认证主机列表	无缺省值

38.2.2 802.1X 的配置注意事项

- 只有支持 802.1x 的产品，才能进行以下设置。

- 802.1x 既可以在二层下又可以在三层下的设备运行。
- 要先设置认证服务器 IP 地址，radius－server 认证方式才可以正常工作。
- 802.1Q TUNNEL 端口不允许打开 1X 认证。
- Aggregate Port 不允许打开 1X 认证。
- 交换机只要有一个端口打开了 1x 功能，所有的端口都会将 1x 协议报文送到 cpu。

38.2.3 配置设备与 RADIUS SERVER 之间的通讯

Radius Server 维护了所有用户的信息：用户名、密码、该用户的授权信息以及该用户的记帐信息。所有的用户集中于 Radius Server 管理，而不必分散于每台设备，便于管理员对用户的集中管理。

设备要能正常地与 RADIUS SERVER 通讯，必须进行如下设置：

Radius Server 端：要注册一个 Radius Client。注册时要告知 Radius Server 设备的 IP、认证的 UDP 端口（若记帐还要添记帐的 UDP 端口）、设备与 Radius Server 通讯的约定密码，还要选上对该 Client 支持 EAP 扩展认证方式）。对于如何在 Radius Server 上注册一个 Radius Client，不同软件的设置方式不同，请查阅相关的文档。

设备端：为了让设备能与 Server 进行通讯，设备端要做如下的设置：设置 Radius Server 的 IP 地址，认证（记帐）的 UDP 端口，与服务器通讯的约定密码。

在特权模式下，按如下步骤设置设备与 Radius Server 之间的通讯：

命令	作用
configure terminal	进入全局配置模式
aaa new-model	打开 AAA 开关
radius-server host *ip-address* [**auth-port** *port*] [**acct-port** *port*]	配置 RADIUS 服务器
Radius-server key *string*	配置 RADIUS Key
end	退出到特权模式
write	保存配置
show radius server	显示 RADIUS 服务器

使用 **no radius-server host** *ip-address* **auth-port** 命令将 Radius Server 认证 UDP 端口恢复为缺省值。使用 **no radius-server key** 命令删除 Radius Server 认证密码。以下例子是设置 server ip 为 192.168.4.12、认证 Udp 端口为 600、以明文方式设置约定密码：

```
Ruijie# configure terminal
Ruijie(config)# radius-server host 192.168.4.12
```

```
Ruijie(config)# radius-server host 192.168.4.12 auth-port 600
Ruijie(config)# radius-server key MsdadShaAdasdj878dajL6g6g
a
Ruijie(config)# end
```

- 官方约定的认证的 UDP 端口为 1812
- 官方约定的记帐的 UDP 端口为 1813
- 设备与 Radius Server 约定的密码的长度建议不少于 16 个字符
- 设备与 Radius Server 连接的端口要设置成非受控口

38.2.4 设置 802.1X 认证的开关

当打开 802.1x 认证时，设备会主动要求受控端口上的主机进行认证，未通过认证的主机不允许访问网络。

在特权模式下，按如下步骤打开 1x 认证：

命令	作用
configure terminal	进入全局配置模式
aaa new-model	打开 AAA 开关
radius-server host *ip-address* [**auth-port** *port*] [**acct-port** *port*]	配置 RADIUS 服务器
Radius-server key string	配置 RADIUS Key
aaa authentication dot1x *auth* **group radius**	配置 dot1x 认证方法列表
dot1x authentication *auth*	dot1x 应用认证方法列表
end	退出到特权模式
write	保存配置
show running-config	显示配置

以下例子是打开 802.1x 认证：

```
Ruijie# configure terminal
Ruijie(config)# aaa new-model
Ruijie(config)# radius-server host 192.168.217.64
Ruijie(config)# radius-server key starnet
Ruijie(config)# aaa authentication dot1x authen group radius
Ruijie(config)# dot1x authentication authen
Ruijie(config)# end
Ruijie# show running-config
!
aaa new-model
!
aaa authentication dot1x authen group radius
```

```
!
username ruijie password 0 starnet
!
radius-server host 192.168.217.64
radius-server key 7 072d172e071c2211
!
!
!
dot1x authentication authen
!
interface VLAN 1
 ip address 192.168.217.222 255.255.255.0
 no shutdown
!
!
line con 0
line vty 0 4
!
end
```

802.1x 在应用 RADIUS 认证方法时，先配置 Radius Server 的 IP 地址，并确保设备与 Radius Server 之间的通讯正常。若没有 Radius Server 的配合，设备无法完成认证功能。如何设置 Radius Server 与设备之间的通讯请见上一章节。

38.2.5 打开/关闭一个端口的认证

在 802.1x 打开的情形下，打开一个端口的认证，则该端口成为受控口，连接在该端口下的用户要通过认证才能访问网络，然而，在非受控端口下的用户可以直接访问网络。

在特权模式下，按如下步骤设置一个端口的认证状态。

命令	作用
configure terminal	进入全局配置模式
interface *interface*	进入接口设置模式,指定要配置的接口
dot1x port-control auto	设置该接口为受控接口（打开接口认证功能）。使用该命令的 no 选项关闭该接口的认证功能。
end	退出到特权模式
write	保存配置
show dot1x port-control	查看 802.1x 接口认证配置

使用该命令的 **no dot1x port-control** 命令关闭接口的认证功能。以下例子是设置

以太网接口 1/1 为受倥接口：

```
Ruijie# configure terminal
Ruijie(config)# interface f 1/1
Ruijie(config-if)# dot1x port-control auto
Ruijie(config)# end
```

当接口被设置为受控口的时候，只允许 EAP 报文过，往 CPU 的报文也在受控范围内。

注意：

在 S3760 交换机中，接口被设为受控口的时候，不能过滤掉发往 CPU 的非 EAP 报文。如 arp 报文和 ping 报文。
如果希望 cpu 不收到任何来自受控口的非 EAP 报文，可以将交换机设置为管理 vlan，与用户 vlan 分开。

38.2.6 打开定时重认证

802.1x 能定时主动要求用户重新认证，这样可以防止已通过认证的用户不会被其他用户冒用，还可以检测用户是否断线，使记费更准确。除了可以设定重认证的开关，我们还可以定义重认证的间隔。默认的重认证间隔是 3600 秒。在根据时常进行记费的场合下，要根据具体的网络规模确定重认证间隔，使之既有足够时间完成一次认证又尽可能精确。

在特权模式下，按如下步骤打开/关闭重认证状态，并且设置重认证时间间隔。

命令	作用
configure terminal	进入全局配置模式
dot1x re-authentication	打开定时重认证功能
dot1x timeout re-authperiod *seconds*	设置重认证时间间隔
End	退出到特权模式
Write	保存配置
show dot1x	显示 dot1x 配置

使用 **no dot1x re-authentication** 命令关闭定时重认证功能，使用 **no dot1x timeout re-authperiod** 命令将重认证时间间隔恢复为缺省值。

以下例子是打开定时重认证功能，并设置重认证时间间隔为 1000 秒：

```
Ruijie# configure terminal
```

```
Ruijie(config)# dot1x re-authentication
Ruijie(config)# dot1x timeout re-authperiod 1000
Ruijie(config)# end
Ruijie# show dot1x
802.1X Status:       Disabled
Authentication Mode:  EAP-MD5
Authed User Number:   0
Re-authen Enabled:    Enabled
Re-authen Period:     1000 sec
Quiet Timer Period:   10 sec
Tx Timer Period:      3 sec
Supplicant Timeout:   3 sec
Server Timeout:       5 sec
Re-authen Max:        3 times
Maximum Request:      3 times
Filter Non-RG Supp:   Disabled
Client Oline Probe:   Disabled
Eapol Tag Enable:     Disabled
Authorization Mode:   Disabled
```

若打开重认证，请注意重认证间隔的合理性。要根据具体的网络规模而设置。

38.2.7 打开/关闭 过滤非锐捷 supplicant 功能的开关

当选用锐捷的 supplicant 产品作为 802.1x 认证的客户端时，如果用户同时也使用了其它的一些 802.1x 认证客户端(比如打开了 WindowsXP 自带的 802.1x 认证功能选项)，则有可能会使认证失败。

这时可以通过打开本功能，将非锐捷 supplicant 发出的 802.1x 报文过滤掉，来保证 supplicant 的认证不受其它 802.1x 客户端的影响。默认情况下，该功能是关闭的。

在特权模式下，按如下步骤打开、关闭过滤功能：

命令	作用
configure terminal	进入全局配置模式
dot1x private-supplicant-only	打开过滤功能
end	退出到特权模式
write	保存配置
show dot1x	显示 dot1x 配置

以下例子是打开锐捷 supplicant 功能的开关的配置：

```
Ruijie# configure terminal
```

```
Ruijie(config)# dot1x private-supplicant-only
Ruijie(config)# end
Ruijie# show dot1x
802.1X Status:       enable
Authentication Mode:  eap-md5
Total User Number:    0(exclude dynamic user)
Authed User Number:   0(exclude dynamic user)
Dynamic User Number:  0
Re-authen Enabled:    enable
Re-authen Period:     2 sec
Quiet Timer Period:   10 sec
Tx Timer Period:      3 sec
Supplicant Timeout:   3 sec
Server Timeout:       5 sec
Re-authen Max:        3 times
Maximum Request:      3 times
Private supplicant only:   enable
Client Online Probe:  disable
Eapol Tag Enable:     disable
Authorization Mode:   disable
```

使用 **no dot1x private-supplicant-only** 命令关闭功能。

38.2.8 改变 QUIET 时间

当用户认证失败时，设备将等待一段时间后，才允许用户再次认证。Quiet Period 的时间长度便是允许再认证的时间间隔。该值的作用是避免设备受恶意攻击。Quiet Period 的默认间隔为 10 秒，

我们可以通过设定较短的 Quiet Period 使用户可以更快地进行再认证。

在特权模式下，按如下步骤设置 Quiet Period 的值

命令	作用
configure terminal	进入全局配置模式
dot1x timeout quiet-period *seconds*	设置 Quiet Period 值
end	退出到特权模式
write	保存配置
show dot1x	显示 dot1x 配置

使用 **no dot1x timeout quiet-period** 命令将 Quiet Period 恢复为缺省值。以下例子是设置 QuietPeriod 值 500 秒：

```
Ruijie# configure terminal
```

```
Ruijie (config)# dot1x timeout quiet-period 500
Ruijie(config)# end
```

38.2.9 设置报文重传间隔

设备发 EAP-request/identity 之后，若在一定的时间内没有收到用户的回应，设备将重传这个报文。该值的默认值为 3 秒，要根据具体的网络规模进行调整。

在特权模式下，按如下步骤设置报文重传间隔

命令	作用
configure terminal	进入全局配置模式
dot1x timeout tx-period *seconds*	设置报文重传间隔
end	退出到特权模式
write	保存配置
show dot1x	显示 dot1x 配置

使用 **no dot1x timeout tx-period** 命令将报文重传间隔恢复为缺省值。以下例子是设置报文重传间隔为 100 秒：

```
Ruijie# configure terminal
Ruijie(config)# dot1x timeout tx-period 100
Ruijie(config)# end
```

38.2.10 设置最大请求次数

设备在RadiusServer 发出认证请求后，若在ServerTimeout 时间内没收到Radius Server 的回应，将重传该报文。最大请求次数指的是设备重传请求的最大数，超过该次数设备将认为本次认证失败。默认的重传次数为 3 次，我们要根据具体的网络环境进行调整。

在特权模式下，按如下步骤设置报文重传次数

命令	作用
configure terminal	进入全局配置模式
dot1x max-req *count*	设置报文重传次数
end	退出到特权模式
write	保存配置
show dot1x	显示 dot1x 配置

```
Ruijie#show dot1x
```

使用 **no dot1x max-req** 命令将报文重传次数恢复为缺省值。以下例子是设置报文重传次数为 5 次：

```
Ruijie# configure terminal
Ruijie(config)# dot1x max-req 5
Ruijie(config)# end
```

38.2.11 设置最大重认证次数

当用户认证失败后，设备会尝试再次与用户进行认证。在认证次数超过最大重认证次数之后，设备就认为这个用户已经断线，结束认证过程。系统默认的次数是 3 次，我们可以重新设置这个值。

在特权模式下，按如下步骤设置最大重认证次数：

命令	作用
configure terminal	进入全局配置模式
dot1x reauth-max *count*	设置最大重认证次数
end	退出到特权模式
write	保存配置
show dot1x	显示 dot1x 配置

使用 **no dot1x reauth-max** 命令将最大重认证次数恢复为缺省值。以下例子是设置最大重认证次数为 3 次：

```
Ruijie# configure terminal
Ruijie(config)# dot1x reauth-max 3
Ruijie(config)# end
Ruijie#
```

38.2.12 设置 Server-timeout

该值指的是 Radius Server 的最大响应时间，若在该时间内，设备没有收到 Radius Server 的响应，将认为本次认证失败。

在特权模式下，按如下步骤设置 Server-timeout、恢复为默认值

命令	作用
configure terminal	进入全局配置模式
dot1x timeout server-timeout *seconds*	设置服务器最大响应时间，使用该命令的 no 选项将其恢复为缺省值

end	退出到特权模式
write	保存配置
show dot1x	显示 dot1x 配置

38.2.13 配置设备主动发起 802.1x 认证

802.1x 是基于端口的安全访问认证，用户要访问网络，就必须先进行认证，绝大多数情况下，认证是由用户端通过发出 EAPOL-START 报文来发起的。关于认证过程中报文的交互，请参考“认证的发起及认证过程中的报文交互”。

但是在某些特殊情况下，认证却需要由设备来发起。比如设备复位、认证端口状态由 linkdown 变为 linkup，这时候为了保证已认证的用户能够继续使用网络，就需要设备主动发起认证。另外，如果用户使用了一些不会主动发起认证请求的 802.1x 客户端软件来进行认证（比如使用 WindowsXP 自带的 802.1x 客户端软件），这时候就需要设备能够主动发起认证，设备通过发出 EAP-request/identity 组播报文来强制要求认证端口下的所有用户进行认证。

以下将介绍如何配置设备主动发起 802.1x 认证，以及在不同的应用环境下用户应该如何合理地进行配置。

打开/关闭设备主动发起认证的开关

当该功能关闭时，设备只在复位和认证端口状态改变的时候发起一次认证请求，这是为了保证在线用户能够继续认证使用网络，其它任何时候设备都不会主动发起认证请求。当该功能打开后， 用户可以配置主动发起认证请求的次数、发送认证请求的间隔、用户认证通过后是否要停止发送请求等附属功能。

进入特权模式，按以下步骤打开主动认证功能：

命令	作用
configure terminal	进入全局配置模式
dot1x auto-req	打开主动认证功能。该功缺省认情况下是关闭的
end	退出到特权模式
write	保存配置
show dot1x	显示 dot1x 配置

使用该命令的 **no** 选项可以关闭该功能，只有该功能打开的情况下，后面的设置才能起作用，设置设备主动发送认证请求的次数，用户可以设置设备主动发起认证请求的次数，这个值可根据实际的网络环境设定。

进入特权模式，按以下步骤设置发送报文的次数：

命令	作用
configure terminal	进入全局配置模式
dot1x auto-req packet-num *num*	设备主动发送 num 个 802.1x 认证请求报文,如果 num 为 0，则设备将持续地发送该报文。缺省值是 0（无限个）
end	退出到特权模式
write	保存配置
show dot1x auto-req	显示配置

使用该命令的 **no** 选项恢复为默认值，下面配置报文发送间隔

进入特权模式，按以下步骤设置主动发送报文的间隔：

命令	作用
configure terminal	进入全局配置模式
dot1x auto-req req-interval *interval*	设置报文发送间隔
end	退出到特权模式
write	保存配置
show dot1x auto-req	显示配置

使用该命令的 **no** 选项恢复为默认值，由于发出认证请求组播报文会导致认证口下的所有用户进行重认证，所以发送间隔不要太小，否则会影响认证效率。

设置在发现用户认证通过后是否停止发送请求报文，在某些应用需求下(比如一个端口下面只接一个用户)，我们可以指定设备在发现用户认证通过后，停止向对应端口发送认证请求，如果用户下线，则继续发送。

进入特权模式，按以下步骤设置：

命令	作用
configure terminal	进入全局配置模式
dot1x auto-req user-detect	端口有认证用户停止发送报文。该功能缺省打开
end	退出到特权模式
write	保存配置
show dot1x auto-req	显示配置

使用该命令的 **no** 选项关闭该功能，用户在设置该功能时要仔细考虑当前的网络应用环境。

应用示例以上三个命令为用户提供了灵活的应用策略，用户可以根据具体的网络应用环境来选择合适的配置命令。为了方便用户进行配置，我们建立了如下的应用配置表，用户可参考该表来进行配置：

	方案 1	方案 2	方案 3
用户环境	一端口对任意用户	一端口对单用户	一端口对多用户
是否要求使用锐捷的 supplicant 作为认证客户端	是	否	否
建议采用的配置命令	不必打开 dot1x auto-req 功能	**dot1x auto-req** **dot1x auto-req packet-num** *num* **dot1x auto-req req-interval** *interval* **dot1x auto-req user-detect**	**dot1x auto-req** **dot1x auto-req packet-num** *0* **dot1x auto-req req-interval** *interval* **no dot1x auto-req user-detect**

38.2.14 配置 802.1x 记帐

锐捷网络公司的 802.1x 实现了记帐功能。该记帐是基于时长的，也就是说 802.1x 记录了用户第一次认证通过到用户主动退出或设备检测到用户中断的时间长度。

在用户第一次认证通过之后，设备会向服务器发一个记帐开始请求，当用户主动离线或设备检测到用户已离线或用户的物理连接已中断，设备将向服务器发一个记帐结束请求。服务器组将会把这些信息记录在服务器组的数据库上。网管便可以根据这些信息提供记帐的依据。

锐捷网络公司的 802.1x 十分重视记帐的可靠性，为了避免记帐服务器的意外情况，特别支持记帐备份服务器。当一个服务器由于各种原因而不能提供记帐服务，设备将自动把记帐信息转发给另一台备份服务器，这大大提高了记帐可靠性。

在用户主动退出的情形下，记帐的时长是精确的；在用户意外中断的情形下，用户记帐的精度以重认证的间隔为准（设备通过重认证检测一个用户是否意外中断）。

要打开设备的记帐工作需对设备做如下的设置：

1. 在 Radius Server 注册这台设备为 Radius Client，如认证时的操作
2. 设置记帐服务器的 IP 地址
3. 设置记帐的 UDP 端口
4. 在 802.1x 打开的前提下，打开记帐服务

在特权模式下，按如下步骤设置记帐服务

命令	作用
configure terminal	进入全局配置模式
aaa new-model	打开 AAA 功能
aaa group server radius *gs*	配置记帐服务器组
server *192.168.4.12* **acct-port** *11*	向服务器组添加服务器
exit	退回全局配置模式
aaa accounting network *acct* **start-stop group** *gs*	配置记账方法列表
dot1x accounting *acct*	为 802.1X 应用记账方法列表
end	退出到特权模式
write	保存配置
show running-config	显示配置

使用 **no aaa accounting network** 命令的将删除记账方法列表。使用 **no dot1x accounting** 命令使用默认的 dot1x 记帐方法。以下例子是设置记帐服务器的 IP 地址为 192.168.4.12、设置备份记帐服务器的 IP 地址为 192.168.4.13、设置记帐服务器的 UDP 端口为 1200、应用 802.1x 记帐功能：

```
Ruijie# configure terminal
Ruijie(config)# aaa new-model
Ruijie(config)# aaa group server radius acct-use
Ruijie(config-gs-radius)# server 192.168.4.12 acct-port 1200
Ruijie(config-gs-radius)# server 192.168.4.13 acct-port 1200
Ruijie(config-gs-radius)# exit
Ruijie(config)# aaa accounting network acct-list start-stop
group acct-use
Ruijie(config)# dot1x accounting acct-list
Ruijie(config)# end
Ruijie# write memory
Ruijie# show running-config
```

注意：

1) 与 Radius Server 的约定记帐密码与认证相同

2) 必须在 AAA 打开的前提下才能打开记帐

3) 802.1X 认证通过后才能进行记账

4) 802.1x 的记帐功能在默认的情形下是关闭的

5) 记帐的数据库格式请见相关的 Radius Server 文档

同时我们支持计费更新功能 ，在 NAS 设备上设定了计费更新包的周期后，NAS 设备即定时定期的向 Radius Server 发送计费更新包，Radius Server 上可以定义，如果多少次（几个周期）没有收到 NAS 设备发过来的某用户的计费更新包时，即认为该 NAS 或该用户不在线，Radius Server 即可停止相关用户的计费，从在线用户表中将其删除等等操作。

在特权模式下，按如下步骤设置记帐更新功能服务

命令	作用
configure terminal	进入全局配置模式
aaa new-model	打开 AAA 功能
aaa accounting update	设置记帐计费更新功能
end	退出到特权模式
write	保存配置
show running-config	显示配置

用 **no aaa accounting update** 可以关闭记账更新功能服务

```
Ruijie# configure terminal
Ruijie(config)# aaa accounting update
Ruijie(config)# end
Ruijie# write memory
Ruijie# show running-config
```

以下章节为锐捷网络产品特有的功能：

为了方便宽带运营商及其他特殊场合的用途，锐捷对 802.1x 的功能在标准的基础上进行了扩展（该扩展是完全基于标准之上，没有任何的与 IEEE 802.1x 不兼容）。

38.2.15 配置 IP 授权模式

锐捷网络实现的 802.1x，可以强制要求已认证的用户使用固定的 IP。管理员通过配置 IP 授权模式来限定用户获得 IP 地址的方式。IP 授权模式有四种：DISABLE 模式、DHCP SERVER 模式、RADIUS SERVER 模式 、SUPPLICANT 模式。下面分别介绍这四种工作模式的特性：

DISABLE 模式（默认）：在该模式下，设备不对用户的 IP 做限制，用户只需认证通过便可以使用网络。

DHCP SERVER 模式：用户的 IP 通过指定的 DHCP SERVER 获得，只有指定的 DHCP SERVER 分配的 IP 才是合法的 IP，对于 DHCP 模式可以使用 DHCP relay option82 实现 802.1X 的更为灵活的 IP 分配策略。典型的方案图如下：

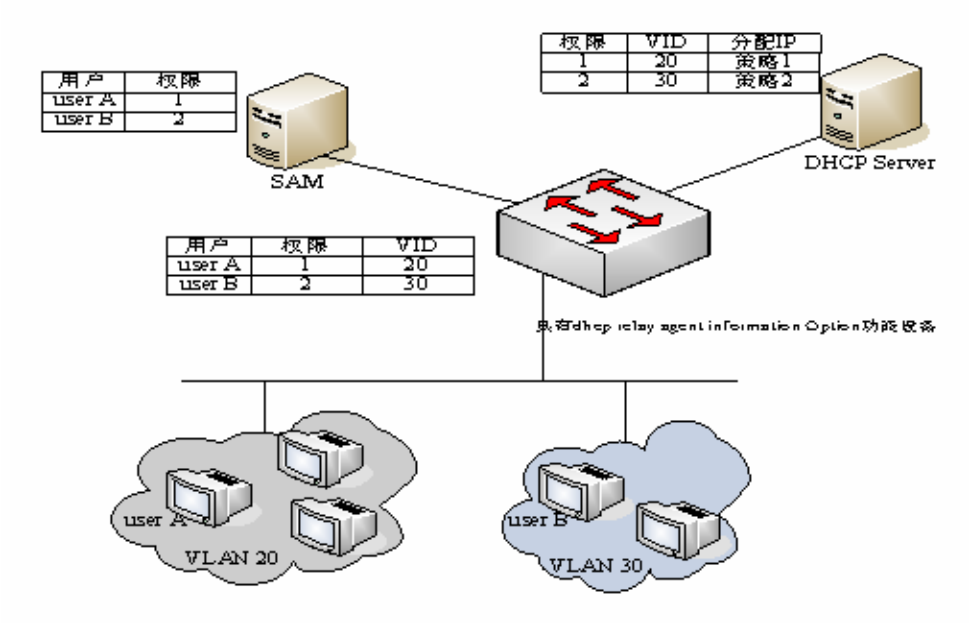


图 5

各用户通过 DHCP Client 发起 IP 请求，具有 dhcp relay option82 的网络设备融合 SAM 服务器上的用户权限，构造 option82 字段封装于 DHCP 请求报文，该 option82 字段由“vid + 权限”组成。DHCP Server 由 option82 字段选择不同的分配策略。

此种模式下需要对 DHCP Relay 及相应的 option82 进行配置，如打开 DHCP relay 功能，并选择 option82 策略，相应的配置见 DHCP Relay 配置指南和命令参考。

RADIUS SERVER 模式：用户的 IP 通过 RADIUS SERVER 指定。用户只能用 RADIUS SERVER 指定的 IP 访问网络。

SUPPLICANT 模式 ：所绑定用户的 IP 为 SUPPLICANT 认证时 PC 的 IP。认证后，用户只能用该 IP 访问网络，不可随意更改 IP。

四种模式下的应用模型：

- DISABLE 模式：适合不对用户限定 IP 的场合。用户只需通过认证便可以访问网络。

- DHCP SERVER 模式：用户 PC 通过 DHCP 获得 IP 地址，管理员通过配置设备的 DHCP RELAY 来限定用户访问的 DHCP SERVER，这样，只有指定的 DHCP SERVER 分配的 IP 才是合法的。

- RADIUS SERVER 模式：用户 PC 使用固定的 IP，RADIUS SERVER 配置了<用户—IP>的对应关系，并通过 RADIUS 的 Framed-IP-Address 属性告知设备，用户只能用该 IP 才能访问网络。

- SUPPLICANT 模式：用户 PC 使用固定的 IP，SUPPLICANT 将该信息告知设备，用户只能用认证时的 IP 才能访问网络。

注意：

用户切换模式时会导致已经认证的用户全部下线，所以建议在用户使用前就配置好认证模式。

在特权模式下，如下配置 IP 授权模式：

命令	作用
configure terminal	进入全局配置模式
aaa new-model	打开 AAA 功能
aaa authorization ip-auth-mode {disabled | dhcp-server | radius-server | supplicant }	配置 IP 授权模式
end	退出到特权模式
write	保存配置
show running-config	显示配置

以下例子是把配置 IP 授权模式为 RADIUS-SERVER 模式：

```
Ruijie# configure terminal
Ruijie(config)# aaa authorization ip-auth-mode radius-serve
r
Ruijie(config)# end
Ruijie# show running-config
!
aaa new-model
!
aaa authorization ip-auth-mode radius-server
!
Ruijie# write memory
```

38.2.16 发布广告信息

锐捷网络实现的 802.1x，可以在 Radius Server 端配置 Reply-Message 字段，当认证成功后，该字段的信息可以在锐捷网络推出的 802.1x 客户端 Star-Supplicant 上显示出来，便于运营商发布一些信息。

该消息只有在用户第一次认证时显示，重认证时不会显示，这样就避免了对用户的频繁打扰。

广告信息的显示窗口支持 HTML，会自动把消息中的 http://XXX.XXX.XX 转换成可直接跳转的连接，便于用户查看详细的信息。

广告信息的发布：

1) 运行商在 Radius Server 端，配置 Reply Message 属性的内容

2) 只有锐捷的客户端 r-supplicant 才支持（对本公司设备的用户免费），其它的客户端看不到信息但不影响使用

3) 在设备端无需设置

38.2.17 某端口下的可认证主机列表

为了增强 802.1x 的安全性，我们在不影响 IEEE 802.1x 的基础上进行了扩展，网管可以限定某个端口的认证的主机列表。如果一个端口下的可认证主机列表为空，则任何用户均可认证；若可认证主机列表不为空，那么只有列表中的主机允许认证。允许认证的主机用 MAC 标识。

添加某一端口下的可认证主机、删除某一端口下的可认证主机

命令	作用
configure terminal	进入全局配置模式
dot1x auth-address-table address *mac-addr* **interface** *interface*	设置可认证主机列表
end	退出到特权模式
write	保存配置
show running-config	显示配置

注意：

若主机列表为空，该端口允许任何主机认证。

38.2.18 授权

为了方便运营商，锐捷的产品可以对不同类型的用户提供不同质量的服务，如：提供给用户的最大带宽不同。而这些信息集中于 Radius Server 上，管理员不必对每台设备进行配置。

由于 Radius 没有标准的属性来表示最大数据率。我们只能通过厂商自定义属性来传递授权信息。

定义的通用格式如下：

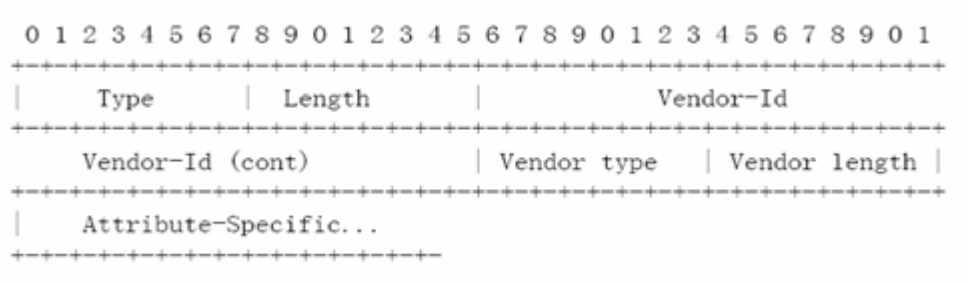


图 6

对于最大数据率应填的值如下：

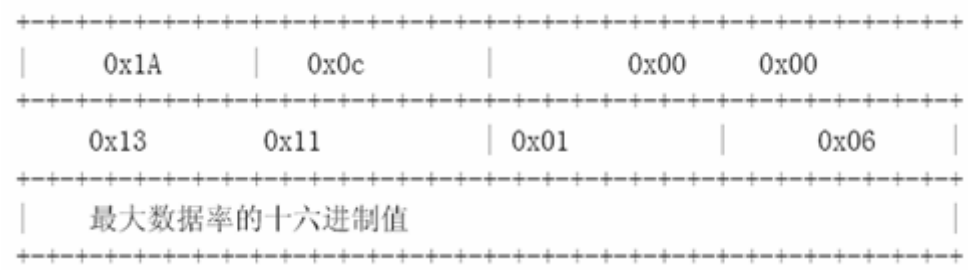


图 7

最大数据率的单位：kbps

对于最大数据率为 10M 的用户应填如下：

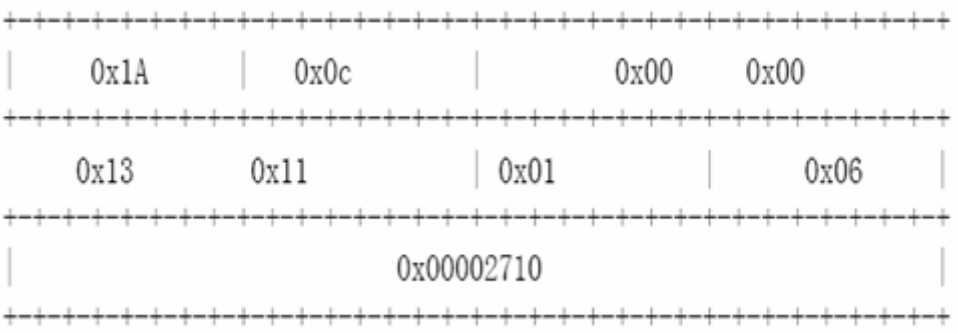


图 8

自定义的头按照以上格式填，最大数据率为 10M，即 10000kbsp，转换成 16 进制则 0x00002710，填在相应的字段上即可。

该功能无须设置设备端。只要设备端支持授权便可。

注意：

S3760 不支持 dscp 属性的下传，下行 rate-limit 属性的下传。

38.2.19 配置认证方式

在标准中，802.1x 是通过 EAP-MD5 进行认证。锐捷网络科技公司实现的 802.1X，不但可以通过 EAP-MD5 方式认证（默认），还可以采用 CHAP 和 PAP 方式认证。采用 CHAP 的好处是可以减少一次设备与 RADIUS SERVER 的通讯，减轻 RADIUS SERVER 的压力。和 CHAP 方式一样，PAP 和 RADIUS SERVER 间的通讯也只有一次，虽然 PAP 认证方式由于其在安全方面存在很大缺陷，所以不建议使用，但是在某些情况下还是可以满足用户的特殊需求，比如当用户使用的安全服务器只支持 PAP 认证模式时，选择 PAP 认证模式就能够充分地利用现有的资源，保护用户的投资。

在特权模式下，按如下步骤设置 802.1x 的认证方式：

命令	作用
configure terminal	进入全局配置模式
dot1x auth-mode mode	配置认证模式
end	退出到特权模式
write	保存配置
show dot1x	显示配置

以下例子是配置成 CHAP 模式的例子：

```
Ruijie# configure terminal
Ruijie (config)# dot1x auth-mode CHAP
Ruijie(config)# end
Ruijie# show dot1x
802.1X Status:       Disabled
Authentication Mode:  CHAP
Authed User Number:   0
Re-authen Enabled:    Disabled
Re-authen Period:     3600 sec
Quiet Timer Period:   10 sec
Tx Timer Period:      3 sec
Supplicant Timeout:   3 sec
Server Timeout:       5 sec
Re-authen Max:        3 times
Maximum Request:      3 times
Filter Non-RG Supp:   Disabled
Client Oline Probe:   Disabled
Eapol Tag Enable:     Disabled
Authorization Mode:   Group Server
```

38.2.20 配置备份认证服务器

锐捷基于 802.1x 的认证系统，可以支持备份服务器。当主服务器因各种原因当机之后，设备将自动向方法列表服务器组下一个服务器提交认证请求。

在特权模式下，按如下方式设置备份认证服务器。

命令	作用
configure terminal	进入全局配置模式
aaa new-model	打开 aaa 开关
aaa group server radius *gs-name*	配置服务器组
server sever	配置服务器
server server-backup	配置备份服务器
end	退出到特权模式
write	保存配置
show dot1x	显示配置

以下例子是配置 192.168.4.12 为备份服务器的例子：

```
Ruijie# configure terminal
Ruijie# aaa new-model
Ruijie(config)# aaa group server radius auth-ll
Ruijie(config-gs-radius)# server 192.168.4.1
Ruijie(config-gs-radius)# server 192.168.4.12
Ruijie(config-gs-radius)# end
Ruijie#
```

38.2.21 对在线用户的配置及管理

锐捷设备提供了通过 SNMP 对已认证用户进行管理的功能。管理员可以通过 SNMP 查看已认证用户的信息，还可以强制使一个用户下线。被强制下线的用户必须再次认证才能使用网络资源。

该功能无需配置设备。

38.2.22 实现用户与 IP 的捆绑

用锐捷网络公司提供的客户端，以及对 Radius Server 的正确配置，可以实现用户与 IP 的唯一绑定。某个用户必须以管理员分配的 IP 进行认证，否则将不能认证成功。

该功能无需配置设备，用户需用锐捷网络公司提供的客户端软件，管理员需配置 Radius Server。

38.2.23 基于端口的流量记费

锐捷网络设备除了提供针对时长的记费外，在设备的每个端口只接入一个用户的情形下，设备还可以提供针对流量的记费功能。

该功能无需配置设备，但需 Radius server 支持。

38.2.24 配置 802.1X 端口动态 VLAN 自动跳转

动态 VLAN 自动跳转功能需要在远端 RADIUS 服务器上设置针对用户的下传 VLAN，RADIUS 服务器通过相应定义的 RADIUS 属性封装下传 VLAN 信息，接入设备收到该信息并在用户认证后就会自动把该用户所在端口加入到 RADIUS 服务器下传 VLAN 中。这个过程无需管理员再手工配置设备。

在接入设备端您可以通过 **show dot1x summary** 这条命令看出该用户所处的真实 VLAN，通过 **show dot1x user id** 命令可以看到 RADIUS 服务器下传 VLAN。

接入设备可以通过扩展 RADIUS 属性及 RADIUS 标准属性两种方式接收 RADIUS 服务器的下传 VLAN。

RADIUS 服务器使用基于标准扩展的属性下传 VLAN 给接入设备时，服务器将扩展属性封装在 26 号 RADIUS 标准属性中，扩展厂商 ID 为十六进制数 0x00001311，默认情况下该扩展属性类型号为 4，您可以在接入设备上通过配置命令 **radius attribute 4 vendor-type** *type* 来接收扩展属性类型号设置为 *type* 的下传 VLAN，配置命令可参见“配置 RADIUS”。

接入设备也能支持 RADIUS 服务器使用 RADIUS 标准属性下传 VLAN，包括以下属性组合：

- 64 号属性 Tunnel-Type
- 65 号属性 Tunnel-Medium-Type
- 81 号属性 Tunnel-Private-Group-ID

并且在应用动态 VLAN 自动跳转功能时，取值范围如下

- Tunnel-Type=VLAN (13)
- Tunnel-Medium-Type=802（6）
- Tunnel-Private-Group-ID=VLAN ID 或 VLAN Name

具体可参见 RFC2868 和 RFC3580。

对于接入设备接收到下传 VLAN 后处理步骤如下：先将下传 VLAN 属性作为 VLAN 名称，查看接入设备上是否配置有相同名称的 VLAN，若存在相同名称的 VLAN，则用户所在端口自动跳转至该 VLAN；若不存在相同名称的 VLAN，则将下传 VLAN 属性作为 VLAN ID，若该字符串表达的 VLAN ID 合法（系统支持的 VLAN ID 范围），则用户所在端口自动跳转至该 VLAN；若该字符串表达的 VLAN ID 为 0，则认为没有下传 VLAN 信息；其他情况用户将认证失败。

接入设备只支持在端口模式为 ACCESS 和 TRUNK 模式下进行 802.1X 认证，其他端口模式下启用动态 VLAN 自动跳转功能将认证失败，在 ACCESS 和 TRUNK 端口模式下，VLAN 自动跳转支持情况如下：

1、认证端口模式为 ACCESS 的端口 VLAN 自动跳转

下传 VLAN 在设备上没有配置时，若下传 VLAN 被设备识别为 VLAN ID 时，设备创建相应 ID 的 VLAN，并使认证端口跳转到新创建的 VLAN 中；若下传 VLAN 被设备识别为 VLAN 名称，认证端口下用户将认证失败。

下传 VLAN 在设备上已经有配置存在时，若下传 VLAN 在设备上被配置为 ACCESS 口不支持跳转的 VLAN（见下面描述）时，则认证端口下用户将认证失败；若下传 VLAN 在设备上被配置为 ACCESS 口支持跳转的 VLAN，则认证成功并使认证端口跳转到下传 VLAN 中。

ACCESS 口不支持跳转的 VLAN 有如下：

- 私有 VLAN；
- Remote VLAN；
- Super VLAN。

2、认证端口模式为 TRUNK 的端口 Native VLAN 设置

TRUNK 口开启认证，使用下传 VLAN 作为 Native VLAN 设置到认证端口。

下传 VLAN 在设备上已经有配置存在时，若下传 VLAN 被设备识别为 VLAN ID，设置认证端口的 Native VLAN 为下传 VLAN；若下传 VLAN 杯设备识别为 VLAN 名称，端口下用户将认证失败。

下传 VLAN 在设备上已经有配置存在时，若下传 VLAN 在设备上被配置为 TRUNK 口不支持跳转的 VLAN(见下面描述)时，则认证端口下的用户将认证失败，若下传 VLAN 在设备上被配置为 TRUNK 口支持跳转的 VLAN，则认证成功并设置认证端口 Native VLAN 为下传 VLAN。

TRUNK 口不支持跳转 VLAN 有如下：

- 私有 VLAN；
- Remote VLAN；
- Super VLAN。

在设备上设置某个端口进行动态 VLAN 自动跳转功能，您可以通过以下步骤配置：

打开 AAA 开关

	命令	作用
Step 1	**configure terminal**	进入全局配置模式
Step 2	**aaa new-model**	打开 AAA 开关

具体参见“配置 AAA”

配置 RADIUS 服务器

	命令	作用
Step 1	**configure terminal**	进入全局配置模式

Step 2	**radius-server host** *host-ip*	配置 RADIUS 服务器
Step 3	**radius-server key** *text*	配置 RADIUS 服务器共享密钥

具体参见“配置 RADIUS”

配置方法列表

	命令	作用
Step 1	**configure terminal**	进入全局配置模式
Step 2	**aaa authentication dot1x** *list1* **group radius**	配置认证方法列表 list1
Step 3	**aaa accounting network** *list2* **start-stop group radius**	配置记账方法列表 list2

具体参见“配置 AAA”

802.1X 选择方法列表

	命令	作用
Step 1	**configure terminal**	进入全局配置模式
Step 2	**dot1x authentication** *list1*	选择认证方法列表为 list1，list1 为步骤 3 中配置
Step 3	**dot1x accounting** *list2*	选择记账方法列表为 list2，list2 为步骤 3 中配置

接口使能 802.1X 认证

	命令	作用
Step 1	**configure terminal**	进入全局配置模式
Step 2	**interface** *interface_id*	进入需要配置的接口模式下，*interface_id* 为所要进入接口
Step 3	**dot1x port-control auto**	接口使能 802.1X 认证

接口使能 VLAN 自动跳转功能

	命令	作用
Step 1	**configure terminal**	进入全局配置模式
Step 2	**interface** *interface_id*	进入需要配置的接口模式下，*interface_id* 为所要进入接口
Step 3	**dot1x dynamic-vlan enable**	接口使能 VLAN 自动跳转功能

注意

VLAN 自动跳转功能必须要在接口下打开动态跳转开关，即需要相应的接口下配置命令 **dot1x dynamic-vlan enable**。否则会忽略封装下传 VLAN 的 RADIUS 属性。

dot1x dynamic-vlan enable 必须在接口配置命令 **dot1x port-control auto** 后才能打开，在使用 **no dot1x port-control auto** 命令之后接口 VLAN 自动跳转功能的使能开关将自动关闭。

动态 vlan 跳转不支持 private vlan，也就是说不能将 private vlan 配置为 dot1x 的动态 vlan。

查看动态 VLAN 自动跳转信息

完成以上配置并且在认证成功后，可以通过下列步骤查看动态 VLAN 自动跳转相关信息.

	命令	作用
Step 1	**show dot1x user id** *session_id*	查看会话号为 *session_id* 的用户信息，包括动态 VLAN 自动跳转信息。
Step 2	**show dot1x summary**	该命令可以看出用户真实所在 VLAN

至此，接入设备上的 VLAN 自动跳转功能配置完成，有关的注意事项请参见“配置 802.1X 其他注意事项”章节。

38.2.25 实现端口的 GUEST VLAN 功能

如果在交换机上，配置了 **guest vlan**，则当端口主动发送一定数量的认证请求，而没有对应的回应或者在此期间没有收到 **eapol** 报文，把端口添加到 **guest vlan** 中。您可以通过 **show running-config** 这条命令看出这条命令的配置，以及通过 **show vlan** 查看到端口是否已经跳转到 **guest vlan**。

设备上可以设置某个端口是否允许 **GUEST VLAN** 跳转，您可以通过以下步骤配置：

命令	作用
configure terminal	进入全局配置模式
interface *interface*	进入接口配置模式
dot1x dynamic-vlan enable	接口允许 Vlan 跳转
[no] dot1x guest-vlan *vid*	配置是否允许 guest vlan g 功能，缺省值是关闭的
end	退出到特权模式
write	保存配置
show running-config	显示配置

注意：

1、 必须配置 **dot1x dynamic-vlan enable，guest vlan 才能生效。**

2、 配置 **guest vlan** 期间最好不要配置二层属性，尤其是不要再手动设置端口 **vlan。**

3、 **guest vlan** 退出，当端口下有 **eapol** 报文的时候则会退出。当端口 **linkdown** 的时候也会退出 **guest vlan**。如果配置了 **guest vlan**，端口 **linkup** 的时候会再次

进行 **guest vlan** 转换条件检查。

4、 Trunk 口上启用 guest vlan 功能会导致该端口上其它 vlan 的用户无需通过 802.1X 认证也能访问网络，因此建议只在 Access 口上开启 guest vlan。

5、 guest vlan 不支持 private vlan，也就是说不能将 private vlan 配置为 dot1x 的 guest vlan。

38.2.26 代理服务器屏蔽和拨号屏蔽

用户自行架设代理服务器以及用户认证后再自行拨号上网，这是网络安全最大的两个隐患。锐捷网络设备提供代理服务器屏蔽和拨号屏蔽功能。

实现该功能设备端不需任何设置，只需在 RADIUS 服务器端配置相应的属性。由于 Radius 没有标准的属性来表示最大数据率，我们只能通过厂商自定义属性来传递授权信息。我们定义的通用格式见授权一节的说明。

代理服务器屏蔽功能定义的 Vendor Type 为 0x20，拨号屏蔽功能定义的 Vendor Type 为 0x21。

Attribute-Specific 字段为四个字节的厂商自定义属性，定义了对使用代理服务器上网和拨号上网行为所采取的动作．0x0000 表示正常连接，不进行检测屏蔽，0x0001 表示进行检测屏蔽。

若要屏蔽通过代理服务器上网，用户应填如下信息：

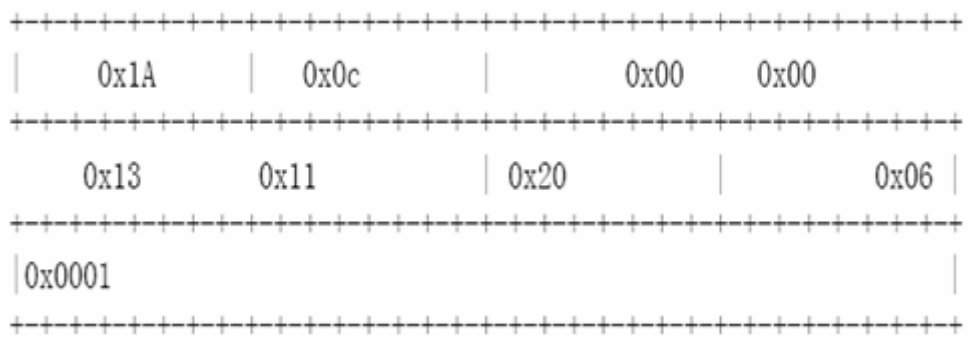


图 9

若要屏蔽通过拨号上网，用户应填如下信息：

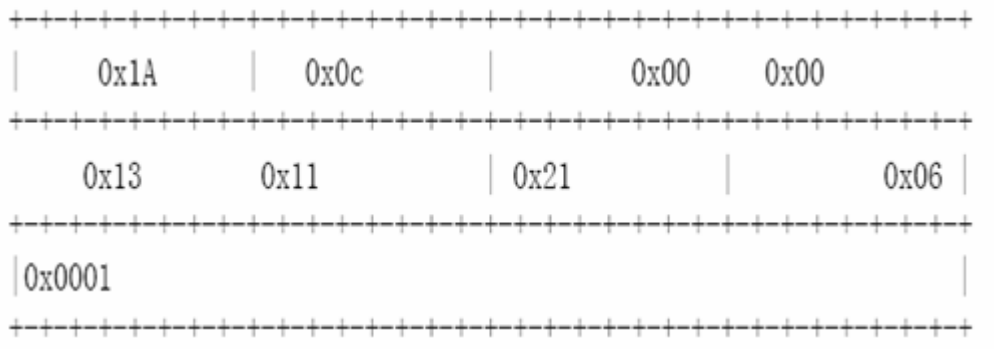


图 10

38.2.27 配置客户端在线探测

为了保证计费的准确性，需要一种在线探测机制能够在短时间内获知用户是否在线。在标准实现中的重认证机制能够满足这种需求，但是标准实现中需要 RADIUS 服务器的参与，要实现准确的探测用户是否在线将会占用设备和 RADIUS 服务器的大量资源。为了满足在占用少量资源的基础上实现计费的准确性，我们采用了一种新的客户端在线探测机制。这种机制只需要在设备和客户端之间交互，并且对网络流量的占用极小，能够实现分钟级的计费精度(用户可以通过配置来确定计费的精度)。

注意:

实现客户端在线监测需要客户端软件支持这一功能。

以下两个定时器影响到在线探测的性能和精度：

- Hello Interval: 客户端定时发送通告的间隔。
- Alive Interval：客户端在线间隔。设备在此间隔期间没有收到客户端的通告，将主动断开同客户端连接，并通知计费服务器。该值必须大于 Hello Interval。

在特权模式下，我们可以按以下方式配置客户端在线探测功能:

命令	作用
configure terminal	进入全局配置模式
dot1x client-probe enable	打开客户端在线探测功能
dot1x probe-timer interval *interval*	配置 Hello Interval
dot1x probe-timer alive interval	配置设备的 Alive Interval
end	退出到特权模式

write	保存配置
show dot1x	显示配置

38.2.28 配置 EAPOL 帧带 TAG 的选项开关

IEEE 802.1x 约定 EAPOL 报文不能加 VLAN TAG，但基于可能的应用需求，我们提供了相应的选项开关，在打开相应开关时，能根据 Trunk Port 的相应输出规则输出 TAG。

典型的应用环境就是在汇聚层打开 802.1x 认证，参见“典型应用的拓扑结构”。

在特权模式下，我们可以按以下方式配置 EAPOL 帧带 TAG 的选项开关。

命令	作用
configure terminal	进入全局配置模式
dot1x eapol-tag	打开 EAPOL 帧带 TAG 的功能。缺省该功能是关闭的
end	退出到特权模式
write	保存配置
show dot1x	显示配置

您可以用 **no dot1x eapol-tag** 命令来关闭该功能。

38.2.29 配置基于端口的用户认证

802.1x 对用户的控制默认情况下是基于用户 MAC 进行控制的，只有通过认证的用户才能使用网络，而对于其他接在同一端口的用户无法使用网络，而基于端口的控制模式及表示当某一端口有一个用户认证通过时，此端口就变成已认证端口，所有接在此端口下的用户都能够正常的使用网络。

要配置端口的控制模式为基于端口的控制模式，从特权模式开始，按步骤进行以下设置。

命令	作用
configure terminal	进入全局配置模式
interface *interface-id*	进入接口模式
dot1x port-control auto	打开受控功能
dot1x port-control-mode {*mac-based*|*port-based*}	选择受控的模式

end	退出到特权模式
write	保存配置
show dot1x port-control	显示端口 802.1X 配置

您可以用 **no dot1x port-control-mode** 命令来恢复默认受控方式。

下面的例子显示了如何配置端口的认证模式

```
Ruijie#configure terminal
Ruijie(config)#interface fastEthernet 0/4
Ruijie(config-if)#dot1x port-control-mode port-base
```

注意:

基于端口的认证模式下，每个端口只能接一个认证用户。

对于基于端口的认证方式下，可以允许或禁止动态用户在多个认证端口之间迁移，默认情况是允许动态用户在不同端口之间迁移，如要禁止动态用户的迁移，从特权模式开始，按步骤进行以下设置。

命令	作用
configure terminal	进入全局配置模式
dot1x stationarity enable	禁止端口间的迁移
end	退到特权模式
write	保存配置

38.2.30 配置基于端口单用户认证

802.1x 对用户的控制默认情况下是基于用户 MAC 进行控制的，只有通过认证的用户才能使用网络，而对于其他接在同一端口的用户无法使用网络，基于端口的控制模式及表示当某一端口有一个用户认证通过时，此端口就变成已认证端口，所有接在此端口下的用户都能够正常的使用网络。

而基于端口单用户的认证是在基于端口的控制模式下，对认证用户的数量控制为单一用户。该端口下只允许单一用户认证通过，此端口就变成已认证端口，能够正常的使用网络，这时，如果发现端口有其它的用户存在，则把端口下的所有用户清除，重新认证。

要配置端口单用户的控制模式为基于端口的控制模式，从特权模式开始，按步骤进行以下设置。

命令	作用
configure terminal	进入全局配置模式
interface *interface-id*	进入接口模式
dot1x port-control auto	打开受控功能
dot1x port-control-mode port-based single-host	基于端口单用户的控制模式
end	退出到特权模式
write	保存配置
show dot1x port-control	显示端口 802.1X 配置
show running-config	显示所有配置

您可以用 **no dot1x port-control-mode** 命令来恢复默认受控方式。

下面的例子显示了如何配置端口的认证模式

```
Ruijie#configure terminal
Ruijie(config)#interface fastEthernet 0/4
Ruijie(config-if)#dot1x port-control-mode port-based single-host
```

注意:

基于端口的认证模式下，每个端口只能接一个认证用户

single-host 是基于端口单用户的 802.1x 访问控制，在 show dot1x port-control 上会显示端口为 port-based，在 show running-config 会显示为 dot1x port-control-mode port-based single-host。

single-host 由于只支持单一用户形式，手动配置端口 default-user-limit，在 single-host 模式无效。如果在配置 single-host 后，端口下配置 default-user-limit，这个数量对 single-host 没有约束，该端口还是只允许单一用户上网。

对于基于端口的认证方式下，可以允许或禁止动态用户在多个认证端口之间迁移，默认情况是允许动态用户在不同端口之间迁移，如要禁止动态用户的迁移，从特权模式开始，按步骤进行以下设置。

命令	作用
configure terminal	进入全局配置模式
dot1x stationarity enable	禁止端口间的迁移
end	退到特权模式

Write	保存配置

38.2.31 配置动态 acl 下发

802.1x 支持从服务器下发 acl，动态安装下发的 acl。我司产品默认支持安装 acl，只要服务器上配置为许可 acl，并能在用户认证成功后下发 acl，我司产品都会动态安装下去。

实现动态 acl 下发，需要把端口配置成为 mac-based 认证模式，或者配置成基于端口单用户认证模式，基于端口的多用户认证模式不支持 ACL 下发，这些模式的配置请参考相关命令配置说明。

注意：

在 single-host 认证模式下，在重认证的时候支持更新 acl。即用户已经认证后，在服务器上配置了 acl，用户重新认证的时候这些 acl 生效。

在 mac-based 认证模式下，不支持重认证的时候更新 ACL,即认证用户的 acl 只能下发一次,如果在重认证时更换了 acl,交换机将会忽略新的 acl,维持原有的 acl。

支持的 acl 类型：在我司交换机上 acl 功能能够解析的扩展类型。

如果需要支持非我司认证服务器动态 acl 下发，需要在配置如下命令：

```
Ruijie#configure terminal
Ruijie(config)# radius vendor-specific extend
```

38.2.32 配置 MAC 旁路认证

GUEST VLAN 提供了一种支持无 802.1X 认证客户端设备接入网络的方法，但是该技术无法判断接入设备是否安全，存在安全隐患。在某些情况下，出于网络管理和安全考虑，即便无 802.1X 认证客户端，网络管理员仍然需要控制这些接入设备的合法性。MAC 旁路认证(MAC Authentication Bypass,简称 MAB)为这种应用提供了一种解决方案。

在 802.1X 认证端口部署了 MAB 功能后，802.1x 会向该端口持续发送认证请求报文并期望得到客户端响应。如果在 tx-period*reauth-max 时间内并无客户端响应，则 802.1x 会监听该认证口下学习到的 MAC 地址，并且以该 MAC 地址为用户名和密码向认证服务器发起认证，通过服务器返回的认证结果判断该 MAC 地址是否允许访问网络。

通过如下的步骤配置 MAB 功能：

	命令	作用
Step 1	**configure terminal**	进入全局配置模式

Step 2	**interface** <*interface-id*>	进入接口模式
Step 3	**dot1x mac-auth-bypass**	配置
Step 4	**end**	退出到特权模式
Step 5	**Write**	保存配置
Step6	**show running-config**	显示所有配置

下面的例子显示了如何配置 MAB 功能：

```
Ruijie# configure terminal
Ruijie(config)# interface fa 0/1
Ruijie(config-if)# dot1x port-control auto
Ruijie(config-if)# dot1x mac-auth-bypass
```

说明

- 服务器配置 MAC 账号的用户名和密码时，均需要使用 XXXXXXXXXXXX 的格式。
- 端口进入 MAB 模式后，只能有一个 MAC 地址做认证，其它 MAC 地址将不会进行认证，这要取决于设备先发现哪个 MAC 地址。
- 端口模式和 MAC 模式均只能支持一个端口一个 MAC 地址认证。
- 任何时刻端口下有客户端响应 802.1x 认证，端口的 MAB 将失效，除非端口经历 down/up 的 link 变化或者重新开关端口的 1x 功能。
- 客户端在线探测功能对 MAB 模式下的认证 MAC 不生效。
- 配置了 MAB 的端口，每隔 tx-period 发出一个认证请求报文，发送 reauth-max 次之后，如果没有客户端响应，则该端口进入 MAB 模式。进入 MAB 模式的端口可以学习 mac 地址,并使用这些 MAC 地址为账号进行认证。
- MAB 支持 PAP、CHAP、EAP-MD5 三种认证方法。如何配置认证方法请参考"配置认证方式"章节。
- MAB 模式下的 mac 地址认证失败后，如果有配置 guest vlan，则该认证口将进入 guest vlan。如果没有配置 guest vlan，则端口维持在原有 vlan。MAB 不支持失败 VLAN,也就是说,即便 MAB 认证失败,且用户配置了失败 vlan，端口也不会进入失败 vlan。
- MAB 支持动态 vlan 下发、ACL 下发等服务器部署功能。
- 如果某个 MAC 地址在一个端口通过 MAB 认证后，又在其它端口出现，则后出现的那个端口将被设置为违例。
- MAB 不支持和安全通道共用。
- 静态地址、过滤地址无法进行 MAB 认证。
- MAB 认证是为无认证客户端软件的设备提供接入认证服务的，这些设备通常都无法识别 802.1Q 的 TAG 标签，因此 MAB 认证功能最好部署在 ACCESS 口上，否则即便认证通过，设备同样有可能无法通信。

38.2.33 配置 MAC 旁路认证超时时间

MAB 模式下的 MAC 地址认证上线后，除非重认证失败、端口 down 或者因为管理策略原因下线，比如管理员强制下线等，否则交换机将认为该 MAC 地址一直是可以在线的。

用户可以配置许可这些认证地址的在线时间，默认是 0，表示允许一直在线。

通过如下的步骤配置 MAB 的超时时间：

	命令	作用
Step 1	**configure terminal**	进入全局配置模式
Step 2	**nterface** <*interface-id*>	进入接口模式
Step 3	**dot1x mac-auth-bypass timeout-activity <value>**	配置 MAB 超时时间，单位秒，无默认值,参数的配置范围 1-65535
Step 4	**end**	退出到特权模式
Step 5	**Write**	保存配置
Step 6	**show running-config**	显示所有配置

下面的例子显示了如何配置 MAB 超时功能：

```
Ruijie# configure terminal
Ruijie(config)# interface fa 0/1
Ruijie(config)# dot1x mac-auth-bypass timeout-activity  3600
```

说明

- 如果服务器也下发了认证 mac 地址的在线时间，则这个时间和 timeout-activity 是独立生效的，也就是说，哪个时间先到，就哪个先生效。
- 超时时间到达后，如果端口有配置 guest vlan，则端口将切换到 guest vlan。但是认证过程中的服务器响应超时不会导致 MAB 端口进入 guest vlan。

38.2.34 配置 MAC 旁路认证违例

默认情况下，有一个 MAC 地址通过 MAB 认证后，该端口下的所有设备的数据都允许被转发。但是在某些安全应用下，管理员会要求一个 MAB 端口下只能有一个 MAC 地址存在，此时可以在该端口上配置 MAB 违例。配置了 MAB 违例后，一旦端口进入了 MAB 模式，如果发现该端口下有超过 1 个 MAC 地址，该端口将产生违例。

通过如下的步骤配置端口的 MAB 违例：

	命令	作用
Step 1	**configure terminal**	进入全局配置模式
Step 2	**nterface** <*interface-id*>	进入接口模式
Step 3	**dot1x mac-auth-bypass violation**	配置 MAB 违例
Step 4	**end**	退出到特权模式
Step 5	**Write**	保存配置
Step 6	**show running-config**	显示所有配置

下面的例子显示了如何配置 MAB 违例：

```
Ruijie# configure terminal
Ruijie(config)# interface fa 0/1
Ruijie(config)# dot1x mac-auth-bypass violation
```

说明

- MAB 违例端口可以通过 errdisable recover 恢复。
- 由于 private vlan 的端口上同个 MAC 地址会同时出现在主从 VLAN 里面,因此不要在 private vlan 端口上配置 MAB 认证违例，否则会导致 MAB 违例判断错误，影响正常的使用。

38.2.35 配置失败 VLAN

如果在交换机上，配置了失败 vlan，当该端口的用户认证失败后，该端口将进入预先配置好的失败 vlan。

失败 vlan 基于接口，您可以通过以下步骤配置：

	命令	作用
Step 1	**configure terminal**	进入全局配置模式
Step 2	**interface** <*interface-id*>	进入接口模式
Step 3	**dot1x auth-fail vlan** < *vid*>	配置该端口的失败 vlan
Step 4	**end**	退出到特权模式
Step 5	**write**	保存配置
Step 6	**show run**	显示配置

下面的例子显示了如何配置失败 vlan：

```
Ruijie# configure terminal
Ruijie(config)# interface fa  0/1
Ruijie(config-if)# dot1x auth-fail vlan 2
```

说明

- 如果配置的 vlan 不存在，则在端口进入失败 vlan 时会被动态创建，端口退出失败 vlan 时自动删除。
- 如果端口 down，则端口自动退出失败 vlan。
- 允许失败 vlan 和 guest vlan 配置为同个 vlan。
- 端口模式下，进入 FAIL-VLAN 后，只允许之前认证失败的用户重新进行认证，其它用户的认证请求被丢弃,MAC 模式则无此限制。
- 失败 vlan 不支持 private vlan, 也就是说不能将 private vlan 配置为 dot1x 的失败 vlan。。

38.2.36 配置失败 VLAN 尝试次数

失败 VLAN 是在客户端尝试认证连续失败一定次数后才会进入的。您可以通过以

下步骤配置这个次数：

	命令	作用
Step 1	**configure terminal**	进入全局配置模式
Step 2	**dot1x auth-fail max-attemp<value>**	配置进入失败 vlan 的尝试次数，默认为 3 次,参数范围 1-3
Step 3	**end**	退出到特权模式
Step 4	**Write**	保存配置
Step5	**show running-config**	显示所有配置

下面的例子显示了如何配置失败 VLAN 尝试次数：

```
Ruijie# configure terminal
        Ruijie(config)# dot1x auth-fail max-attemp   2
```

38.3 查看 802.1x 的配置及当前的统计值

本公司实现的 802.1X 可以查看丰富的状态机信息，为网管提供了强有力的管理依据，便于管理员对用户状态的时时监控，且方便解决故障。

- 查看 Radius 认证及记帐相关配置
- 查看当前的用户数
- 查看可认证地址列表
- 查看用户认证状态信息
- 查看 1X 客户端探测定时器设置

38.3.1 查看 Radius 认证及记帐相关配置

用 **show radius server** 命令查看 Radius Server 的相关配置，用 **show aaa user** 查看用户相关信息

```
Ruijie# sh radius server
Server IP:       192.168.5.11
Accounting Port:  1813
Authen  Port:    1812
Server State:     Ready
```

38.3.2 查看当前的用户数

本公司实现的 802.1X 可以查看两类当前的用户数，一是当前用户数，二是已认证用户数。当前用户数指的是当前认证用户总数（无论是否认证成功）；已认证用户数，指的是已认证通过的用户的总数。

在特权模式下，查看当前用户数及已认证用户数，查看 1x 配置，包括当前用户数和已认证用户数使用 **show dot1x**。

以下例子是查看 802.1x 配置：

```
Ruijie# show dot1x
802.1X Status:       Disabled
Authentication Mode:  EAP-MD5
Authed User Number:   0
Re-authen Enabled:    Disabled
Re-authen Period:     3600 sec
Quiet Timer Period:   10 sec
Tx Timer Period:      3 sec
Supplicant Timeout:   3 sec
Server Timeout:       5 sec
Re-authen Max:        3 times
Maximum Request:      3 times
Filter Non-RG Supp:   Disabled
Client Oline Probe:   Disabled
Eapol Tag Enable:     Disabled
Authorization Mode:   Disabled
```

38.3.3 查看可认证地址列表

本公司实现的 802.1x，对功能进行了扩展，可以设置在某些端口上只有哪些主机可以认证。查看可认证主机列表功能，可以让管理员查看目前已有的设置。

在特权模式下，按如下操作查看可认证主机列表

命令	作用
configure terminal	进入全局配置模式
dot1x auth-address-table address *mac-addr* **interface** *interface*	配置可认证主机列表
end	退出到特权模式
write	保存配置
show dot1x auth-address-table	查看可认证主机列表

使用 **no dot1x auth-address-table address** 命令删除指定的可认证主机列表。以下例子是查看可认证主机列表：

```
Ruijie# show dot1x auth-address-table
interface:g3/1
-----------------------------------
mac addr: 00D0.F800.0001
```

38.3.4 查看用户认证状态信息

管理员可以查看本设备的当前用户的认证状态，便于排解故障。

在特权模式下，按如下操作查看用户认证状态信息

命令	作用
show dot1x summary	查看用户认证状态信息

以下例子是查看用户认证状态信息：

```
Ruijie# show dot1x summary
ID    MAC          Interface   VLAN   Auth-State   Backend-State
Port-Status
---- ------------ --------- ----  ----------  ------------
1     00d0f8000001  Gi3/1          1     Authenticated    IDLE
Authed
```

38.3.5 查看 1x 客户端探测定时器设置

在特权模式下，按如下操作查看 1x 定时器设置

命令	作用
show dot1x probe-timer	查看 1x 定时器设置

以下例子是查看 1x 定时器设置:

```
Ruijie# show dot1x probe-timer
Hello Interval: 20 Seconds
Hello Alive: 250 Seconds
Ruijie#
```

38.3.6 配置 802.1x 其它注意事项

1、 1X 和 ACL 同时应用

在非 IP 授权模式下，如果您打开某一个端口的 802.1x 认证功能，又将某一个 ACL 关联到某一个接口。则 ACL 在 MAC 地址的基础上生效。也就是说，只有源 MAC 为认证通过的用户 MAC 的报文才会经过 ACL 过滤，其它源 MAC 地址报文被丢弃，ACL 只能在该 MAC 地址的基础上进行限制。

比如，认证通过的 MAC 地址为 00d0.f800.0001，则所有源 MAC 为 00d0.f800.0001 的报文可以交换。如果该端口再关联 ACL，则 ACL 在这些能够交换的报文基础上将进一步过滤。如拒绝源 MAC 为该地址的 ICMP 报文。

2、 对认证端口上有用户进行认证或有已经认证成功用户情况下的限制说明：

- 端口模式不能修改，例如使用命令 switchport mode trunk；
- 端口在 ACCESS 模式下不能修改端口 Access VLAN；
- 端口在 TRUNK 模式下不能修改端口 Allowed VLAN 及 Native VLAN。

3、 在 VLAN 中有用户正在进行认证或有已经认证成功用户情况下的限制说明：

- VLAN 不能删除；
- VLAN 的类型不能修改，例如使用命令 private-vlan primary。

4、 对同一认证端口下多个用户认证情况的限制说明：

- 第一个用户没有下发 VLAN，使用端口缺省 VLAN 作为第一个用户的下发 VLAN；
- 后续认证用户没有下发 VLAN，使用第一个用户下发 VLAN；
- 后续认证用户下发的 VLAN 必须同第一个用户一致，否则无法通过认证；
- 第 一个用户重认证后下发 VLAN 必须与该用户上次认证通过时下发 VLAN 相同，否则无法通过认证。

5、 GVRP 不能和动态 VLAN 自动跳转功能同时使用。

6、 由于 VLAN 跳转功能是 802.1X 认证通过后将整个端口切换到另外一个 VLAN 进行通信，因此该功能最适宜的部署场景是 ACCESS 口下连单用户,如果在 TRUNK 口上部署该功能，虽然允许配置，但是实际认证会失败，因此请不要在 TRUNK 口上部署 VLAN 跳转功能。

802.1x 可以和其他接入控制功能比如端口安全，IP+MAC 绑定等共用。当这些接入控制功能共用时，报文必须同时满足所有的接入控制才可以进入交换机。

39 AAA 配置

访问控制是用来控制哪些人可以访问网络服务器以及用户在网络上可以访问哪些服务的。身份认证、授权和记帐（AAA）是进行访问控制的一种主要的安全机制。

39.1 AAA 基本原理

AAA 是 Authentication Authorization and Accounting（认证、授权和记账）的简称，它提供了对认证、授权和记账功能进行配置的一致性框架，锐捷网络设备产品支持使用 AAA。

AAA 以模块方式提供以下服务：

- 认证：验证用户是否可获得访问权，可选择使用 RADIUS 协议、TACACS+协议或 Local（本地）等。身份认证是在允许用户访问网络和网络服务之前对其身份进行识别的一种方法。
- 授权：授权用户可使用哪些服务。AAA 授权通过定义一系列的属性对来实现，这些属性对描述了用户被授权执行的操作。这些属性对可以存放在网络设备上，也可以远程存放在安全服务器上。
- 记帐：记录用户使用网络资源的情况。当 AAA 记帐被启用时，网络设备便开始以统计记录的方式向安全服务器发送用户使用网络资源的情况。每个记帐记录都是以属性对的方式组成，并存放在安全服务器上，这些记录可以通过专门软件进行读取分析，从而实现对用户使用网络资源的情况进行记帐、统计、跟踪。

说明：

部分产品的 AAA 仅提供认证功能。所有涉及产品规格的问题，可以通过向福建星网锐捷网络有限公司市场人员或技术支援人员咨询得到。

尽管 AAA 是最主要的访问控制方法，我司产品同时也提供了在 AAA 范围之外的简单控制访问，如本地用户名身份认证、线路密码身份认证等。不同之处在于它们提供对网络保护程度不一样，AAA 提供更高级别的安全保护。

使用 AAA 有以下优点：

- 灵活性和可控制性强
- 可扩充性
- 标准化认证
- 多个备用系统

39.1.1 AAA 基本原理

AAA 可以对单个用户（线路）或单个服务器动态配置身份认证、授权以及记帐类型。通过创建方法列表来定义身份认证、记帐、授权类型，然后将这些方法列表应用于特定的服务或接口。

39.1.2 方法列表

由于对用户进行认证、授权和记账可以使用不同的安全方法，您需要使用方法列表定义一个使用不同方法对用户进行认证、授权和记账的前后顺序。方法列表可以定义一个或多个安全协议，这样可以确保在第一个方法失败时，有备用系统可用。我司产品使用方法列表中列出的第一个方法时，如果该方法无应答，则选择方法列表中的下一个方法。这个过程一直持续下去，直到与列出的某种安全方法成功地实现通信或用完方法列表。如果用完方法列表而还没有成功实现通信，则该安全功能宣告失败。

注意:

只有在前一种方法没有应答的情况下，我司产品才会尝试下一种方法。例如在身份认证过程中，某种方法拒绝了用户访问，则身份认证过程结束，不再尝试其他的身份认证方法。

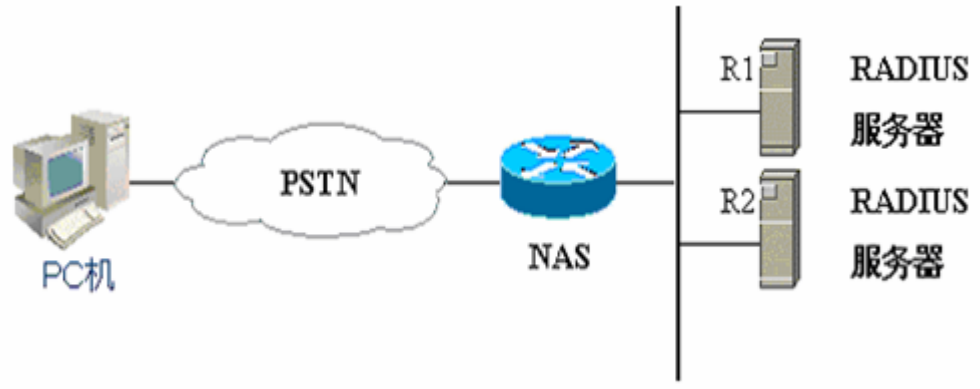


图 1 典型的 AAA 网络配置图

上图说明了一个典型的 AAA 网络配置，它包含两台安全服务器：R1 和 R2 是 RADIUS 服务器。

假设系统管理员已定义了一个方法列表，在这个列表中，R1 首先被用来获取身份信息，然后是 R2，最后是访问服务器上的本地用户名数据库。如果一个远程 PC 用户试图拨号进入网络，网络访问服务器首先向 R1 查询身份认证信息，假如用户通过了 R1 的身份认证，R1 将向网络访问服务器发出一个 ACCEPT 应答，这样用户即获准访问网络。如果 R1 返回的是 REJECT 应答，则拒绝用户访问网络，断

开连接。如果 R1 无应答，网络访问服务器就将它看作 TIMEOUT，并向 R2 查询身份认证信息。这个过程会一直在余下的指定方法中持续下去，直到用户通过身份认证、被拒绝或对话被中止。如果所有的方法返回 TIMEOUT，则认证失败，连接将被断开。

注意：

REJECT 应答不同于 TIMEOUT 应答。REJECT 意味着用户不符合可用身份认证数据库中包含的标准，从而未能通过身份认证，访问请求被拒绝。TIMEOUT 则意味着安全服务器对身份认证查询未作应答，当检测到一个 TIMEOUT 时，AAA 选择身份认证方法列表中定义的下一个身份认证方法将继续进行身份认证过程。

说明：

在本文中，与 AAA 安全服务器相关的认证、授权和记账配置，均以 RADIUS 为例，而与 TACACS+有关的内容请另外参考“配置 TACACS+”。

39.2 AAA 配置基本步骤

首先您必须决定要采用哪种安全解决方案，而且需要评估特定网络中的潜在安全风险，并选择适当的手段来阻止未经授权的访问。我们建议，在可能的情况下，尽量使用 AAA 确保网络安全。

39.2.1 AAA 配置过程概述

如果理解了 AAA 运作的基本过程，配置 AAA 就相对简单了。在锐捷网络设备上配置 AAA 地步骤如下：

1) 启用 AAA，使用全局配置层命令 **aaa new-model**。

2) 如果决定使用安全服务器，请配置安全协议的参数，如 RADIUS。

3) 定义身份认证方法列表，使用 **aaa authentication** 命令。

4) 如有需要，可将该方法列表应用于特定的接口或线路。

注意：

在应用特定方法列表时，如果没有明确指定使用命名的方法列表，则使用默认的身份认证方法列表进行身份认证。
因此，如果不准备使用默认的身份认证方法列表，则需要指定特定的方法列表。

对于本章中使用的命令的完整描述，请参见安全配置命令参考中的相关章节。

39.2.2 启用 AAA

要使用 AAA 安全特性，必须首先启用 AAA。

要启用 AAA，在全局配置模式下执行以下命令：

命令	作用
aaa new-model	启用 AAA

39.2.3 停用 AAA

要停用 AAA，在全局配置模式下执行以下命令：

命令	作用
no aaa new-model	停用 AAA

39.2.4 后续的配置过程

启用 AAA 以后，便可以配置与安全方案相关的其他部分，下表说明了可能要完成的配置任务以及相关内容所在的章节。

AAA 访问控制安全解决方案方法

配置任务	步骤	所在章节
配置 RADIUS 安全协议参数	2	配置 RADIUS
配置本地登录(Login)身份认证	3	配置认证
定义认证方法列表	4	配置认证
将方法列表应用于特定的接口或线路	5	配置认证
启用 RADIUS 授权	6	配置授权
启用 RADIUS 记账	7	配置记账

如果使用 AAA 实现身份认证，请参考“配置认证”中的相关部分。

39.3 配置认证

身份认证是在允许用户使用网络资源以前对其进行识别，在大多数情况下，身份认证是通过 AAA 安全特性来实现的。我们建议，在可能的情况下，最好使用 AAA 来实现身份认证。

39.3.1 定义 AAA 认证方法列表

要配置 AAA 身份认证，首先得定义一个身份认证方法的命名列表，然后各个应用使用已定义列表进行认证。方法列表定义了身份认证的类型和执行顺序。对于已定义的身份认证方法，必须有特定的应用才会被执行。默认方法列表是唯一的例外。所有应用在未进行配置时使用默认方法列表。

方法列表仅是定义将要被依次查询的、并用于认证用户身份的一系列安全方法。方法列表使您能够指定一个或多个用于身份认证的安全协议，这样确保在第一种方法失败的情况下，可以使用身份认证备份系统。我司产品使用第一种方法认证用户的身份，如果该方法无应答，将选择方法列表中的下一种方法。 这个过程一直持续下去，直到与列出的某种身份认证方法成功地实现通信或用完方法列表。如果用完方法列表而还没有成功实现通信，则身份认证宣告失败。

注意:

只有在前一种方法没有应答的情况下，我司产品才会尝试下一种方法。如果在身份认证过程中，某种方法拒绝了用户访问，则身份认证过程结束，不再尝试其他的身份认证方法。

39.3.2 方法列表举例

在典型 AAA 网络配置图中，它包含 2 个服务器：R1 和 R2 是 RADIUS 服务器。假设系统管理员已选定一个安全解决方案，NAS 认证采用一个身份认证方法对 Telnet 连接进行身份认证：首先使用 R1 对用户进行认证，如果无应答，则使用 R2 进行认证；如果 R1、R2 都没有应答，则身份认证由访问服务器的本地数据库完成，要配置以上身份认证列表，执行以下命令：

命令	作用
configure terminal	进入全局配置模式
aaa authentication login default group radius local	配置一个默认身份认证方法列表，其中“default”是方法列表的名称。这个方法列表中包含的协议，按照它们将被查询的顺序排列在名称之后。为所有应用的默认方法列表。

如果系统管理员希望该方法列表仅应用于一个特定的 Login 连接，必须创建一个命名方法列表，然后将它应用于特定的连接。下面的例子说明了如何将身份认证方法列表仅应用于线路 2：

命令	作用

configure terminal	进入全局配置模式
aaa new-model	打开 AAA 开关
aaa authentication login *test* **group radius local**	在全局配置模式下，定义了一个名为“test”的方法列表。
line vty *2*	进入线路 2 配置层
login authentication *test*	在 line 配置模式下，将名为“test”方法列表应用于线路。

当远程 PC 用户试图 Telnet 访问网络设备(NAS)，NAS 首先向 R1 查询身份认证信息，假如用户通过了 R1 的身份认证，R1 将向 NAS 发出一个 ACCEPT 应答，这样用户即获准访问网络。如果 R1 返回的是 REJECT 应答，则拒绝用户访问网络，断开连接。如果 R1 无应答，NAS 就将它看作 TIMEOUT，并向 R2 查询身份认证信息。这个过程会一直在余下的指定方法中持续下去，直到用户通过身份认证、被拒绝或对话被中止。如果所有的服务器(R1、R2)返回 TIMEOUT，则认证由 NAS 本地数据库完成。

注意:

REJECT 应答不同于 TIMEOUT 应答。REJECT 意味着用户不符合可用的身份认证数据库中包含的标准，从而未能通过身份认证，访问请求被拒绝。TIMEOUT 则意味着安全服务器对身份认证查询未作应答，当验证 TIMEOUT 时，AAA 选择认证方法列表中定义的下一个认证方法继续进行认证过程。

39.3.3 认证类型

我司产品目前支持以下认证类型：

- Login（登录）认证
- Enable 认证
- PPP 认证
- DOT1X（IEEE802.1x）认证

其中 Login 认证针对的是用户终端登录到 NAS 上的命令行界面（CLI），在登录时进行身份认证；Enable 认证针对的是用户终端登录到 NAS 上的 CLI 界面以后，提升 CLI 执行权限时进行认证；PPP 认证针对 PPP 拨号用户进行身份认证；DOT1X 认证针对 IEEE802.1x 接入用户进行身份认证。

39.3.4 配置 AAA 身份认证的通用步骤

要配置 AAA 身份认证，都必须执行以下任务：

- 使用 **aaa new-model** 全局配置命令启用 AAA。
- 如果要使用安全服务器，必须配置安全协议参数，如 RADIUS 和 TACACS+。具体的配置请参见“配置 RADIUS”和“配置 TACACS+”。
- 使用 **aaa authentication** 命令定义身份认证方法列表。
- 如果可能，将方法列表应用于某个特定的接口或线路。

注意：

我司产品 DOT1X 认证目前不支持 TACACS+。

39.3.5 配置 AAA Login 认证

本节将具体介绍如何配置我司产品所支持的 AAA Login（登录）身份认证方法：

注意：

只有在全局配置模式下执行 **aaa new-model** 命令启用 AAA，AAA 安全特性才能进行配置使用（下同）。关于更详细的内容，请参见“AAA 概述”。

在很多情况下，用户需要通过 Telnet 访问网络访问服务器(NAS)，一旦建立了这种连接，就可以远程配置 NAS，为了防止网络未经授权的访问，要对用户进行身份认证。

AAA 安全服务使网络设备对各种基于线路的 Login（登录）身份认证变得容易。不论您要决定使用哪种 Login 认证方法，只要使用 **aaa authentication login** 命令定义一个或多个身份认证方法列表，并应用于您需要进行 Login 认证的特定线路就可以了。

要配置 AAA Login 认证，在全局配置模式下执行以下命令：

命令	作用
configure	terminal
aaa new-model	启用 AAA。
aaa authentication login {**default** | *list-name*} *method1* [*method2*...]	定义一个认证方法列表，如果需要定义多个方法列表，重复执行该命令。

line vty *line-num*	进入您需要应用 AAA 身份认证的线路。
login authentication {**default** | *list-name*}	将方法列表应用线路。

关键字 *list-name* 用来命名创建身份认证方法列表，可以是任何字符串；关键字 *method* 指的是认证实际算法。仅当前面的方法返回 ERROR（无应答），才使用后面的其他认证方法；如果前面的方法返回 FAIL(失败)，则不使用其他认证方法。为了使认证最后能成功返回，即使所有指定方法都没有应答，可以在命令行中将 **none** 指定为最后一个认证方法。

例如，在下例中，即使 RADIUS 服务器超时(TIMEOUT)，仍然能够通过身份认证：**aaa authentication login default group radius none**

注意：

由于关键字 **none** 使得拨号的任何用户在安全服务器没有应答情况下都能通过身份认证，所以仅将它作为备用的身份认证方法。我们建议：一般情况下，不要使用 **none** 身份认证，在特殊情况下，如所有可能的拨号用户都是可信任的，而且用户的工作不允许有由于系统故障造成的耽搁，可以在安全服务器无应答的情况下，将 **none** 作为最后一种可选的身份认证方法，建议在 **none** 认证方法前加上本地身份认证方法。

关键字	描述
local	使用本地用户名数据库进行身份认证
none	不进行身份认证
group radius	使用 RADIUS 服务器组进行身份认证
group tacacs+	使用 TACACS+服务器组进行身份认证

上表列出了我司产品支持的 AAA Login 认证方法。

39.3.5.1 使用本地数据库进行 Login 认证

要配置使用本地数据库进行 Login 认证时，首先需要配置本地数据库，我司产品支持基于本地数据库的身份认证，建立用户名身份认证，请在全局配置模式下，根据具体需求执行以下命令：

命令	作用
configure terminal	进入全局配置模式

username *name* [**password** *password*]	建立本地用户，设置口令
end	退出到特权模式
show running-config	确认配置

定义本地 Login 认证方法列表并应用认证方法列表，可使用以下命令：

命令	作用
configure terminal	进入全局配置模式
aaa new-model	打开 AAA 开关
aaa authentication login {**default** | *list-name*} **local**	定义使用本地认证方法
end	退出到特权模式
show aaa method-list	确认配置的方法列表
configure terminal	进入全局配置模式
line vty *line-num*	进入线路配置模式
login authentication {**default** | *list-name*}	应用方法列表
end	退出到特权模式
show running-config	确认配置

39.3.5.2 使用 RADIUS 进行 Login 认证

要配置用 RADIUS 服务器进行 Login 认证，首先要配置 RADIUS 服务器。配置 RADIUS 服务器，请在全局配置模式下，根据具体需求执行以下命令：

命令	作用
configure terminal	进入全局配置模式
aaa new-model	打开 AAA 开关
radius-server host *ip-address* [**auth-port** *port*] [**acct-port** *port*]	配置 RADIUS 服务器
end	退出到特权模式
show radius server	显示 RADIUS 服务器

配置好 RADIUS 服务器后，在配置 RADIUS 进行身份认证以前，请确保与 RADIUS 安全服务器之间已经成功进行了通信，有关 RADIUS 服务器配置的信息，请参见“配置 RADIUS”。

现在就可以配置基于 RADIUS 服务器的方法列表了，在全局配置模式下执行以下命令：

命令	作用
configure terminal	进入全局配置模式
aaa new-model	打开 AAA 开关
aaa authentication login {**default** | *list-name*} **group radius**	定义使用 RADIUS 认证方法
end	退出到特权模式
show aaa method-list	确认配置的方法列表
configure terminal	进入全局配置模式
line vty *line-num*	进入线路配置模式
login authentication {**default** | *list-name*}	应用方法列表
end	退出到特权模式
show running-config	确认配置

39.3.6 配置 AAA Enable 认证

本节将具体介绍如何配置我司产品所支持的 AAA Enable 认证方法：

在很多情况下，用户需要通过 Telnet 等方法访问网络访问服务器(NAS)，在进行了身份认证以后，就可以进入命令行界面（CLI），此时会被赋予一个初始的执行 CLI 命令的权限（0~15 级）。不同的级别，可以执行的命令是不同的，可以使用 **show privilege** 命令查看当前的级别。关于更详细的内容，请参见“使用命令行界面”。

如果登录到命令行界面后，由于初始权限过低，不能执行某些命令，则可以使用 enable 命令来提升权限。为了防止网络未经授权的访问，在提升权限的时候，需要进行身份认证，即 Enable 认证。

要配置 AAA Enable 认证，在全局配置模式下执行以下命令：

命令	作用
configure terminal	进入全局配置模式
aaa new-model	启用 AAA。
aaa authentication enable default *method1* [*method2*...]	定义一个 Enable 认证方法列表，可以指定采用的认证方法，例如 RADIUS。

Enable 认证方法列表全局只能定义一个，因此不需要定义方法列表的名称；关键字 *method* 指的是认证实际方法。仅当前面的方法返回 ERROR（无应答），才使用后面的其他身份方法；如果前面的方法返回 FAIL(失败)，则不使用其他认证方法。为了使认证最后能成功返回，即使所有指定方法都没有应答，可以在命令行中将 **none** 指定为最后一个认证方法。

Enable 认证方法配置以后立即自动生效。此后，在特权模式下执行 **enable** 命令的时候，如果要切换的级别比当前级别要高，则会提示进行认证。如果要切换的级别小于或等于当前级别，则直接切换，不需要进行认证。

注意：

如果进入 CLI 界面的时候经过了 Login 身份认证（**none** 方法除外），将记录当前使用的用户名。此时，进行 Enable 认证的时候，将不再提示输入用户名，直接使用与 Login 认证相同的用户名进行认证，注意输入的口令要与之匹配。

如果进入 CLI 界面的时候没有进行 Login 认证，或在 Login 认证的时候使用了 **none** 方法，将不会记录用户名信息。此时，如果进行 Enable 认证，将会要求重新输入用户名。这个用户名信息不会被记录，每次进行 Enable 认证都要重新输入。

一些认证方法，在认证时可以绑定安全级别。这样，在认证过程中，除了根据安全协议返回的成功或失败的应答外，还需要检查绑定的安全级别。如果服务协议能绑定安全级别，则需要在认证时校验绑定的级别。如果绑定的级别大于或等于要切换的目的级别，则 Enable 认证成功，并切换到目的级别；而如果绑定的级别小于要切换的目的级别，则 Enable 认证失败，提示失败信息，维持当前的级别不变。如果服务协议不能绑定安全级别，则不校验绑定的级别，就可以切换到目的级别。

注意：

目前能够绑定安全级别的认证方法只有 RADIUS 和本地认证，因此只对这两种方法进行检查，如果采用其他认证方法则不进行检查。

39.3.6.1 使用本地数据库进行 Enable 认证

使用本地数据库进行 Enable 认证时，可以在设置本地用户时，为用户设置权限级别。如果没有设置，则默认的用户级别为 1 级。要配置使用本地数据库进行 Enable 身份认证时，首先需要配置本地数据库，并为用户设置权限级别，请在全局配置模式下，根据具体需求执行以下命令：

命令	作用
configure terminal	进入全局配置模式
username *name* [**password** *password*]	建立本地用户，设置口令
username *name* [**privilege** *level*]	为用户设置权限级别（可选）

end	退出到特权模式
show running-config	确认配置

定义本地 Enable 认证方法列表，可使用以下命令：

命令	作用
configure terminal	进入全局配置模式
aaa new-model	打开 AAA 开关
aaa authentication enable default local	定义使用本地认证方法
end	退出到特权模式
show aaa method-list	确认配置的方法列表
show running-config	确认配置

39.3.6.2 使用 RADIUS 进行 Enable 认证

标准的 RADIUS 服务器可以通过 Service-Type 属性（标准属性号为 6）绑定权限，可以指定 1 级或 15 级权限；我司扩展的 RADIUS 服务器（例如 SAM）可以设置设备管理员的级别（私有属性号为 42），可以指定 0~15 级权限。有关 RADIUS 服务器配置的信息，请参见“配置 RADIUS”中的“指定 RADIUS 私有属性类型”章节。

要配置用 RADIUS 认证服务器进行 Enable 认证，首先要配置 RADIUS 服务器，然后再配置基于 RADIUS 服务器的 Enable 方法列表。在全局配置模式下执行以下命令：

命令	作用
configure terminal	进入全局配置模式
aaa new-model	打开 AAA 开关
aaa authentication enable default group radius	定义使用 RADIUS 认证方法
end	退出到特权模式
show aaa method-list	确认配置的方法列表
show running-config	确认配置

39.3.7 配置 PPP 用户使用 AAA 认证

PPP 协议是提供在点到点链路上承载网络层数据包的一种链路层协议。在很多情况下，用户需要通过异步或 ISDN 拨号访问 NAS（网络访问服务器），一旦建立了这种连接，将启动 PPP 协商，为了防止网络未经授权的访问，PPP 在协商过程中要对拨号用户进行身份认证。

本节将具体介绍如何配置我司产品所支持的 AAA PPP 认证方法，要配置 AAA PPP 认证，在全局配置模式下执行以下命令：

命令	作用
configure terminal	进入全局配置模式
aaa new-model	启用 AAA。
aaa authentication ppp {**default** | *list-name*} *method1* [*method2*...]	定义一个 PPP 认证方法列表，可以指定采用的认证方法，目前支持 RADIUS 和 TACACS+远程认证，以及使用本地数据库进行认证。
interface *interface-type interface-number*	进入需要应用 AAA 身份认证的异步或 ISDN 接口。
ppp authentication {**chap** | **pap**} {**default** | *list-name*}	将身份认证方法列表应用于异步或 ISDN 接口。

PPP 协议更具体的配置方式参见“PPP、MP 协议配置”中相关章节。

39.3.8 配置 802.1x 用户使用 AAA 认证

IEEE802.1x（Port-Based Network Access Control）是一个基于端口的网络存取控制标准，为 LAN 提供点对点式的安全接入，提供一种对连接到局域网设备的用户进行认证的手段。

本节将具体介绍如何配置我司产品所支持的 802.1x 认证方法，要配置 802.1x 认证，在全局配置模式下执行以下命令：

命令	作用
configure terminal	进入全局配置模式
aaa new-model	启用 AAA。
aaa authentication dot1x {**default** | *list-name*} *method1* [*method2*...]	定义一个 IEEE802.1x 认证方法列表，可以指定采用的认证方法，目前支持 RADIUS 远程认

	证，以及使用本地数据库进行认证。
dot1x authentication *list-name*	802.1x 应用认证方法列表。

IEEE802.1x 协议更具体的配置方式参见“802.1x 配置”中相关章节。

39.3.9 配置认证示例

下面示例演示如何配置网络设备，使用“RADIUS+本地身份”进行认证。

```
Ruijie(config)# aaa new-model
Ruijie(config)# username Ruijie password starnet
Ruijie(config)# radius-server host 192.168.217.64
Ruijie(config)# radius-server key test
Ruijie(config)# aaa authentication login test group radius
local
Ruijie(config)# line vty 0
Ruijie(config-line)# login authentication test
Ruijie(config-line)# end
Ruijie# show running-config
!
aaa new-model
!
!
aaa authentication login test group radius local
username Ruijie password 0 starnet
!
radius-server host 192.168.217.64
radius-server key 7 093b100133
!
line con 0
line vty 0
login authentication test
line vty 1 4
!
!
```

上面例子中，访问服务器使用 RADIUS 服务器（IP 为 192.168.217.64）对登录的用户进行认证，如果 RADIUS 服务器没有应答，则使用本地数据库进行身份认证。

39.3.10 终端服务应用下认证示例

在终端服务的应用环境下，终端接到设备的异步串口上，再通过 IP 网络连接到网络中心服务器进行业务作业。但是，如果打开了 AAA 功能，则所有的线路都需要进行登录（Login）认证，那么终端需要先通过了设备的登录认证，才能连接到服务器，这样对终端服务的业务产生了影响。我们可以通过配置将两种线路分开，既让使用终端服务的线路不进行登录认证，直接连接服务器；同时连接到本设备的线路又可以使用登录认证来保证设备的安全。即：设置一个终端服务专用的登录认证列表，但认证方法为 none；然后将这个登录认证列表应用在打开终端服务的线路上（其他可以连接到本地的线路不变）；这样该终端就不需要进行本地的登录认证了。配置步骤如下：

```
Ruijie(config)# aaa new-model
Ruijie(config)# username Ruijie password starnet
Ruijie(config)# radius-server host 192.168.217.64
Ruijie(config)# radius-server key test
Ruijie(config)# aaa authentication login test group radius
local
Ruijie(config)# aaa authentication login terms none
Ruijie(config)# line tty 1 4
Ruijie(config-line)# login authentication terms
Ruijie(config-line)# exit
Ruijie(config)# line tty 5 16
Ruijie(config-line)# login authentication test
Ruijie(config-line)# exit
Ruijie(config)# line vty 0 4
Ruijie(config-line)# login authentication test
Ruijie(config-line)# end
Ruijie# show running-config
!
aaa new-model
!
!
aaa authentication login test group radius local
aaa authentication login terms none
username Ruijie password 0 starnet
!
radius-server host 192.168.217.64
radius-server key 7 093b100133
!
line con 0
line aux 0
line tty 1 4
login authentication terms
line tty 5 16
```

```
login authentication test
line vty 0 4
login authentication test
!
!
```

上面例子中，访问服务器使用 RADIUS 服务器（IP 为 192.168.217.64）对登录的用户进行认证，如果 RADIUS 服务器没有应答，则使用本地数据库进行身份认证。我们使用 tty 1-4 作为终端服务使用的线路，不需要登录认证，而其他 tty 和 vty 线路需要进行登录认证。

39.4 配置授权

AAA 授权使管理员能够对用户可使用的服务或权限进行控制。启用 AAA 授权服务以后，网络设备通过本地或服务器中的用户配置文件信息对用户的会话进行配置。完成授权以后，该用户只能使用配置文件中允许的服务或只具备许可的权限。

39.4.1 授权类型

我司产品目前支持以下 AAA 授权类型：

- Exec 授权
- Command（命令）授权
- Network（网络）授权

其中 Exec 授权针对的是用户终端登录到 NAS 上的 CLI 界面时，授予用户终端的权限级别（分为 0~15 级）；命令授权针对的是用户终端登录到 NAS 上的 CLI 界面以后，针对具体命令的执行授权；而网络授权针对的是授予网络连接上的用户会话可使用的服务。

说明：

命令授权功能目前仅 TACACS+协议支持，具体内容请参考“配置 TACACS+”。

39.4.2 授权的准备工作

在配置 AAA 授权以前，必须完成下述任务：

- 启用 AAA 服务。关于如何启用 AAA 服务，请参见“AAA 概述”。

- （可选）配置 AAA 认证。授权一般是在用户通过认证之后进行，但在某些情况下，没有经过认证也可以单独授权。关于 AAA 认证的信息，请参见“配置认证”。
- （可选）配置安全协议参数。如果需要使用安全协议进行授权，需要配置安全协议参数。我司产品 Network 授权仅支持 RADIUS 协议，Exec 授权支持 RADIUS 和 TACACS+协议，关于 RADIUS 协议的信息，请参见“配置 RADIUS”，关于 TACACS+协议的信息，请参见“配置 TACACS+”。
- （可选）如果需要使用本地授权，则需要使用 **username** 命令定义用户权限。

39.4.3 配置授权列表

要启用 AAA 授权，请在全局配置模式下执行以下命令：

命令	作用
configure terminal	进入全局配置模式
aaa new-model	打开 AAA 开关
aaa authorization exec {**default** | *list-name*} *method1* [*method2*…]	定义 AAA Exec 授权方法
aaa authorization network{**default** | *list-name*} *method1* [*method2*…]	定义 AAA Network 授权方法

39.4.4 配置 AAA Exec 授权

我司产品支持针对登录到网络访问服务器（NAS）的用户终端，授予其执行命令的权限，即 Exec 授权。体现为用户登录到 NAS 的 CLI 界面时（例如通过 Telnet），具备的级别（成功登录后，可以使用 **show privilege** 命令查看）。

不论您要决定使用哪种 Exec 授权方法，只要使用 **aaa authorization exec** 命令定义一个或多个 Exec 授权方法列表，并应用于您需要进行 Exec 授权的特定线路就可以了。

要配置 AAA Exec 授权，在全局配置模式下执行以下命令：

命令	作用
configure terminal	进入全局配置模式
aaa new-model	启用 AAA。
aaa authorization exec {**default** | *list-name*} *method1* [*method2*...]	定义一个授权方法列表，如果需要定义多个方法列表，重复执行该命令。
line vty *line-num*	进入您需要应用 AAA Exec 授权的线路。

authorization exec {**default** | *list-name*}	将方法列表应用线路。

关键字 *list-name* 用来命名创建授权方法列表，可以是任何字符串；关键字 *method* 指的是授权实际算法。仅当前面的方法返回 ERROR（无应答），才使用后面的其他方法；如果前面的方法返回 FAIL(失败)，则不使用其他授权方法。为了使授权最后能成功返回，即使所有指定方法都没有应答，可以在命令行中将 **none** 指定为最后一个授权方法。

例如，在下例中，即使 RADIUS 服务器超时(TIMEOUT)，仍然能够通过 Exec 授权：**aaa authorization exec default group radius none**

关键字	描述
local	使用本地用户名数据库进行 Exec 授权
none	不进行 Exec 授权
group radius	使用 RADIUS 服务器组进行 Exec 授权
group tacacs+	使用 TACACS+服务器组进行 Exec 授权

上表列出了我司产品支持的 AAA Exec 授权方法。

注意：

Exec 授权通常结合 Login 认证一起使用，并可以在同一个线路上同时使用 Login 认证和 Exec 授权。但是要注意，由于授权和认证可以采用不同的方法和不同的服务器，因此对于相同的用户，认证和授权可能有不同的结果。用户登录时，如果 Exec 授权失败，即使已经通过了 Login 认证，也不能进入到 CLI 界面。

39.4.4.1 使用本地数据库进行 Exec 授权

要配置使用本地数据库进行 Exec 授权时，首先需要配置本地数据库。可以在设置本地用户时，为用户设置权限级别。如果没有设置，则默认的用户级别为 1 级。请在全局配置模式下，根据具体需求执行以下命令：

命令	作用
configure terminal	进入全局配置模式
username *name* [**password** *password*]	建立本地用户，设置口令
username *name* [**privilege** *level*]	为用户设置权限级别（可选）

end	退出到特权模式
show running-config	确认配置

定义本地 Exec 授权方法列表并应用授权方法列表，可使用以下命令：

命令	作用
configure terminal	进入全局配置模式
aaa new-model	打开 AAA 开关
aaa authorization exec {**default** | *list-name*} **local**	定义使用本地认证方法
end	退出到特权模式
show aaa method-list	确认配置的方法列表
configure terminal	进入全局配置模式
line vty *line-num*	进入线路配置模式
authorization exec {**default** | *list-name*}	应用方法列表
end	退出到特权模式
show running-config	确认配置

39.4.4.2 使用 RADIUS 进行 Exec 授权

要配置用 RADIUS 服务器进行 Exec 授权，首先要配置 RADIUS 服务器，有关 RADIUS 服务器配置的信息，请参见“配置 RADIUS”。

配置好 RADIUS 服务器后，就可以配置基于 RADIUS 服务器的方法列表了，在全局配置模式下执行以下命令：

命令	作用
configure terminal	进入全局配置模式
aaa new-model	打开 AAA 开关
aaa authorization exec {**default** | *list-name*} **group radius**	定义使用 RADIUS 认证方法
end	退出到特权模式
show aaa method-list	确认配置的方法列表
configure terminal	进入全局配置模式
line vty *line-num*	进入线路配置模式
authorization exec {**default** | *list-name*}	应用方法列表
end	退出到特权模式

show running-config	确认配置

39.4.4.3 配置 Exec 授权示例

下例演示如何进行 Exec 授权。我们设置 VTY 线路 0~4 上的用户登录时采用 Login 认证，并且进行 Exec 授权。其中 Login 认证采用本地认证，Exec 授权先采用 RADIUS、如果没有响应可以采用本地授权。远程 RADIUS 服务器地址为 192.168.217.64，共享密钥为 test；本地用户名为 ruijie，口令为 ruijie，绑定级别是 6 级。如下：

```
Ruijie# configure terminal
Ruijie(config)# aaa new-model
Ruijie(config)# radius-server host 192.168.217.64
Ruijie(config)# radius-server key test
Ruijie(config)# username ruijie password ruijie
Ruijie(config)# username ruijie privilege 6
Ruijie(config)# aaa authentication login mlist1 local
Ruijie(config)# aaa authorization exec mlist2 group radius
local
Ruijie(config)# line vty 0 4
Ruijie(config-line)# login authentication mlist1
Ruijie(config-line)# authorization exec mlist2
Ruijie(config)# end
Ruijie# show running-config
aaa new-model
!
aaa authorization exec mlist2 group radius local
aaa authentication login mlist1 local
!
username ruijie password ruijie
username ruijie privilege 6
!
radius-server host 192.168.217.64
radius-server key 7 093b100133
!
line con 0
line vty 0 4
 authorization exec mlist2
 login authentication mlist1
!
end
```

39.4.5 配置 AAA Network 授权

我司产品支持对包括 PPP、SLIP 等网络连接进行 Network（网络）授权，这些网络连接通过 Network 授权，可以获得诸如流量、带宽、超时等服务配置。Network 授权仅支持通过 RADIUS 协议进行，服务器下发的授权信息封装在 RADIUS 属性里，针对不同的网络连接应用，服务器下发的授权信息可能不相同。

注意:

目前该配置不支持 802.1X 的 AAA 授权，802.1X 通过另外的命令完成，具体参见“802.1X 配置”。

要配置 AAA Network 授权，在全局配置模式下执行以下命令：

命令	作用
configure terminal	进入全局配置模式
aaa new-model	启用 AAA。
aaa authorization network {**default** | *list-name*} *method1* [*method2*...]	定义一个授权方法列表，如果需要定义多个方法列表，重复执行该命令。

关键字 *list-name* 用来命名创建授权方法列表，可以是任何字符串；关键字 *method* 指的是授权实际算法。仅当前面的方法返回 ERROR（无应答），才使用后面的其他授权方法；如果前面的方法返回 FAIL(失败)，则不使用其他授权方法。为了使授权最后能成功返回，即使所有指定方法都没有应答，可以在命令行中将 **none** 指定为最后一个授权方法。

39.4.5.1 使用 RADIUS 进行 Network 授权

要配置用 RADIUS 服务器进行 Network 授权，首先要配置 RADIUS 服务器，有关 RADIUS 服务器配置的信息，请参见“配置 RADIUS”。

配置好 RADIUS 服务器后，就可以配置基于 RADIUS 服务器的方法列表了，在全局配置模式下执行以下命令：

命令	作用
configure terminal	进入全局配置模式
aaa new-model	打开 AAA 开关
aaa authorization network {**default** |	定义使用 RADIUS 授权方法。

list-name} **group radius**	

39.4.5.2 配置 Network 授权示例

下例演示如何进行网络授权。

```
Ruijie# configure terminal
Ruijie(config)# aaa new-model
Ruijie(config)# radius-server host 192.168.217.64
Ruijie(config)# radius-server key test
Ruijie(config)# aaa authorization network test group radius
none
Ruijie(config)# end
Ruijie# show running-config
aaa new-model
!
aaa authorization network test group radius none
!
radius-server host 192.168.217.64
radius-server key 7 093b100133
!
```

39.5 配置记帐

AAA 记帐功能够跟踪用户使用的服务和网络资源。启用记帐功能以后，网络访问服务器或设备实时地以属性对的方式将用户访问网络的情况发送给安全服务器。您可以使用一些分析软件对这些数据进行分析，实现对用户的活动进行计费、审计以及跟踪等功能。

39.5.1 记帐类型

我司产品目前支持以下记帐类型：

- Exec 记账
- Command（命令）记账
- Network（网络）记帐

其中 Exec 记账针对的是用户终端登录到 NAS 上的命令行界面（CLI），在登入和登出时分别进行记账；命令记账针对的是用户终端登录到 NAS 上的 CLI 界面以后，记录其具体执行的命令；而网络记账针对的与网络连接上的用户会话有关的信息。

.

说明:

命令记账功能目前仅 TACACS+协议支持，具体内容请参考“配置 TACACS+”。

39.5.2 记帐的准备工作

在配置 AAA 记帐以前，必须完成以下任务：

- 启用 AAA 安全服务。关于如何启用 AAA 的信息，请参见“AAA 概述”。
- 定义安全协议参数。进行记账，需要配置安全协议参数。我司产品 Network 记账仅支持 RADIUS 安全协议；Exec 记账支持 RADIUS 和 TACACS+协议；Command 记账仅支持 TACACS+协议。关于 RADIUS 协议的信息，请参见“配置 RADIUS”， 关于 TACACS+协议的信息，请参见“配置 TACACS+”。
- （可选）配置 AAA 认证。某些记账需要在用户通过认证之后才进行（例如 Exec 记账），其他情况下，没有经过认证也可以进行记账。关于 AAA 认证的信息，请参见“配置认证”。

39.5.3 配置 AAA Exec 记帐

Exec 记帐对于已登录到 NAS 的用户终端，可以记录其进入和退出 CLI 界面的信息。在用户终端登录并进入到 NAS 的 CLI 界面时，会向安全服务器发送一个记账开始（Start）信息，在退出 CLI 界面时，会向服务器发送一个记账结束（Stop）信息。

注意:

只有登录到 NAS 的用户终端通过了 Login 认证，才会进行 Exec 记账。如果没有设置 Login 认证，或者认证时候采用了 none 方法，则不会进行 Exec 记账。针对同一个用户终端的登录，登入时如果没有进行过 Start 记账，登出时也就不会进行 Stop 记账。

要配置 AAA Exec 记账，在全局配置模式下执行以下命令：

命令	作用
configure terminal	进入全局配置模式
aaa new-model	启用 AAA。
aaa accounting exec {default | *list-name*}	定义一个记账方法列表，如果需

start-stop *method1* [*method2*...]	要定义多个方法列表，重复执行该命令。
line vty *line-num*	进入您需要应用 AAA Exec 记账的线路。
accounting exec {**default** | *list-name*}	将方法列表应用线路。

关键字 *list-name* 用来命名创建记账方法列表，可以是任何字符串；关键字 *method* 指的是记账实际算法。仅当前面的方法返回 ERROR（无应答），才使用后面的其他记账方法；如果前面的方法返回 FAIL(失败)，则不使用其他记账方法。为了使记账最后能成功返回，即使所有指定方法都没有应答，可以在命令行中将 **none** 指定为最后一个记账方法。

说明:

使用关键字 **start-stop**，网络访问服务器在用户开始和结束访问网络服务时都给安全服务器发送记帐信息。

39.5.3.1 使用 RADIUS 进行 Exec 记账

要配置用 RADIUS 服务器进行 Exec 记账，首先要配置 RADIUS 服务器，有关 RADIUS 服务器配置的信息，请参见“配置 RADIUS”。

配置好 RADIUS 服务器后，就可以配置基于 RADIUS 服务器的方法列表了，在全局配置模式下执行以下命令：

命令	作用
configure terminal	进入全局配置模式
aaa new-model	打开 AAA 开关
aaa accounting exec {**default** | *list-name*} **start-stop group radius**	定义使用 RADIUS 记账方法。
end	退出到特权模式
show aaa method-list	确认配置的方法列表
configure terminal	进入全局配置模式
line vty line-num	进入线路配置模式
accounting exec {**default** | *list-name*}	应用方法列表
end	退出到特权模式
show running-config	确认配置

39.5.3.2 配置 Exec 记帐示例

下例演示如何进行 Exec 记账。我们设置 VTY 线路 0~4 上的用户登录时采用 Login 认证，并且进行 Exec 记账。其中 Login 认证采用本地认证，Exec 记账采用 RADIUS。远程 RADIUS 服务器地址为 192.168.217.64，共享密钥为 test。本地用户名为 ruijie，口令为 ruijie。如下：

```
Ruijie# config
Ruijie(config)# aaa new-model
Ruijie(config)# radius-server host 192.168.217.64
Ruijie(config)# radius-server key test
Ruijie(config)# username ruijie password ruijie
Ruijie(config)# aaa authentication login auth local
Ruijie(config)# aaa accounting exec acct start-stop group
radius
Ruijie(config)# line vty 0 4
Ruijie(config-line)# login authentication auth
Ruijie(config-line)# accounting exec acct
Ruijie(config)# end
Ruijie# show running-config
!
aaa new-model
!
aaa accounting exec acct start-stop group radius
aaa authentication login auth local
!
username ruijie password ruijie
!
radius-server host 192.168.217.64
radius-server key 7 093b100133
!
line con 0
line vty 0 4
 accounting exec acct
 login authentication auth
!
end
```

39.5.4 配置 AAA Network 记帐

Network（网络）记帐提供了关于用户会话的记账信息，包括报文的个数及字节数、IP 地址、用户名等。Network 记账目前只支持 RADIUS 协议。

说明:

RADIUS 记帐信息的格式随不同的 RADIUS 安全服务器而变化。记帐记录中的内容可能会由于我司产品版本的不同而有些变化。

要配置 AAA Network 记账，在全局配置模式下执行以下命令：

命令	作用
configure terminal	进入全局配置模式
aaa new-model	启用 AAA。
aaa accounting network {**default** | *list-name*} **start-stop** *method1* [*method2*...]	定义一个记账方法列表，如果需要定义多个方法列表，重复执行该命令。

关键字 *list-name* 用来命名创建记账方法列表，可以是任何字符串；关键字 *method* 指的是记账实际算法。仅当前面的方法返回 ERROR（无应答），才使用后面的其他记账方法；如果前面的方法返回 FAIL(失败)，则不使用其他记账方法。为了使记账最后能成功返回，即使所有指定方法都没有应答，可以在命令行中将 **none** 指定为最后一个记账方法。

39.5.4.1 使用 RADIUS 进行 Network 记账

要配置用 RADIUS 服务器进行 Network 记账，首先要配置 RADIUS 服务器，有关 RADIUS 服务器配置的信息，请参见“配置 RADIUS”。

配置好 RADIUS 服务器后，就可以配置基于 RADIUS 服务器的方法列表了，在全局配置模式下执行以下命令：

命令	作用
configure terminal	进入全局配置模式
aaa new-model	打开 AAA 开关
aaa accounting network {**default** | *list-name*} **start-stop group radius**	定义使用 RADIUS 记账方法。

39.5.4.2 配置 Network 记帐示例

以下是使用 RADIUS 进行 Network 记帐的一个例子：

```
Ruijie# config
Ruijie(config)# aaa new-model
Ruijie(config)# radius-server host 192.168.217.64
Ruijie(config)# radius-server key test
```

```
Ruijie(config)# aaa accounting network acct start-stop group
radius
Ruijie(config)# end
Ruijie# show running-config
!
aaa new-model
!
aaa accounting network acct start-stop group radius
!
username Ruijie password 0 starnet
username Ruijie privilege 6
!
radius-server host 192.168.217.64
radius-server key 7 093b100133
```

39.6 监视 AAA 用户

要查看当前登录用户的信息，请在特权用户模式下执行命令：

命令	作用
show aaa user { *id* | **all** }	查看当前 AAA 用户信息

39.7 配置支持 VRF 的 AAA 组

Virtual Private Networks (VPNs)为用户提供了一种安全的方式在 ISP 骨干网上共享带宽。一个 VPN 即是共享路由的站点集。用户站点通过一到多个接口链接到服务提供商网络，VPN 路由表也叫 VPN routing/forwarding (VRF) table，AAA 可以为每个自定义服务器组指定 VRF。

要配置 AAA 组的 VRF，请在全局配置模式下执行以下命令：

命令	作用
configure terminal	进入全局配置模式
aaa new-model	打开 AAA 开关
aaa group server radius *gs_name*	配置 RADIUS 服务器组 进入服务器组配置模式
ip vrf forwarding *vrf_name*	为组选择 vrf

说明:

需要产品支持 vrf 功能

39.8 配置 Login 的用户认证失败锁定

Login 登录交换机，为了防止 Login 用户破解密码，提供配置命令用于限制用户尝试密码的失败次数；超过尝试失败次数，用户被锁定多长时间不能登录。

要配置 Login 登录参量，请在全局配置模式下执行以下命令：

命令	作用
configure terminal	进入全局配置模式
aaa new-model	打开 AAA 开关
aaa local authentication attempts *num*	配置 login 登录，用户尝试失败次数
aaa local authentication lockout-time *num*	配置 login 登录，用户尝试超过配置的失败次数，被锁定的时间长度（小时）
end	退出到特权模式
show aaa user lockout {all | user-name *name-string*}	显示当前被锁定的用户列表
clear aaa local user lockout {all | user-name *name-string* }	清除被锁定的用户列表

说明:

默认情况下，Login 尝试失败次数为 3 次，被锁定的时间限制为 15 小时。

40 RADIUS 配置

40.1 RADIUS 概述

RADIUS 是远程认证拨号用户服务(Remote Authentication Dial-In User Service)的简称，它是一种分布式的客户机/服务器系统，与 AAA 配合对试图连接的用户进行身份认证，防止未经授权的访问。在 RGOS 的实现中，RADIUS 客户端运行在设备或网络访问服务器（NAS）上，并向中央 RADIUS 服务器发出身份认证请求，中央服务器包含了所有的用户身份认证和网络服务信息。

由于 RADIUS 是一种完全开放的协议，很多系统如 UNIX、WINDOWS 2000 等均将 RADIUS 服务器作为一个组件安装，因此 RADIUS 是目前应用最广泛的安全服务器。

RADIUS 的运行过程如下：

- 提示用户输入用户名和密码。
- 用户名和加密的密码通过网络发送到 RADIUS 服务器。
- RADIUS 返回下述响应之一：
- ACCEPT：通过用户身份认证。
- REJECT：身份认证失败，提示重新输入用户名和密码。
- CHALLENGE：RADIUS 服务器发出询问请求，从用户那里收集更多认证信息。
- 在 ACCEPT 的响应中包含用户的授权信息。

下图是一个典型的 RADIUS 拓扑图：

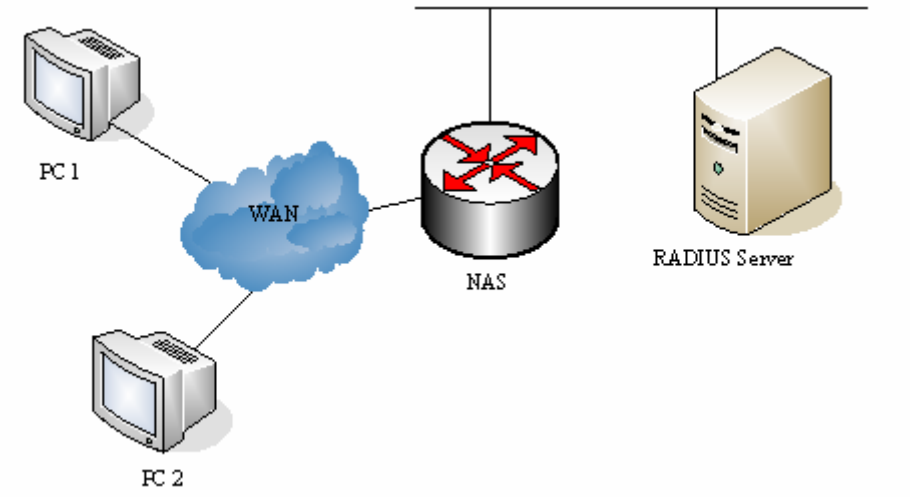


图 1 典型 RADIUS 网络配置

40.2 RADIUS 配置任务

要在网络设备上配置 RADIUS，必须先执行以下任务：

- 启用 AAA。关于如何启用 AAA 的信息，请参见“AAA 概述”。
- 使用 **aaa authentication** 命令定义 RADIUS 身份认证方法列表。关于使用 **aaa authentication** 命令定义身份认证列表的信息，请参见“配置认证”。
- 在特定线路上应用定义的认证列表，否则使用默认身份认证列表进行认证。关于更详细信息，请参见“配置认证”。

完成以上任务后，就可进行配置 RADIUS。配置 RADIUS 包括以下几个部分：

- 配置 RADIUS 协议参数
- 指定 RADIUS 认证

40.2.1 配置 RADIUS 协议参数

在网络设备上配置 RADIUS 之前，应确保 RADIUS 服务器的网络通讯良好。要配置 RADIUS 协议参数，请执行以下命令：

命令	作用
configure terminal	进入全局配置模式
radius-server host *ip-address* [**auth-port** *port*] [**acct-port** *port*]	配置远程 RADIUS 安全服务器的 IP 地址或主机名，指定认证端口和记帐端口。
radius-server key *string*	配置设备和 RADIUS 服务器进行通讯的共享密码。
radius-server retransmit *retries*	指定设备在确认 RADIUS 无效以前发送请求的次数（默认值为 3）。
radius-server timeout *seconds*	指定设备重传请求以前等待的时间（默认值为 2 秒）。
radius-server deadtime *minutes*	指定设备发出请求没有回应而作为服务器死亡的需要等待时间（默认值为 5minutes）

注意：

配置 RADIUS，必须要配置 RADIUS Key。网络设备上的共享密码和 RADIUS 服务器上的共享密码必须一致。

40.2.2 指定 RADIUS 身份认证

指定 RADIUS 服务器并定义了 RADIUS 身份认证共享密码后，需要为 RADIUS 定义身份认证方法列表。由于 RADIUS 身份认证是通过 AAA 来进行的，所以需要执行 **aaa authentication** 命令来定义身份认证方法列表，并指定身份认证方法为 RADIUS。关于更详细的信息，可以参考 AAA 配置。

40.2.3 指定 RADIUS 标准属性类型

通过该节可以配置标准属性的类型，目前支持设置 RADIUS Calling-Station-ID 属性（属性类型为 31）。

40.2.3.1 配置 Calling-Station-ID 格式

RADIUS Calling-Station-ID 属性用于 NAS 向 RADIUS Server 发送请求报文时候，标识 NAS 的身份。Calling-Station-ID 属性内容是字符串，可以有多种组成格式，由于要求必须能唯一标识一个 NAS，因此常选择使用 NAS 的 MAC 地址作为其内容。关于这 MAC 地址的格式，有以下几种：

格式	说明
ietf	IETF（RFC3580）规定的标准格式，使用‘-’作为分隔符。例如： 00-D0-F8-33-22-AC
normal	常用的表示 MAC 地址的格式（点分十六进制格式），使用‘.’作为分隔符。例如： 00d0.f833.22ac
unformatted	无格式，没有任何分隔符，默认使用这个格式。例如： 00d0f83322ac

要配置 RADIUS Calling-Station-ID 属性（基于 MAC）的格式，请执行以下命令：

命令	作用
configure terminal	进入全局配置模式
radius-server attribute 31 mac format {ietf | normal | unformatted}	配置 RADIUS Calling-Station-ID 属性（基于 MAC）的格式，默认使用 **unformatted** 格式。

40.2.4 指定 RADIUS 私有属性类型

通过该节可以自由配置私有属性的类型。默认配置如下：

锐捷产品私有属性识别默认配置：

ID	功能	TYPE
1	max down-rate	1
2	qos	2
3	user ip	3
4	vlan id	4
5	version to client	5
6	net ip	6
7	user name	7
8	password	8
9	file-diractory	9
10	file-count	10
11	file-name-0	11
12	file-name-1	12
13	file-name-2	13
14	file-name-3	14
15	file-name-4	15
16	max up-rate	16
17	version to server	17
18	flux-max-high32	18
19	flux-max-low32	19
20	proxy-avoid	20
21	dailup-avoid	21
22	ip privilige	22
23	login privilige	42
24	limit to user number	50

扩展厂商 ID 默认配置：

ID	功能	TYPE
1	max down-rate	76
2	qos	77
3	user ip	3
4	vlan id	4
5	version to client	5

6	net ip	6
7	user name	7
8	password	8
9	file-diractory	9
10	file-count	10
11	file-name-0	11
12	file-name-1	12
13	file-name-2	13
14	file-name-3	14
15	file-name-4	15
16	max up-rate	75
17	version to server	17
18	flux-max-high32	18
19	flux-max-low32	19
20	proxy-avoid	20
21	dailup-avoid	21
22	ip privilige	22
23	login privilige	42
24	limit to user number	50

说明：

两个功能不能配置为一个类型号

以下是网络设备配置私有类型的具体实例：

```
Ruijie# show radius vendor-specific
id    vendor-specific     type-value
----- -------------------- ----------
1     max down-rate       76
2     qos                 77
3     user ip             3
4     vlan id             4
5     version to client    5
6     net ip              6
7     user name           7
8     password            8
9     file-diractory      9
```

```
10   file-count          10
11   file-name-0         11
12   file-name-1         12
13   file-name-2         13
14   file-name-3         14
15   file-name-4         15
16   max up-rate         75
17   version to server    17
18   flux-max-high32      18
19   flux-max-low32       19
20   proxy-avoid         20
21   dailup-avoid         21
22   ip privilige         22
23   login privilige      42
24   limit to user number 50
```

```
Ruijie# configure
Ruijie(config)# radius attribute 24 vendor-type 67
Ruijie(config)# show radius vendor-specific
id    vendor-specific     type-value
----- -------------------- ----------
1     max down-rate        76
2     qos                 77
3     user ip              3
4     vlan id              4
5     version to client    5
6     net ip              6
7     user name            7
8     password             8
9     file-diractory       9
10    file-count          10
11    file-name-0         11
12    file-name-1         12
13    file-name-2         13
14    file-name-3         14
15    file-name-4         15
16    max up-rate         75
17    version to server    17
18    flux-max-high32      18
19    flux-max-low32       19
20    proxy-avoid         20
21    dailup-avoid         21
22    ip privilige         22
23    login privilige      42
24    limit to user number 50
Ruijie(config)#
```

```
Ruijie(config)#
```

40.3 监控 RADIUS

要监控 RADIUS，请在特权用户模式执行以下命令：

命令	作用
debug radius event	打开 RADIUS 调试开关，查看 RADIUS 调试信息。

40.4 RADIUS 配置示例

典型 RADIUS 网络配置图中，RADIUS 服务器对访问用户进行身份认证，并为访问用户启用记帐功能，记录用户使用网络服务情况。

说明：

RADIUS 服务器可以是 Windows 2000/2003 Server（IAS）、UNIX 系统所带组件，也可以是一些厂商通过的专用服务器软件。

以下是网络设备配置 RADIUS 的具体实例：

```
Ruijie# configure terminal
Ruijie(config)# aaa new-model
Ruijie(config)# radius-server host 192.168.12.219
auth-port 1645 acct-port 1646
Ruijie(config)# radius-server key aaa
Ruijie(config)# aaa authentication login test group radius
Ruijie(config)# end
Ruijie# show radius server
Server IP:       192.168.12.219
Accounting Port: 1646
Authen  Port:    1645
Server State:    Ready

Ruijie# configure terminal
Ruijie(config)# line vty 0
Ruijie(config-line)# login authentication test
Ruijie(config-line)# end
Ruijie# show running-config
!
aaa new-model
!
!
```

```
aaa authentication login test group radius
!
username ruijie password 0 starnet
!
radius-server host 192.168.12.219 auth-port 1645 acct-port
1646
!
line con 0
line vty 0
login authentication test
line vty 1 4
!
```

41 TACACS+配置

41.1 TACACS+概述

TACACS+（是在 TACACS（RFC 1492 Terminal Access Controller Access Control System））基础上进行了功能增强的安全协议。主要是通过 Client-Server 模式与 TACACS 服务器通信来实现多种用户的 AAA 功能，可用于终端用户的认证、授权和计费，在配置使用 TACACS+服务器前，需要先配置好 TACACS+服务器的相关内容。

TACACS+支持用户认证、授权和计费分析，即我们可以使用一台服务器进行认证，然后使用另外一台服务器进行授权，同时可以使用第三台服务器进行计费，每台服务器可以拥有自己的用户数据信息，可以对立进行用户的认证、授权和计费。

TACACS+的报文格式如下：

4	8	16	24	32 bit
Major	Minor	Packet type	Sequence no.	Flags
Session ID				
Length				

图 1

- Major Version — 主要 TACACS+ 版本号。
- Minor Version — 次要 TACACS+ 版本号。
- Packet Type — 可能值包括：
  - TAC_PLUS_AUTHEN： = 0x01 （认证）；
  - TAC_PLUS_AUTHOR：= 0x02 （授权）；
  - TAC_PLUS_ACCT：= 0x03 （计费）。
- Sequence Number — 当前会话中的数据包序列号。会话中的第一个 TACACS+ 数据包序列号必须为 1，其后的每个数据包序列号逐次加 1。因此客户机只发送奇序列号数据包，而 TACACS+ Daemon 只发送偶序列号数据包。
- Flags — 该字段包括各种位图格式的标志（flag）。Flag 值表明数据包是否进行加密。
- Session ID — 该 TACACS+ 会话的 ID。
- Length — TACACS+ 数据包主体长度（不包括头部），报文全部以加密形式在网络上传输。

41.2 TACACS+应用

TACACS+的典型应用为终端用户的登陆管理控制，交换机作为 TACACS+的客户端，将用户名和密码发给 TACACS+服务器进行验证，验证通过并得到授权之后可以登录到交换机上进行操作。如图 1 所示：

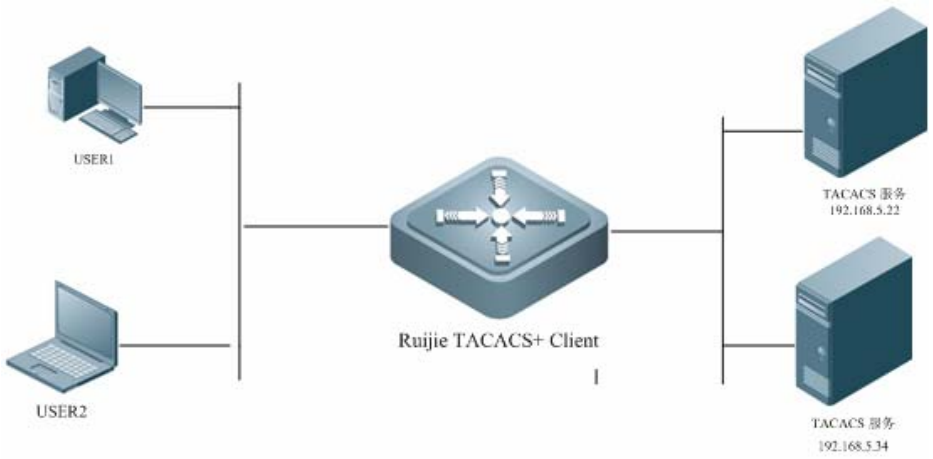


图 2

下边以 login 登陆的基本认证、授权和计费说明 TACACS+运行中的数据报文的交互：

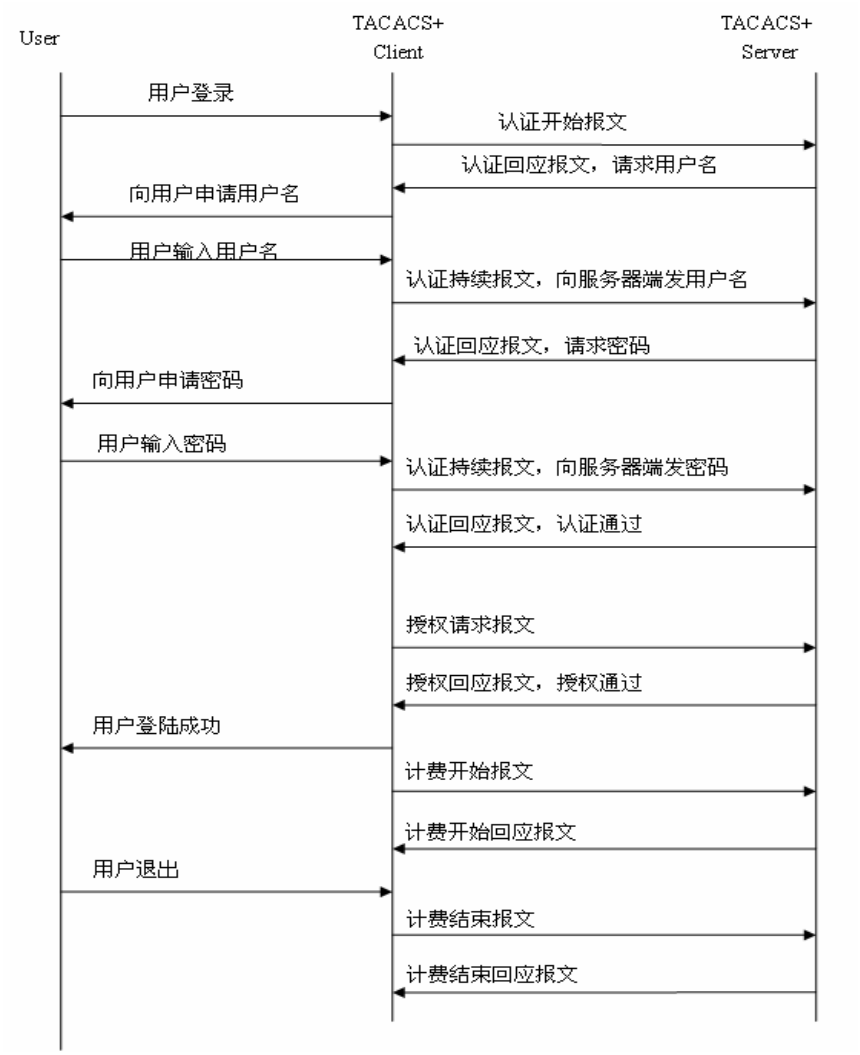


图 3

在整个过程中的基本消息交互流程可以分为三个部分：

1. 认证过程包含：

1） 用户请求登录交换机。

2） TACACS+客户端收到请求之后，向 TACACS+服务器发送认证开始报文。

3） TACACS+服务器发送认证回应报文，请求用户名；

4） TACACS+客户端向用户询问用户名。

5） 用户输入登陆的用户名信息。

6） TACACS+客户端收到用户名后，向 TACACS+服务器发送认证持续报文，其中包括了用户名。

7） TACACS+服务器发送认证回应报文，请求登录密码；

8） TACACS+客户端收到向用户询问登录密码。

9） 用户输入登陆的密码信息。

10） TACACS+客户端收到登录密码后，向 TACACS+服务器发送认证持续报

文，其中包括了登录密码。

11） TACACS+服务器发送认证回应报文，指示用户通过认证。

2、认证通过后对用户进行授权：

1） TACACS+客户端向 TACACS+服务器发送授权请求报文。

2） TACACS+服务器发送授权回应报文，指示用户通过授权。

3） TACACS+客户端收到授权回应成功报文，向用户输出交换机的配置界面。

3、授权通过后，需要对登陆的用户进行计费，审计。

1） TACACS+客户端向 TACACS+服务器发送计费开始报文。

2） TACACS+服务器发送计费回应报文，指示计费开始报文已经收到。

3） 用户退出。

4） TACACS+客户端向 TACACS+服务器发送计费结束报文。

5） ACACS+服务器发送计费回应报文，指示计费结束报文已经收到。

41.3 TACACS+配置任务

要在网络设备上配置 TACACS+，必须先执行以下任务：

- 使用 **aaa new-mode** 命令启用 AAA。在使用 TACACS+前必须先启用 AAA，关于如何启用 **aaa new-mode** 的信息，请参见“AAA 概述”。
- 使用 **tacacs-server host** 命令配置一个或多个 tacacs+服务器。
- 使用 **tacacs-server key** 命令指定服务器和 NAS 共享的 key。
- 使用 **tacacs-server timeout** 命令指定等待服务器响应的超时时间。
- 如果需要认证，使用 **aaa authentication** 命令定义使用 TACACS+的身份认证方法列表。关于使用 a**aa authentication** 命令定义身份认证列表的信息，请参见“配置认证”。
- 如果需要授权，使用 **aaa authorization** 命令定义使用 TACACS+的授权方法列表。关于使用 **aaa authorization** 命令定义授权方法列表的信息，请参见“配置授权”。
- 如果需要记帐，使用 **aaa accounting** 命令定义使用 TACACS+的记帐方法列表。关于使用 **aaa accounting** 命令定义记帐方法列表的信息，请参见“配置记帐”。
- 在特定线路上应用定义的认证列表，否则使用默认列表。

41.3.1 配置 TACACS+协议参数

在网络设备上配置 TACACS+之前，应确保 TACACS+服务器的网络通讯良好。要配置 TACACS+协议参数，请执行以下命令：

命令	作用
configure terminal	进入全局配置模式
aaa group server tacacs+ *group-name*	配置 TACACS+服务器组，将不同的TACACS+服务器划分到不同的组。
server *ip-address*	配置 TACACS+服务器组中的服务器地址
ip vrf forwarding *vrf-name*	配置 TACACS+服务器组使用的 vrf 名字(支持 VRF 的设备存在此命令)
tacacs-server host *ip-address* [**port** *integer*] [**timeout** *integer*] **key** [**0** | **7**]*string* []	配置远程 TACACS+安全服务器的 IP 地址，通过不同的参数组合份配置该服务器的不同的参数： ● *ip-address* ：配置服务器的地址。 ● **port** *intege* [可选] ：确定服务器使用的端口，默认使用 49，可配置范围 1 到 65535。 ● **timeout** *integer* [可选] ：用于配置本服务器的超时响应的时间，默认 5s，可配置范围 1 到 1000s ● **key** *string* [可选]：用于配置和对应 ip 的服务器的共享的密钥。
tacacs-server key [**0** | **7**]*string*	配置设备和TACACS+服务器进行通讯的共享密钥，当对应的主机没有单独配置 key 时，则使用全局的配置。
tacacs-server timeout *seconds*	指定设备重传请求以前等待的时间（默认值为 5 秒），当特定主机没有配置特定的超时时间时，则使用全局的配置时间。
ip tacacs source-interface *interface*	指定发送 tacacs+请求到服务器使用的源 IP，默认为不指定

注意：

配置 TACACS+，必须要配置 TACACS+ Key。网络设备上的共享密码和 TACACS+服务器上的共享密码必须一致。

41.4 使用 TACACS+ 进行认证、授权、记帐配置示例

典型 TACACS+网络配置图中，TACACS+服务器对访问用户进行身份认证、授权和记帐，下边分别通过 login 认证、授权和记帐，演示如何配置使用 TACACS+进行认证、授权和记帐。

41.4.1 Login 认证使用 TACACS+的配置示例：

1. 首先配置启用 aaa：

```
Ruijie# configure terminal
Ruijie(config)# aaa new-model
```

2. 然后配置 tacacs+ server 信息：

```
Ruijie(config)# tacacs-server host 192.168.12.219
Ruijie(config)# tacacs-server key aaa
```

3. 然后配置使用 tacacs+的认证方法：

```
Ruijie(config)# aaa authentication login test group tacacs+
```

4. 在接口上应用配置认证方法:

```
Ruijie(config)# line vty 0 4
Ruijie (config-line)# login authentication test
```

通过以上配置就实现了配置 login 的 tacacs+认证，配置显示如下。

```
Ruijie#show running-config
!
aaa new-model
!
aaa authentication login test group tacacs+
!
tacacs-server host 192.168.12.219
tacacs-server key aaa
!
line con 0
line vty 0
login authentication test
line vty 1 4
login authentication test
!
```

41.4.2 ENABLE 认证使用 TACACS+的配置示例：

1. 首先配置启用 aaa：

```
Ruijie# configure terminal
Ruijie(config)# aaa new-model
```

2. 然后配置 tacacs+ server 信息：

```
Ruijie(config)# tacacs-server host 192.168.12.219
Ruijie(config)# tacacs-server host 192.168.12.218
```

```
Ruijie(config)# tacacs-server host 192.168.12.217
Ruijie(config)# tacacs-server key aaa
```

3. 配置 tacacs+ server group 使用服务器列表中一部分服务器：

```
Ruijie(config)#aaa group server tacacs+ tacgroup1
Ruijie(config-gs-tacacs)#server 192.168.12.219
Ruijie(config-gs-tacacs)#server 192.168.12.218
```

4. 然后配置使用 tacgroup1 的认证方法：

```
Ruijie(config)# aaa authentication enable default group tacgroup1
```

通过以上配置就实现了配置 enable 的部分 tacacs+服务器认证，配置显示如下。

```
Ruijie#show running-config
!
aaa new-model
!
!
aaa group server tacacs+ tacgroup1
server 192.168.12.219
server 192.168.12.218
!
aaa authentication enable default group tacgroup1
!
!
tacacs-server host 192.168.12.219
tacacs-server host 192.168.12.218
tacacs-server host 192.168.12.217
tacacs-server key aaa
!
line con 0
line vty 0
line vty 1 4
!
```

41.4.3 登陆授权使用 TACACS+的配置示例：

1. 首先配置启用 aaa：

```
Ruijie# configure terminal
Ruijie(config)# aaa new-model
```

2. 然后配置 tacacs+ server 信息：

```
Ruijie(config)# tacacs-server host 192.168.12.219
Ruijie(config)# tacacs-server key aaa
```

3. 然后配置使用 tacacs+的授权方法：

```
Ruijie(config)# aaa authorization exex test group tacacs+
```

4. 在接口上应用配置授权：

```
Ruijie(config)# line vty 0 4
Ruijie (config-line)#authorization exec test
```

通过以上配置就实现了配置登陆授权使用 tacacs+，配置显示如下。

```
Ruijie#show running-config
!
aaa new-model
!
!
aaa authorization exec test group tacacs+
!
tacacs-server host 192.168.12.219
tacacs-server key aaa
!
line con 0
line vty 0
 authorization exec test
line vty 1 4
 authorization exec test
!
```

41.4.4 15 级 Commans 审计使用 TACACS+的配置示例

1) 首先配置启用 aaa：

```
Ruijie# configure terminal
Ruijie(config)# aaa new-model
```

2) 然后配置 tacacs+ server 信息：

```
Ruijie(config)# tacacs-server host 192.168.12.219
Ruijie(config)# tacacs-server key aaa
```

3) 然后配置使用 tacacs+的命令审计方法：

```
Ruijie(config)# aaa accounting commands 15 test start-stop group tacacs+
```

4) 在接口上应用配置授权:

```
Ruijie(config)# line vty 0 4
Ruijie (config-line)#accounting commands 15 test
```

5) 通过以上配置就实现了配置 enable 的部分 tacacs+服务器认证，配置显示如下。

```
Ruijie#show running-config
!
aaa new-model
!
!
aaa accounting commands 15 default group tacacs+
!
!
tacacs-server host 192.168.12.219
tacacs-server key aaa
!
line con 0
line vty 0 4
 accounting commands 15 test
```

42 SSH 终端服务

42.1 SSH 简介

SSH 是英文 Secure Shell 的简写形式。SSH 连接所提供的功能类似于一个 Telnet 连接，与 Telnet 不同的是基于该连接所有的传输都是加密的。当用户通过一个不能保证安全的网络环境远程登录到设备时，SSH 特性可以提供安全的信息保障和强大的认证功能，以保护设备不受诸如 IP 地址欺诈、明文密码截取等攻击。

42.2 锐捷 SSH 支持算法

支持算法	SSH1	SSH2
签名认证算法	RSA	RSA、DSA
密钥交换算法	基于 RSA 公钥加密的密钥交换算法。	KEX_DH_GEX_SHA1 KEX_DH_GRP1_SHA1 KEX_DH_GRP14_SHA1
加密算法	DES、3DES、Blowfish	DES 、 3DES 、 AES-128 、 AES-192、AES-256
用户认证算法	基于用户口令的认证方式	基于用户口令的认证方式
消息认证算法	不支持	MD5、SHA1、SHA1-96、MD5-96
压缩算法	NONE（无压缩）	NONE（无压缩）

42.3 锐捷 SSH 支持

注意：

锐捷网络产品仅支持 SSH 服务器（兼容 SSHv1 与 SSHv2），不支持 SSH 客户端。

42.4 SSH 配置

42.4.1 缺省的 SSH 配置

项目	缺省值
SSH 服务端状态	关闭
SSH 版本	兼容模式（支持版本 1 和 2）
SSH 用户认证超时时间	120s
SSH 用户重认证次数	3 次

42.4.2 用户认证配置

1. 基于 SSH 连接安全性的考虑，禁止使用无认证方式登录。因此在用户登录认证时，所使用的登录认证方式必须设置密码；（Telnet 可以设置无认证登录）
2. 每次输入的用户名(Username)与密码(Password)长度必须大于零。如果当前认证方式不需要用户名时，用户名可以任意输入，但是输入长度必须大于零。

42.4.3 打开 SSH Server

缺省情况下，SSH Server 处于关闭状态。打开 SSH Server，需要在全局配置模式下，执行 **enable service ssh-server** 命令，同时需要生成 SSH 密钥，使 SSH Server 的状态成为 ENABLE。

命令	说明
configure terminal	进入配置模式
enable service ssh-server	打开 SSH Server
crypto key generate {rsa|dsa}	生成密钥

注意:

删除密钥时，不存在命令**[no] crypto key generate**；而是通过命令 **crypto key zeroize** 命令删除密钥。

42.4.4 关闭 SSH Server

关闭 SSH Server，需要在全局配置模式下，执行 **no enable service ssh-server**

命令，使 SSH Server 的状态成为 DISABLE。

命令	说明
configure terminal	进入配置模式
no enable service ssh-server	关闭 SSH Server

42.4.5 配置 SSH server 支持版本

缺省情况下，SSH Server 兼容版本 1 和 2。使用以下命令配置 SSH 使用版本。

命令	说明
configure terminal	进入配置模式
ip ssh version {1 | 2}	配置 SSH 支持的版本
no ip ssh version	恢复 SSH 为缺省配置，支持 SSHv1 与 SSHv2；

42.4.6 配置 SSH 用户认证超时时间

缺省情况下，SSH Server 的用户认证超时时间为 120 秒，。使用以下命令配置 SSH 的用户认证超时时间。

命令	说明
configure terminal	进入配置模式
ip ssh time-out *time*	配置 SSH 的超时时间（范围 1-120sec）
no ip ssh time-out	恢复 SSH 的缺省用户认证超时时间为 120 秒；

42.4.7 配置 SSH 重认证次数

该配置命令用来设置 SSH 用户请求连接的认证重试次数，防止恶意猜测等非法行为。缺省情况下，SSH Server 的重认证次数为 3 次，即可以允许用户尝试三次输入用户名与密码进行认证尝试。使用以下命令配置 SSH 的重认证次数。

命令	说明
configure terminal	进入配置模式
ip ssh authentication-retries *retry times*	配置 SSH 的重认证次数（范围 0-5）
no ip ssh authentication-retries	恢复 SSH 的缺省重认证次数为 3 次；

注：上述命令具体配置请参见[SSH 命令参考手册]。

42.5 使用 SSH 进行设备管理

您可以使用 SSH 对设备进行管理，前提是必须打开 SSH Server 功能，默认情况下是关闭该功能的。由于 Windows 自带的 Telnet 组件不支持 SSH，因此必须使用第三方客户端软件，当前兼容性较好的客户端包括：Putty，Linux，SecureCRT。下面以客户端软件 SecureCRT 为例介绍 SSH 客户端的配置，配置界面如下图：

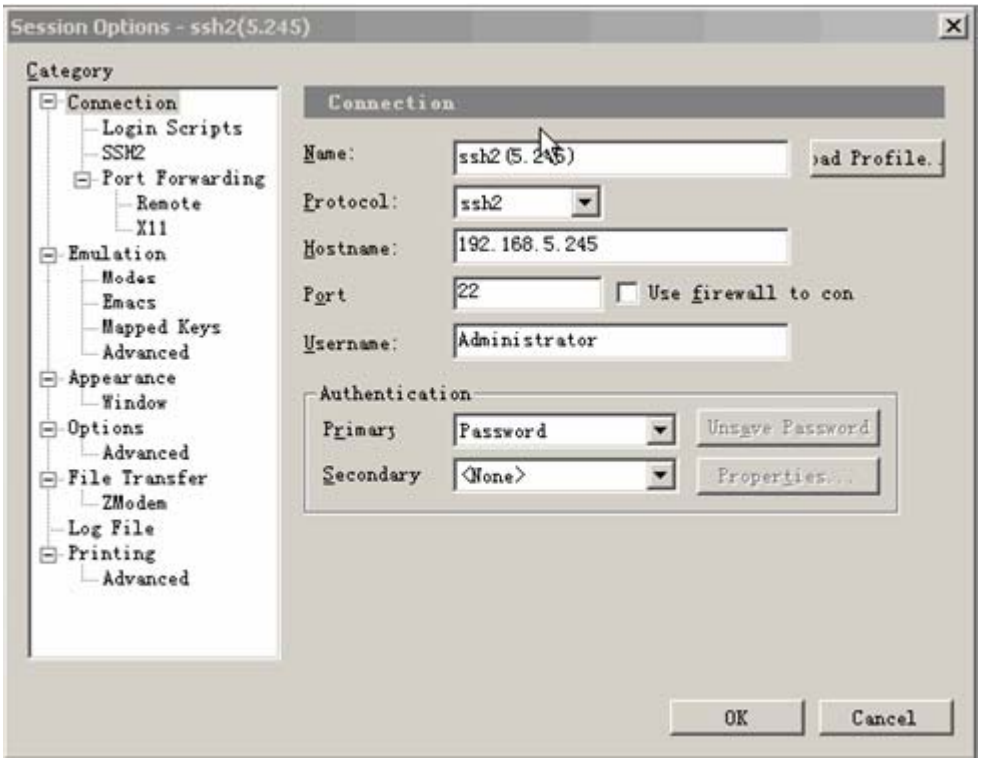


图 1

如图 1 使用协议 2 进行登陆，因此在 Protocol 选择 SSH2，Hostname 就是要登陆的主机的 IP 地址，这里为 192.168.5.245，端口为 22 即 SSH 监听的默认端口号，Username 为用户名，当设备只要求密码时，该用户名不会起作用，Authentication 为认证方式，我们只支持用户名密码的认证方式。使用的密码和 Telnet 密码是一致的。

然后点击 OK，进入出现以下的对话框：

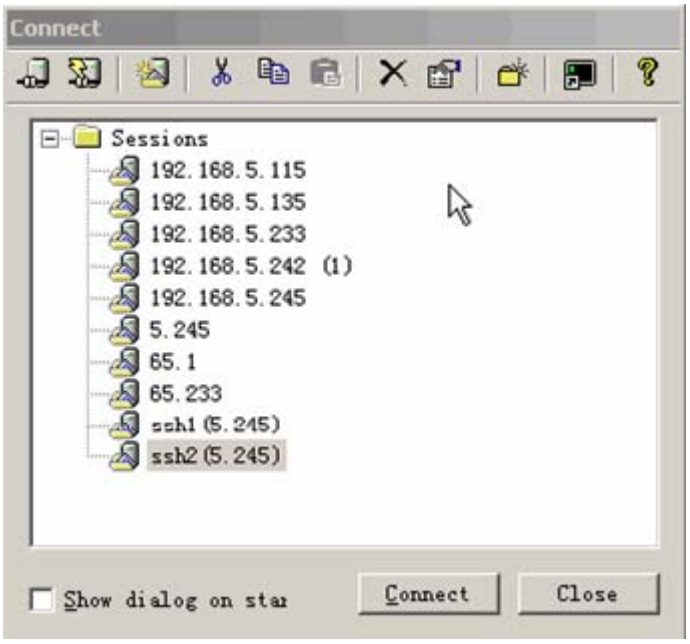


图 2

点击 Connect，登陆我们刚才配置的主机，如下图：

图 3

询问正在登陆主机 192.168.5.245 的机器，是否接收服务端发送过来的密钥，选择 Accept & Save（接受而且保存）或者 Accept Once(只接受一次)，接着会出现下面的密码认证对话框，如下图：

图 4

此时输入 Telnet 登陆密码就可以进入和 Telnet 一样的界面了。如下图显示：

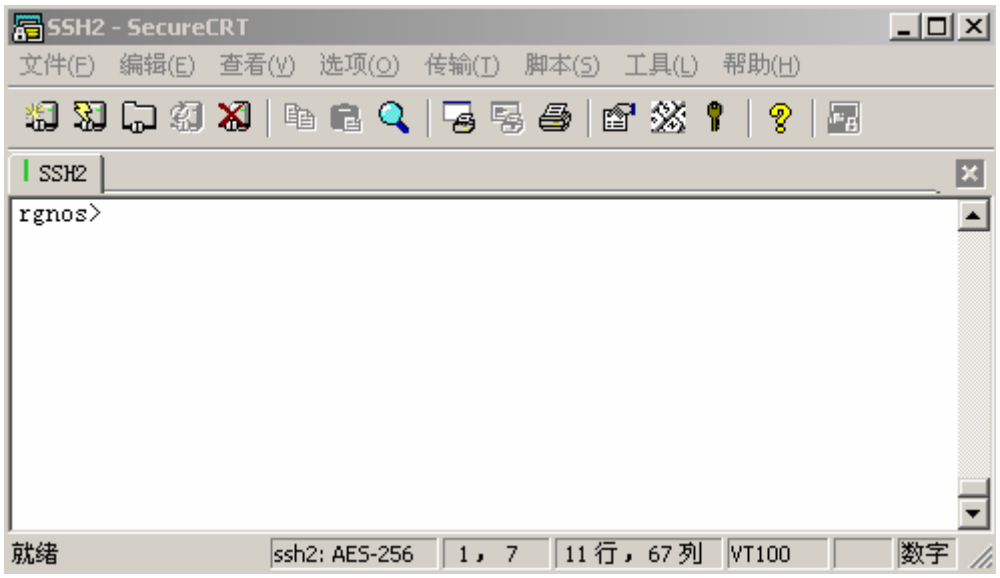


图 5

43 CPU 保护配置

43.1 概述

在网络环境中，有各种攻击报文在网络上传播，其会导致交换机的 CPU 利用率过高，影响协议运行，甚至无法正常管理交换机。针对这种情况，必须对交换机的 CPU 进行保护，即对送往交换机的 CPU 处理的各种报文进行流量控制和优先级处理，保护其正常处理能力。

CPU Protect 的实现模型分为 Classifying、Queuing、Scheduling 和 Shaping 四个阶段，下面对这四个阶段进行详细的说明。

Classifying：

Classifying 对每个需要送到 CPU 的报文进行分类，分类是根据报文的 L2、L3 以及 L4 信息，具体如下表所示：

S3760 系列交换机

报文类型	分类标准
BPDU	目的 MAC 地址为 01-80-C2-00-00-00 的报文
ARP	ARP request 报文
IGMP	IPV4 IGMP V1/V2/V3 报文
802.1X	目的 MAC 地址为 01-80-C2-00-00-03 的报文
GVRP	目的 MAC 地址为 01-80-C2-00-00-21 的报文
DHCP	DHCP 协议报文
Error_TTL	IPV4 TTL = 0 或 1 的报文
Unicast	目的 MAC 地址是交换机三层接口的 MAC 地址 例如 ARP reply,管理协议报文如：snmp，telnet，http 报文
Multicast	非 IGMP 的组播报文
Broadcast	非 DHCP 的广播报文
ipv4-ctrl	Ospf，vrrp 等 ipv4 协议控制报文
ripv1	目的 ip 是广播，udp 端口号为 520 的 ripv1 报文
route	IPv4/IPv6 单播路由打通报文
route-local	IPv4 目的 IP 是本机 IP 的报文
Route6-local	IPv6 目的 IP 是本机 IP 的报文
Other	非上述分类的需要送到 CPU 的报文

Queuing：

Queuing 动作负责将各种不同类型的报文送到指定的队列，在不同队列的报文具有不同的传输优先级。

CPU 端口共有 8 个优先级队列，您可以配置每种类型的报文对应的队列，Queuing 根据您的配置自动地将这种类型的报文的送到指定队列。

Scheduling：

当多个队列有报文需要传输时，Scheduling 负责从中选择一个队列并传输这个队列的报文。

CPU 端口 Scheduling 采用严格优先级(SP)算法，队列 7 具有最高优先级，队列 6 次之，以此类推，队列 0 具有最低优先级，高优先级队列报文总是先于低优先级队列的报文被传输。这样您可以根据每种报文的重要性将它们对应到不同的优先级的队列，确保重要的报文总是优先被传输。

说明：

S3760 系列交换机中 CPU 端口的队列调度：

当 buffer 控制处于 fc 模式时，对 S3760-24 和 S3760-48，只有当不超过两个队列有报文流时，才能保证队列调度严格优先；

当 buffer 控制处于 qos 模式时，对 S3760-12SFP/GT，只有当不超过四个队列有报文流时，才能保证队列调度严格优先。

关于 buffer 控制模式，请参见“QOS”的 buffer 控制章节。

Shaping：

Shaping 控制每个传输队列的最大速率，超过最大速率的报文将被丢弃。您可以根据网络实际情况配置每个队列的最大速率，同时，您还可以配置整个 CPU 端口的最大速率。

43.2 配置 CPU Protect

我们将从以下几个章节描述如何配置 CPU Protect

- CPU Protect 默认配置
- CPU Protect 配置指导
- 配置每种类型报文的队列
- 配置每个队列的最大速率
- 配置 CPU 端口最大速率
- 配置每种报文的最大速率

- 配置地址学习风暴控制

43.2.1 CPU Protect 默认配置

由于不同类型的交换机应用在不同的网络环境中，具有不同的网络攻击，CPU 保护也应该采用不同策略对应这些工具。所以对不同的交换机产品，我们精心地给您准备了不同的 CPU 的保护缺省配置。

S3760 系列交换机

报文和队列的对应关系：

报文类型	队列
BPDU	6
ARP	5
IGMP	3
802.1X	3
GVRP	3
DHCP	2
Error_TTL	0
Unicast	4
Multicast	1
Broadcast	0
error_ttl	0
route	5
route-local	7
route6-local	7
ripv1	7
ipv4-ctrl	7
other	0

每个队列最大速率：

队列	最大速率(kbps)
7	1000
6	1000
5	1000
4	1000
3	1000

2	1000
1	1000
0	1000
CPU 端口	3000

43.2.2 CPU Protect 配置指导

注意：

S3760 交换机不支持对 GVRP 和 802.1X 报文分别进行控制，即配置 GVRP 和 802.1X 报文对应的队列时，配置 GVRP 报文对应的队列会相应的改变 802.1X 报文对应的队列；配置 802.1X 报文对应的队列也会相应的改变 GVRP 报文对应的队列。

在配置 S3760 系列交换机的 CPU 端口和每个队列最大速率时，速率的粒度是 651（kbps）最小速率为 651(kbps)。

S3760 交换机，SVI 成员口上送往 CPU 的 ARP 报文，指定到队列 0，不能修改。此种报文不包含在 ARP 分类中。

43.2.3 配置报文和队列的对应关系

在配置模式下，按如下步骤设置报文和队列的对应关系：

命令	作用
Ruijie(config)# **cpu-protect type** { **bpdu** | **arp** | **igmp** | **dot1x** | **gvrp** | **dhcp** | **unicast** | **multicast** | **broadcast** | **error_ttl** | **other** } **traffic-class** *traffic-class-num*	设置报文对应的队列，*traffic-class-num* 的取值范围为 0－7。

如果要恢复其中一种类型报文对应的队列，可以用 **no cpu-protect type** { **bpdu** | **arp** | **igmp** | **dot1x** | **gvrp** | **dhcp** | **unicast** | **multicast** | **broadcast** | **error_ttl** | **other** } **traffic-class** 来执行。

以下例子是表示报文对应的队列的配置过程：

```
Ruijie(config)# cpu-protect type bpdu traffic-class 7
Ruijie(config)# end
Ruijie # show cpu-protect type bpdu traffic-class
%**********packet type      traffic-class**********
               bpdu           7
```

按以上配置，bpdu 报文就对应到队列 7。

43.2.4 配置每个队列的最大速率

在配置模式下，按如下步骤设置一个队列的最大速率：

命令	作用
Ruijie(config)# **cpu-protect traffic-class id** *id_num* **bandwidth** *bandwidth_value*	配置每个队列的最大速率（kbps），*id_num* 取值范围为 0－7，*bandwidth-value* 取值范围为 32－131072(kbps)
Ruijie(config)# **cpu-protect traffic-class all bandwidth** *bandwidth_value*	配置所有队列的最大速率（kbps），*bandwidth-value* 取值范围为 32－131072(kbps)

如果要恢复每个队列的缺省最大速率值，可以用 **no cpu-protect traffic-class** 来执行

以下例子是设置队列 7 的最大速率为 312(kbps)：

```
Ruijie#configure terminal
Ruijie(config)# cpu-protect traffic-class id 7  bandwidth 312
Ruijie(config)#end
Ruijie# show cpu-protect traffic-class id 7
%*********traffic class     bandwidth(kbps)**********
            7                 312
```

43.2.5 配置 CPU 端口的最大速率

在配置模式下，按如下步骤配置 CPU 端口最大速率：

命令	作用
Ruijie(config)# **cpu-protect cpu bandwidth** *bandwidth_value*	配置 CPU 端口的最大速率（kbps），*bandwidth-value* 取值范围为 64－1000000(kbps)。

您可以通过 **no cpu-protect cpu** 命令来恢复 cpu 端口的最大速率值。

以下是配置 CPU 端口最大速率为 2000（kbps）的实例：

```
Ruijie#configure terminal
Ruijie(config)#cpu-protect cpu bandwidth 2000
Ruijie(config)#end
Ruijie#show cpu-protect cpu
```

```
%cpu port bandwidth: 2000(kpbs)
```

43.3 显示 CPU Protect 配置

我们提供的可查看相关 CPU Protect 的信息如下：

- 查看每种类型报文对应的队列
- 查看每个队列的最大速率
- 查看 CPU 端口最大速率
- 查看报文的最大速率
- 查看地址学习风暴控制

43.3.1 查看每种类型报文对应的队列

在特权模式下使用如下命令查看每种类型报文对应的队列：

命令	作用
Ruijie# **show cpu-protect type** { **bpdu** | **arp** | **igmp** | **dot1x** | **gvrp** | **dhcp** | **unicast** | **multicast** | **broadcast** | **error_ttl** | **other** }	查看每种类型报文对应的队列。

以下例子使用 **show cpu-protect type all** 命令查看所有报文对应的队列：

```
%**********packet type      traffic-class**********
              bpdu              6
              arp               5
              igmp              3
              dot1x             3
              gvrp              3
              dhcp              2
              unicast           4
              multicast         1
              broadcast         0
              error_ttl         0
              co-operate        6
              other             0
```

43.3.2 查看每个队列的最大速率

在特权模式下使用如下命令查看每个队列的最大速率：

命令	作用

Ruijie# **show cpu-protect traffic-class id** *id_num*	查看每个队列的最大速率。id_num 的取值范围为 0－7
Ruijie# **show cpu-protect traffic-class all**	查看所有队列的最大速率

以下例子使用 **show cpu-protect traffic-class all** 命令查看所有队列的最大速率：

```
Ruijie# show cpu-protect traffic-class all
%*********traffic class     bandwidth(kbps)**********
            0                1000
            1                1000
            2                1000
            3                1000
            4                1000
            5                1000
            6                1000
            7                100000
```

43.3.3 查看 CPU 端口最大速率

在特权模式下使用如下命令查看 CPU 端口最大速率：

命令	作用
Ruijie# **show cpu-protect cpu**	查看 CPU 端口最大速率。

以下例子是查看 CPU 端口最大速率：

```
Ruijie# show cpu-protect cpu
%cpu port bandwidth: 100000(kbps)
```

44 防攻击的系统保护配置

44.1 概述

众所周知，许多黑客攻击、网络病毒入侵都是从扫描网络内活动的主机开始的。因此大量的扫描报文急剧占用了网络带宽，导致网络通讯无法正常进行。

为此，锐捷三层设备提供了防扫描的功能，用以防止黑客扫描和类似“冲击波”病毒的攻击，还能减少三层设备的 CPU 负担。

目前发现的扫描攻击主要有两种：

1) 目的 IP 地址变化的扫描，我们称为“Scan Dest Ip Attack”。这种扫描是最危害网络的，不但消耗网络带宽，增加设备的负担，而且更是大部分黑客攻击手段的起手。

2) 目的 IP 地址不存在，却不断的发送大量报文，我们称为“Same Dest Ip Attack”。这种攻击主要是针对设备 CPU 的负担来设计。对三层设备来说，如果目的 IP 地址存在，则报文的转发会通过交换芯片直接转发，不会占用设备 CPU 的资源，而如果目的 IP 不存在，设备 CPU 会定时的尝试连接，而如果大量的这种攻击存在，也会消耗着 CPU 资源。当然，这种攻击的危害比第一种小得多了。

对于以上这两种攻击，可以通过在锐捷设备的接口上调整相应的攻击阈值、攻讦主机隔离时间等参数来减轻网络或设备的负担。管理员可以根据具体网络状况来细化设备的管理配置。如果每个接口的配置都一样，管理员也可通过 Interface Range 功能进行一批端口的设置。

44.2 防攻击的系统保护配置

防攻击的系统保护配置工作在设备的全局配置模式下完成，在进行防攻击的系统保护配置前，请先进入全局配置模式。

44.2.1 IP 防扫描配置任务列表

- 打开接口的防攻击系统保护功能
- 设置非法攻击 IP 的隔离时间
- 设置判断一个 IP 是否为非法攻击 IP 的阈值
- 设置监控的 IP 的最大数
- 设置不进行监控的特例 IP
- 清除已被隔离的 IP 隔离状态

- 查看防攻击的系统保护的相关信息

44.2.2 打开接口的防攻击系统保护功能

您可以在接口模式下打开系统保护。系统保护功能只支持物理端口。

命令	含义
configure terminal	进入全局配置模式。
interface *interface-id*	进入该 **interface** 的配置模式。 合法的 **interface** 包括物理端口
system-guard enable	打开系统保护功能
end	退回到特权模式。
show system-guard	核对配置条目
copy running-config startup-config	保存配置。

如果您要关闭该接口下的系统保护，请在接口模式下用 **no system-guard** 进行设置。

44.2.3 设置非法攻击 IP 的隔离时间

非法攻击 IP 的隔离时间基于端口，您可以在接口模式下配置非法攻击用户的隔离时间。隔离这段时间后该 IP 会自动恢复通讯。

命令	含义
configure terminal	进入全局配置模式。
interface *interface-id*	进入该 **interface** 的配置模式。 合法的 **interface** 包括物理端口
system-guard isolate-time *seconds*	配置非法用户的隔离时间。 取值范围为 30 秒到 3600 秒，缺省值为 120 秒。
end	退回到特权模式。
show system-guard	核对配置条目
copy running-config startup-config	保存配置。

如果您要恢复隔离时间的缺省值，请在接口模式下用 **no system-guard isolation-time** 进行设置。

另外，当非法用户被隔离时，设备会发送一个 LOG 记录到日志系统中，以备管理员查询，非法用户隔离解除时也会发送一个 LOG 通知。

44.2.4 设置判断一个 IP 是否为非法攻击 IP 的阈值

有两种攻击方法都可能会影响设备的性能：

1. 对一批 IP 网段进行扫描。
2. 对某个不存在的 IP 不断的发 IP 报文进行攻击。

对此我们设备做了这两个限制，在用户发送的一批报文中，只要其中一条超出了管理员控制的报文限制，就认为该用户是非法攻击者，即把它隔离。非法攻击 IP 的判断阈值也基于端口，您可以在接口模式下进行配置。

命令	含义
configure terminal	进入全局配置模式。
interface *interface-id*	进入该 **interface** 的配置模式。 合法的 **interface** 包括物理端口
system-guard same-dest-ip-attack-packets *number*	配置对某个不存在的 IP 不断的发 IP 报文进行攻击的最大阈值。 取值范围为每秒 0-2000 个报文，缺省值为 20 个。设置成 0 表示不对这种攻击进行监控。
system-guard scan-dest-ip-attack-packets *number*	配置对一批 IP 网段进行扫描攻击的最大阈值。 取值范围为每秒 0-1000 个报文，缺省值为 10 个。设置成 0 表示不对这种攻击进行监控。
end	退回到特权模式。
show system-guard	核对配置条目
copy running-config startup-config	保存配置。

注意：

值设置得越小，判断攻击的主机的准确度就可能越差，容易误隔离正常上网的主机，建议管理员根据实际的网络环境的安全程度配置相应的阈值。

对于第二攻击方法，三层交换机的硬件都能够自动过滤多余的攻击报文，所以一般情况下，交换机无法检查到第二种的攻击。但是 Sys-Guard 功能仍旧有效，在硬件容量满等极端情况下，交换机无法自动过滤攻击报文的时候，Sys-Guard 功能会成功交换机的第二道防线，保护交换机的 CPU 不受攻击。

如果您要恢复相应参数的缺省值，可以在接口模式下用 **no system-guard same-dest-ip-attack-packets** 和 **no system-guard scan-dest-ip-attack-packets** 进行设置。

44.2.5 设置监控的 IP 的最大数

您可以设置设备监控攻击主机的最大数。一般来说，这个数目保持在“实际运行的主机数的%20”左右即可。但是，如果您发现被隔离的主机数已经达到或接近监控的主机数最大数，那可以扩大监控主机的数目，以达到更好的保护系统的要求。

您可以通过以下步骤设置监控攻击主机的最大数目：

命令	含义
configure terminal	进入全局配置模式。
system-guard detect-maxnum *number*	设置监控主机的最大数，此值基于线卡。取值范围为 1-500，缺省值为 100。
end	退回到特权模式。
show system-guard	核对配置条目
copy running-config startup-config	保存配置。

注意:

您如果把监控主机的数目改成比原来小，会引起当前监控的主机数据清空。当被隔离的 IP 数目很多时，系统可能会提示“Chip Resource Full”，这是因为设备隔离了较多的用户，导致设备的硬件芯片资源占满（根据实际的设备运作及 ACL 设置，这个数目大约是每端口可隔离 100－120 个 IP 地址），这时这些用户并没有实际的被隔离，管理员需要采取其他措施来处理这些攻击者。

如果您要恢复监控主机最大数目的缺省值，可以在全局配置模式下使用 **no system-guard detect-maxnum** 进行设置。

44.2.6 设置不进行监控的特例 IP

您可以设置不进行监控的特例 IP，符合该特例的 IP 报文将允许被发往 CPU。

命令	含义
configure terminal	进入全局配置模式。
system-guard exception-ip *ip mask*	添加防攻击功能的特例 IP 码.最多支持 255 条特例 IP 表项
end	退回到特权模式。
show system-guard exception-ip	显示所有的特例 IP 表项
copy running-config startup-config	保存配置。

在全局配置模式下，可以使用该命令的 **no** 选项删除一条特例 IP 表项。使用命令的 **no** 和 **all-eip** 选项可以删除所有特例 IP

例如删除所有特例 ip：

```
Ruijie(config)# no system-guard exception-ip all-eip
```

或删除单条特例 ip：

```
Ruijie(config)# no system-guard exception-ip 192.168.5.145/32
```

注意：

对于已经被隔离的 IP，即使该 IP 在配置的特例 IP 范围内，在该 IP 被老化之前仍会处于被隔离状态，如果您想要允许该 IP 的报文发往 CPU，可以用 **clear system-guard** 命令来解除对该 IP 的隔离。

S3760 系列允许任意 IP 地址配置成特例 IP，如：127.0.0.0。

44.2.7 清除已被隔离的 IP 隔离状态

已被隔离的用户会在一段时间后自动解除隔离，如果你要手动清除该用户，可以在特权模式下用以下命令清除。

命令	含义
clear system-guard [**interface** *interface-id* [**ip-address** *ip-address*]]	清除已被隔离用户。 其中 **clear system-guard** 表示清除所有已被隔离用户； **clear system-guard interface** *interface-id* 表示清除该端口下的所有用户； **clear system-guard interface** *interface-id* **ip-address** *ip-address* 表示清除该接口下的指定 IP 用户。

44.2.8 查看系统保护的相关信息

44.2.8.1 查看系统保护的相关信息

使用 **show system-guard** 查看系统保护的配置参数：

命令	含义
show system-guard [**interface** *interface-id*]	查看系统保护配置参数

下面是一个例子：

```
Ruijie# show system-guard
detect-maxnum number   : 100            //设备监控的主机最大数目
isolated host number   : 11              //设备已隔离的主机数目
inteface  state   isolate time  same-attack-pkts
scan-attack-pkts
-----------------  --------------------   --------------
Fa 0/1    ENABLE    120             20                 10
Fa 0/2    DISABLE   110             21                 11
……
Ruijie# show system-guard interface Fa 0/1
detect-maxnum number   : 100            //设备监控的主机最大数目
isolated host number   : 11              //设备已隔离的主机数目
intefacestate solate time ame-attack-pkts   scan-attack-pkts
-----------------   -----------------   ---------------
Fa 0/1     ENABLE     120         20                 10
```

44.2.8.2 查看系统保护被隔离的 IP 的信息

命令	含义
show system-guard isolate-ip [**interface** *interface-id*]	查看 IP 防扫描各接口被隔离的 IP 的信息

```
Ruijie# show system-guard isolated-ip

interface ip-address    isolate reason   remain-time(second)
-----------------   -----------------  -----------------
Fa 0/1     192.168.5.119   scan ip attack      110
Fa 0/1     192.168.5.109   same ip attack      61
```

以上几栏分别表示：已隔离的 IP 地址出现的端口、已隔离的 IP 地址，隔离原因，隔离的剩余时间。

44.2.8.3 查看正在被监控的用户

命令	含义
show system-guard detect-ip [**interface** *interface-id*]	查看正在被监控的 IP 情况

```
Ruijie# show system-guard detect-ip
interface ip-address ame ip attack packets  scan ip attack
packets
------------------   ------------------  ----------------
Fa 0/1     192.168.5.118         0              8
```

```
Fa 0/1    192.168.5.108        12              2
```

44.2.8.4 显示不进行监控的特例 IP

显示在防攻击功能中允许访问设备的特例 IP

命令	含义
show system-guard exception-ip	查看所有的特例 IP

```
Ruijie# show system-guard exception-ip
Exception IP Address    Exception Mask
--------------------    ---------------
192.168.5.145           255.255.255.0
192.168.4.11             255.255.255.0
```

45 GSN 配置

45.1 锐捷 GSN 安全方案简介

锐捷安全解决方案由以下四个元素组成：

1) 锐捷安全策略管理平台（RG Security policy Management Platform）
2) 锐捷安全客户端（RG Security Agent）
3) 锐捷安全修复系统（RG Restore System）
4) 锐捷安全设备（RG Security Switch）

45.1.1 锐捷安全策略管理平台（RG SMP）

锐捷安全策略管理平台通过配置策略，来决定是否允许或者禁止某种条件范围内的数据报文通过安全设备进行传输。安装策略就是将策略设置到设备上，卸载策略就是将策略从设备上移除。

45.1.2 锐捷安全客户端

锐捷安全客户端是运行在企业网络中的每一台接入网络的主机上的一个软件，它负责收集客户端信息、认知用户的网络行为、监控客户机的网络通讯和安全状态并将收集到的信息发送到安全策略管理平台以便管理员针对性地制定相应的安全策略。同时安全客户端也将自动地从安全策略管理平台中下载新的安全策略并在本地执行指定的安全策略。

45.1.3 锐捷安全修复系统

安全修复系统对异常行为做如下操作：

对于不符合企业安全策略的用户，管理员在安全策略管理平台上预先配置相应的策略，这些策略可屏蔽这些“非法用户”的绝大部分的网络访问权限，只留下一条绿色的安全通道。这个安全通道连接到的主机只能是企业安全策略升级服务器。包括：Windows 补丁升级服务器、防病毒软件的病毒升级库服务器，或是企业的其他升级服务器。

当安全客户端在检测到自己的安全策略不符合管理平台制定的安全等级之后，安全代理会立即将自己的安全日志上传至安全策略管理平台，安全策略管理平台根据接收到安全客户端传来的报警日志从预先配置好的策略集中选择相应的一条，将此条策略下发到所有的安全设备，安全设备接受最新的策略配置后立即应用配置，使该报警的用户只能根据策略服务器规定的恢复动作访问指定的升级服务器，自动安装

这些补丁程序。

当用户已经完成了策略服务器规定的一切恢复动作之后，安全客户端会再次对客户端操作平台进行安全检测，如果此时客户端满足所有的安全策略集，安全客户端会通知安全策略管理平台解除对此客户端的访问列表限制，将客户端设置成正常的用户。

45.1.4 锐捷安全设备

做为锐捷安全解决方案的一部分，锐捷安全设备负责从锐捷安全策略管理平台上接收策略，并且实现策略的安装，并根据安装的策略对用户进行控制。

45.2 GSN 安全设备配置

45.2.1 配置设备支持安全方案的开关

缺省情况下，设备的 GSN 安全方案的开关是关闭的

命令	说明
configure terminal	进入配置模式
[**no**] **security gsn enable**	打开 GSN 全局配置开关

下边是配置打开设备的 GSN 功能：

```
Ruijie# configure terminal
Ruijie(config)# security gsn enable
```

45.2.2 配置同 SMP server 之间的通讯

设备必须配置 SMP Server 的 IP 地址以及安全认证名才实现同 SMP Server 的通讯。

命令	说明
Configure terminal	进入配置模式
[**no**] **security** { [**v1** | **v2**] **community** *community* | **v3** **user** *username* }	配置和 smp 服务器通信的安全名.该命令支持 snmp v1、v2、v3.默认使用没有配置通讯 community。security v1 community 和 security community 效果一样,都是配置 v1，只是为了方便用户配置.如果选择了 v3,则需要在 snmp-server 命令下配置相应的 v3 用户,相关配置命令请参看《SNMP 配置》章节。
[**no**] **smp-server host** *ip-address*	配置 SMP 服务器地址

说明：

在配置 security v3 user 时,需要在 SNMP 部分配置对应的 v3 user。

45.2.3 配置安全事件传输最小时间间隔

为了避免非法用户通过伪造安全事件，频繁地发送安全事件信息对安全设备和 SMP 进行攻击，用户可以通过配置安全事件传输最小时间间隔来限制用户通告安全事件的最小时间间隔。

根据实际情况，用户可以通过以下命令设置安全事件传输最小时间间隔：

命令	说明
Configure terminal	进入配置模式
[no] security event interval *interval*	配置安全事件传输最小时间间隔配置，*interval*：1-65535s 缺省情况下该时间间隔为 5 秒，

45.2.4 配置端口支持地址绑定开关

用户可以通过该命令来控制是否在端口产生地址绑定策略，当认证端口连接着多个用户时管理员必须打开该认证端口的地址绑定：

命令	说明
Configure terminal	进入配置模式
interface *interface*	进入接口配置模式
[no] security address-bind enable	配置端口启用地址绑定策略

说明：

只有全局 GSN 支持功能打开并且配置的端口是认证端口时该功能才能起作用。另外需要注意的是当使用该功能时，需要关闭 802.1x 的 IP 授权功能，否则会影响安全策略的实际运行效果。

45.3 GSN 配置显示

45.3.1 显示 smp server

您可以通过以下步骤显示 smp sverver 内容

命令	说明
show smp-server	显示 smp server

例如：

```
Ruijie# show smp-server
SMP-Server IP:192.168.217.220
```

45.3.2 显示 security event interval

您可以通过以下步骤显示 policy-map 内容

命令	说明
show security event interval	查看安全事件传输最小时间间隔的配置

例如：

```
Ruijie# show security event interval
Event sending interval(Seconds):5
```

45.4 GSN 配置使用相关注意事项

45.4.1 GSN 支持的表项数目

由于 GSN 的策略安装实现是通过设置到硬件，由硬件过滤来实现的，所以 GSN 能够实际支持的策略数目由于各产品的芯片不同而不同，另外由于 GSN 使用的硬件表项资源有可能被其他模块所占用，所以当开启相关功能（如端口的防 ARP 欺骗功能等）时也会减少可用的表项数，GSN 支持的表项数也相应减少。为了能够支持更多的动态策略，增强 GSN 的控制能力，所以在开启 GSN 时尽量不要开启其他耗用硬件表项的功能。

45.4.2 与 GSN 使用冲突的一些功能

由于 GSN 应用的特性，GSN 与下边的一些功能存在着使用上的冲突，需要在应用时加以注意，避免同时打开出现功能异常。

1) GSN 不能与 1x 的 ip 授权共用。

2) GSN 不能同端口安全共用。

3) GSN 不支持在 AP 口上安装策略，而安装了策略的端口不能加入 AP 口，如果把已经安装了 GSN 策略的端口加入 AP 口可能会导致对应的口的应用功能出现错误，如果一个已经安装了策略的端口需要加入 AP 口需要先删除对应端口上已经安装的策略。

45.4.3 GSN 使用的其他注意事项

由于 GSN 最终实现是通过硬件过滤来实现用户的配置，所以对于下边的一些配置可能会影响 GSN 的使用：

全局 IP+MAC 绑定：如配置了某个全局 IP+MAC 的绑定后，放过了某个 MAC 的用户，而 GSN 又设置对此 MAC 的用户进行隔离，这时候由于全局 IP+MAC 绑定已经放过用户，从而导致 GSN 的隔离可能失效。

ACL：当 ACL 设置的表项内容与 GSN 下放的策略出现冲突时也会导致 GSN 的功能失效，在 GSN 启用时建议不要启用 ACL。

46 动态 ARP 检测配置

46.1 了解 DAI

DAI 的全称是 Dynamic ARP Inspection，中文名为动态 ARP 检测。即对接收到的 ARP 报文进行合法性检查。不合法的 arp 报文会被丢弃。

46.1.1 了解 ARP 欺骗攻击

由于 ARP 协议本身的缺陷，ARP 协议本身不对收到的 ARP 报文进行合法性检查。这就造成了攻击者利用协议的漏洞轻易的进行 ARP 欺骗攻击。这其中，最典型的就是中间人攻击。中间人攻击描述如下：

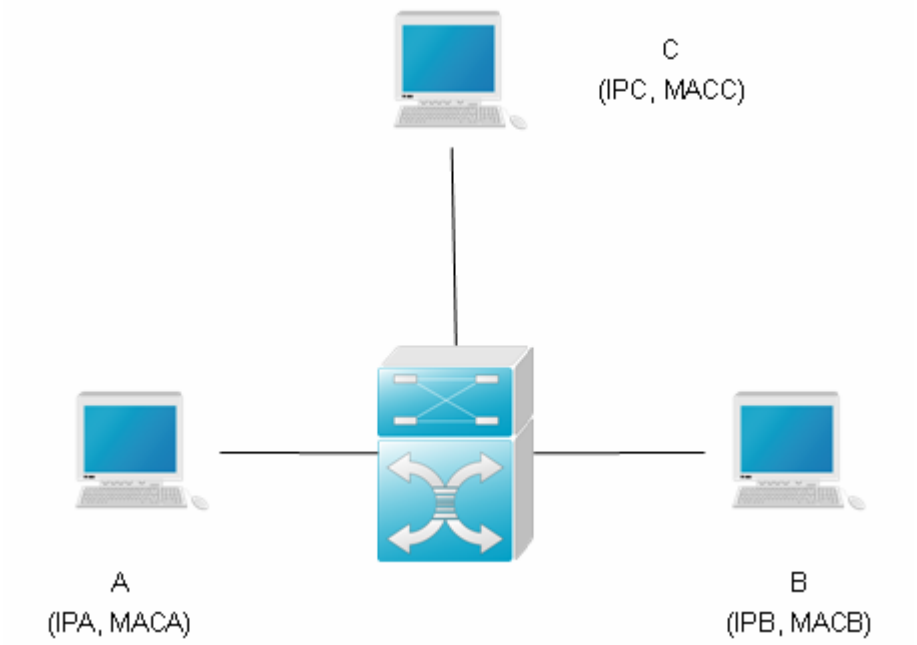


图 1

如图所示：设备 A,B,C 均连接在锐捷设备上，并且它们位于同一个子网，它们的 IP 和 MAC 分别表示为（IPA, MACA）(IPB, MACB) (IPC, MACC),当设备 A 需要和设备 B 进行网络层通信时，设备 A 将会在子网内广播一个 ARP 请求，询问设备 B 的 MAC 值。当设备 B 接收到此 ARP 请求报文时，会更新自己的 ARP 缓存，使用的是 IPA 和 MACA,并发出 ARP 应答。设备 A 收到此应答后，会更新自己的 ARP 缓存，使用的是 IPB 和 MACB。

在这种模型下，设备 C 可以使设备 A 和设备 B 中的对应 ARP 表项对应关系不正确。使用的策略是，不断向网络中广播 ARP 应答，此应答使用的 IP 地址是 IPA/IPB,而 MAC 地址是 MACC，这样，设备 A 中就会存在 ARP 表项(IPB、MACC),设备 B

中就会存在 ARP 表项(IPA, MACC).这样，设备 A 和 B 之间的通信就变成了和设备 C 之间的通信，而设备 A、B 对此都一无所知。设备 C 充当了中间人的角色，只需要把发给自己的报文做合适的修改，转给另一方即可。这就是有名的中间人攻击。

46.1.2 了解 DAI 和 ARP 欺骗攻击

DAI 确保了只有合法的 ARP 报文才会被设备转发。它主要执行以下几个步骤：

- 在打开 DAI 检查功能的 VLAN 所对应的非信任端口上拦截住所有 ARP 请求和应答报文
- 在做进一步相关处理之前，根据 DHCP 数据库的设置，对拦截住的 ARP 报文进行合法性检查。
- 释放没有通过检查的报文。
- 检查通过的报文继续做相应的处理，送给相应的目的地。

ARP 报文是否合法的依据是 DHCP snooping binding 数据库，具体请参考相应的配置指导《DHCP snooping 配置》。

46.1.3 接口信任状态和网络安全

基于设备上每一个端口的信任状态，对 ARP 报文作出相应的检查，从受信任端口接收到的报文将跳过 DAI 检查，被认为是合法的 ARP 报文； 而从非信任端口接收到的 ARP 报文，将严格执行 DAI 检查。

在一个典型的网络配置中，应该将连接到网络设备的二层端口设置为受信任端口，连接到主机设备的二层端口设置为非信任端口。

注意:

将一个二层端口错误的配置成非信任端口可能会影响到网络正常通信。

具体配置命令请参考 ip arp inspection trust, show ip arp inspection interface。

46.1.4 限制 ARP 报文的速率

由于设备执行 DAI 合法性检查，需要消耗一定的 CPU 资源，所以对 ARP 报文限速，即限制每秒钟接收的 ARP 报文的数量，可以有效避免针对 DAI 功能而产生的拒绝服务攻击。默认情况下，非信任端口每秒钟 ARP 报文的最大个数是 15。信任端口不受此限制。可以在二层接口配置模式下通过 ip arp inspection limit-rate 来配置此速率限制。

具体配置命令请参考 ip arp inspection limit-rate。show ip arp inspection interface

46.2 配置 DAI

DAI 是一个基于 **ARP** 协议的安全过滤技术，简而言之就是配置一系列的过滤策略使得经过设备的 ARP 报文的合法性得到比传统更加有效的检验。

要使用 DAI 相关功能，可选择性地执行以下各项任务：

- 启用指定 VLAN 的 DAI 功能（必须）
- 设置端口的信任状态（可选）
- 设置端口的 ARP 报文最大接收速率（可选）
- DHCP snooping database 相关配置（可选）

46.2.1 启用指定 VLAN 的 DAI 报文检查功能

缺省情况下，所有 VLAN 的 DAI 报文检查功能是关闭的。

如果没有启用了 VLAN vid 的 DAI 报文检查功能，vlan-id = vid 的 ARP 报文会跳过 DAI 相关的安全检查（不会跳过 ARP 报文限速）。

可以通过 **show ip arp inspection vlan** 查看所有 VLAN 是否启用了 DAI 报文检查功能

要配置 VLAN 的 DAI 报文检查功能，在接口配置模式中执行以下命令：

命令	作用
Ruijie(config)**# ip arp inspection vlan** *vlan-id*	启用 VLAN *vlan-id* 的 DAI 报文检查功能开关
Ruijie(config)**# no ip arp inspection vlan** [*vlan-id*]	关闭 VLAN *vlan-id* 的 DAI 报文检查功能开关省略 *vlan-id* 时关掉所有 VLAN 的 DAI 报文检查功能

46.2.2 设置端口的信任状态

此命令应用在二层接口配置模式，且此二层接口为一个 SVI 的成员口。

缺省情况下，所有二层端口都是不可信任的。

如果端口是可信任的，ARP 报文将跳过进一步的检查，否则，会使用 DHCP snooping 数据库的信息来检查当前 ARP 报文的合法性

要设置端口信任状态，在接口配置模式中执行以下命令：

命令	作用
Ruijie(config-if)**# ip arp inspection trust**	设置端口是可信任的
Ruijie(config-if)**# no ip arp inspection trust**	设置端口是不可信任的

46.2.3 设置端口的 ARP 报文最大接收速率

此命令应用在二层接口模式，且此二层接口为一个 SVI 的成员口。

缺省情况下，每一个非信任交换端口的默认接收 ARP 报文速率为每秒 15 个 ARP 报文。信任交换端口默认不做限制。

如果 1 秒钟内此端口收到的 ARP 报文超过限制值，后面收到的报文会被丢弃，不会做进一步处理。

可以通过 **show ip arp inspection interface** 命令查看各二层接口的速率限制。

要设置端口 ARP 报文最大接收速率，在接口配置模式中执行以下命令：

命令	作用
Ruijie(config-if)**# ip arp inspection limit-rate{** *<1-2048>* **| none}**	设置端口 ARP 报文最大接收速率，单位：报文/秒 **none**：表示不限制
Ruijie(config-if)**# no ip arp inspection limit-rate**	恢复缺省值

46.2.4 DHCP snooping database 相关配置

参考《DHCP Snooping 配置》。

如果没有配置 DHCP Snooping database，则所有 ARP 报文通过检查

46.3 显示 DAI 配置

46.3.1 显示 VLAN 是否启用 DAI 功能

要显示各 VLAN 的启用状态，在全局配置模式中执行以下命令：

命令	作用
Ruijie(config)**# show ip arp inspection vlan**	显示各 VLAN 的启用状态

46.3.2 显示各二层接口 DAI 配置状态

要显示各二层接口 DAI 配置状态，在全局配置模式中执行以下命令：

命令	作用

Ruijie(config)# **show ip arp inspection interface**	显示各二层接口的 DAI 配置（包括信任状态，和速率限制）

46.4 DAI 典型配置用例

46.4.1.1 拓扑图

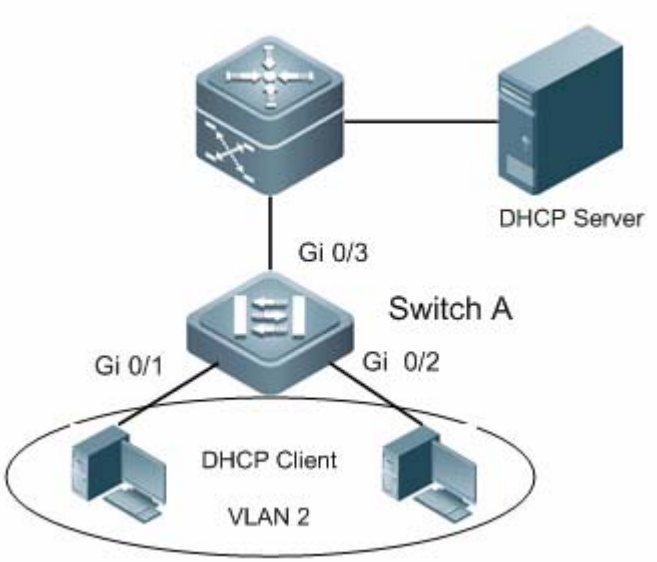


图 12 DHCP 部署环境

46.4.1.2 应用需求

如上图所示，用户 PC 的 IP 地址是 DHCP 服务器自动分配的，为了保证用户能够正常上网，有如下要求：

1. 用户 PC 只能从指定的 DHCP 服务器获取 IP 地址，不允许私设 DHCP 服务器。
2. 用户 PC 只能使用合法 DHCP 服务器分配的 IP 地址上网，不允许随意设置 IP 地址。

46.4.1.3 配置要点

- 配置要点

1. 在接入交换机(本例为 Switch A)上启用 DHCP Snooping 并将连接合法 DHCP 服务器的上链口（本例为 GigabitEthernet 0/3）设置为信任口可满足第一个需求。
2. 在接入设备（本例为 Switch A）启用 DHCP Snooping 基础上再开启 DAI，可满足第二需求。

- 注意事项

1、在汇聚或核心交换上如有其他 PC 接入并存在私设 DHCP 服务器可能，也需要开启 DHCP Snooping。

2、接入交换机下接用户比较多，应将接入设备的上链口（本例为 Switch A 的 GigabitEthernet 0/3）设置为 DAI 的信任口，降低 DAI 对系统资源的消耗。

46.4.1.4 配置步骤

- 配置 Switch A

第一步，设置直连用户 PC 的端口的 VLAN。

```
Ruijie#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Ruijie(config)#interface range gigabitEthernet 0/1-2
Ruijie(config-if-range)#switchport access vlan 2
```

第二步，开启 DHCP SNOOPING 功能。

```
Ruijie(config-if-range)#exit
Ruijie(config)#ip dhcp snooping
```

第三步，在对应的 VLAN 上开启 DAI 功能。

```
Ruijie(config)#ip arp inspection vlan 2
```

第四步，将上链口设置为 DHCP SNOOPING 信任口

```
Ruijie(config)#interface gigabitEthernet 0/3
Ruijie(config-if-GigabitEthernet 0/3)#switchport mode trunk
Ruijie(config-if-GigabitEthernet 0/3)#ip dhcp snooping trust
```

第五步，将上链口设置为 DAI 信任口。

```
Ruijie(config-if-GigabitEthernet 0/3)#ip arp inspection trust
```

46.4.1.5 配置验证

第一步，确认配置是否正确，关注点为 DHCP Snooping/DAI 是否启用，信任接口是否正确。

```
Ruijie#show running-config
ip dhcp snooping
!
ip arp inspection vlan 2
```

```
!
interface GigabitEthernet 0/1
 switchport access vlan 2
!
interface GigabitEthernet 0/2
 switchport access vlan 2
!
interface GigabitEthernet 0/3
 switchport mode trunk
 ip dhcp snooping trust
 ip arp inspection trust
```

第二步，查看 DHCP SNOOPING 的使能状态以及对应的信任端口，关注点为上链口是否设置为可信任接口。

```
Ruijie#show ip dhcp snooping
Switch DHCP snooping  status                    :   ENABLE
DHCP snooping  Verification of hwaddr status    :   DISABLE
DHCP snooping database write-delay time         :   0 seconds
DHCP snooping option 82 status                  :   DISABLE
DHCP snooping Support bootp bind status         :   DISABLE
Interface                    Trusted      Rate limit (pps)
------------------------      -------      ----------------
GigabitEthernet 0/3           YES          unlimited
```

第三步，查看 DAI 状态，关注点为对应的 VLAN 的使能情况和上链口是否设置为可信任接口。

```
Ruijie#show ip arp inspection vlan
Vlan    Configuration
----    -------------
2       Enable
Ruijie#show ip arp inspection interface
      Interface        Trust State
---------------------    -----------
GigabitEthernet 0/1     Untrusted
GigabitEthernet 0/2     Untrusted
GigabitEthernet 0/3     Trusted
```

如果需要查看 DHCP Snooping 形成的数据库绑定信息，可以通过 **show ip dhcp snooping binding** 命令，在此不再列举。

47 防网关 Arp 欺骗配置

47.1 概述

在二层交换机上，默认情况下 arp 报文是在本 vlan 内广播的，因此这就给针对网关的 arp 欺骗提供了机会。

针对网关的 arp 欺骗是指用户 A 发送 arp 报文请求网关的 mac 地址，这时处于同一 vlan 的用户 B 也会收到该 arp 报文，因此用户 B 可以发送 arp 响应报文，将报文的源 ip 填为网关 ip，而源 mac 填为自己的 mac 地址。用户 A 收到该 arp 响应后，就会认为用户 B 的机器就是网关，因此用户 A 通讯中发往网关的报文都将发往用户 B， 这样用户 A 的通讯实际上都被截取了，造成 arp 欺骗的效果。

因此我们可以在二层交换机上配置防网关 arp 欺骗来防止针对网关的 arp 欺骗。防网关 arp 欺骗配置后，可以在端口上检查 arp 报文的源 ip 是否是我们配置的网关 ip，如果是，则将该报文丢弃，防止用户收到错误的 arp 响应报文。 这样只有交换机上连设备能够下发网关的 ARP 报文，其它 pc 就不能发送假冒网关的 arp 响应报文。

47.2 配置防网关 arp 欺骗

47.2.1 防网关 arp 欺骗的缺省配置

缺省没有设置任何网关地址。

47.2.2 设置防网关 arp 欺骗

设置防网关 arp 欺骗地址：

命令	作用
Ruijie(config-if)**#anti-arp-spoofing ip** *ip-address*	在该端口下配置防网关 arp 欺骗，网关 ip 地址为指定 ip。

你可以在端口配置模式下通过命令 **no anti-arp-spoofing ip** *ip-address* 来将防网关 arp 配置清除。

注意：

在上链口上不能配置防网关 arp 欺骗。

配置防网关 arp 欺骗或者 arp check 后，ipv6 acl 不能再使用，反之亦然。

47.3 查看防网关 arp 欺骗信息

查看交换机的防网关 arp 欺骗信息：

命令	作用
Ruijie **#show anti-arp-spoofing**	显示所有接口下的防网关 arp 欺骗信息

配置案例

47.3.1 拓扑图

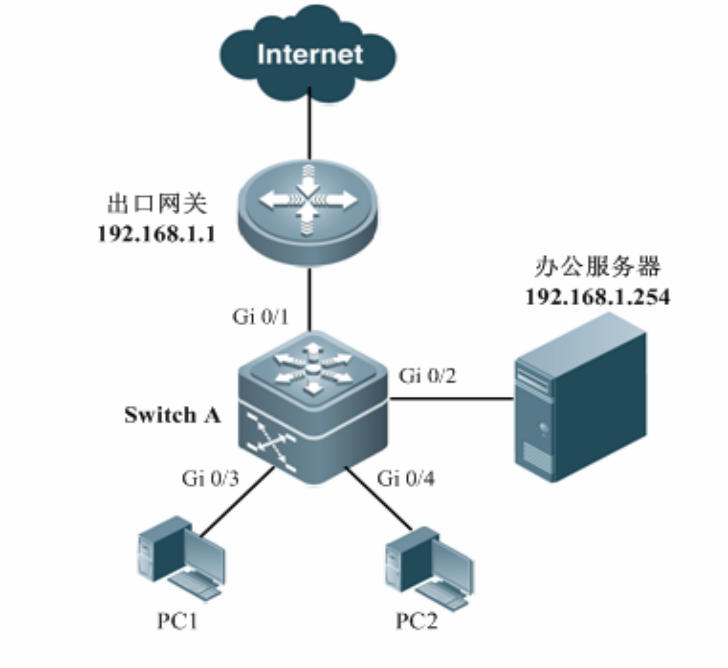


图 1 典型拓扑

47.3.2 应用需求

上图为中小型公司或单位的简化典型网络拓扑，PC 用户通过接入设备 Switch A 访问办公服务器，以及通过网关设备连接外网。如果存在非法用户冒充网关 IP 地址或服务器 IP 地址进行 ARP 欺骗，将导致其它用户无法正常上网以及访问服务器。

基于以上应用分析，组网需求如下：

- 阻断伪造网关和内网服务器 ARP 欺骗报文，保证用户能正常上网

47.3.3 配置要点

- **配置要点**

在接入交换机（本例为 Switch A）直连 PC 的端口(本例为 Gi 0/3, Gi 0/4)上启用防网关欺骗，网关地址为内网网关地址和内网服务器地址。

- **配置注意**

1、 上链口、连接出口网关或服务器的端口不能启用防网关欺骗，否则将导致源地址为网关 IP 或服务器 IP 的 ARP 报文被阻断，造成网络不通

2、 如果端口不够，需在接入交换机下挂一台 8 口的 hub（集线器），也可在接入交换机上开启防网关欺骗，缺陷是同一台 hub 上的电脑之间的网关 ARP 欺骗无法阻断。

47.3.4 配置步骤

- 配置 SwitchA

第一步，在直连电脑的端口上启用防网关欺骗

```
SwitchA# configure terminal
Enter configuration commands, one per line.  End with
CNTL/Z.
SwitchA(config)#interface range gigabitEthernet 0/2-4
SwitchA(config-if-range)# anti-arp-spoofing ip
192.168.1.1
SwitchA(config-if-range)# anti-arp-spoofing ip
192.168.1.254
```

47.3.5 配置验证

第一步，确认配置是否正确，关注点：是否启用防网关欺骗、网关地址是否正确、上链口是否启用防网关欺骗

```
SwitchA (config-if)#show running-config
interface GigabitEthernet 0/1
!
interface GigabitEthernet 0/2
 anti-arp-spoofing ip 192.168.1.1
 anti-arp-spoofing ip 192.168.1.254
!
interface GigabitEthernet 0/3
 anti-arp-spoofing ip 192.168.1.1
```

```
 anti-arp-spoofing ip 192.168.1.254
!
interface GigabitEthernet 0/4
 anti-arp-spoofing ip 192.168.1.1
 anti-arp-spoofing ip 192.168.1.254
!
```

第二步，查看防网关欺骗状态，再次确认。关注点同第一步

```
SwitchA#show anti-arp-spoofing
Anti-arp-spoofing
port          ip
-------       -------
Gi 0/2        192.168.1.1
Gi 0/2        192.168.1.254
Gi 0/3        192.168.1.1
Gi 0/3        192.168.1.254
Gi 0/4        192.168.1.1
Gi 0/4        192.168.1.254
```

第三步，如果可能，将 PC1 的 IP 地址配置为网关的 IP 地址，然后观察网关是否报告 IP 地址冲突，PC2 是否能正常上网。

如果一切正常，说明防网关欺骗配置生效。

48 IP Source Guard 配置

48.1 IP Source Guard 简介

48.1.1 理解 DHCP

在典型的 DHCP 应用环境中，DHCP 服务器负责网络中主机地址的分配和管理，网络中的主机通过向 DHCP 服务器申请合法的网络地址。通过 DHCP 技术可以便于管理员对网络地址的管理，避免地址冲突。

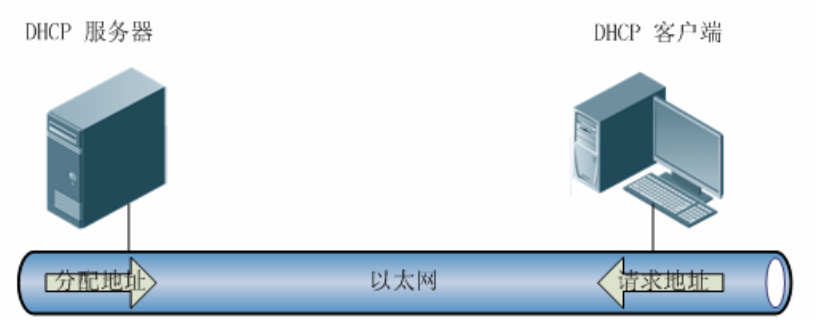


图 1 正常的 DHCP 地址分配

然而仅仅依靠服务器/客户端这种应用模型还是无法保证网络地址管理的有效性和安全性。网络中可能存在各种伪装服务器（如图 2），以及客户端的非法报文甚至攻击报文（如图 3），这些隐患对传统的 DHCP 应用模型提出了更高的安全要求。

DHCP Snooping 功能的引入解决了这一问题，在连接 DHCP 服务器与客户端的设备上开启 DHCP Snooping 功能，可以解决传统 DHCP 应用模型中的安全问题。DHCP Snooping 将网络分为两个部分：非信任网络，屏蔽该网络上所有的 DHCP 服务器响应报文，对来自该网络上的客户端请求进行安全检测；信任网络，将收到的合法的客户端请求转发到信任网络中的服务器，由指定服务器完成地址的统一管理分配。

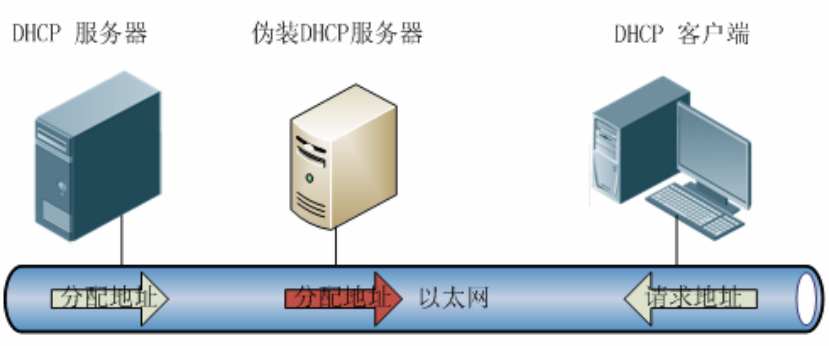


图 2 存在伪装 DHCP 服务器的网络

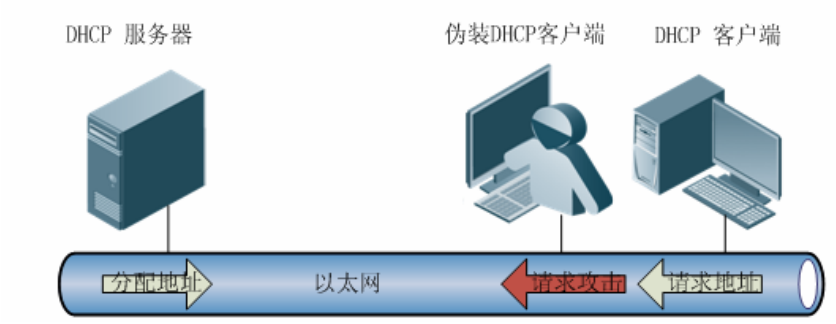


图 3 存在伪装 DHCP 客户端进行攻击的网络

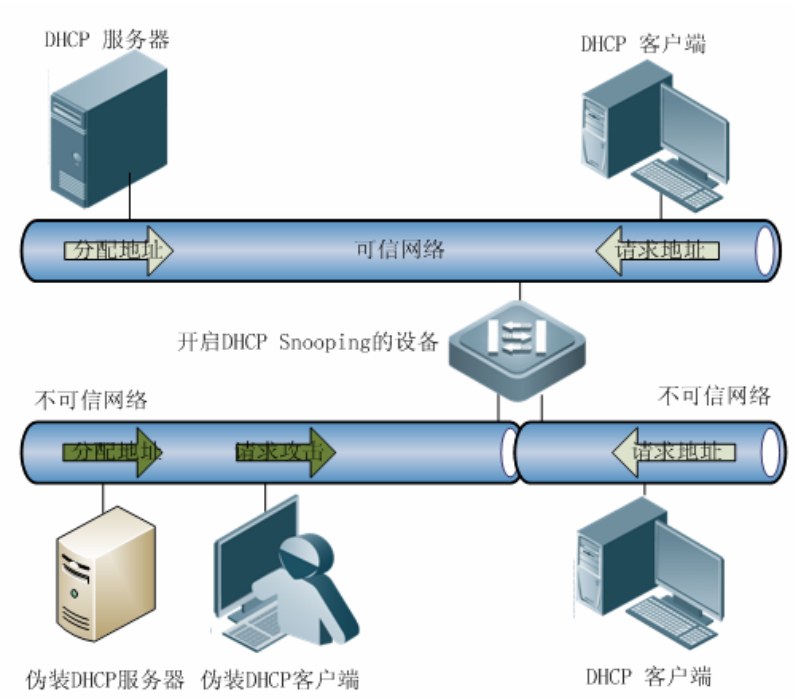


图 4 通过 DHCP Snooping 进行保护的网络

通过 DHCP Snooping 对 DHCP 报文的过滤，可屏蔽伪装服务器、拦截客户端攻击，但却无法实现对私设 IP 用户的控制。在 DHCP 网络应用中，私设 IP 用户的出现极易造成网络地址冲突，不利于网络地址的管理。为了防止在 DHCP 网络中客户端私设 IP，可以在连接服务器与客户端网络之间的设备上开启 IP Source Guard 功能。IP Source Guard 基于 DHCP Snooping 功能，在 DHCP 模型中可以有效的保证网络中 DHCP 客户端能够正常使用网络，杜绝私设 IP 用户使用网络。

48.1.2 理解 IP Source Guard

IP Source Guard 维护一个 IP 报文硬件过滤数据库，可以通过硬件对 IP 报文进行过滤，从而保证只有 IP 报文硬件过滤数据库中存在对应信息的用户才能正常使用网络，防止了用户私设 IP 地址。

IP Source Guard 之所以能够在 DHCP 应用中进行有效的安全控制，主要取决于 IP 报文硬件过滤数据库。IP 报文硬件过滤数据库以 DHCP Snooping 数据库为依据，

当启动 IP Source Guard 功能后，DHCP Snooping 数据库信息将同步到 IP Source Guard 的 IP 报文硬件过滤数据库中，这样 IP Source Guard 就可以在打开 DHCP Snooping 功能的设备上对客户端的 IP 报文进行严格过滤。

默认情况下，在端口上打开 IP Source Guard 功能后，检查所有经过该端口的 IP 报文（DHCP 报文除外）。只有通过 DHCP 获取 IP 地址的合法用户，以及管理员配置的静态绑定用户可以正常使用网络。

IP Source Guard 支持基于源 IP + 源 MAC 或者仅基于源 IP 的 IP 报文过滤策略。如果打开基于源 IP + 源 MAC 的过滤，IP Source Guard 会对所有 IP 报文的源 IP + 源 MAC 进行检测，仅允许 IP 报文硬件过滤数据库中存在的用户报文通过；而基于源 IP 的过滤，仅对报文的源 IP 地址进行检测。

48.1.3 IP Source Guard 配置的其他注意事项

IP Source Guard 功能基于 DHCP Snooping 功能，也就是说基于端口的 IP Source Guard 仅在 DHCP Snooping 控制范围内的非信任口上生效。在信任口或者非 DHCP Snooping 控制 VLAN 范围内的接口上配置该功能，将不会生效。

48.2 IP Source Guard 配置

48.2.1 配置端口的 IP Source Guard 功能

缺省情况下，端口上的 IP Source Guard 功能是关闭的。该端口下连接的所有用户都可以使用网络。打开该端口的 IP Source Guard 功能后，将会根据 IP 报文硬件过滤数据库对端口下的用户进行 IP 报文过滤。

命令	说明
Ruijie(config)# **interface** *interface-id*	进入接口配置模式
Ruijie(config-if)# **[no] ip verify source [port-security]**	打开端口上的 IP Source Guard 功能，port-security 将配置 IP 报文为基于 IP + MAC 的过滤

下边是配置打开端口 1 的 IP Source Guard 功能：

```
Ruijie(config)# interface FastEthernet 0/1
Ruijie(config-if)# ip verify source
Ruijie(config-if)# end
```

注意：

IP Source Guard 功能和 DHCP Snooping 功能结合应用，该功能仅在 DHCP Snooping 控制范围内的非信任端口上生效；

48.2.2 配置静态绑定用户

缺省情况下，设备中不存在静态绑定用户。在某些应用情况下，用户希望配置静态 IP 地址访问网络，可以通过配置静态绑定用户功能来实现。

命令	说明
Ruijie**# configure terminal**	进入配置模式
Ruijie(config)**# [no] ip source binding mac-addrees vlan** *vlan_id ip-address* **interface** *interface-id*	配置静态绑定用户

下面是添加一个静态绑定用户：

```
Ruijie# configure terminal
Ruijie(config)# ip source binding 00d0.f801.0101 vlan 1
192.168.4.243 interface FastEthernet 0/9
Ruijie(config)# end
```

48.3 IP Source Guard 其他配置命令

48.3.1 显示 IP Source Guard 过滤表项

此命令用于显示 IP Source Guard 过滤表项：

命令	说明
Ruijie**# show ip verify source** [**interface** *interface*]	显示 IP Source Guard 过滤表项

例如：

```
Ruijie # show ip verify source

Interface  Filter-type Filter-mode Ip-address  Mac-address

VLAN

-------------------- ----------- ----------- ---------------
FastEthernet    0/1    ip    active     192.168.4.243
00d0.f801.0101 1
```

48.3.2 显示 IP 报文硬件过滤数据库信息

此命令用于显示 IP 报文硬件过滤数据库信息的相关内容：

命令	说明
Ruijie**# show ip source binding** [*ip-address*] [*mac-address*] [*dhcp-snooping*] [*static*] [**vlan** *vlan-id*] [**interface** *interface-id*]	查看 IP 报文硬件过滤数据库

例如：

```
Ruijie# show ip source binding
MacAddress    IpAddress   Lease(sec)   Type   VLAN
Interface
------------------ --------------- ------------ -------------
00d0.f801.0101  192.168.4.243  infinite   static   1
FastEthernet  0/1
Total number of bindings: 1
```

48.3.3 IP Source Guard 调试开关

此命令用于打开 IP Source Guard 的调试信息开关。

命令	说明
Ruijie**# debug ip source bind**	打开/关闭 IP Source Guard 调试信息开关

例如：

```
Ruijie# debug ip source bind
```

48.4 IP Source Guard 配置用例

48.4.1 拓扑图

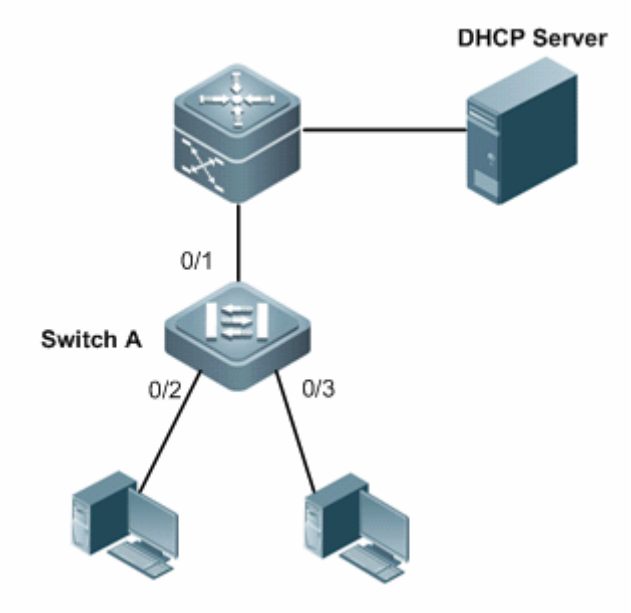


图 13 DHCP 部署环境

48.4.2 应用需求

1、用户只能使用合法 DHCP 服务器动态分配或管理员静态分配的 IP 地址访问网络，源 IP 地址与列表中的 IP 地址不匹配的 IP 报文被阻断，保证网络安全

48.4.3 配置要点

在接入设备（本例为 Switch A）上配置 IP Source Guard 和 DHCP Snooping 组合功能可满足需求：

1、 设置上链口（本例为 GigabitEthernet 0/1）为信任口，避免 DHCP 服务器欺骗

2、 在直连 PC 的端口上（本例为 GigabitEthernet 0/2 、GigabitEthernet 0/3）启用 IP Source Guard 功能

3、 管理员指定 IP 地址的用户可通过 IP Source Guard 静态绑定来实现（本例静态绑定 IP 地址为 192.168.216.4、MAC 地址为 0000.0000.0001）

48.4.4 配置步骤

- **配置 Swtich A**

第一步，开启 DHCP Snooping 功能。

```
Ruijie#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Ruijie(config)#ip dhcp snooping
```

第二步，将上链口设置为 DHCP SNOOPING 信任口。

```
Ruijie(config)#interface gigabitEthernet 0/1
Ruijie(config-if-GigabitEthernet 0/1)#ip dhcp snooping trust
Ruijie(config-if-GigabitEthernet 0/1)#exit
```

第三步，在直连 PC 的端口上开启 IP Source Guard 功能

```
Ruijie(config)#interface range gigabitEthernet 0/2-3
Ruijie(config-if-range)#ip verify source port-security
Ruijie(config-if-range)#exit
```

第四步，配置静态绑定用户

```
Ruijie(config)#ip source binding 0000.0000.0001 vlan 1
192.168.216.4 interface gigabitEthernet 0/2
```

48.4.5 配置验证

第一步，查看 Switch A 的配置情况，关注点：是否开启 DHCP Snooping、是否将上链口配置为信任口、是否在直连用户的端口上开启 IP Source Guard 功能、静态绑定用户的表项是否正确

```
Ruijie#show running-config
ip dhcp snooping
!
ip source binding 0000.0000.0001 vlan 1 192.168.216.1 interface
GigabitEthernet 0/2
!
interface GigabitEthernet 0/1
 ip dhcp snooping trust
!
interface GigabitEthernet 0/2
 ip verify source port-security
!
interface GigabitEthernet 0/3
 ip verify source port-security
```

第二步，查看 DHCP Snooping 用户绑定数据库

```
Ruijie#show ip dhcp snooping binding
Total number of bindings: 2
```

```
MacAddress        IpAddress       Lease(sec)   Type         VLAN
Interface
------------------ --------------- ------------ -------------
0013.2049.9014    192.168.216.4   86233        dhcp-snooping 1
GigabitEthernet 0/3
00e0.4c70.b7e2    192.168.216.3   86228        dhcp-snooping 1
GigabitEthernet 0/2
```

第三步，查看通过 DHCP Snooping 用户绑定数据库和静态绑定用户共同生成的 IP 硬件过滤库：

```
Ruijie#show ip source binding
MacAddress        IpAddress       Lease(sec)   Type         VLAN
Interface
------------------ --------------- ------------ -------------
0000.0000.0001    192.168.216.4   infinite     static       1
GigabitEthernet 0/2
0013.2049.9014    192.168.216.4   86176        dhcp-snooping 1
GigabitEthernet 0/3
00e0.4c70.b7e2    192.168.216.3   86171        dhcp-snooping 1
GigabitEthernet 0/2
Total number of bindings: 3
```

第四步，查看 IP Source Guard 过滤表项如下：

```
Ruijie#show ip verify source
Interface          Filter-type Filter-mode Ip-address
Mac-address    VLAN
-------------------- ----------- ----------- ---------------
-------------- --------
GigabitEthernet 0/2  ip+mac     active     192.168.216.4
0000.0000.0001  1
GigabitEthernet 0/2  ip+mac     active     192.168.216.3
00e0.4c70.b7e2  1
GigabitEthernet 0/2  ip+mac     active     deny-all
deny-all
GigabitEthernet 0/3  ip+mac     active     192.168.216.4
0013.2049.9014  1
GigabitEthernet 0/3  ip+mac     active     deny-all
deny-all
```

49 访问控制列表配置

49.1 概述

作为锐捷产品安全解决方案的一部分，使用访问控制列表提供强大的数据流过滤功能。锐捷产品目前支持以下访问列表：

- 标准 IP 访问控制列表
- 扩展 IP 访问控制列表
- MAC 访问控制列表
- MAC 扩展访问控制列表
- Expert 扩展访问控制列表
- IPV6 扩展访问控制列表

您可以根据网络具体情况选择不同的访问控制列表对数据流进行控制。

49.1.1 访问控制列表简介

ACLs 的全称为接入控制列表(Access Control Lists)，也称为访问列表（Access Lists），俗称为防火墙，在有的文档中还称之为包过滤。ACLs 通过定义一些规则对网络设备接口上的数据报文进行控制：允许通过或丢弃。按照其使用的范围，可以分为安全 ACLs 和 QoS ACLs。

对数据流进行过滤可以限制网络中的通讯数据的类型，限制网络的使用者或使用的设备。安全 ACLs 在数据流通过网络设备时对其进行分类过滤，并对从指定接口输入或者输出的数据流进行检查，根据匹配条件(Conditions)决定是允许其通过(Permit)还是丢弃(Deny)。

总的来说，安全 ACLs 用于控制哪些数据流允许从网络设备通过，Qos 策略对这些数据流进行优先级分类和处理。

ACLs 由一系列的表项组成，我们称之为接入控制列表表项(Access Control Entry：ACE)。每个接入控制列表表项都申明了满足该表项的匹配条件及行为。

访问列表规则可以针对数据流的源地址、目标地址、上层协议，时间区域等信息。

49.1.2 为什么要配置访问列表

配置访问列表的原因比较多，主要有以下一些：

- 限制路由更新：控制路由更新信息发往什么地方，同时希望在什么地方收到路由更新信息。

- 限制网络访问：为了确保网络安全，通过定义规则，可以限制用户访问一些服务（如只需要访问 WWW 和电子邮件服务，其他服务如 TELNET 则禁止），或者仅允许在给定的时间段内访问，或只允许一些主机访问网络等等。

图 1 是一个案例，在该案例中，只允许主机 A 访问财务网络，而主机 B 禁止访问。如下图 1 所示。

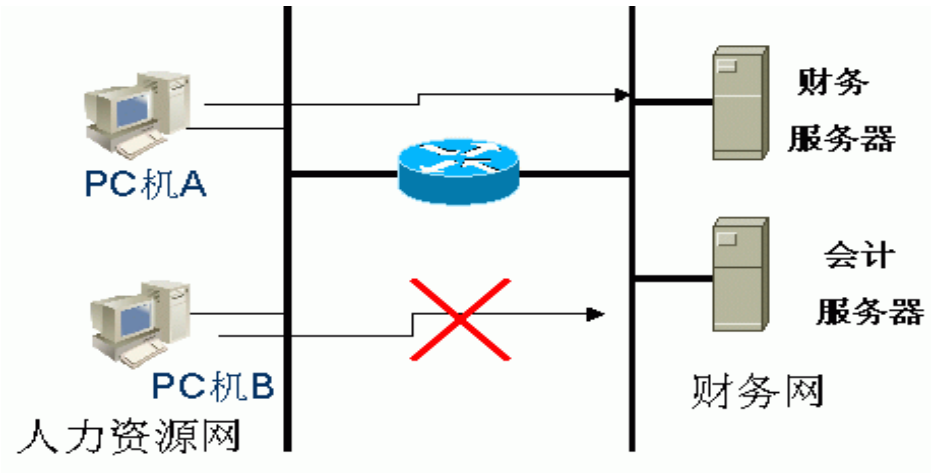


图 1　使用访问列表控制网络访问

49.1.3 什么时候配置访问列表

您可以根据需要选择基本访问列表或动态访问列表。一般情况下，使用基本访问列表已经能够满足安全需要。但经验丰富的黑客可能会通过一些软件假冒源地址欺骗设备，得以访问网络。而动态访问列表在用户访问网络以前，要求通过身份认证，使黑客难以攻入网络，所以在一些敏感的区域可以使用动态访问列表保证网络安全。

说明：

通过假冒源地址欺骗设备即电子欺骗是所有访问列表固有的问题，使用动态列表也会遭遇电子欺骗问题：黑客可能在用户通过身份认证的有效访问期间，假冒用户的地址访问网络。解决这个问题的方法有两种，一种是尽量将用户访问的空闲时间设置小些，这样可以使黑客更难以攻入网络，另一种是使用 IPSEC 加密协议对网络数据进行加密，确保进入设备时，所有的数据都是加密的。

访问列表一般配置在以下位置的网络设备上：

- 内部网和外部网（如 INTERNET）之间的设备
- 网络两个部分交界的设备
- 接入控制端口的设备.

访问控制列表语句的执行必须严格按照表中语句的顺序，从第一条语句开始比较，

一旦一个数据包的报头跟表中的某个条件判断语句相匹配，那么后面的语句就将被忽略，不再进行检查。

49.1.4 输入/输出 ACL、过滤域模板及规则

输入 ACL 在设备接口接收到报文时，检查报文是否与该接口输入 ACL 的某一条 ACE 相匹配；输出 ACL 在设备准备从某一个接口输出报文时，检查报文是否与该接口输出 ACL 的某一条 ACE 相匹配。

在制定不同的过滤规则时，多条规则可能同时被应用，也可能只应用其中几条。只要是符合某条 ACE，就按照该 ACE 定义的处理报文(Permit 或 Deny)。ACL 的 ACE 根据以太网报文的某些字段来标识以太网报文的，这些字段包括：

二层字段(Layer 2 Fields)：

- 48 位的源 MAC 地址(必须申明所有 48 位)
- 48 位的目的 MAC 地址(必须申明所有 48 位)
- 16 位的二层类型字段

三层字段(Layer 3 Fields)：

- 源 IP 地址字段(可以申明全部源 IP 地址值，或申明您所定义的子网来定义一类流)
- 目的 IP 地址字段(可以申明全部目的 IP 地址值，或申明您所定义的子网来定义一类流)
- 协议类型字段

四层字段(Layer 4 Fields)：

- 可以申明一个 TCP 的源端口、目的端口或者都申明
- 可以申明一个 UDP 的源端口、目的端口或者都申明

过滤域指的就是您在生成一条 ACE 时，根据报文中的哪些字段用以对报文进行识别、分类。过滤域模板就是这些字段组合的定义。比如，您在生成某一条 ACE 时希望根据报文的目的 IP 字段对报文进行识别、分类，而在生成另一条 ACE 时，希望根据的是报文的源 IP 地址字段和 UDP 的源端口字段，这样，这两条 ACE 就使用了不同的过滤域模板。

规则(Rules)，指的是 ACE 过滤域模板对应的值。 比如有一条 ACE 内容如下：

permit tcp host *192.168.12.2* **any eq telnet**

在这条 ACE 中，过滤域模板为以下字段的集合：源 IP 地址字段、IP 协议字段、目的 TCP 端口字段。对应的值(Rules)分别为：源 IP 地址＝Host 192.168.12.2；IP 协议＝TCP；TCP 目的端口 ＝ Telnet。

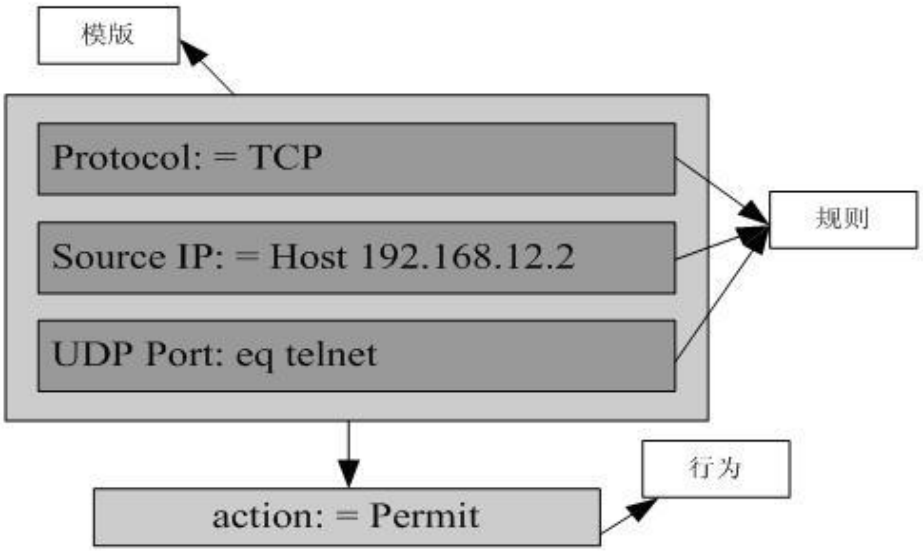


图 2　对 ACE：permit tcp host 192.168.12.2 any eq telnet 的分析

说明：

过滤域模板可以是三层字段(Layer 3 Field)和四层字段(Layer 4 Field)字段的集合，也可以是多个二层字段(Layer 2 Field)的集合，但标准与扩展的 ACL 的过滤域模板不能是二层和三层、二层和四层、二层和三层、四层字段的集合。要使用二层、三层、四层字段集合，可以应用 Expert 扩展访问控制列表（Expert ACLs）。

注意：

对于 S3760 系列，ACL 应用在 private vlan 的 SVI 输出方向时，community vlan 和 isolate vlan 上的报文不受 ACL 限制。

49.2 IP 访问列表配置

要在设备上配置访问列表，必须为协议的访问列表指定一个唯一的名称或编号，以便在协议内部能够唯一标识每个访问列表。下表列出了可以使用编号来指定访问列表的协议以及每种协议可以使用的访问列表编号范围。

协议	编号范围
标准 IP	1-99，1300 - 1999

扩展 IP	100-199，2000 - 2699

49.2.1 IP 访问列表配置指导

创建访问列表时，定义的规则将应用于设备上所有的分组报文，设备通过判断分组是否与规则匹配来决定是否转发或阻断分组报文。

基本访问列表包括标准访问列表和扩展访问列表，访问列表中定义的典型规则主要有以下：

- 源地址
- 目标地址
- 上层协议
- 时间区域

标准 IP 访问列表（编号为 1 - 99，1300 - 1999）主要是根据源 IP 地址来进行转发或阻断分组的，扩展 IP 访问列表（编号为 100 – 199，2000 - 2699）使用以上四种组合来进行转发或阻断分组的。其他类型的访问列表根据相关代码来转发或阻断分组的。

对于单一的访问列表来说，可以使用多条独立的访问列表语句来定义多种规则，其中所有的语句引用同一个编号或名字，以便将这些语句绑定到同一个访问列表。不过，使用的语句越多，阅读和理解访问列表就越来越困难。

49.2.1.1 隐含“拒绝所有数据流”规则语句

在每个访问列表的末尾隐含着一条“拒绝所有数据流”规则语句，因此如果分组与任何规则都不匹配，将被拒绝。

如下例：

access-list 1 **permit host** 192.168.4.12

此列表只允许源主机为 192.168.4.12 的报文通过，其它主机都将被拒绝。因为这条访问列表最后包含了一条规则语句：access-list 1 deny any。

又如：

Access-list 1 **deny host** 192.168.4.12

如果列表只包含以上这一条语句，则任何主机报文通过该端口时都将被拒绝。

注意：

1、在定义访问列表的时候，要考虑到路由更新的报文。由于访问列表末尾“拒绝所有数据流”，可能导致所有的路由更新报文被阻断。

i. 对于 S3760 系列，访问列表末尾没有隐含“拒绝所有数据流”，而是隐含“通

过所有数据流”语句，请注意在配置时，显式配置“拒绝所有数据流”或者“通过所有数据流”语句。

49.2.1.2 输入规则语句的顺序

加入的每条规则都被追加到访问列表的最后，语句被创建以后，就无法单独删除它，而只能删除整个访问列表。所以访问列表语句的次序非常重要。设备在决定转发还是阻断分组时，设备按语句创建的次序将分组与语句进行比较，找到匹配的语句后，便不再检查其他规则语句。

假设创建了一条语句，它允许所有的数据流通过，则后面的语句将不被检查。

如下例：

```
access-list 101 deny ip any any
access-list 101 permit tcp 192.168.12.0 0.0.0.255 eq telnet any
```

由于第一条规则语句拒绝了所有的 IP 报文，所以 192.168.12.0/24 网络的主机 Telnet 报文将被拒绝，因为设备在检查到报文和第一条规则语句匹配，便不再检查后面的规则语句。

49.2.2 配置 IP 访问列表

基本访问列表的配置包括以下两步：

1. 定义基本访问列表
2. 将基本访问列表应用于特定接口

要配置基本访问列表，有以下两种方式：

方式一 在全局配置模式下执行以下命令：

命令	功能
Ruijie(config)# **access-list** *id* {**deny** | **permit**} {src *src-wildcard* | **host** *src* | **any** } [**time-range** *tm-rng-name*]	定义访问列表
Ruijie(config)# **interface** *interface*	选择选择要应用访问列表的接口
Ruijie(config-if)# **ip access-group** *id* { **in** | **out** }	将访问列表应用特定接口

方式二 在 ACL 配置模式下执行以下命令：

命令	功能

Ruijie(config)# **ip access-list** { **standard** | **extended** } { *id* | *name* }	进入配置访问列表模式
Ruijie (config-xxx-nacl)# [*sn*] { **permit** | **deny** } {src *src-wildcard* | **host** *src* | **any** } [**time-range** *tm-rng-name*]	为 ACL 添加表项,具体内容请参见命令参考
Ruijie(config-xxx-nacl)# **exit** Ruijie(config)# **interface** *interface*	退出访问控制列表模式，选择选择要应用访问列表的接口
Ruijie(config-if)# **ip access-group** *id* { **in** | **out** }	将访问列表应用特定接口

说明:

方式一只对数值 ACL 进行配置，方式二可以对命名和数值 ACL 进行配置，还可以指定表项的优先级(在支持 ACE 优先级的设备中)。

49.2.3 显示 IP 访问列表的配置

要监控访问列表，请在特权用户模式执行以下命令：

show access-lists [*id* | *name*]

此命令可以查看基本访问列表

49.2.4 IP 访问列表示例

- **配置要求**

有两台设备 Switch A 和 Switch B，如图 1-3：

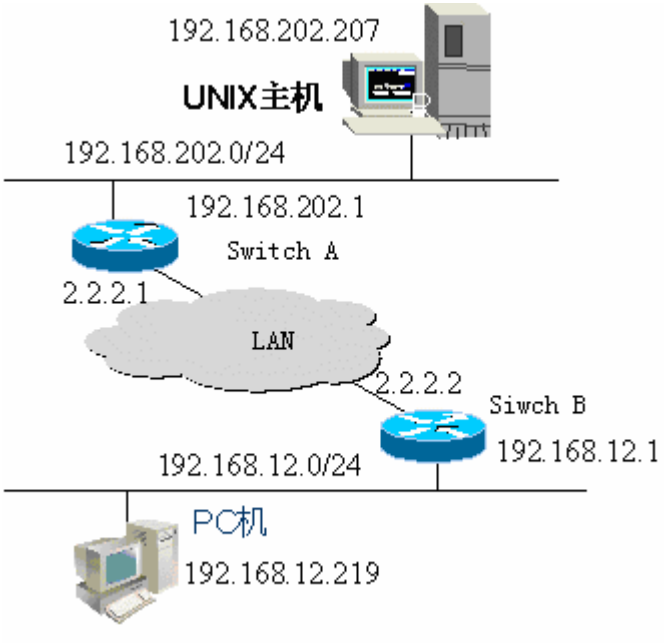


图 3　基本访问列表示例

要求通过在 Switch B 上配置访问列表，实现以下安全功能：

1. 192.168.12.0/24 网段的主机只能在正常上班时间访问远程 UNIX 主机 TELNET 服务，拒绝 PING 服务。

2. 在 Switch B 控制台上不能访问 192.168.202.0/24 网段主机的所有服务。

说明：

以上案例是银行系统应用的简化，即只允许分行或储蓄点局域网上的主机访问中心主机，不允许在设备上访问中心主机。

- **设备配置**

Switch B 的配置：

```
Ruijie(config)# interface GigabitEthernet 0/1
Ruijie(config-if)# ip address 192.168.12.1 255.255.255.0
Ruijie(config-if)# exit
Ruijie(config)# interface GigabitEthernet 0/2
Ruijie(config-if)# ip address 2.2.2.2 255.255.255.0
Ruijie(config-if)# ip access-group 101 in
Ruijie(config-if)# ip access-group 101 out
```

按照要求，配置一个编号为 101 的扩展访问列表

```
Ruijie(config)# access-list 101 permit tcp 192.168.12.0 0.0.0.255 any eq telnet time-range check
Ruijie(config)# access-list 101 deny icmp 192.168.12.0 0.0.0.255 any
```

```
Ruijie(config)# access-list 101 deny ip 2.2.2.0 0.0.0.255 any
Ruijie(config)# access-list 101 deny ip any any
```

配置 Time-Range 时间区

```
Ruijie(config)# time-range check
Ruijie(config-time-range)# periodic weekdays 8:30 to 17:30
```

说明：

访问列表 101 最后一条规则语句 access-list 101 deny ip any any 可以不要，因为访问列表最后隐含一条拒绝所有的规则语句。

Switch A 的配置：

```
Ruijie(config)# hostname Ruijie
Ruijie(config)# interface GigabitEthernet 0/1
Ruijie(config-if)# ip address 192.168.202.1 255.255.255.0
Ruijie(config)# interface GigabitEthernet 0/2
Ruijie(config-if)# ip address 2.2.2.1 255.255.255.0
```

49.3 MAC 扩展访问列表的配置

要在设备上配置 MAC 访问列表，必须给协议的访问列表指定一个唯一的名称或编号，以便在协议内部能够唯一标识每个访问列表。下表列出可以使用编号来指定 MAC 访问列表编号范围。

协议	编号范围
MAC 扩展访问列表	700-799

49.3.1 MAC 扩展访问列表配置指导

创建 MAC 访问列表时，定义的规则将可以应用于所有的分组报文，通过判断分组是否与规则匹配来决定是否转发或阻断分组报文。

MAC 访问列表中定义的典型规则主要有以下：

- 源 MAC 地址
- 目标 MAC 地址
- 以太网协议类型
- 时间区

MAC 扩展访问列表（编号 700 -799）主要是根据源和目的 MAC 地址来进行转发或阻断分组的，也可以对以太网协议类型匹配。

对于单一的 MAC 访问列表来说，可以使用多条独立的访问列表语句来定义多种规

则，其中所有的语句需引用同一个编号或名字，以便将这些语句绑定到同一个访问列表。

49.3.2 配置 MAC 扩展访问列表

MAC 访问列表的配置包括以下两步：

1. 定义 MAC 访问列表
2. 应用列表于特定接口

要配置 MAC 访问列表，有以下两种方式：

方式一　在全局配置模式下执行以下命令：

命令	功能
Ruijie(config)# **access-list** *id* {**deny** | **permit**}{**any** | **host** *src-mac-addr*} {**any** | **host** *dst-mac-addr*} [*ethernet-type*] [**cos** *cos*]	定义访问列表,命令的具体内容请参见命令参考
Ruijie(config)# **interface** *interface*	选择选择要应用访问列表的接口
Ruijie(config-if)# **mac access-group** *id* **in**	将访问列表应用特定接口

方式二， 在 ACL 配置模式下执行以下命令：

命令	功能
Ruijie(config)# **mac access-list extended** {*id* | *name*}	进入配置访问列表模式
Ruijie (config-mac-nacl)# [*sn*] { **permit** | **deny** }{**any** | **host** *src-mac-addr*} {**any** | **host** *dst-mac-addr*} [*ethernet-type*] [**cos** *cos*]	为 ACL 添加表项,命令的具体内容请参见命令参考
Ruijie(config-mac-nacl)# **exit** Ruijie(config)# **interface** *interface*	退出访问控制列表模式，选择选择要应用访问列表的接口
Ruijie(config-if)# **mac access-group** {*id* | *name*} **in**	将访问列表应用特定接口

说明:

方式一只对数值 ACL 进行配置；方式二可以对命名和数值 ACL 进行配置，还可以指定表项的优先级。

49.3.3 显示 MAC 扩展访问列表的配置

要监控访问列表，请在特权模式执行以下命令：

Ruijie**# show access-lists** [*id* | *name*]

可以查看基本访问列表

49.3.4 MAC 扩展访问列表示例

要求通过配置 MAC 访问列表，实现以下安全功能：

1. 使用 IPX 协议的主机 0013.2049.8272 不能访问设备 giga 0/1 端口。
2. 其他可以访问

配置以太口，在以太口上应用访问列表 101,对以太口上进出的所有报文进行检查

```
Ruijie> enable
Ruijie# configure terminal
Ruijie(config)# mac access-list extended mac-list
Ruijie(config-mac-nacl)# deny host 0013.2049.8272 any ipx
Ruijie(config-mac-nacl)# permit any any
Ruijie(config-mac-nacl)# exit
Ruijie(config)# interface gigabitEthernet 0/1
Ruijie(config-if)# mac access-group mac-list in
Ruijie(config-if)# end
Ruijie# show access-lists
mac access-list extended mac-list
deny host 0013.2049.8272 any ipx
permit any any
Ruijie#
```

说明：

访问列表语句 permit any any 不能不要，因为访问列表最后隐含一条拒绝所有的规则语句。

49.4 Expert 扩展访问列表的配置

要在设备上配置 Expert 扩展访问列表，必须给协议的访问列表指定一个唯一的名称或编号，以便在协议内部能够唯一标识每个访问列表。下表列出 Expert 访问列表的编号范围。

协议	编号范围
Expert 扩展访问列表	2700-2899

49.4.1 Expert 扩展访问列表配置指导

创建 expert 扩展访问列表时，定义的规则可以应用于所有的分组报文，通过判断分组是否与规则匹配来决定是否转发或阻断分组报文。

Expert 访问列表中定义的典型规则主要有以下：

- 基本访问列表和 MAC 扩展访问列表所有的信息
- VLAN ID

Expert 扩展访问列表（编号 2700 -2899）为基本访问列表和 MAC 扩展访问列表的综合体，并且能对 VLAN ID 进行过滤。

对于单一的 Expert 访问列表来说，可以使用多条独立的访问列表语句来定义多种规则，其中所有的语句需引用同一个编号或名字，以便将这些语句绑定到同一个访问列表。

49.4.2 配置 Expert 扩展访问列表

Expert 访问列表的配置包括以下两步：

1. 定义 Expert 访问列表
2. 应用列表于特定接口(应用特例)

要配置 Expert 访问列表，有以下两种方式

方式一 在全局配置模式下执行以下命令：

命令	功能
Ruijie (config)**# access-list** *id* {**deny** | **permit**} [*prot* | {[*ethernet-type*] [**cos** *cos*]}] [**VID** *vid*] {src *src-wildcard* | **host** *src* } {**host** *src-mac-addr* | **any**} {dst *dst-wildcard* | **host** *dst* | **any**}{**host** *dst-mac-addr* | **any**}] [**precedence** *precedence*] [**tos** *tos*] [**dscp** *dscp*] [**fragment**] [**time-range** *tm-rng-name*]	定义访问列表,命令的具体内容请参见命令参考
Ruijie(config)**# interface** *interface*	选择选择要应用访问列表的接口
Ruijie(config-if)**# expert access-group** *id* **in**	将访问列表应用特定接口

方式二， 在 ACL 配置模式下执行以下命令：

命令	功能
Ruijie(config)# **expert access-list extended** {*id*|*name*}	进入配置访问列表模式
Ruijie (config-exp-nacl)# [*sn*]{ **permit** | **deny** }[*prot* | {[*ethernet-type*] [**cos** *cos*]}] [**VID** *vid*] {src *src-wildcard* | **host** *src* }{**host** *src-mac-addr* | **any**} {dst *dst-wildcard* | **host** *dst* | **any**}{**host** *dst-mac-addr* | **any**}][**precedence** *precedence*] [**tos** *tos*] [**dscp** *dscp*] [**fragment**] [**time-range** *tm-rng-name*]	为 ACL 添加表项,命令的具体内容请参见命令参考
Ruijie(config-exp-nacl)# **exit** Ruijie(config)# **interface** *interface*	退出访问控制列表模式，选择选择要应用访问列表的接口
Ruijie(config-if)# **expert access-group** {*id*|*name*} **in**	将访问列表应用特定接口

说明:

方式一只对数值 ACL 进行配置；方式二可以对命名和数值 ACL 进行配置，在支持优先级表项的版本中，还可以指定表项的优先级(命令中[sn]选项)。

当只有-E 卡的情况下，对于 expert ACL，应用在 OUT 方向时，如果同时匹配 IP 字段和以太网字段，匹配以太网字段的 ACE 不会生效。

49.4.3 显示 Expert 扩展访问列表的配置

要监控访问列表，请在特权用户模式执行以下命令：

show access-lists [*id* | *name*]

可以查看 Expert 访问列表

49.4.4 Expert 扩展访问列表示例

要求通过配置 Expert 访问列表，实现以下安全功能：

1. VLAN20 的 0013.2049.8272 的主机能访问设备 Giga 0/1 端口。
2. 其他的不能访问

```
Ruijie> enable
```

```
Ruijie# config terminal
Ruijie(config)# expert access-list extended expert-list
Ruijie(config-exp-nacl)#  permit  ip  vid  20  any  host
0013.2049.8272 any any
Ruijie(config-exp-nacl)# deny any any any any
Ruijie(config-exp-nacl)# exit
Ruijie(config)# interface gigabitEthernet 0/1
Ruijie(config-if)# expert access-group expert-list in
Ruijie(config-if)# end
Ruijie# show access-lists
expert access-list extended expert-list
petmit ip vid 20 any host 0013.2049.8272 any any
deny any any
Ruijie#
```

49.5 IPv6 扩展访问列表的配置

49.5.1 配置 IPv6 扩展访问列表

IPv6 访问列表的配置包括以下两步：

1. 定义 IPv6 访问列表
2. 应用列表于特定接口(应用特例)

要配置基本访问列表，有以下方式，在 ACL 配置模式下执行以下命令：

命令	功能
Ruijie(config)# **ipv6 access-list** *name*	进入配置访问列表模式
Ruijie (config-ipv6-nacl)# [*sn*] {**permit** | **deny** }*prot* {*src-ipv6-prefix/prefix-len* | **host** *src-ipv6-addr* | **any**} {*dst-ipv6-pfix/pfix-len* | **any** | **host** *dst-ipv6-addr*} [**dscp** *dscp*] [**flow-label** *flow-label*] [**fragments**] [**time-range** *tm-rng-name*]	为 ACL 添加表项,命令的具体内容请参见命令参考
Ruijie(config-exp-nacl)# **exit** Ruijie(config)# **interface** *interface*	退出访问控制列表模式，选择选择要应用访问列表的接口
Ruijie(config-if)# **ipv6 traffic-filter** *name* **in**	将访问列表应用特定接口

49.5.2 显示 IPv6 扩展访问列表的配置

要监控访问列表，请在特权用户模式执行以下命令：

Ruijie**# show access-lists** [*name*]
此命令可以查看基本访问列表

49.5.3 IPv6 扩展访问列表示例

要求通过配置访问列表，实现以下安全功能：

1. 192.168.4.12 的主机能访问设备 gi 0/1 端口。
2. 其他的不能访问

```
Ruijie> enable
Ruijie# config terminal
Ruijie(config)# ipv6 access-list v6-list
Ruijie(config-ipv6-nacl)# permit ipv6 ::192:68:4:12/24 any
Ruijie(config-ipv6-nacl)# deny ipv6 any any
Ruijie(config-ipv6-nacl)# exit
Ruijie(config)# interface gigabitEthernet 0/1
Ruijie(config-if)# ipv6 traffic-filter v6-list in
Ruijie(config-if)# end
Ruijie# show access-lists
ipv6 access-list extended v6-list
petmit ipv6 ::192.168.4.12 any
deny any any
Ruijie#
```

49.6 配置 TCP Flag 过滤控制

TCP Flag 过滤特性提供了一种灵活机制，当前 TCP Flag 过滤控制支持 Match-All 选项，接收到的报文匹配到有 TCP Flag 与 ACL 表项中定义的 TCP Flag 每一位均吻合，就由 ACL 规则来检验，使用者可以定义 TCP Flag 的任意组合，用来过滤某些具有特定 TCP Flag 的报文。

例如：

permit tcp any any match-all rst

允许 TCP Flag RST 置位，其他位为 0 的报文通过。

说明：

这种过滤特性可以在命名 ACL，数值 ACL 配置协议号为 TCP 的情况下可选配置。

MAC 扩展和 IP 标准没有此项功能。

请按照如下步骤配置 TCP Flag

命令	功能
Ruijie(config)# **ip access-list extended** { *id* | *name* }	进入配置访问列表模式
Ruijie(config-ext-nacl)# [*sn*] [**permit** | **deny**] **tcp** *source source-wildcard* [**operator port** [*port*]] *destination destination-wildcard* [**operator port** [*port*]] [**match-all** *flag-name*][**precedence** *precedence*]	为 ACL 添加表项,命令的具体内容请参见命令参考
Ruijie(config-exp-nacl)# **exit** Ruijie(config)# **interface** *interface*	退出访问控制列表模式，选择选择要应用访问列表的接口
Ruijie(config-if)# **ip access-group** {*id* | *name*} {**in** | **out**}	将访问列表应用特定接口

下面的例子说明如何配置 TCP Flag

1) enable 权限和密码

```
Ruijie> enable
Ruijie#
```

2) 进入全局配置模式

```
Ruijie# configure terminal
Ruijie(config)#
```

3) 进入 ACL 配置模式

```
Ruijie(config)# ip access-list extended test-tcp-flag
Ruijie(config-ext-nacl)#
```

4) 添加 ACL 表项

```
Ruijie(config-ext-nacl)# permit tcp any any match-all rst
```

5) 添加 deny 表项

```
Ruijie(config-ext-nacl)# deny tcp any any match-all fin
```

6) 重复添加删除表项，

7) end

```
Ruijie(config-ext-nacl)# end
```

8) 8 显示

```
Ruijie# show  access-list test-tcp-flag
```

```
ip access-lists extended test-tcp-flag
10 permit tcp any any match-all rst
20 deny tcp any any match-all fin
```

49.7 按优先级配置 ACL 表项

为了体现 ACE 的优先级，对每个 ACL 列表提出标准，以规范该 ACL 列表下的 ACE 的编排方式，采用序号的起点-增量方式，具体描述如下：

- ACE 在链表中以序号自小至大方式排列；
- 以起点序号开始，如不指定序号均以前一 ACE 的序号为基础以增量递增。
- 指定序号时将该 ACE 以排序方式插入，增量保证了在两个相邻 ACE 之间能够插入新的 ACE。
- ACL 列表指定序号起点和序号增量。

提供 **ip access-list resequence** {*acl-id*| *acl-name*} *sn-start sn-inc* 命令，相关命令见命令参考。

每运行以上命令，便对 ACL list 下的 ace 重新排列，如名字为 tst_acl 的 ACL 下 ace 序号为：

初始时

```
ace1: 10
ace2: 20
ace3: 30
```

运行 **ip access-list resequence** *tst_acl 100 3*, ACE 的序号如下

```
Ruijie(config)# ip access-list resequence tst_acl 100 3
ace1: 100
ace2: 103
ace3: 106
```

不输入 sn-num 添加 ace4 时，序号如下

```
Ruijie(config-std-nacl)# permit 。。。
ace1: 100
ace2: 103
ace3: 106
ace4: 109
```

输入 seq-num = 105 添加 ace5 时，序号如下

```
Ruijie(config-std-nacl)# 105 permit 。。。
ace1: 100
ace2: 103
ace5: 105
ace3: 106
ace4: 109
```

序号的引用是为了实现 4 中的优先级添加 ace 的方式。
删除 ACE

```
Ruijie(config-std-nacl)# no 106
ace1: 100
ace2: 103
ace5: 105
ace4: 109
```

如上序号也可以方便 ACE 的删除。

49.8 配置基于时间区的 ACL

您可以使 ACL 基于时间运行，比如让 ACL 在一个星期的某些时间段内生效等。为了达到这个要求，您必须首先配置一个 Time-Range。

Time-Range 的实现依赖于系统时钟，如果您要使用这个功能，必须保证系统有一个可靠的时钟。

从特权模式开始，您可以通过以下步骤来设置一个 Time-Range：

命令	功能
Ruijie# **configure terminal**	进入全局配置模式。
Ruijie(config)# **time-range** *time-range-name*	通过一个有意义的显示字符串作为名字来标识一个
Ruijie(config-time-range)# **absolute** [**start time** *date*] **end time** *date*	设置绝对时间区间(可选)，具体可参见 time range 的配置指南
Ruijie(config-time-range)# **periodic** *day-of-the-week time* **to** [*day-of-the-week*] *time*	设置周期时间(可选)，具体可参见 time range 的配置指南
Ruijie# **show time-range**	验证您的配置
Ruijie# **copy running-config startup-config**	保存配置
Ruijie(config)# **ip access-list extended** *101*	进入 ACL 配置模式
Ruijie(config-ext-nacl)# **permit ip any any time-range** *time-range-name*	配置时间区的 ACE

说明：

Time Range 名字的长度为 1－32 个字符，不能包含空格

绝对的运行时间区间只能设置一个或不设置，基于 Time Range 的应用将仅在这个

时间区间内有效

您可以设置一个或多个周期性运行的时间段。如果您已经为这个 Time Range 设置了一个运行时间区间，则将在时间区间内周期性的生效

下面的例子以时间区 ACL 应用为例，说明如何在每周工作时间段内禁止 HTTP 的数据流：

```
Ruijie(config)# time-range no-http
Ruijie(config-time-range)# periodic weekdays 8:00 to 18:00
Ruijie(config)# end
Ruijie(config)# ip access-list extended limit-udp
Ruijie(config-ext-nacl)# deny tcp any any eq www time-range
no-http
Ruijie(config-ext-nacl)# exit
Ruijie(config)# interface gigabitEthernet 0/1
Ruijie(config-if)# ip access-group no-http in
Ruijie(config)# end
```

下面为 Time Range 的显示范例:

```
Ruijie# show time-range
time-range entry: no-http(inactive)
periodic Weekdays 8:00 to 18:00
time-range entry: no-udp
periodic Tuesday 15:30 to 16:30
```

49.9 配置注释

为了便于浏览和理解 ACL 配置，ACL 模块提供了针对 ACL 和 ACE 的注释功能。

说明：

- 一个 ACL 中最多配置 1 个 ACL 注释和 2048 条 ACE 注释。
- 每个注释长度为 100 字节。
- 仅在路由器上支持 ACE 注释。

从特权模式开始，您可以通过以下步骤给 ACL 配置注释：

命令	功能
Ruijie**# configure terminal**	进入全局配置模式。
Ruijie(config)**# ip access-list standard** *id*	进入 ACL 配置模式(其他类型 ACL 类似)
Ruijie(config-std-nacl)**# list-remark** *comment*	给 ACL 配置注释

也可以通过以下步骤给 ACL 配置注释：

命令	功能
Ruijie**# configure terminal**	进入全局配置模式。
Ruijie(config)**# access-list** *id* **list-remark** *comment*	直接给 ACL 配置注释(其他类型 ACL 类似)

从特权模式开始，您可以通过以下步骤给 ACE 配置注释：

命令	功能
Ruijie**# configure terminal**	进入全局配置模式。
Ruijie(config)**# ip access-list standard** *id*	进入 ACL 配置模式(其他类型 ACL 类似)
Ruijie(config-std-nacl)**# remark** *comment*	给 ACL 配置一条 ACE 注释

也可以通过以下步骤给 ACE 配置注释：

命令	功能
Ruijie**# configure terminal**	进入全局配置模式。
Ruijie(config)**# access-list** *id* **remark** *comment*	直接给 ACL 配置一条 ACE 注释(其他类型 ACL 类似)

下面的例子说明了如何配置 ACL 注释和 ACE 注释：

```
Ruijie(config)#ip access-list standard 1
Ruijie(config-std-nacl)#remark ace_remark_permit_62_start
Ruijie(config-std-nacl)#permit 192.168.197.62 0.0.0.0
Ruijie(config-std-nacl)#remark ace_remark_permit_62_end
Ruijie(config-std-nacl)#list-remark acl_remark_foo
Ruijie(config-std-nacl)#end
Ruijie#write
Ruijie#show access-lists 1
ip access-list standard 1
 remark ace_remark_permit_62_start
 10 permit host 192.168.197.62
 remark ace_remark_permit_62_end
list-remark acl_remark_foo
Ruijie#
```

49.10 配置举例

49.10.1 配置 TCP 单向连接

通过配置 TCP Flag 过滤实现单向 ACL 的功能。

49.10.1.1 配置要求

为了在一定程度上保证网络 A 的安全，要求只允许网络 A 的主机向网络 B 主机发起的 TCP 通信请求，但是不允许网络 B 的主机发起到网络 A 的 TCP 通信请求。

49.10.1.2 拓扑图

图 4

如上图所示，通过三层交换机连接两个网络，网络 A 与交换机的 G3/1 口相连，网络 B 与交换机的 G3/2 口相连。

49.10.1.3 分析

环境要求阻止网络 B 的主机发起到网络 A 的 TCP 通信请求，只要过滤从网络 B 发起经过交换机 G3/2 口转发的 TCP 连接请求报文即可。分析 TCP 连接过程可知，TCP 初始请求报文的 TCP 首部标志字段的 SYN 置位且 ACK 标志位为 0。因此，我们可以通过配置扩展访问控制列表的 Match-all 选项，在 G3/2 端口的入口方向，对 TCP 首部 SYN 标志位置 1 且 ACK 标志位为 0 的报文进行过滤，即可实现网络 A 单向访问网络 B 的应用。

49.10.1.4 配置步骤

1) 定义访问控制列表

进入交换机的配置模式

```
Ruijie# configure terminal
```

在配置模式下创建扩展访问列表 ACL101

```
Ruijie(config)# ip access-list extended 101
```

拒绝TCP Flag的SYN=1，且其它（包含ACK标志位）标志位为0的报文通过

```
Ruijie(config-ext-nacl)# deny tcp any any match-all syn
```

允许其它IP报文通过

```
Ruijie(config-ext-nacl)# permit ip any any
```

2) 把访问控制列表应用在接口上

退出访问控制列表模式

```
Ruijie(config-ext-nacl)# exit
```

进入应用此访问列表的接口 G3/2

```
Ruijie(config)# interface gigabitEthernet 3/2
```

将 ACL 101 应用于 G3/2 入方向的包过滤

```
Ruijie(config-if)# ip access-group 101 in
```

3) 显示访问列表的配置

在特权模式下，使用 **show** 命令显示 ACL 相关配置

```
Ruijie# show access-lists 101

ip access-list extended 101
10 deny tcp any any match-all syn
20 permit ip any any
```

49.10.2 企业网 ACL 典型应用

49.10.2.1组网图

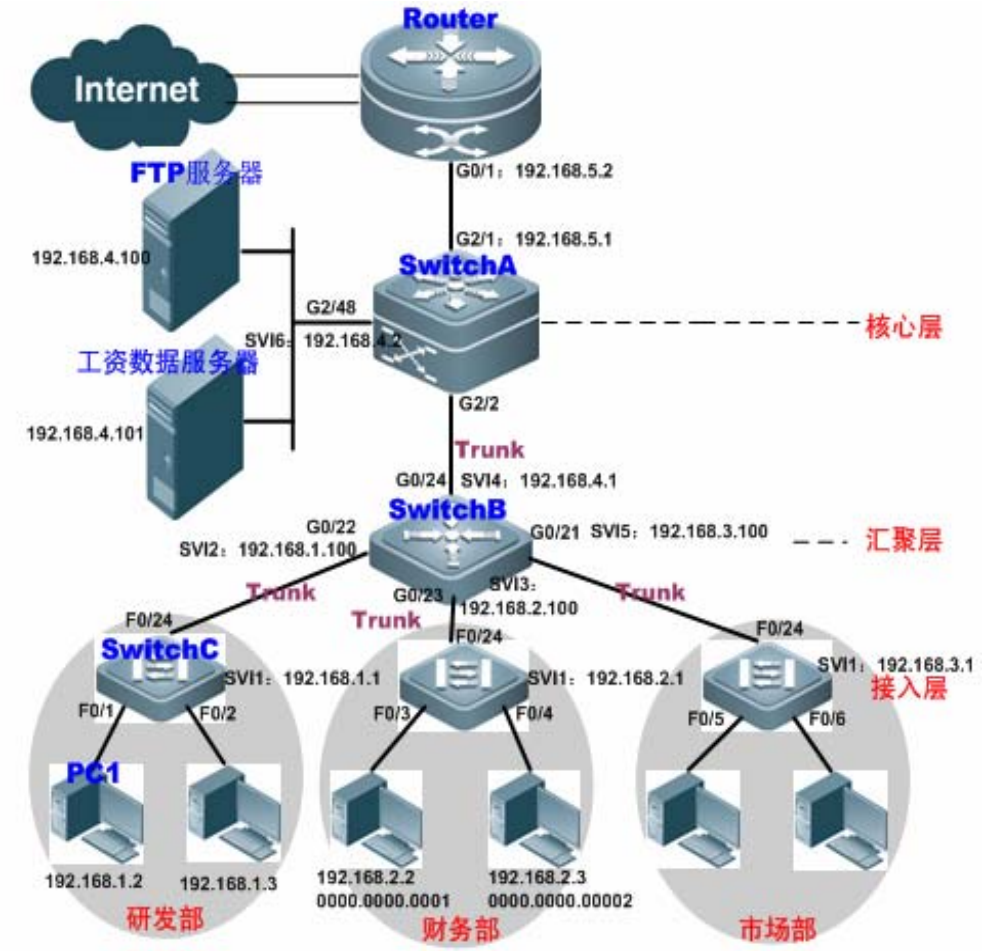


图 1 企业网 ACL 应用场景拓扑图

上图是典型的企业网络拓扑：

接入交换机 SwitchC：连接各部门的 PC，通过千兆光纤(trunk 方式)连接汇聚交换机。

汇聚交换机 SwitchB：划分多个 VLAN，每个部门为一个 VLAN，通过万兆光纤(trunk 方式)上连核心交换机。

核心交换机 SwitchA：连接各种服务器，如 FTP，HTTP 服务器等，通过防火墙与 Internet 相连。

49.10.2.2应用需求

上述企业网 ACL 应用场景主要有以下组网需求：

1、Internet 病毒无处不在，需要封堵各种病毒的常用端口，以保障内网安全。

2、只允许内部 PC 访问服务器，不允许外部 PC 访问服务器；

3、不允许非财务部门 PC 访问财务部 PC；不允许非研发部门 PC 访问研发部 PC

4、不允许研发部门人员在上班时间（即 9:00~18:00）使用 QQ、MSN 等聊天工具；

49.10.2.3配置要点

1. 通过在核心交换机（本例为 SwitchA）上联 Router 的端口（本例为 G2/1 口）上设置扩展 ACL 来过滤相关端口的数据包来达到防病毒的目的
2. 要求内部 PC 对服务器进行访问，不允许外部 PC 访问服务器，可以通过定义 IP 扩展 ACL 并应用到核心交换机（本例为 SwitchA）的下联汇聚交换机和服务器的接口（本例为 G2/2 口/SVI 2）上实现
3. 要求特定部门间不能互访，可通过定义 IP 扩展 ACL 实现（本例中分别在 SwitchB 的 G0/22、G0/23 上应用 IP 扩展 ACL）；
4. 可通过配置时间 IP 扩展 ACL，限制研发部门在特定时间内使用 QQ/MSN 等聊天工具（本例中在 SwichB 的 SVI2 上应用时间 IP 扩展 ACL）。

49.10.2.4配置步骤

- **配置核心交换机 SwitchA**

第一步：定义阻断病毒访问控制列表 *Virus_Defence*

☞ 配置导读

网络中的蠕虫病毒会在本地的 udp/69 端口上建立一个 tftp 服务器，用来向其它受侵害的系统上传送病毒的二进制程序。蠕虫选择目标 IP 地址的时候会首先选择受感染系统所在子网的 IP，然后再按照一定算法随机在互连网上选择目标攻击。一旦连接建立，蠕虫会向目标的 TCP 的 136、445、593、1025、5554、9995、9996，UDP 的 136、445、593、1433、1434，UDP/TCP 的 135、137、138、139 端口发送攻击数据。如果攻击成功，会监听目标系统的 TCP/4444 端口作为后门。然后蠕虫会连接到这个端口，发送 tftp 命令，回连到发起进攻的主机，将病毒文件传到目标系统上，然后运行它。中毒的服务器会向网络发送大量无效的数据包，浪费有效的网络带宽，甚至使交换机等网络设备死机，导致网络瘫痪。此时可以使用扩展访问列表来过滤这些端口的数据包来达到防病毒的目的。

```
SwitchA#configure terminal
SwitchA(config)#ip access-list extended Virus_Defence
```

```
！阻止来自内网外网可能被病毒利用的 TCP 端口报文
SwitchA(config-ext-nacl)#deny tcp any any eq 135
```

```
SwitchA(config-ext-nacl)#deny tcp any eq 135 any
SwitchA(config-ext-nacl)#deny tcp any any eq 136
SwitchA(config-ext-nacl)#deny tcp any eq 136 any
SwitchA(config-ext-nacl)#deny tcp any any eq 137
SwitchA(config-ext-nacl)#deny tcp any eq 137 any
……！中间的配置类似，此处省略说明
SwitchA(config-ext-nacl)#deny tcp any any eq 9996
SwitchA(config-ext-nacl)#deny tcp any eq 9996 any

！阻止来自内网外网可能被病毒利用的 UDP 端口报文
SwitchA(config-ext-nacl)#deny udp any any eq 69
SwitchA(config-ext-nacl)#deny udp any eq 69 any
SwitchA(config-ext-nacl)#deny udp any any eq 135
SwitchA(config-ext-nacl)#deny udp any eq 135 any
SwitchA(config-ext-nacl)#deny udp any any eq 137
SwitchA(config-ext-nacl)#deny udp any eq 137 any
……！中间的配置类似，此处省略说明
SwitchA(config-ext-nacl)#deny udp any any eq 1434
SwitchA(config-ext-nacl)#deny udp any eq 1434 any

！阻止 ICMP 报文
SwitchA(config-ext-nacl)#deny icmp any any

！允许其它所有 ip 数据包
SwitchA(config-ext-nacl)#permit ip any any

SwitchA(config-ext-nacl)#exit
```

第二步：将访问控制列表 *Virus_Defence* 应用在核心交换机上联 Router 的接口上

```
SwitchA(config)#interface gigabitEthernet 2/1
SwitchA(config-if)#no switchport
SwitchA(config-if)#ip address 192.168.5.1 255.255.255.0

！将 ACL Virus_Defence 应用于 G2/1 入方向，阻断外网的病毒报文
SwitchA(config-if)#ip access-group Virus_Defence in

SwitchA(config-if)#exit
```

第三步，定义只允许内网 PC 访问服务器的访问控制列表 *access_server*

```
SwitchA(config)#ip access-list extended access_server

！只允许指定内网 IP 网段 PC 访问服务器(IP 地址为 192.168.4.100)
SwitchA(config-ext-nacl)#permit ip 192.168.2.0 0.0.0.255 host
192.168.4.100
SwitchA(config-ext-nacl)#permit ip 192.168.1.0 0.0.0.255 host
192.168.4.100
SwitchA(config-ext-nacl)#permit ip 192.168.3.0 0.0.0.255 host
192.168.4.100
SwitchA(config-ext-nacl)#deny ip any any
```

第四步，将访问控制列表 *access_server* 应用在下联汇聚交换机和服务器的接口上

```
SwitchA(config)#interface gigabitEthernet 2/2
SwitchA(config-if)#switch mode trunk

！应用于连接汇聚交换机接口的入方向
SwitchA(config-if)#ip access-group access_server in
SwitchA(config-if)#exit

！创建 vlan
SwitchA(config)#vlan 2
SwitchA(config-vlan)#exit
SwitchA(config)#interface gigabitEthernet 2/48

！连接服务器的接口 G2/48 属于 vlan2
SwitchA(config-if)#switch access vlan 2
SwitchA(config-if)#exit

！应用于连接服务器接口的入方向
SwitchA(config)#interface vlan 2
SwitchA(config-if-VLAN 2)# ip access-group access_server in
SwitchA(config-if-VLAN 2)# ip address 192.168.4.2
255.255.255.0
SwitchA(config-ext-nacl)#end
```

- **配置汇聚交换机 SwitchB**

第一步，创建 vlan2-4

```
SwitchB#configure terminal

！创建 vlan2-4
SwitchB(config)#vlan range 2-4
SwitchB(config-vlan-range)#exit
```

第二步，定义访问控制列表

```
！定义 IP 扩展 ACL（vlan_access1 和 vlan_access2）
SwitchB(config)#ip access-list extended vlan_access1
！不允许财务部、市场部访问研发部
SwitchB(config-ext-nacl)#deny ip 192.168.2.0 0.0.0.255
192.168.1.0 0.0.0.255
SwitchB(config-ext-nacl)#deny ip 192.168.3.0 0.0.0.255
192.168.1.0 0.0.0.255
SwitchB(config-ext-nacl)#permit ip any any
SwitchB(config)#ip access-list extended vlan_access2
！不允许研发部、市场部访问财务部
SwitchB(config-ext-nacl)#deny ip 192.168.1.0 0.0.0.255
192.168.2.0 0.0.0.255
SwitchB(config-ext-nacl)#deny ip 192.168.3.0 0.0.0.255
```

```
192.168.2.0 0.0.0.255
SwitchB(config-ext-nacl)#permit ip any any
SwitchB(config-ext-nacl)#exit
```

第三步，将访问控制列表(vlan_access1 和 vlan_access2)应用在对应接口上

！配置 G0/22 口为 trunk 口，并应用 vlan_access1

```
SwitchB(config)#interface GigabitEthernet 0/22
SwitchB(config-if)#switchport mode trunk
SwitchB(config-if)#ip access-group vlan_access1 in
```

！配置 G0/23 口为 trunk 口，并应用 vlan_access2

```
SwitchB(config)# interface GigabitEthernet 0/23
SwitchB(config-if)# switchport mode trunk
SwitchB(config-if)# ip access-group vlan_access2 in
```

！配置 G0/24 为 trunk 口。

```
SwitchB(config)#interface GigabitEthernet 0/24
SwitchB(config-if)#switchport mode trunk
```

！配置 SVI2 的 IP 地址。

```
SwitchB(config)#interface vlan 2
SwitchB(config-if)#ip address 192.168.1.100 255.255.255.0
```

！配置 SVI3 的 IP 地址。

```
SwitchB(config)#interface vlan 3
SwitchB(config-if)#ip address 192.168.2.100 255.255.255.0
```

！配置 SVI4 的 IP 地址

```
SwitchB(config)#interface vlan 4
SwitchB(config-if)#ip address 192.168.4.1 255.255.255.0
```

第四步，定义时间段

！定义周一至周五的 9：00~18：00 的周期时间段

```
SwitchB#configure terminal
SwitchB(config)#time-range worktime
SwitchB(config-time-range)#periodic weekdays 9:00 to 18:00
```

第五步，定义研发部门数据流向规则

```
SwitchB#configure terminal
```

！在配置模式下创建扩展访问列表 ACL *yanfa*

```
SwitchB(config)#ip access-list extended yanfa
```

！禁止研发部的所有主机在工作日的 9：00 至 18：00 使用 QQ、MSN 等聊天工具。

```
SwitchB(config-ext-nacl)#deny tcp 192.168.1.0 0.0.0.255 eq
8000 any time-range worktime
SwitchB(config-ext-nacl)#deny tcp 192.168.1.0 0.0.0.255 eq
8001 any time-range worktime
```

```
SwitchB(config-ext-nacl)#deny tcp 192.168.1.0 0.0.0.255 eq 443
any time-range worktime
SwitchB(config-ext-nacl)#deny tcp 192.168.1.0 0.0.0.255 eq
1863 any time-range worktime
SwitchB(config-ext-nacl)#deny tcp 192.168.1.0 0.0.0.255 eq
4000 any time-range worktime
SwitchB(config-ext-nacl)#deny udp 192.168.1.0 0.0.0.255 eq
8000 any time-range worktime
SwitchB(config-ext-nacl)#deny udp 192.168.1.0 0.0.0.255 eq
1429 any time-range worktime
SwitchB(config-ext-nacl)#deny udp 192.168.1.0 0.0.0.255 eq
6000 any time-range worktime
SwitchB(config-ext-nacl)#deny udp 192.168.1.0 0.0.0.255 eq
6001 any time-range worktime
SwitchB(config-ext-nacl)#deny udp 192.168.1.0 0.0.0.255 eq
6002 any time-range worktime
SwitchB(config-ext-nacl)#deny udp 192.168.1.0 0.0.0.255 eq
6003 any time-range worktime
SwitchB(config-ext-nacl)#deny udp 192.168.1.0 0.0.0.255 eq
6004 any time-range worktime

！允许其它 IP 流量
SwitchB(config-ext-nacl)#permit ip any any

！将列表应用在 SVI2 的入方向上
SwitchB(config)#interface vlan 2
SwitchB(config-if)#ip access-group yanfa in
```

49.10.2.5配置验证

第一步，确认 ACE 条目是否正确，关注点为配置项先后顺序是否正确和是否生效。

```
SwitchA#show access-lists

ip access-list extended Virus_Defence
 10 deny tcp any any eq 135
 20 deny tcp any eq 135 any
 30 deny tcp any eq 4444 any
 40 deny tcp any any eq 5554
 50 deny tcp any eq 5554 any
 60 deny tcp any any eq 9995
 70 deny tcp any eq 9995 any
 80 deny tcp any any eq 9996
 90 deny tcp any eq 9996 any
 100 deny udp any any eq tftp
 110 deny udp any eq tftp any
```

```
 120 deny udp any any eq 135
 130 deny udp any eq 135 any
 140 deny udp any any eq netbios-ns
 150 deny udp any eq netbios-ns any
 160 deny udp any any eq netbios-dgm
 170 deny udp any eq netbios-dgm any
 180 deny udp any any eq netbios-ss
 190 deny udp any eq netbios-ss any
 200 deny udp any any eq 445
 210 deny udp any eq 445 any
 220 deny udp any any eq 593
 230 deny udp any eq 593 any
 240 deny udp any any eq 1433
 250 deny udp any eq 1433 any
 260 deny udp any any eq 1434
 270 deny udp any eq 1434 any
 280 deny tcp any any eq 136
 290 deny tcp any eq 136 any
 300 deny tcp any any eq 137
 310 deny tcp any eq 137 any
 320 deny tcp any any eq 138
 330 deny tcp any eq 138 any
 340 deny tcp any any eq 139
 350 deny tcp any eq 139 any
 360 deny tcp any any eq 445
 370 deny tcp any eq 445 any
 380 deny tcp any any eq 593
 390 deny tcp any eq 593 any
 400 deny tcp any eq 1025 any
 410 deny tcp any any eq 4444
 420 deny icmp any any
 430 permit tcp any any
 440 permit udp any any
 450 permit ip any any

ip access-list extended access_server
 10 permit ip 192.168.2.0 0.0.0.255 host 192.168.4.100
 20 permit ip 192.168.1.0 0.0.0.255 host 192.168.4.100
 30 permit ip 192.168.3.0 0.0.0.255 host 192.168.4.100
 40 deny ip any any

SwitchB#show access-lists
ip access-list extended vlan_access1
 10 deny ip 192.168.2.0 0.0.0.255 192.168.1.0 0.0.0.255
 20 deny ip 192.168.3.0 0.0.0.255 192.168.1.0 0.0.0.255
 30 permit ip any any
```

```
ip access-list extended vlan_access2
 10 deny ip 192.168.1.0 0.0.0.255 192.168.2.0 0.0.0.255
 20 deny ip 192.168.3.0 0.0.0.255 192.168.2.0 0.0.0.255
 30 permit ip any any

ip access-list extended yanfa
 10 deny tcp 192.168.1.0 0.0.0.255 eq 8000 any time-range
worktime (active)
 20 deny tcp 192.168.1.0 0.0.0.255 eq 8001 any time-range
worktime (active)
 30 deny tcp 192.168.1.0 0.0.0.255 eq 443 any time-range
worktime (active)
 40 deny tcp 192.168.1.0 0.0.0.255 eq 1863 any time-range
worktime (active)
 50 deny tcp 192.168.1.0 0.0.0.255 eq 4000 any time-range
worktime (active)
 60 deny udp 192.168.1.0 0.0.0.255 eq 8000 any time-range
worktime (active)
 70 deny udp 192.168.1.0 0.0.0.255 eq 1429 any time-range
worktime (active)
 80 deny udp 192.168.1.0 0.0.0.255 eq 6000 any time-range
worktime (active)
 90 deny udp 192.168.1.0 0.0.0.255 eq 6001 any time-range
worktime (active)
 100 deny udp 192.168.1.0 0.0.0.255 eq 6002 any time-range
worktime (active)
 110 deny udp 192.168.1.0 0.0.0.255 eq 6003 any time-range
worktime (active)
 120 deny udp 192.168.1.0 0.0.0.255 eq 6004 any time-range
worktime (active)
```

第二步，确认 ACL 配置是否完整，关注点为是否将正确的 ACL 应用到指定的接口上。

SwitchA 配置：

```
SwitchA#show run
interface GigabitEthernet 2/1
 no switchport
 no ip proxy-arp
 ip access-group Virus_Defence in
 ip address 192.168.5.1 255.255.255.0
!
interface GigabitEthernet 2/2
 switchport mode trunk
 ip access-group access_server in
```

```
!
interface VLAN 2
 no ip proxy-arp
 ip access-group access_server in
 ip address 192.168.4.2 255.255.255.0
```

SwitchB 配置：

```
SwitchB#show run
!
interface GigabitEthernet 0/22
 switchport mode trunk
 ip access-group vlan_access1 in
!
interface GigabitEthernet 0/23
 switchport mode trunk
 ip access-group vlan_access2 in
!
interface VLAN 2
 no ip proxy-arp
 ip access-group yanfa in
 ip address 192.168.1.100 255.255.255.0
```

49.10.3 专家级 ACL&ACL80 应用

49.10.3.1 拓扑图

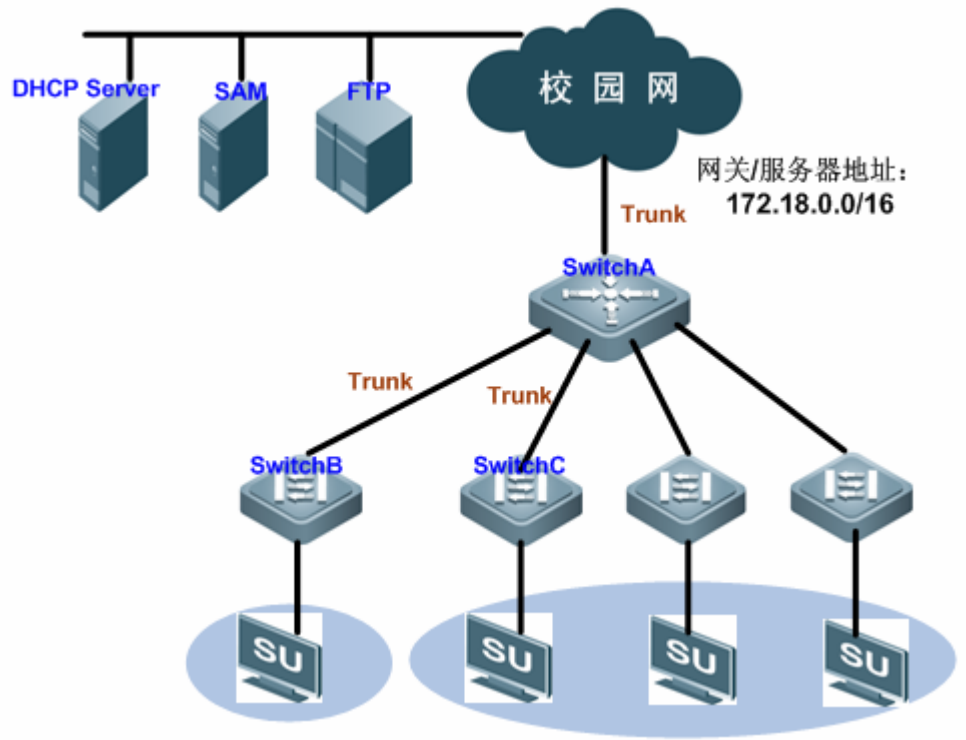


图 2 专家级 ACL&ACL80 应用拓扑图

上图为校园网的简化拓扑：

SwitchA 作为汇聚设备，划分多个 VLAN，每个院系为一个 VLAN，通过万兆光纤(trunk 方式)上连校园网络。

SwitchB、SwitchC 作为接入设备，连接各院系的 PC，通过千兆光纤(trunk 方式)连接汇聚交换机。

每台 PC 必须安装 SU 客户端并通过 802.1x 认证之后才能使用网络。

49.10.3.2 应用需求

SU 软件不是 windows 自带的，PC 必须要到服务器下载 SU 客户端安装才能进行认证，但 PC 没通过 802.1x 认证不能使用网络下载软件，为了解决这个矛盾，提出如下需求：

1) 允许访问网关/服务器网段地址（172.18.0.0/16）的 IP 报文、 ARP 报文可以免认证通过，以便用户 PC 在认证前可以到指定服务器下载软件或访问网关；
2) 允许 DHCP 报文(UDP 端口号 67/68)免认证通过，以便用户 PC 可以获取 IP 地址以便进行认证；

49.10.3.3 配置要点

在接入层设备（本例中的 SwitchB/SwitchC）配置 ACL80 或专家级 ACL，并结合安全通道功能，可以实现对特定报文的免认证通过满足上述需求。

本案例在 SwitchB 上配置 ACL80，在 SwitchC 上配置专家级 ACL 来分别说明其应用。

49.10.3.4 配置步骤

- **配置 SwitchB**

☞ 配置导读

自定义访问列表可以让用户对报文的前 80 个字节中的指定 64 个字节按比特（bit）位进行匹配过滤。将用户自定义的字符串和从报文提取出来的字符串进行比较（1 表示匹配对应过滤规则的比特位，0 表示不匹配），进行相应的处理。

第一步，配置自定义访问控制列表

```
SwitchB#configure terminal
```

!创建名为 *tongdao* 的自定义访问控制列表
```
SwitchB(config)#expert access-list advanced tongdao
```

！允许源 IP（ARP 报文的源 IP 地址偏移值为 40）为 172.18.0.0（十六进制值 ac12）网段的所有 ARP 报文(协议号 0806，偏移值为 24)通过
```
SwitchB(config-exp-dacl)#permit 0806 ffff 24 ac12 ffff 40
```

！允许源 IP（IP 报文的源 IP 地址偏移值为 38）为 172.18.0.0（十六进制值 ac12）网段的所有 IP 报文(协议号 0800，偏移值为 24)通过
```
SwitchB(config-exp-dacl)#permit 0800 ffff 24 ac12 ffff 38
```

！允许 UDP 端口号为 67（Bootstrap Protocol Server）、68（Bootstrap Protocol Client）的 DHCP 报文通过（协议号偏移值为 35，十六进制值 11 的表示 UDP，端口的偏移值为 46，十六进制值 43/44 分别对应端口号 67、68）。

```
SwitchB(config-exp-dacl)# permit 11 ff 35 00440043 ffffffff 46
SwitchB(config-exp-dacl)#exit
```

第二步，在全局上配置安全通道应用 ACL

！配置安全通道应用 ACL *tongdao*
```
SwitchB(config)# security global access-group tongdao
```

- **配置 SwitchC**

第一步，配置专家级访问控制列表

```
SwitchC#configure terminal
```

！在配置模式下创建名为 *tongdao1* 的专家级访问控制列表

```
SwitchC(config)#expert access-list extended tongdao1
```

！允许源 IP 为 172.18.0.0 网段的所有 IP 报文通过

```
SwitchC(config-exp-dacl)#permit ip 172.18.0.0 0.0.255.255 any
any any
```

！允许 udp 端口号为 67（Bootstrap Protocol Server）、68（Bootstrap Protocol Client）的报文通过。

```
SwitchC(config-exp-dacl)# permit udp any any eq bootpc any any
eq bootps
SwitchC(config-exp-dacl)#exit
```

第二步，在全局上配置安全通道应用 ACL

! 配置安全通道应用 ACL *tongdao1*

```
SwitchC(config)# security global access-group tongdao1
```

49.10.3.5 配置验证

第一步，确认 ACE 条目是否正确，关注点为配置项先后顺序是否正确和是否生效

```
SwitchB# show access-lists
expert access-list advanced tongdao
 10 permit 0806 FFFF 24 AC12 FFFF 40
 20 permit 0800 FFFF 24 AC12 FFFF 38
 30 permit 11 FF 35 00440043 FFFFFFFF 46
```

```
SwitchC# show access-lists
expert access-list extended tongdao1
 10 permit ip 172.18.0.0 0.0.255.255 any any any
 20 permit udp any any eq bootpc any any eq bootps
```

通过上述命令可以查看相应类型的访问列表配置的 ACE 条目是否正确。

第二步，确认 ACL 配置是否完整，关注点为是否将正确的 ACL 应用在全局模式下：

```
SwitchB#show run

expert access-list advanced tongdao
!
security global access-group tongdao
!
SwitchC#show run

!
expert access-list advanced tongdao1
!
security global access-group tongdao1
!
```

50 QOS 配置

50.1 QOS 概述

随着 Internet 的飞速发展，人们对于在 Internet 上传输多媒体流的需求越来越大，一般说来，用户对不同的多媒体应用有着不同的服务质量要求，这就要求网络应能根据用户的要求分配和调度资源，因此，传统所采用的“尽力而为”转发机制，已经不能满足用户的要求。QOS 应运而生。

QOS（Quality of Service，服务质量）是用来评估服务方满足客户需求的能力。在因特网中，为了提高网络服务质量，引入 QOS 机制，用 QOS 评估网络投递分组的能力。我们通常所说的 QOS，是对分组投递过程中为延迟、抖动、丢包等核心需求提供支持的服务能力的评估。

50.1.1 QoS 基础框架

不支持 QoS 功能的设备不具有提供传输品质服务的能力，它同等对待所有的交通数据流，并不保证某一特殊的数据流会受到特殊的转发待遇。当网络带宽充裕的时候，所有的数据流都得到了较好的处理，而当网络拥塞发生的时候，所有的数据流都有可能被丢弃。这种转发策略被称做提供最佳效果服务，因为这时设备是尽最大能力转发数据，设备本身的交换带宽得到了充分的利用。

本设备支持 QoS 功能，能够提供传输品质服务。针对某种类别的数据流，您可以为它赋予某个级别的传输优先级，来标识它的相对重要性，并使用设备所提供的各种优先级转发策略、拥塞避免等机制为这些数据流提供特殊的传输服务。配置了 QoS 的网络环境，增加了网络的性能可预知性，并能够有效地分配网络带宽，更加合理地利用网络资源。

本设备的 QoS 实现以 IETF（Internet Engineering Task Force） 的 DiffServ（Differentiated Servece Mode），差分服务模型） 体系为基础。DiffServ 体系规定网络中的每一个传输报文将被划分成不同的类别，分类信息被包含在了 IP 报文头中，DiffServ 体系使用了 IPv4 报文头中的 TOS（Type Of Service）或者 Ipv6 报文头中的 Traffic Class 字段的前 6 个比特来携带报文的分类信息。当然分类信息也可以被携带在链路层报文头上。一般地，附带在报文中的分类信息有：

- 携带在 802.1Q 帧头的 Tag Control Information 中的前 3 个比特，它包含了 8 个类别的优先级信息，通常称这三个比特为 User Priority bits。

- 携带在 IPv4 报文头中的 TOS 或者 IPv6 报文头中的 Traffic Class 字段的前 3 个比特，称作 IPprecedence value；或者携带在 IPv4 报文头中的 TOS 或者 IPv6 报文头中的 Traffic Class 字段的前 6 个比特，称作 Differentiated Services Code Point (DSCP) value。

在遵循 DiffServ 体系的网络中，各设备对包含相同分类信息的报文采取相同的传

输服务策略，对包含不同分类信息的报文采取不同的传输服务策略。报文的分类信息可以由网络上的主机、设备或者其它网络设备赋予。可以基于不同的应用策略或者基于报文内容的不同为报文赋予类别信息。识别报文的内容以便为报文赋予类别信息的做法往往需要消耗网络设备的大量处理资源，为了减少骨干网络的处理开销，一般这种赋予类别信息的方式都使用在网络边界。设备根据报文所携带的类别信息，为各种交通流提供不同的传输优先级，或者为某种交通流预留带宽，或者适当地丢弃一些优先级较低的报文、或者采取其他一些操作等等。这些独立设备的这种行为在 DiffServ 体系中被称作每跳行为（Per-hop Behavior）。

如果网络上的所有设备提供了一致的每跳行为，那么对于 DiffServ 体系来说，这个网络就可以构成 End-to-end QoSsolution。

50.1.2 QOS 处理流程

50.1.2.1 Classifying

Classifying 即分类，其过程是根据信任策略或者根据分析每个报文的内容来确定将这些报文归类到以 CoS 值来表示的各个数据流中，因此分类动作的核心任务是确定输入报文的 CoS 值。分类发生在端口接收输入报文阶段，当某个端口关联了一个表示 QoS 策略的 Policy-map 后，分类就在该端口上生效，它对所有从该端口输入的报文起作用。

对于一般非 IP 报文，设备将根据以下准则来归类报文：

- 如果报文本身不包含 QoS 信息，即报文的第二层报文头中不包含 User Priority bits，那么可以根据报文输入端口的缺省 CoS 值来获得报文的 QoS 信息。端口的缺省 CoS 值和报文的 User Priority bits 一样，取值范围为 0～7。

- 如果报文本身包含 QoS 信息，报文的第二层报文头中包含 User Priority bits，那么可以直接从报文中获得 CoS 值。

说明：

以上两种归类准则只有当端口的 QoS 信任模式打开的时候才起作用。打开端口的 QoS 的信任模式意味着不通过分析报文的内容，而直接从报文中或报文的输入端口上获得报文 QoS 信息。

- 如果端口关联的 Policy-map 中使用了基于 Mac access-list extended 的 ACLs 归类，那么在该端口上，将通过提取报文的源 MAC 地址、目的 MAC 地址以及 Ethertype 域来匹配关联的 ACLs，以确定报文的 DSCP 值。要注意的是，如果端口关联了某个 Policy-map，但又没有为其设置相应的 DSCP 值，则设备将按照缺省行为为符合这种归类的报文分配优先级：即根据报文第二层报文头中包含的优先级信息或端口的缺省优先级。

说明:

上面三种归类准则可能会同时作用于一个端口上。在这种情况下，上面三种归类准则按 3、2、1 的优先级起作用。即先根据 ACLs 归类，在归类失败的情况下，才有可能选择归类准则 2、1，在这个时候，如果端口的 QoS 信任模式打开，则根据准则 2 和 1 直接从报文中或者从端口上获得 QoS 信息；如果端口的 QoS 信任模式关闭，那么那些归类失败的报文将被赋予 DSCP 的缺省值 0。

对于 IP 报文，可以将根据以下准则来归类报文：

- 如果端口信任模式为 Trust ip-precedence，则直接从 IP 报文的 Ip precedence 字段（3 个比特）提取出来，填充到输出报文的 CoS 字段（3 个比特）。
- 如果端口信任模式为 Trust cos，则将报文的 CoS 字段（3 个比特）直接提取出来覆盖报文 Ip Precedence 字段（3 个比特）。这有两种情况，一是第二层报文头中不包含 User Priority bits，那么可以根据报文输入端口的缺省 CoS 值来获得报文的 CoS 值。另外一种是第二层报文头中包含 User Priority bits，则直接从报文头中取得 CoS 值。
- 如果端口关联的 Policy-map 中使用了基于 Ip access-list (Extended)的 ACLs 归类，那么该在该端口上，将通过提取报文的源 IP 地址、目的 IP 地址、Protocol 字段、以及第四层 TCP/UDP 端口字段来匹配相关联的 ACLs，以确定报文的 DSCP 值。要注意的是，如果端口关联了某个 Policy-map，但又没有为其设置相应的 DSCP 值，则设备将按照按照前面的规则 1、2 确定优先级。

和非 IP 报文归类准则一样，以上几种归类准则同样可以同时作用于一个端口上。在这种情况下，上面的归类准则按照 3、2、1 的优先级起作用。

有关上面提到的 CoS-to-DSCP map、IP-precedence-to-DSCP map 映射表的详细描述见随后描述。

50.1.2.2 Policing

Policing 即策略，发生在数据流分类完成后，用于约束被分类的数据流所占用的传输带宽。Policing 动作检查被归类的数据流中的每一个报文，如果该报文超出了作用于该数据流的 Police 所允许的限制带宽，那么该报文将会被做特殊处理，它或者要被丢弃，或者要被赋予另外的 DSCP 值。

在 QoS 处理流程中，Policing 动作是可选的。如果没有 Policing 动作，那么被分类的数据流中的报文的 DSCP 值将不会作任何修改，报文也不会在送往 Marking 动作之前被丢弃。

说明:

S3760 系列交换机不支持输入方向限速，输出方向限速的最小值和粒度均为 651Kbps。且输出方向限速不计算帧间距；

50.1.2.3 Marking

Marking 即标识，经过 Classifying 和 Policing 动作处理之后，为了确保被分类报文对应 DSCP 的值能够传递给网络上的下一跳设备，需要通过 Marking 动作将为报文写入 QoS 信息，可以使用 QoS ACLs 改变报文的 QoS 信息，也可以使用 Trust 方式直接保留报文中 QoS 信息，例如，选择 Trust DSCP 从而保留 IP 报文头的 DSCP 信息。

50.1.2.4 Queueing

Queueing 即队列，负责将数据流中报文送往端口的某个输出队列中，送往端口的不同输出队列的报文将获得不同等级和性质的传输服务策略。

每一个端口上都拥有 8 个输出队列，通过设备上配置的 DSCP-to-CoS Map 和 Cos-to-Queue Map 两张映射表来将报文的 DSCP 值转化成输出队列号，以便确定报文应该被送往的输出队列。

50.1.2.5 Scheduling

Scheduling 即调度，为 QoS 流程的最后一个环节。当报文被送到端口的不同输出队列上之后，设备将采用 WRR 或者其它算法发送 8 个队列中的报文。

可以通过设置 WRR 算法的权重值来配置各个输出队列在输出报文的时候所占用的每循环发送报文个数,从而影响传输带宽。或通过设置 DRR 算法的权重值来配置各个输出队列在输出报文的时候所占用的每循环发送报文字节数,从而影响传输带宽。

50.1.3 交换机 buffer 控制

802.3x flow-control 标准规定了一个报文无丢弃的交换实现，然而在大多数情况下，flow-control 是有害的，例如：恶意用户可以发动流控攻击，故意发流控报文而影响交换机的正常转发。同时在实现 QoS 功能的交换机中，流控会产生 HOL（头部阻塞）现象。导致高优先级的报文不能优先传输。

为了解决这个问题，我们提供灵活的 buffer 控制功能。您可以配置交换机处于 802.3xflow-control 状态，实现一个无丢弃的网络环境；您也可以配置交换机处于 QoS 状态，这时避免 HOL 现象发生。

当交换机处于 flow-control 状态时，如果入端报文速率过大，来不及转发时(比如多个入口，一个出口)。交换机会向入端的反方向发送 Pause 帧，通知那个方向的网络设备暂停发送帧。避免丢弃。

当交换机处于 QoS 状态时，优先传输高优先级报文，并且缓冲低优先级报文，如果高优先级报文还拥塞，则缓存高优先级报文，并且给高优先级的报文提供更多的缓存能力。对待来不及缓存处理的报文，则是直接采用丢弃的办法，而不会发出 Pause 帧。

我们推荐您如果关闭交换机 QoS 功能时，让交换机 buffer 控制处于 802.3xflow-control 状态；打开交换机 QoS 功能时，让交换机 buffer 控制处于 QoS 状态。对于多台设备组成堆叠系统情况下，推荐使用 QOS 模式，只有在该模式下才能保证堆叠拓扑稳定可靠。

比如：文件传输应用中，服务器的处理能力高于客户端，服务器的网卡是 1000m，而服务端只有 100m，那么在 QOS 模式下，服务端持续发出的数据速率可能大于 100m，从而导致大量的报文丢失，进而导致了大量的数据包重传，因此出现实际有效传输速率过低的情况，这中情况下应该使用 FC 模式，避免报文丢失。

但需要注意的是，并不是打开 QoS 功能时，就只能采用 QoS 状态，关闭 QoS 功能时就只能采用 flow-control 状态。要根据实际应用的需求来判断。

比如：当采用 Police 进行限速时(打开了 QoS 功能)。如果这时使 buffer 控制处于 QOS 模式，不会产生流控，但如果多个入口同时往一个出口发包，会造成硬件资源不足，从而速率限制不准。如果采用 flow-control 控制，就可以避免这个问题。

50.1.4交换机拥塞队列数控制

交换机的每个端口支持 8 个输出队列，队列在输出报文时需要使用输出缓冲区资源，交换机的输出缓冲区资源由所有端口队列共享，因此当交换机上多个端口的多个队列同时发生输出拥塞时，缓冲区资源可能出现无法满足当前所有输出队列的需求，端口输出队列将由于获取不到足够的缓冲区，此时拥塞控制策略将会不精确（如输出调度算法不准确等）。

我司交换机提供根据端口输出拥塞队列数分配输出缓区的机制，您可以使用动态配置与静态配置两种模式：

动态模式

设备支持动态规划所有端口的拥塞队列数，规则如下：

- 千兆端口配置满足 8 个队列同时拥塞的缓冲区资源。
- 百兆端口配置均分剩余的所有缓冲区资源。

静态模式

设备支持手工静态配置端口的拥塞队列数，保障端口在特定数目个输出队列拥塞时各队列有足够的输出缓冲区以满足输出拥塞控制的需求。

通过手工静态配置的手段，您可以结合网络规划中各端口实际所使用的输出队列数，配置各端口的拥塞队列数，以更加有效地利用交换机的输出缓冲区资源。

说明：

1. 仅下列交换机支持拥塞队列数配置，其余产品所有端口均支持 8 个拥塞队列。
   - S3760-48 交换机
2. 配置仅在交换机的 Buffer 控制为 Qos 模式时才生效。
3. 拥塞队列控制仅有物理端口与 AP 成员口支持。
4. 端口仅配置一个拥塞队列数时，端口产生输出拥塞时，各队列的报文输出时不表现出优先级的差异。

50.1.5 QOS 逻辑端口组

可以指定一系列端口为一个 QOS 逻辑端口组（这里端口可以是 AP，也可以是物理口，下文简称为逻辑端口组），并针对这个逻辑端口组关联 Policy-map 进行 QOS 处理，以限速为例，对符合限速条件的报文，在同一个逻辑端口组内所有的端口共享 Policy-map 所限定的带宽值。

注意：

加入逻辑端口组的成员必须是物理口或者是 Aggregate Port。
交换机逻辑端口组的支持数量为 128 个。

50.2 配置 QOS

50.2.1 缺省 QOS 设置

用户在进行 QoS 配置之前，需要清楚和 QoS 有关的几点信息，如下：

- 一个接口最多关联 1 个 Policy-map
- 一个 Policy-map 可以拥有多个 Class-map
- 一个 Class-map 最多关联 1 个 ACL，该 ACL 的所有 ACE 必须具有相同过滤域模板
- 一个接口上关联的 ACE 的个数服从“配置安全 ACL”章节的限制

缺省情况下，QoS 功能是关闭的，即设备对所有的报文同等处理。但当您将一个 Policy Map 关联到某一个接口上，并设置了接口的信任模式时，该接口的 QoS 功能即被打开。要关闭该接口的 QoS 功能，您可以通过解除该接口的 Policy Map 设置，并将接口的信任模式设为 Off 即可。以下为 QOS 的缺省配置：

缺省 CoS 值	0
队列个数	8
队列轮转算法	WRR
QueueWeight	1:1:1:1:1:1:1:1
WRR Weight Range	1:15
DRR Weight Range	1:15
信任模式	No Trust

Cos 值到队列的默认映射表

CoS 值	0	1	2	3	4	5	6	7
队列	1	2	3	4	5	6	7	8

CoS to DSCP 默认映射表

CoS 值	0	1	2	3	4	5	6	7
DSCP 值	0	8	16	24	32	40	48	56

IP-Precedence to DSCP 默认映射表

IP-Precedence	0	1	2	3	4	5	6	7
DSCP	0	8	16	24	32	40	48	56

DSCP to CoS 的默认映射表

DSCP	0	8	16	24	32	40	48	56
CoS	0	1	2	3	4	5	6	7

50.2.2 配置接口的 Qos 信任模式

缺省情况下，接口的 Qos 信任模式是不信任

命令	说明
configure terminal	进入配置模式
interface *interface*	进入接口配置模式
mls qos trust {cos | ip-precedence | dscp}	配置接口的 Qos 信任模式 cos，dscp 或 ip-precedence
no mls qos trust	恢复接口默认 Qos 信任模式

以下命令将端口 **interface** GigabitEthernet 0/4 信任模式设置为 DSCP：

```
Ruijie(config)# interface gigabitEthernet 0/4
Ruijie(config-if)# mls qos trust dscp
Ruijie(config-if)# end
```

```
Ruijie# show mls qos interface g0/4
Interface: GigabitEthernet 0/4
Attached input  policy-map:
Default COS: trust dscp
Default COS: 0
Ruijie#
```

50.2.3 配置接口的缺省 CoS 值

您可以通过下面的设置步骤来配置每一个接口的缺省 CoS 值

缺省情况下，接口的缺省 CoS 值为 0

命令	说明
configure terminal	进入配置模式
interface *interface*	进入接口配置模式
mls qos cos default-cos	配置接口的缺省 CoS 值，default-cos 为要设置的缺省 CoS 值，取值范围为 0～7
no mls qos cos	默认的缺省 CoS 值

下面的例子将接口 Interface g0/4 缺省 CoS 值设置为 6

```
Ruijie# configure terminal
Ruijie(config)# interface g 0/4
Ruijie(config-if)# mls qos cos 6
Ruijie(config-if)# end
Ruijie# show mls qos interface g 0/4
Interface: GigabitEthernet 0/4
Attached input  policy-map:
Default COS: trust dscp
Default COS: 6
Ruijie#
```

50.2.4 配置逻辑端口组

在接口配置模式下，请按如下步骤将端口加入逻辑端口组：

命令	作用
Ruijie(config-if)# [no] **virtual-group** *virtual-group-number*	将该接口加入一个逻辑端口组或退出一个逻辑端口组。*virtual-group-number* 表示逻辑端口组成员端口组的编号，即逻辑端口组号。

在接口配置模式下使用 **no virtual-group** *virtual-group-number* 命令将一个物理端口退出逻辑端口组。

下面的例子是将以太网接口 0/1 配置成逻辑端口组 5 的成员：

```
Ruijie# configure terminal
Ruijie(config)# interface gigabitEthernet 0/1
Ruijie(config-if-range)# virtual-group 5
Ruijie(config-if-range)# end
```

50.2.5 配置 Class Maps

您可以通过下面的设置步骤来创建并配置 Class Maps

命令	说明
configure terminal	进入配置模式
ip access-list extended {*id* | *name*} **…** **ip access-list standard** {*id* | *name*} **…** **mac access-list extended** {*id* | *name*} **…** **expert access-list extended** {*id* | *name*} **…** **ipv6 access-list extended** *name* **…** **access-list** *id* [**…**]	创建 ACL 请参见 ACL 章节
[**no**] **class-map class-map-name**	创建并进入 class map 配置模式,class-map-name 是要创建的 class map 的名字 no 选项 删除一个已经存在的 class map
[**no**] **match access-group** {*acl-num* | *acl-name* }	设置匹配 ACL, acl-name 为已经创建的 ACL 名字， acl-num 为已经创建的 ACL id， no 选项删除该匹配

例如，以下设置步骤创建了一个名为 Class1 的 Class-map，它关联一个 ACL:acl_1。这个 Class-map 将分类所有端口号为 80 的 TCP 报文

```
Ruijie(config)# ip access-list extended acl_1
Ruijie(config-ext-nacl)# permit tcp any any eq 80
Ruijie(config-ext-nacl)# exit
Ruijie(config)# class-map class1
Ruijie(config-cmap)# match access-group acl_1
Ruijie(config-cmap)# end
```

50.2.6 配置 Policy Maps

您可以通过下面的设置步骤来创建并配置 Policy Maps

命令	说明
configure terminal	进入配置模式
[**no**] **policy-map** *policy-map-name*	创建并进入 policymap 配置模式，policy-map-name 是要创建的 policymap 的名字 no 选项 删除一个已经存在的 policy map
[**no**] **class** *class-map-name*	创建并进入数据分类配置模式，class-map-name 是已经创建的 class map 名字 no 选项 删除该数据分类
[**no**]**set ip dscp** *new-dscp*	为该数据流中的 IP 报文设置新的 ip dscp 值；对于非 IP 报文，该设置不起作用； new-dscp 是要设置的新 DSCP 值，取值范围依产品不同而不同
police *rate-bps burst-byte* [**exceed-action** {**drop** | **dscp** *dscp-value*}] **no police**	限制该数据流的带宽和为带宽超限部分指定处理动作，rate-bps 是每秒钟带宽限制量(kbps)，burst-byte 猝发流量限制值(Kbyte)，drop 来丢弃带宽超限部分的报文，dscp dscp-value 改写带宽超限部分报文的 DSCP 值，dscp-value 取值范围依产品不同而不同

说明:

S3760 系列设备 Police 设置的速率带宽是可以被多个接口共享。若同一个 Policy Maps 被应用到多个接口上，则其中的 police 设置的速率带宽是多个接口共享的。要达到一个接口独占 Policy Maps 中的 police 设置的速率带宽，可通过配置一个被接口唯一关联的 Policy Maps（只需要 Policy Maps 的名称与其它的不一样，其关联的 Class 都可以和其它一样）。

S3760 系列设备不支持对带宽超限部分报文改写 DSCP 值。在 S3760 系列设备中，带宽超限部分报文只能被丢弃。

例如，以下的设置步骤创建了一个名为 Policy1 的 Policy-map，并将该 Policy-map 关联接口 Gigabitethernet 1/1

```
Ruijie(config)# policy-map policy1
Ruijie(config-pmap)# class class1
Ruijie(config-pmap-c)# set ip dscp 48
Ruijie(config-pmap-c)# exit
Ruijie(config-pmap)# exit
Ruijie(config)# interface gigabitethernet 1/1
Ruijie(config-if)# switchport mode trunk
Ruijie(config-if)# mls qos trust cos
Ruijie(config-if)# service-policy input policy1
```

50.2.7 配置接口应用 Policy Maps

您可以通过下面的设置步骤将 Policy Maps 应用到端口上

命令	说明
configure terminal	进入配置模式
interface *interface*	进入接口配置模式
[**no**] **service-policy** {**input** | **output**} **policy-map-name**	将创建的 Policy Map 应用到接口上；policy-map-name 是已经创建的 policy map 的名字，input 为输入限速,output 为输出限速

50.2.8 配置逻辑端口组应用 Policy Maps

您可以通过下面的设置步骤将 Policy Maps 应用到逻辑端口组上

命令	说明
configure terminal	进入配置模式
virtual-group *virtual-group-number*	进入逻辑端口组配置模式
[**no**] **service-policy** {**input** | **output**} **policy-map-name**	将创建的 Policy Map 应用到逻辑端口组上；policy-map-name 是已经创建的 policy map 的名字，input 为输入限速,output 为输出限速

🕮 **说明：**

目前 output 方向应用在逻辑端口组上未被支持。由于 class map 需要关联 acl，所以 acl 配置的所有限制均适用于 qos，具体请参考 acl 配置指南。

50.2.9 配置输出队列调度算法

您可以为端口的输出队列调度算法：WRR，SP 和 DRR，缺省情况下，输出队列算法为 WRR（带权重的队列轮转)

您可以通过以下步骤对端口优先级队列调度方式进行设置,详细算法请参照 QOS 概述。

命令	说明
configure terminal	进入配置模式
mls qos scheduler {sp |wrr | drr }	端口优先级队列调度方式，**sp** 为绝对优先级调度， **wrr** 为带帧数量权重轮转调度，**drr** 为带帧长度权重轮转调度
no mls qos scheduler	恢复为缺省 **wrr** 调度

例如，以下的设置步骤将端口的输出轮转算法设置成 SP：

```
Ruijie# configure terminal
Ruijie(config)# mls qos scheduler sp
Ruijie(config)# end
Ruijie# show mls qos scheduler
Global Multi-Layer Switching scheduling
Strict Priority
Ruijie#
```

说明：

S3760 系列产品支持 SP+DRR 调度。在使用 SP+DRR 调度方式时，SP、DRR 分组之间是严格优先级调度，调度优先级按队列号从大到小依次降低，DRR 分组内部按差额轮转进行调度，同时要求在 DRR 分组中的队列必须是连续的。例如可以配置队列 1，2，3，4 属于一个 DRR 分组，但不能配置队列 1，3，4 属于一个 DRR 分组。

50.2.10 配置输出轮转权重

您可以通过以下步骤设置端口的输出轮转权重

命令	说明
configure terminal	进入配置模式
{wrr-queue | drr-queue} bandwidth weight1...weightn	weight1...weightn 为指定的输出队列的权重值，个数及取值范围见缺省 QOS 设置
no {wrr-queue | drr-queue} bandwidth	no 选项恢复权重的缺省值

下面的例子将 wrr 调度权重设置为 1:2:3:4:5:6:7:8

```
Ruijie# configure terminal
Ruijie(config)# wrr-queue bandwidth 1 2 3 4 5 6 7 8
Ruijie(config)# end
Ruijie# show mls qos queueing
Cos-queue map:
cos qid
--- ---
0   1
1   2
2   3
3   4
4   5
5   6
6   7
7   8
wrr bandwidth weights:
qid weights
--- -------
0   1
1   2
2   3
3   4
4   5
5   6
6   7
7   8
Ruijie(config)#
```

50.2.11 配置 Cos-Map

您可以通过设置 Cos-Map 来选择报文输出时进入哪个输出队列，Cos-Map 的缺省

设置见缺省 QOS 配置

命令	说明
configure terminal	进入配置模式
priority-queue Cos-Map qid *cos0* [*cos1* [*cos2* [*cos3* [*cos4* [*cos5* [*cos6* [*cos7*]]]]]]]	qid 为队列 id,cos0..cos7 为指定和这个队列关联的 CoS 值。
no priority-queue cos-map	Cos-Map 恢复成缺省值

下面是设置 CoS Map 的例子

```
Ruijie# configure terminal
Ruijie(config)# priority-queue Cos-Map 1 2 4 6 7 5
Ruijie(config)# end
Ruijie# show mls qos queueing
Cos-queue map:
cos qid
--- ---
0   1
1   2
2   1
3   4
4   1
5   1
6   1
7   1

wrr bandwidth weights:
qid weights
--- -------
0   1
1   2
2   3
3   4
4   5
5   6
6   7
7   8
```

50.2.12 配置 CoS-to-DSCP Map

CoS-to-DSCP Map 用于将报文的 CoS 值映射到内部 DSCP 值,您可以通过以下步骤对 CoS-to-DSCP Map 进行设置 ,CoS-to-DSCP Map 的缺省设置见缺省

QOS 配置

命令	说明
configure terminal	进入配置模式
mls qos map cos-dscp *dscp1...dscp8* **no mls qos map cos-dscp**	修改 CoS-to-DSCP Map 的设置,dscp1...dscp8 是对应于 CoS 值 0～7 的 DSCP 值,DSCP 取值范围依产品不同而不同

例如如下配置：

```
Ruijie# configure terminal
Ruijie(config)# mls qos map cos-dscp 56 48 46 40 34 32 26 24
Ruijie(config)# end
Ruijie# show mls qos maps cos-dscp
cos dscp
--- ----
0   56
1   48
2   46
3   40
4   34
5   32
6   26
7   24
```

50.2.13 配置 DSCP-to-CoS Map

DSCP-to-CoS 用于将报文的内部 DSCP 值映射到 CoS 值，以便为报文选择输出队列

DSCP-to-CoS Map 的缺省设置见缺省 QOS 配置，您可以通过以下步骤对 DSCP-to-CoS Map 进行设置:

命令	说明
configure terminal	进入配置模式
mls qos map dscp-cos *dscp-list* **to** *cos*	设置 DSCP to COS Map， *dscp-list* ：要设置的 DSCP 值的列表，DSCP 值之间用空格分隔，取值范围依产品不同而不同,*cos* ：对应 DSCP 值的 CoS 值，取值范围为：0～7；
no mls qos map dscp-cos	设置为默认值

例如，以下的设置步骤将 DSCP 值 0、32、56 设置对应成 6：

```
Ruijie# configure terminal
Ruijie(config)# mls qos map dscp-cos 0 32 56 to 6
Ruijie(config)# show mls qos maps dscp-cos
dscp cos    dscp cos    dscp cos    dscp cos
---- ---    ---- ---    ---- ---    ---- ---
 0   6       1   0       2   0       3   0
 4   0       5   0       6   0       7   0
 8   1       9   1      10   1      11   1
12   1      13   1      14   1      15   1
16   2      17   2      18   2      19   2
20   2      21   2      22   2      23   2
24   3      25   3      26   3      27   3
28   3      29   3      30   3      31   3
32   6      33   4      34   4      35   4
36   4      37   4      38   4      39   4
40   5      41   5      42   5      43   5
44   5      45   5      46   5      47   5
48   6      49   6      50   6      51   6
52   6      53   6      54   6      55   6
56   6      57   7      58   7      59   7
60   7      61   7      62   7      63   7
```

50.2.14 配置端口速率限制

您可以通过以下步骤对端口速率限制进行设置

命令	说明
configure terminal	进入配置模式
interface *interface*	进入接口配置模式
rate-limit output bps *burst-size*	端口速率限制， output 为输出限速，bps 是每秒钟的带宽限制量(kbps)， burst-size 猝发流量限制值(Kbyte)
no rate-limit	取消端口限速

例如：

```
Ruijie# configure terminal
Ruijie(config)# interface gigabitEthernet 0/4
Ruijie(config-if)# rate-limit output 64 1024
Ruijie(config-if)# end
Ruijie#
```

50.2.15 配置 IPpre to DSCP Map

IPpre-to-Dscp 用于将报文的 IPpre 值映射到内部 DSCP 值, IPpre-to-DSCP Map 的缺省设置见缺省 QOS 配置,您可以通过以下步骤对 IPpre-to-Dscp Map 进行设置:

命令	说明
configure terminal	进入配置模式
mls qos map ip-precedence-dscp *dscp1...dscp8*	修改 IP-Precedence-to-Dscp Map 的设置,dscp1...dscp8 是对应于 IP-Precedence 值 0～7 的 DSCP 值
no mls qos map ip-prec-dscp	

例如如下配置：

```
Ruijie# configure terminal
Ruijie(config)# mls qos map ip-precedence-dscp 56 48 46 40 34 32 26 24
Ruijie(config)# end
Ruijie# show mls qos maps ip-prec-dscp
ip-precedence dscp
------------- ----
0     56
1     48
2     46
3     40
4     34
5     32
6     26
7     24
```

50.2.16 配置交换机 buffer

您可以配置交换机 buffer 管理处于 802.3x flow-control 状态或处于 QoS 状态。

命令	说明
configure terminal	进入配置模式
buffer management { **fc** | **qos** }	配置交换机的 buffer 管理模式 FC：802.3xflow-control QoS：QoS 模式
no buffer management	

例如如下配置交换机处于 qos 模式：

```
Ruijie# configure terminal
Ruijie(config)# buffer management qos
Ruijie(config)# end
Ruijie# show buffer management

%current port's buffer management mode: qos
```

50.2.17 配置交换机拥塞队列数控制

您可以配置交换机端口的拥塞队列数：

命令	说明
configure terminal	进入配置模式
Interface *interface*	进入接口配置模式
buffer management qos queue *queue-number*	配置交换机的端口拥塞队列数 queue-number： 端口支持的拥塞队列数，1～8 1 表示拥塞时各队列不区分优先级 8 表示拥塞时支持 8 个队列拥塞时，报文按优先级输出
[no|defaut]buffer management qos queue	配置端口的拥塞队列数为缺省值

例如如下配置交换机的 fastEthernet 0/4 端口的拥塞队列数为 8

```
Ruijie# configure terminal
Ruijie(config)# interface fastEthernet 0/4
Ruijie(config-if)# buffer management qos queue 8
Ruijie(config-if)# end
```

50.3 QOS 显示

50.3.1 显示 class-map

您可以通过以下步骤显示 class-map 内容

命令	说明
show class-map [*class-name*]	显示 class map 实体的内容

例如：

```
Ruijie# show class-map
Class Map cc
Match access-group 1
Ruijie#
```

50.3.2 显示 policy-map

您可以通过以下步骤显示 Policy-map 内容

命令	说明
show policy-map [*policy-name* [**class** *class-name*]]	显示 QoS policy map，policy-name 为选定的 policy map 名，指定 class class-name 时显示相应 policy map 绑定的 class map。

例如：

```
Ruijie# show policy-map
Policy Map pp
Class cc
Ruijie#
```

50.3.3 显示 mls qos interface

您可以通过以下步骤显示所有端口 qos 信息

命令	说明
show mls qos interface [*interface* | *policers*]	显示接口的 QoS 信息，Policers 选项显示接口应用的 Policy map

例如：

```
Ruijie# show mls qos interface gigabitEthernet 0/4
Interface: GigabitEthernet 0/4
Attached input  policy-map: pp
Default COS: trust dscp
Default COS: 6
Ruijie#show mls qos interface policers
Interface: GigabitEthernet 0/4
Attached input  policy-map: pp
Ruijie#
```

50.3.4 显示 mls qos virtual-group

您可以通过以下步骤显示所有端口 qos 信息

命令	说明
show mls qos virtual-group [*virtual-group-number* | **policers**]	显示逻辑端口组关联的 police 信息 Policers 选项显示逻辑端口组关联的 police

例如：

```
Ruijie# show mls qos virtual-group 1
Virtual-group: 1
Attached input  policy-map: pp
Ruijie# show mls qos virtual-group policers
Virtual-group: 1
Attached input  policy-map: pp
Ruijie#
```

50.3.5 显示 mls qos queueing

您可以通过以下步骤显示 qos 队列信息

命令	说明
show mls qos queueing	显示 QoS 队列信息, CoS-to-queue map， wrr weight 及 drr weight;

举例如下：

```
Ruijie# show mls qos queueing
Cos-queue map:
cos qid
--- ---
```

```
0   1
1   2
2   1
3   4
4   1
5   1
6   1
7   1
wrr bandwidth weights:
qid weights
--- -------
0   1
1   2
2   3
3   4
4   5
5   6
6   7
7   8
```

50.3.6 显示 mls qos scheduler

您可以通过以下步骤显示 QOS 调度方式

命令	说明
show mls qos scheduler	显示端口优先级队列调度方式

举例如下：

```
Ruijie# show mls qos scheduler
Global Multi-Layer Switching scheduling
Strict Priority
Ruijie#
```

50.3.7 显示 mls qos maps

您可以通过以下步骤显示 **mls qos maps** 对应表

命令	说明
show mls qos maps **[cos-dscp |dscp-cos|ip-prec-dscp]**	显示 **dscp-cos maps** **dscp-cos maps** **ip-prec-dscp maps**

举例如下：

```
Ruijie# show mls qos maps cos-dscp
cos dscp
--- ----
0   0
1   8
2   16
3   24
4   32
5   40
6   48
7   56
```

```
Ruijie# show mls qos maps dscp-cos
dscp cos    dscp cos    dscp cos    dscp cos
---- ---    ---- ---    ---- ---    ---- ---
0   6       1   0       2   0       3   0
4   0       5   0       6   0       7   0
8   1       9   1      10   1      11   1
12   1      13   1      14   1      15   1
16   2      17   2      18   2      19   2
20   2      21   2      22   2      23   2
24   3      25   3      26   3      27   3
28   3      29   3      30   3      31   3
32   6      33   4      34   4      35   4
36   4      37   4      38   4      39   4
40   5      41   5      42   5      43   5
44   5      45   5      46   5      47   5
48   6      49   6      50   6      51   6
52   6      53   6      54   6      55   6
56   6      57   7      58   7      59   7
60   7      61   7      62   7      63   7
```

```
Ruijie# show mls qos maps ip-prec-dscp
ip-precedence dscp
------------- ----
0     56
1     48
2     46
3     40
4     34
5     32
6     26
7     24
```

50.3.8 显示 mls qos rate-limit

您可以通过以下步骤显示端口速率限制信息

命令	说明
show mls qos rate-limit [**interface** *interface*]	显示[端口] 速率限制

```
Ruijie# show mls qos rate-limit
Interface: GigabitEthernet 0/4
rate limit input bps = 100 burst = 100
```

50.3.9 显示 show policy-map interface

您可以通过以下步骤显示端口 policymap 的配置

命令	说明
show policy-map interface *interface*	显示[端口] policymap 配置

```
Ruijie# show policy-map interface f0/1
FastEthernet 0/1  input (tc policy): pp
Class cc
set ip dscp 22
mark count 0
```

说明:

交换机设备目前不支持 mark count 的统计

50.3.10 显示交换机 buffer 管理模式

您可以通过以下步骤显示交换机 buffer 管理模式

命令	说明
show buffer management	显示交换机 buffer 管理模式

```
Ruijie# show buffer management
%current port's buffer management mode: qos
```

50.3.11 显示交换机拥塞队列数

您可以通过以下步骤显示交换机的拥塞队列数

命令	说明
show buffer management qos queue	查看各端口的拥塞队列数

```
Ruijie#show buffer management qos queue

Interface            Status  Queue

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

FastEthernet    0/1    Auto   8

FastEthernet    0/2    Admin  8

....

GigabitEthernet 0/52   Auto   8
```

50.3.12 显示 virtual-group

在特权模式下，请按如下步骤显示 virtual-group 设置。

命令	作用
show virtual-group [*virtual-group-number* | **summary**]	显示逻辑端口组信息。

```
Ruijie#show virtual-group 1
  virtual-group       member
  -------------       -------------------------
  1                   Gi0/2 Gi0/3 Gi0/4 Gi0/5
                      Gi0/6 Gi0/7 Gi0/8 Gi0/9 Gi0/10
Ruijie#show virtual-group summary
  virtual-group       member
  -------------       -------------------------
  1                   Gi0/1 Gi0/2 Gi0/3 Gi0/4
                      Gi0/5 Gi0/6 Gi0/7 Gi0/8 Gi0/9
  2                   Gi0/11 Gi0/12 Gi0/13 Gi0/14
              Gi0/15 Gi0/16 Gi0/17 Gi0/18 Gi0/19
```

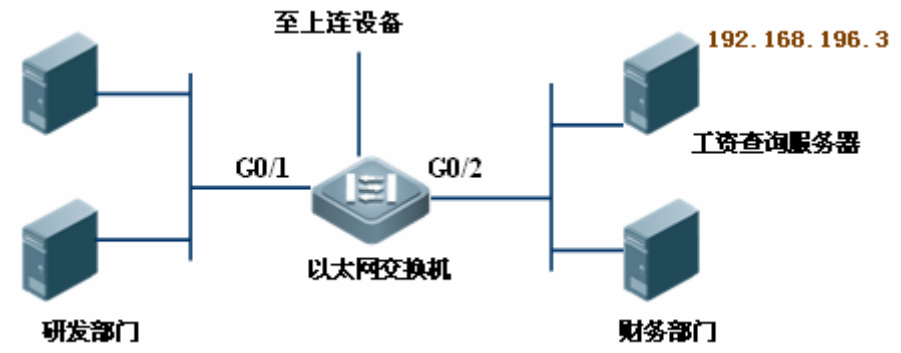
50.4 QOS 配置用例

50.4.1 基于分类报文的流量限速

50.4.1.1 应用需求

公司内网通过以太网交换机实现各部门之间的互连，其中财务部门由交换机的 G0/2 端口接入。现要求限制该部门工资查询服务器向外发送的最大流量不超过 512Kbps，并且对超出规格的流量做丢弃处理。

50.4.1.2 拓扑图



50.4.1.3 配置步骤

说明：

以下配置过程只列出了与 QOS ACL 相关的配置命令

#进入全局配置模式

```
Ruijie#configure
Enter configuration commands, one per line.  End with CNTL/Z.
```

#定义名为 salary_acl 的标准 ACL

```
Ruijie(config)#ip access-list standard salary_acl
```

#定义规则，允许工资服务器的流量

```
Ruijie(config-std-nacl)#permit host 192.168.217.223
```

#退回到全局配置模式

```
Ruijie(config-std-nacl)#exit
```

#创建名为 salaryclass 的 class map，进入 class-map 配置模式

```
Ruijie(config)#class-map salaryclass
```

#定义匹配的规则

```
Ruijie(config-cmap)#match access-group salary_acl
```

#退回到全局配置模式

```
Ruijie(config-cmap)#exit
```

#创建名为 salarypolicy 的策略，进入 policy-map 配置模式

```
Ruijie(config)#policy-map salarypolicy
```

#指定该策略执行的分类规则为 salaryclass

```
Ruijie(config-pmap)#class salaryclass
```

#限制工资查询服务器向外发送的最大流量不超过 512Kbps，猝发流量限制值为 32 Kbyte，并且对超出规格的流量做丢弃处理。

```
Ruijie(config-pmap-c)#police 512 32  exceed-action drop
```

#退回到 class-map 配置模式

```
Ruijie(config-pmap-c)#exit
```

#退回到全局配置模式

```
Ruijie(config-pmap)#exit
```

#进入 G0/2 的接口配置模式

```
Ruijie(config)#interface gigabitEthernet 0/2
```

#把 salarypolicy 策略应用在 G0/2 口的入口方向

```
Ruijie(config-if)#service-policy input salarypolicy
```

#退回到特权模式

```
Ruijie(config-if)#end
```

#在特权模式下用 show 命令查看相应的配置

```
Ruijie#show mls qos interface policers
Interface: GigabitEthernet 0/2
Attached input  policy-map: salarypolicy
Ruijie#show policy-map salarypolicy
 Policy Map salarypolicy
    Class salaryclass
      police 512 32 exceed-action drop
```

```
Ruijie#show class-map salaryclass
 Class Map salaryclass
   Match access-group salary_acl

Ruijie#show access-lists salary_acl
ip access-list standard salary_acl
 10 permit host 192.168.217.223
```

51 VRRP 配置

51.1 概述

VRRP (Virtual Router Redundancy Protocol，虚拟路由冗余协议)设计采用主备模式，以保证当主路由设备发生故障时，备份路由设备可以在不影响内外数据通信的前提下进行功能切换，且不需要再修改内部网络的参数。VRRP 组内多个路由设备都映射为一个虚拟的路由设备。VRRP 保证同时有且只有一个路由设备在代表虚拟路由设备进行包的发送，而主机则是把数据包发向该虚拟路由设备，这个转发数据包的路由设备被选择成为主路由设备。如果这个主路由设备在某个时候由于某种原因而无法工作的话，则处于备份状态的路由设备将被选择来代替原来的主路由设备。VRRP 使得局域网内的主机看上去只使用了一个路由设备，并且即使在它当前所使用的首跳路由设备失败的情况下仍能够保持路由的连通性。

RFC 2338 中定义了 VRRP 类型的 IP 报文格式及其运作机制，VRRP 报文是一类指定目的地址的组播报文，该报文由主路由设备定时发出来标志其运行正常同时该报文也用于选举主路由设备。VRRP 允许为 IP 局域网承担路由转发功能的路由设备失效后，局域网中另外一个路由设备将自动接管失效的路由设备，从而实现 IP 路由的热备份与容错，同时也保证了局域网内主机通讯的连续性和可靠性。一个 VRRP 应用组通过多台路由设备来实现冗余，但是任何时候只有一台路由设备作为主路由设备来承担路由转发功能，其他的为备份路由设备，VRRP 应用组中不同路由设备间的切换对局域网内的主机则是完全透明的。RFC 2338 规定了路由设备的切换规则：

- VRRP 协议采用简单竞选的方法选择主路由设备。首先比较同一个 VRRP 组内的各台路由设备对应接口上设置的 VRRP 优先级的大小，优先级最大的为主路由设备，它的状态变为 Master。若路由设备的优先级相同，则比较对应网络接口的主 IP 地址，主 IP 地址大的就成为主路由设备，由它提供实际的路由转发服务。

- 主路由设备选出后，其它路由设备作为备份路由设备(状态变为 Backup)，并通过主路由设备定时发出的 VRRP 报文监测主路由设备的状态。当正常工作时，主路由设备会每隔一段时间发送一个 VRRP 组播报文，称为通告报文，以通知备份路由设备：主路由设备处于正常工作状态。如果组内的备份路由设备在设定的时间段没有接收到来自主路由设备的报文，则将自己状态转为 Master。当组内有多台状态为 Master 路由设备时，重复 1)的竞选过程。通过这样一个过程就会将优先级最大的路由设备选成新的主路由设备，从而实现 VRRP 的备份功能。

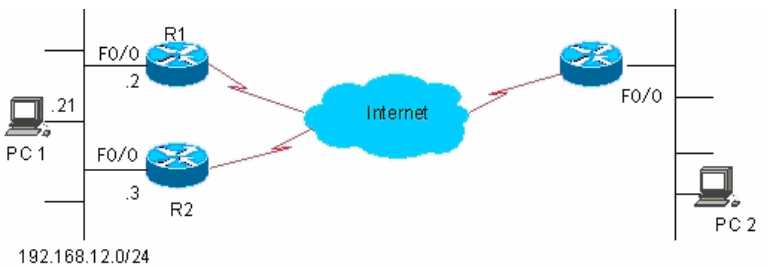


图 1　VRRP 工作原理图

一旦在一个 VRRP 备份组选举出它的主路由设备，局域网内的主机将通过主路由设备进行路由转发。通讯过程可以由图 1 来说明。在图 1 中，路由设备 R1 和 R2 均通过以太网口 Fa0/0 与局域网 192.168.12.0/24 连接，路由设备 R1 与 R2 的 Fa0/0 接口上设置了 VRRP，局域网内的主机都以该 VRRP 组的虚拟路由设备 IP 地址作为默认网关。对于局域网内的主机而言，它们只能感受到由 VRRP 组的虚拟路由设备，而实际承担路由转发功能的 VRRP 组的主路由设备对它们而言则是透明的。譬如，局域网内的主机 PC 1 如果要与其它网络内的主机 PC 2 通讯，PC 1 会以虚拟路由设备为默认网关来发送通向 PC 2 的网络数据包，VRRP 组中的主路由设备在接收到该数据包后会将该数据包转发给 PC 2。在这个通讯过程中，PC 1 只能感受到虚拟路由设备而不知道扮演虚拟路由设备角色的主路由设备究竟是 R1 还是 R2，在这个 VRRP 组中的主路由设备是在 R1 与 R2 之间选举产生的，一旦主路由设备失效，那么另外一台将自动成为主路由设备。

51.2 VRRP 的应用

VRRP 有两种应用模式：基本应用与高级应用。其中基本应用是使用单备份组实现简单路由冗余，高级应用是使用多备份组同时实现路由冗余与负载均衡。

51.2.1 路由冗余

VRRP 的基本应用可以通过图 2 示例来说明。

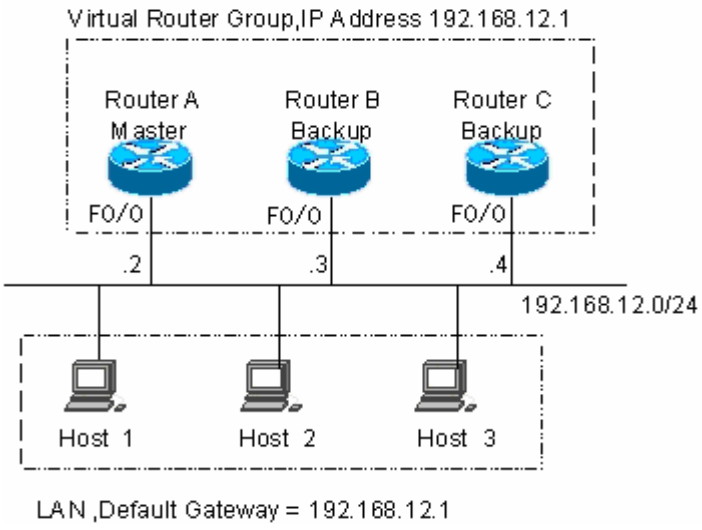


图 2　VRRP 基本应用示意图

如图 2，路由设备 A、B 以及 C 均使用以太网口与局域网连接，并在其连接局域网的以太网接口上设置了 VRRP，它们处于同一个 VRRP 组并且该 VRRP 组的虚拟 IP 地址为 192.168.12.1，其中路由设备 A 经选举为 VRRP 的主路由设备，路由设备 B 与 C 作为备份路由设备。局域网内的主机 1、2 以及 3 以虚拟路由设备的 IP 地址 192.168.12.1 作为网关。局域网内主机发往其它网络的数据包将由主路由设备(在图 2-2 中是路由设备 A)进行路由转发。一旦路由设备 A 失效，将在路由设备

B 与 C 之间选举出主路由设备来承担虚拟路由设备的路由转发功能，由此实现了简单路由冗余。

51.2.2 负载均衡

VRRP 的高级应用可以通过图 3 示例来说明。

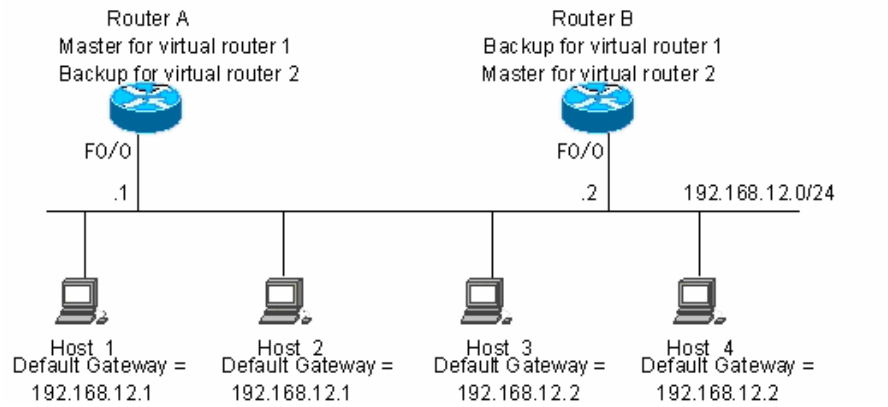


图 3　VRRP 高级应用示意图

如图 3，设置了两个虚拟路由设备。对于虚拟路由设备 1，路由设备 A 使用以太网口 Fa0/0 的 IP 地址 192.168.12.1 作为虚拟路由设备的 IP 地址，这样路由设备 A 就成为主路由设备，而路由设备 B 成为备份路由设备。对于虚拟路由设备 2，路由设备 B 使用以太网口 Fa0/0 的 IP 地址 192.168.12.2 作为虚拟路由设备的 IP 地址，这样路由设备 B 就成为主路由设备，而路由设备 A 成为备份路由设备。在局域网内，主机 1 和主机 2 使用虚拟路由设备 1 的 IP 地址 192.168.12.1 作为默认网关，主机 3 和主机 4 使用虚拟路由设备 2 的 IP 地址 192.168.12.2 作为默认网关。在 VRRP 这个应用中，路由设备 A 和路由设备 B 实现了路由冗余，并同时分担了来自局域网的流量即实现了负载平衡。

51.3 VRRP 的配置

51.3.1 VRRP 配置任务列表

VRRP 协议是适用于多播或者广播的局域网如以太网等。VRRP 的配置就集中在以太网接口上。其配置任务有：

- 启动 VRRP 备份功能(必须)；
- 设置 VRRP 备份组的验证字符串(可选)；
- 设置 VRRP 备份组的通告发送间隔(可选)；
- 设置路由设备在 VRRP 备份组中的抢占模式(可选)；
- 设置路由设备在 VRRP 备份组中的优先级(可选)；
- 设置 VRRP 备份组监视的接口(可选)；

- 设置 VRRP 备份组监视的 IP 地址(可选)；
- 设置 VRRP 通告定时设备学习功能(可选)；
- 设置路由设备在 VRRP 备份组的描述字符串(可选)；
- 设置 VRRP 备份组延迟启动(可选)。

当然，并不是这里的每一项都要设置。对一个 VRRP 备份组设置哪些项取决于用户的使用需要。

51.3.2 启动 VRRP 备份功能

通过设置备份组号和虚拟 IP 地址可以在指定的局域网段上添加一个备份组从而启动对应的以太网接口的 VRRP 备份功能。

命令	目的
Ruijie(config-if)# **vrrp** *group* **ip** *ipaddress* [**secondary**]	启用 VRRP
Ruijie(config-if)# **no vrrp** *group* **ip** *ipaddress* [**secondary**]	关闭 VRRP

备份组号 Group 取值范围为 1~255。如果不指定虚拟 IP 地址，路由设备就不会参与 VRRP 备份组。如果不使用 Secondary 参数，那么设置的 IP 地址将成为虚拟路由设备的主 IP 地址。

说明：

如果 VRRP 组的虚拟 IP 地址(Primary 或者 Secondary)与所在以太网接口上的 IP 地址(Primary 或者 Secondary)一致，那么就认为该 VRRP 组占用(Own)了以太网接口实际 IP 地址，此时该 VRRP 组的优先级为 255，如果对应的以太网接口可用，那么该 VRRP 组将自动处于 Master 状态。

在 NMX-2GEH 线卡上，每个接口最多支持 14 个 VRRP 备份组。如果配置的 VRRP 组数超过 14，则系统会提示错误。

51.3.3 设置 VRRP 备份组的验证字符串

VRRP 支持明文密码验证模式以及无验证模式。设置 VRRP 备份组的验证字符串的同时也设定该 VRRP 组处于明文密码验证模式。VRRP 备份组成员必须处于相同的验证模式下才可能正常通讯。明文密码验证模式下，在同一个 VRRP 组中的路由设备必须设置相同的验证口令。明文验证口令不能保证安全性，它只是用来防止/提示错误的 VRRP 配置。

命令	目的
Ruijie(config-if)# **vrrp** *group* **authentication** *string*	设置 VRRP 的验证字符串
Ruijie(config-if)# **no vrrp** *group* **authentication**	设置 VRRP 处于无验证模式

缺省状态下，VRRP 处于无验证模式。在明文密码验证模式下，明文密码长度不能超过 8 个字节。

51.3.4 设置 VRRP 备份组的通告发送间隔

命令	目的
Ruijie(config-if)# **vrrp** *group* **timers advertise** *interval*	设置主路由设备 VRRP 通告间隔
Ruijie(config-if)# **no vrrp** *group* **timers advertise**	恢复主路由设备 VRRP 通告间隔的系统默认设置

如果当前路由设备 VRRP 组中的主路由设备，它将以设定的间隔发送 VRRP 通告来通告自己的 VRRP 状态、优先级以及其它信息。缺省状态下，系统默认主路由设备的 VRRP 通告发送间隔为 1 秒。

说明：

在没有设置 VRRP 定时设备学习功能的时候，同一个 VRRP 备份组要设置相同的 VRRP 通告发送间隔，否则处于备份状态的路由设备将会丢弃接收到的 VRRP 通告。

51.3.5 设置路由设备在 VRRP 备份组中的抢占模式

如果 VRRP 组工作在抢占模式下，一旦它发现自己的优先级高于当前 Master 的优先级，它将抢占成为该 VRRP 组的主路由设备。如果 VRRP 组工作在非抢占模式下，即便它发现自己的优先级高于当前 Master 的优先级，它也不会抢占成为该 VRRP 组的主路由设备。VRRP 组使用以太网接口 IP 地址情况下，抢占模式是否设置意义不大，因为此时该 VRRP 组具有最大优先级，它自动成为该 VRRP 组中的主路由设备。

命令	目的
Ruijie(config-if)# **vrrp** *group* **preempt** [**delay** *seconds*]	设置 VRRP 备份组处于抢占模式

Ruijie(config-if)# **no vrrp** group **preempt** [**delay**]	设置 VRRP 备份组处于非抢占模式

可选参数 Delay Seconds 定义了处于备份状态的 VRRP 路由设备准备宣告自己拥有 Master 身份之前的延迟，缺省值为 0 秒。一旦启用 VRRP 功能，VRRP 组默认工作在抢占模式下。

51.3.6 设置路由设备在 VRRP 备份组中的优先级

VRRP 协议规定根据路由设备的优先级参数来确定在备份组中每台路由设备的地位。工作在抢占模式下具有最高优先级并且已获得虚拟 IP 地址的路由设备将成为该备份组的活动的(或主)路由设备，同一个备份组中低于该路由设备优先级的其它路由设备将成为备份的(或监听的)路由设备。一旦启用 VRRP 功能，VRRP 组默认其优先级为 100。

命令	目的
Ruijie(config-if)# **vrrp** *group* **priority** *level*	设置 VRRP 备份组的优先级
Ruijie(config-if)# **no vrrp** *group* **priority**	恢复 VRRP 优先级的默认值

优先级 Level 的取值范围为 1~254。如果 VRRP 虚拟 IP 地址与所在以太网接口上真实 IP 地址一致，对应的 VRRP 组的优先级就为 255，此时无论 VRRP 组是否处于抢占模式，对应的 VRRP 组都会自动处于 Master 状态(只要对应的以太网接口可用)。

51.3.7 设置 VRRP 备份组监视的接口

在配置了 VRRP 备份组监视的接口后，系统将根据所监视接口的状态动态地调整本路由设备的优先级。一旦所监视的接口状态变为不可用就按照设置的数值减少本路由设备在 VRRP 备份组中的的优先级，而此时同一个备份组中接口状态更稳定并且优先级更高的其它路由设备就可以成为该 VRRP 备份组的活动的(或主)路由设备。

命令	目的
Ruijie(config-if)# **vrrp** *group* **track** *interface-type number* [*interface –priority*]	设置 VRRP 备份组监视的接口
Ruijie(config-if)# **no vrrp** *group* **track** *interface-type number*	取消 VRRP 备份组监视接口设置

缺省状态下，系统没有设置 VRRP 备份组监视的接口。参数 Interface -Priority 取值范围为 1~255。如果参数 Interface -Priority 缺省，系统会取默认值即 10。

说明：

被监视的接口只允许是三层可路由的逻辑接口(如 Routed Port ，SVI ，Loopback，Tunnel 等等)。

51.3.8 设置 VRRP 备份组监视的 IP 地址

在配置了 VRRP 备份组监视的 IP 地址后，系统将根据所监视的地址是否可达来动态地调整本设备的优先级。一旦所监视的 IP 地址变为不可达，即 ping 不通，就按照设置的数值减少本设备在 VRRP 备份组中的的优先级，而此时同一个备份组中优先级更高的其它路由设备就可以成为该 VRRP 备份组的活动的(或主)路由设备。该命令的可选参数 interval 是探测该目标地址是否可达的间隔时间，可选参数 timeout 是判定超时、即目标不可达的时间。

命令	目的
Ruijie(config-if)# **vrrp** *group* **track** *ip-address* [[[**interval** *interval-value*] **timeout** *timeout-value*] *priority*]	设置 VRRP 备份组监视的 IP 地址
Ruijie(config-if)# **no vrrp** *group* **track** *ip-address*	取消 VRRP 备份组监视地址设置

缺省状态下，系统没有设置 VRRP 备份组监视的地址。参数 interval-value 取值范围为 1~3600 秒，如果该参数缺省，系统会取其默认值即 3 秒。参数 timeoute-value 取值范围为 1~60 秒，如果该参数缺省，系统会取其默认值即 1 秒，注意：该值必须小于或等于 interval-value。参数 priority 取值范围为 1~255，如果该参数缺省，系统会取默认值即 10。

51.3.9 设置 VRRP 通告定时设备学习功能

一旦启用了定时设备学习功能，如果当前路由设备是 VRRP 备份路由设备，在设置了定时设备学习功能后，它会从主路由设备的 VRRP 通告中学习 VRRP 通告发送间隔，并由此来计算 Master 路由设备失效判断间隔，而不是使用自己本地设置的 VRRP 通告发送间隔来计算。本命令可以实现 Backup 路由设备与 Master 路由设备的 VRRP 通告发送定时设备同步。

命令	目的
Ruijie(config-if)# **vrrp** *group* **timers learn**	设置定时设备学习功能
Ruijie(config-if)# **no vrrp** *group* **timers learn**	取消定时设备学习功能

缺省状态下，系统没有为 VRRP 组设置定时设备学习功能。

说明：

在 VRRP 备份路由设备接收到的 VRRP 通告中的通告发送间隔与本地设置的通告发送间隔不一致的时候，如果 VRRP 备份路由设备上没有设置定时设备学习功能，VRRP 备份路由设备将会丢弃该 VRRP 通告否则 VRRP 备份路由设备会接收该 VRRP 通告并依据其中通告间隔来计算 VRRP 的 Master 路由设备失效判断间隔。

51.3.10 设置路由设备在 VRRP 备份组的描述字符串

本命令为 VRRP 组设置描述符，可以便于区分 VRRP 组。

命令	目的
Ruijie(config-if)# **vrrp** *group* **description** *text*	设置 VRRP 组描述字符串
Ruijie(config-if)# **no vrrp** *group* **description**	取消 VRRP 组描述字符串设置

缺省状态下，VRRP 备份组没有设置任何描述字符串。VRRP 备份组描述字符串长度不超过 80。

说明：

如果 VRRP 备份组描述字符串中包含空格，就必须使用”“”以及”” ”来标识描述字符串。

51.3.11 设置 VRRP 备份组延迟启动

本命令配置某个接口上 VRRP 备份组的延迟启动时间；延迟时间有两种：系统启动时的延迟时间，与接口状态变为活动时的延迟时间，可以分别配置，也可同时配置。

在非抢占模式下，优先级较高的 VRRP 备份组启动时，不会抢占同一备份组的主设备。但在某些情况下，即使配置了非抢占模式，刚启动的 VRRP 备份组也会抢占成为 VRRP 主设备。这是因为设备启动或者接口刚变为活动时，该接口上的 VRRP 备份组没有及时收到同一备份组的主设备发出的 VRRP 报文。

这时可以通过配置本命令，使 VRRP 备份组延迟启动。配置本命令后，当系统启动，或者接口状态变为活动时，该接口上的 VRRP 备份组不会立即启动；而是等待相应的延迟时间后再启动 VRRP 备份组，保证非抢占配置不会失效。

如果在延迟启动 VRRP 时该接口上接收到 VRRP 报文，则会取消延迟，立即启动 VRRP 协议。

命令	目的
Ruijie(config-if)**# vrrp delay** { **minimum** *min-seconds* | **reload** *reload-seconds* }	设置接口上 VRRP 备份组延迟启动时间
Ruijie(config-if)**# no vrrp delay**	取消接口上 VRRP 备份组延迟启动设置

缺省状态下，接口没有配置 VRRP 备份组延迟启动。两种 VRRP 备份组延迟启动时间的取值范围均为 0~60 秒。

51.4 VRRP 的监控与维护

锐捷产品提供命令 Show Vrrp 与 Debug Vrrp 来监控与维护 VRRP。使用命令 **show vrrp** 可以考察本地路由设备的 VRRP 状态，使用 Debug Vrrp 可以考察 VRRP 组的状态变化、VRRP 通告收发以及 VRRP 事件等信息。

51.4.1 show vrrp

锐捷产品提供以下的 **show vrrp** 命令来考察本地路由设备的 VRRP 状态。

命令	目的
Ruijie**# show vrrp** [**brief** | *group*]	查看当前的 VRRP 状态
Ruijie**# show vrrp interface** *type number* [**brief**]	显示指定网络接口上 VRRP 状态

下面给出使用这些命令的示例：

1. **show vrrp** 命令

```
Ruijie# show vrrp
FastEthernet 0/0 - Group 1
State is Backup
Virtual IP address is 192.168.201.1 configured
Virtual MAC address is 0000.5e00.0101
Advertisement interval is 3 sec
Preemption is enabled
min delay is 0 sec
Priority is 100
Master Router is 192.168.201.213 , pritority is 120
Master Advertisement interval is 3 sec
Master Down interval is 9 sec
FastEthernet 0/0 - Group 2
```

```
State is Master
Virtual IP address is 192.168.201.2 configured
Virtual MAC address is 0000.5e00.0102
Advertisement interval is 3 sec
Preemption is enabled
min delay is 0 sec
Priority is 120
Master Router is 192.168.201.217 (local), priority is 120
Master Advertisement interval is 3 sec
Master Down interval is 9 sec
```

上面的显示信息包括以太网口名称、接口上设置的 VRRP 备份组号、状态、优先级、抢占方式、VRRP 通告间隔、虚拟 IP 地址、虚拟 MAC 地址、Master 路由设备 IP 地址、Master 路由设备优先级、Master 路由设备通告间隔、Master 路由设备失效判断间隔、当前 VRRP 备份组监视的接口以及对应的优先级改变尺度。

2. **show vrrp brief** 命令

```
Ruijie# show vrrp brief
Interface      Grp Pri Time Own Pre State   Master addr     Group
addr
FastEthernet0/0 1  100  -    -    P  Backup 192.168.201.213
192.168.201.1
FastEthernet0/0 2  120  -    -   P  Master  192.168.201.217
192.168.201.2
```

上面的显示信息包括以太网口名称、接口上设置的 VRRP 备份组号、状态、优先级、抢占方式、虚拟 IP 地址、Master 路由设备 IP 地址。

3. **show vrrp interface** 命令

```
Ruijie# show vrrp interface FastEthernet 0/0
FastEthernet 0/0 - Group 1
State is Backup
Virtual IP address is 192.168.201.1 configured
Virtual MAC address is 0000.5e00.0101
Advertisement interval is 3 sec
Preemption is enabled
min delay is 0 sec
Priority is 100
Master Router is 192.168.201.213 , pritority is 120
Master Advertisement interval is 3 sec
Master Down interval is 9 sec
FastEthernet 0/0 - Group 2
State is Master
Virtual IP address is 192.168.201.2 configured
Virtual MAC address is 0000.5e00.0102
Advertisement interval is 3 sec
Preemption is enabled
```

```
min delay is 0 sec
Priority is 120
Master Router is 192.168.201.217 (local), priority is 120
Master Advertisement interval is 3 sec
Master Down interval is 9 sec
Ruijie#
```

上面的显示信息包括指定的以太网口名称、接口上设置的 VRRP 备份组号、状态、优先级、抢占方式、VRRP 通告间隔、虚拟 IP 地址、虚拟 MAC 地址、Master 路由设备 IP 地址、Master 路由设备优先级、Master 路由设备通告间隔、Master 路由设备失效判断间隔、当前 VRRP 备份组监视的接口以及对应的优先级改变尺度。

51.4.2 debug vrrp

锐捷产品提供以下的 **debug vrrp** 命令来提供本地路由设备的 VRRP 状态调试信息：

命令	目的
Ruijie**# debug vrrp errors**	打开 VRRP 出错提示调试开关
Ruijie**# no debug vrrp errors**	关闭 VRRP 出错提示调试开关
Ruijie**# debug vrrp events**	打开 VRRP 事件调试开关
Ruijie**# no debug vrrp events**	关闭 VRRP 事件调试开关
Ruijie**# debug vrrp packets**	打开 VRRP 报文调试开关
Ruijie**# no debug vrrp packets**	关闭 VRRP 报文调试开关
Ruijie**# debug vrrp state**	打开 VRRP 状态调试开关
Ruijie**# no debug vrrp state**	关闭 VRRP 状态调试开关
Ruijie**# debug vrrp**	打开 VRRP 调试开关
Ruijie**# no debug vrrp**	关闭 VRRP 调试开关

下面给出使用这些命令的示例：

1. **debug vrrp** 命令

```
Ruijie# debug vrrp
Ruijie#
VRRP: Grp 1 Advertisement priority 120, ipaddr 192.168.201.213
VRRP: Grp 1 Event - Advert higher or equal priority
%VRRP-6-STATECHANGE: FastEthernet 0/0 Grp 1 state Master ->
Backup
```

```
VRRP: Grp 1 Advertisement from 192.168.201.213 has invalid
virtual address 192.168.1.1
%VRRP-6-STATECHANGE: FastEthernet 0/0 Grp 1 state Backup ->
Master
Ruijie#
```

debug vrrp 命令相当于同时执行命令 debug vrrp errors、debug vrrp events、debug vrrp packets 以及命令 debug vrrp state。

2. **debug vrrp errors** 命令

```
Ruijie# debug vrrp errors
Ruijie#
VRRP: Grp 1 Advertisement from 192.168.201.213 has invalid
virtual address 192.168.1.1
VRRP: Grp 1 Advertisement from 192.168.201.213 has invalid
virtual address 192.168.1.1
VRRP: Grp 1 Advertisement from 192.168.201.213 has invalid
virtual address 192.168.1.1
```

上面的显示信息表明接收到来自 192.168.201.213 针对 VRRP 组 1 的 VRRP 通告，在该通告中的虚拟 IP 地址 192.168.1.1 不存在于本地 VRRP 组 1。

3. **debug vrrp events** 命令

```
Ruijie# debug vrrp events
Ruijie#
VRRP: Grp 1 Event - Advert higher or equal priority
VRRP: Grp 1 Event - Advert higher or equal priority
VRRP: Grp 1 Event - Advert higher or equal priority
Ruijie#
```

上面的显示信息表明本地 VRRP 组接收到的 VRRP 通告(Advertisement)中的优先级不低于本地优先级。

4. **debug vrrp packets** 命令

```
Ruijie# debug vrrp packets
Ruijie#
VRRP: Grp 2 sending Advertisement checksum DD4D
VRRP: Grp 2 sending Advertisement checksum DD4D
VRRP: Grp 2 sending Advertisement checksum DD4D
```

上面的显示信息表明本地 VRRP 组 2 正在发送 VRRP 通告，其 VRRP 校验和为 0XDD4D。

```
Ruijie# debug vrrp packets
Ruijie#
VRRP: Grp 1 Advertisement priority 120, ipaddr 192.168.201.213
VRRP: Grp 1 Advertisement priority 120, ipaddr 192.168.201.213
VRRP: Grp 1 Advertisement priority 120, ipaddr 192.168.201.213
```

上面的显示信息表明本地接收到来自 192.168.201.213 针对 VRRP 组 1 的 VRRP 通告，其优先级为 120。

5. **debug vrrp state** 命令

```
Ruijie# debug vrrp state
VRRP State debugging is on
Ruijie#
%VRRP-6-STATECHANGE: FastEthernet 0/0 Grp 2 state Master ->
Backup
%VRRP-6-STATECHANGE: FastEthernet 0/0 Grp 2 state Backup ->
Master
Ruijie# config terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Ruijie(config)# interface fastethernet 0/0
Ruijie(config-if)# no shutdown
Ruijie(config-if)# end
Ruijie#
%VRRP-6-STATECHANGE: FastEthernet 0/0 Grp 2 state Master ->
Init
Ruijie#
```

上面的显示信息表明 Fastethernet 0/0 上的 VRRP 组状态在 Master，Backup 以及 Init 之间转换。

51.5 VRRP 的典型配置示例

在图 4 所示的连接中，在路由设备 R1 与 R2 上配置了 VRRP 备份组来为内部网段 192.168.201.0 /24 提供 VRRP 服务，而在路由设备 R3 上没有配置 VRRP 而只是配置了普通路由功能。下面的配置中将只给出路由设备 R1 与 R2 的 VRRP 相关配置。

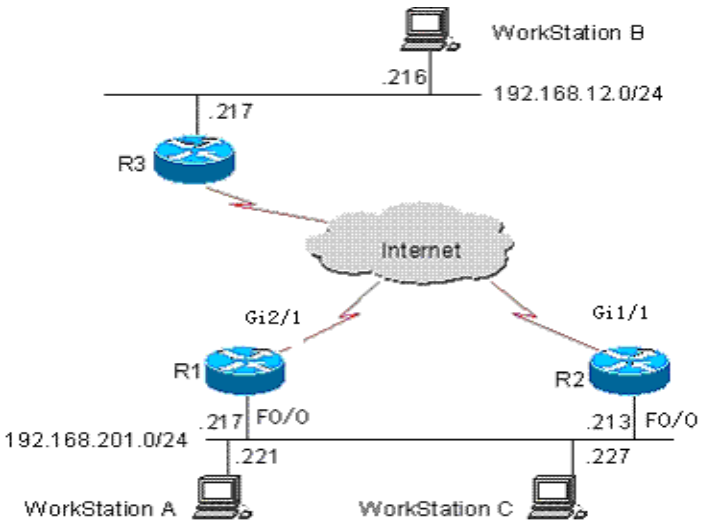


图 4　建立 VRRP 环境的网络连接示意图

在下面的配置示例中，路由设备 R3 的配置是不变的。下面给出路由设备 R3 的配置：

```
!
!
hostname "R3"
!
!
!
interface FastEthernet 0/0
no switchport
ip address 192.168.12.217 255.255.255.0
!
interface GigabitEthernet 1/1
no switchport
ip address 60.154.101.5 255.255.255.0
!
interface GigabitEthernet 2/1
no switchport
ip address 202.101.90.61 255.255.255.0
!
router ospf
network 202.101.90.0 0.0.0.255 area 10
network 192.168.12.0 0.0.0.255 area 10
network 60.154.101.0 0.0.0.255 area 10
!
!
!
end
```

51.5.1 VRRP 单备份组配置示例

按照图 4 建立连接。在这个配置示例中，用户工作站群(192.168.201.0/24)使用路由设备 R1 与 R2 组成的备份组，并将其网关指向该备份组设置的虚拟路由设备的 IP 地址 192.168.201.1，经由虚拟路由设备 192.168.201.1 访问远程用户工作站群(其工作网络为 192.168.12.0 /24)。在这里 R1 被设置成 VRRP 的 Master 路由设备。正常情况下，路由设备 R1 作为活动路由设备提供网关(192.168.201.)的功能，当路由设备 R1 由于关机或者出现故障而不可到达时，路由设备 R2 将替代它来提供网关(192.168.201.1)的功能。下面分别给出路由设备 R1 与 R2 的相关配置。

路由设备 R1 的配置：

```
!
!
hostname "R1"
!
```

```
!
interface FastEthernet 0/0
no switchport
ip address 192.168.201.217 255.255.255.0
vrrp 1 priority 120
vrrp 1 timers advertise 3
vrrp 1 ip 192.168.201.1
!
interface GigabitEthernet 2/1
no switchport
ip address 202.101.90.63 255.255.255.0
!
router ospf
network 202.101.90.0 0.0.0.255 area 10
network 192.168.201.0 0.0.0.255 area 10
!
```

路由设备 R2 的配置：

```
!
hostname "R2"
!
interface FastEthernet 0/0
no switchport
ip address 192.168.201.213 255.255.255.0
vrrp 1 ip 192.168.201.1
vrrp 1 timers advertise 3
!
interface GigabitEthernet 1/1
no switchport
ip address 60.154.101.3 255.255.255.0
!
!
router ospf
network 60.154.101.0 0.0.0.255 area 10
network 192.168.201.0 0.0.0.255 area 10
!
!
end
```

可见，路由设备 R1 与 R2 同处于 VRRP 备份组 1 中，指向相同的虚拟路由设备的 IP 地址(192.168.201.1)并且均处于 VRRP 的抢占模式下。由于路由设备 R1 的 VRRP 备份组优先级为 120，而路由设备 R2 的 VRRP 备份组优先级取默认值 100，所以路由设备 R1 在正常情况下充当 VRRP 的 Master 路由设备。

51.5.2 使用 VRRP 监视接口配置示例

按照图 4 建立连接。在这个配置示例中，用户工作站群(192.168.201.0/24)使用路由设备 R1 与 R2 组成的备份组，并将其网关指向该备份组设置的虚拟路由设备的 IP 地址 192.168.201.1，经由虚拟路由设备 192.168.201.1 访问远程用户工作站群(其工作网络为 192.168.12.0/24)。在这里 R1 被设置成 VRRP 的 Master 路由设备。与单备份组配置示例不同的是，在这个配置示例中，路由设备 R1 中设置了 VRRP 监视接口 GigabitEthernet 2/1。正常情况下，路由设备 R1 作为活动路由设备提供虚拟网关(192.168.201.1)的功能，当路由设备 R1 由于关机或者出现故障而不可到达时，路由设备 R2 将替代它来提供虚拟网关(也就是虚拟路由设备的地址 192.168.201.1)的功能。特别的是在路由设备 R1 的与广域网的接口 GigabitEthernet 2/1 不可用的时候，路由设备 R1 将会按照设置降低自己的 VRRP 备份组的优先级，从而使得路由设备 R2 有机会成为主动路由设备并提供虚拟网关(192.168.201.1)的功能；如果此后路由设备 R1 与广域网的接口 GigabitEthernet 2/1 恢复正常，那么路由设备 R1 将恢复自己的 VRRP 备份组优先级再次成为主动路由设备并提供虚拟网关的功能。下面分别给出路由设备 R1 与 R2 的相关配置。

路由设备 R1 的配置：

```
!
!
hostname "R1"
!
!
interface FastEthernet 0/0
no switchport
ip address 192.168.201.217 255.255.255.0
vrrp 1 priority 120
vrrp 1 timers advertise 3
vrrp 1 ip 192.168.201.1
vrrp 1 track GigabitEthernet 2/1 30
!

interface GigabitEthernet 2/1
no switchport
ip address 202.101.90.63 255.255.255.0
!
router ospf
network 202.101.90.0 0.0.0.255 area 10
network 192.168.201.0 0.0.0.255 area 10
!
!
end
```

路由设备 R2 的配置：

```
!
```

```
!
hostname "R2"
!
interface FastEthernet 0/0
no switchport
ip address 192.168.201.213 255.255.255.0
vrrp 1 ip 192.168.201.1
vrrp 1 timers advertise 3
!
interface GigabitEthernet 1/1
no switchport
ip address 60.154.101.3 255.255.255.0
!
router ospf
network 60.154.101.0 0.0.0.255 area 10
network 192.168.201.0 0.0.0.255 area 10
!
!
end
```

可见，路由设备 R1 与 R2 同处于 VRRP 备份组 1 中，使用相同的 VRRP 备份组验证模式(无验证模式)、指向相同的虚拟 IP 地址(192.168.201.1)并且均处于 VRRP 的抢占模式下。路由设备 R2 与路由设备 R2 的 VRRP 的通告(Advertisement)间隔均为 3 秒。正常情况下，由于路由设备 R1 的 VRRP 备份组优先级为 120，而路由设备 R2 的 VRRP 备份组优先级取默认值 100，所以路由设备 R1 在正常情况下充当 Master 路由设备。如果路由设备 R1 在作为 Master 路由设备状态下发现与广域网的接口 GigabitEthernet 2/1 不可用，路由设备 R1 将降低自己的 VRRP 备份组优先级 30 而成为 90，这样路由设备 R2 就会成为 Master 路由设备。如果在此后，路由设备 R1 发现自己的与广域网的接口 GigabitEthernet 2/1 恢复可用，就增加自己的 VRRP 备份组优先级 30 而恢复到 120，这样路由设备 R1 将再次成为主路由设备。

51.5.3 VRRP 多备份组配置示例

除了单备份组，锐捷产品还允许在同一个以太网接口上配置多个 VRRP 备份组。使用多备份组，有着显而易见的好处：可以实现负载均衡同时通过互相备份来提供更稳定可靠的网络服务。

按照图 4 建立连接。在这个配置示例中，用户工作站群(192.168.201.0/24)使用路由设备 R1 与 R2 组成的备份组，其中部分用户工作站(如 A)将其网关指向备份组 1 的虚拟 IP 地址 192.168.201.1，部分用户工作站(如 C)则将其网关指向备份组 2 的虚拟 IP 地址 192.168.201.2。路由设备 R1 在备份组 2 中作为主路由设备，在备份组 1 中作为备份路由设备；而路由设备 R2 在备份组 2 中作为备份路由设备，在备份组 1 中作为主路由设备。下面给出路由设备 R1 与 R2 相关的配置。

路由设备 R1 的配置：

```
!
!
hostname "R1"
!
interface FastEthernet 0/0
no switchport
ip address 192.168.201.217 255.255.255.0
vrrp 1 timers advertise 3
vrrp 1 ip 192.168.201.1
vrrp 2 priority 120
vrrp 2 timers advertise 3
vrrp 2 ip 192.168.201.2
vrrp 2 track GigabitEthernet 2/1 30
!
interface GigabitEthernet 2/1
no switchport
ip address 202.101.90.63 255.255.255.0
!
router ospf
network 202.101.90.0 0.0.0.255 area 10
network 192.168.201.0 0.0.0.255 area 10
!
!
end
```

路由设备 R2 的配置：

```
!
!
hostname "R2"
!
!
interface Loopback 0
ip address 20.20.20.5 255.255.255.0
!
interface FastEthernet 0/0
no switchport
ip address 192.168.201.213 255.255.255.0
vrrp 1 ip 192.168.201.1
vrrp 1 timers advertise 3
vrrp 1 priority 120
vrrp 2 ip 192.168.201.2
vrrp 2 timers advertise 3
!
interface GigabitEthernet 1/1
no switchport
ip address 60.154.101.3 255.255.255.0
```

```
!
router ospf
network 60.154.101.0 0.0.0.255 area 10
network 192.168.201.0 0.0.0.255 area 10
!
!
!
end
```

可见路由设备 R2 与 R2 相互备份，而且二者分别在 VRRP 备份组 1 与 2 中成为主路由设备提供不同的虚拟网关功能。

51.6 VRRP 的故障诊断与排除

VRRP 的如果出现故障可以通过考察配置以及调试信息来分析和解决。以下是常见的故障的分析说明：

故障现象：虚拟 IP 地址 Ping 不通

故障分析：

- 必须确保备份组内至少有一台路由设备处于活动状态；
- 如果在其它网络设备上 Ping 虚拟 IP 地址不通则可能由于以下原因引起: 由于 VRRP 状态的转换需要短暂时间这可以通过 **show vrrp** 命令考察 VRRP 信息来确认；
- 如果本地网络设备与虚拟路由设备位于同一网段就需要考察本地网络设备的 ARP 表中是否有虚拟 IP 地址的 ARP 项，如果没有就需要检查网络线路；
- 如果本地网络设备与虚拟路由设备不在同一网段则需要确认在本地网络设备上是否有到虚拟 IP 地址的路由；

故障现象：同一个 VRRP 备份组内出现多个 Master 路由设备

故障分析：

- 同一 VRRP 备份组内各路由设备以太网接口上 VRRP 组验证模式不同；
- 同一 VRRP 备份组内各路由设备以太网接口上 VRRP 组验证模式相同均为明文密码模式，但是验证字符串不一致；
- 同一 VRRP 备份组内存在路由设备的以太网口电缆断开，但是路由设备未能检测到以太网口电缆已经断开；
- 同一个 VRRP 备份组内路由设备上 VRRP 的通告发送间隔不一致，并且未设置定时设备学习功能；
- 同一个 VRRP 备份组内路由设备上 VRRP 的虚拟 IP 地址不一致。

52 BFD 配置

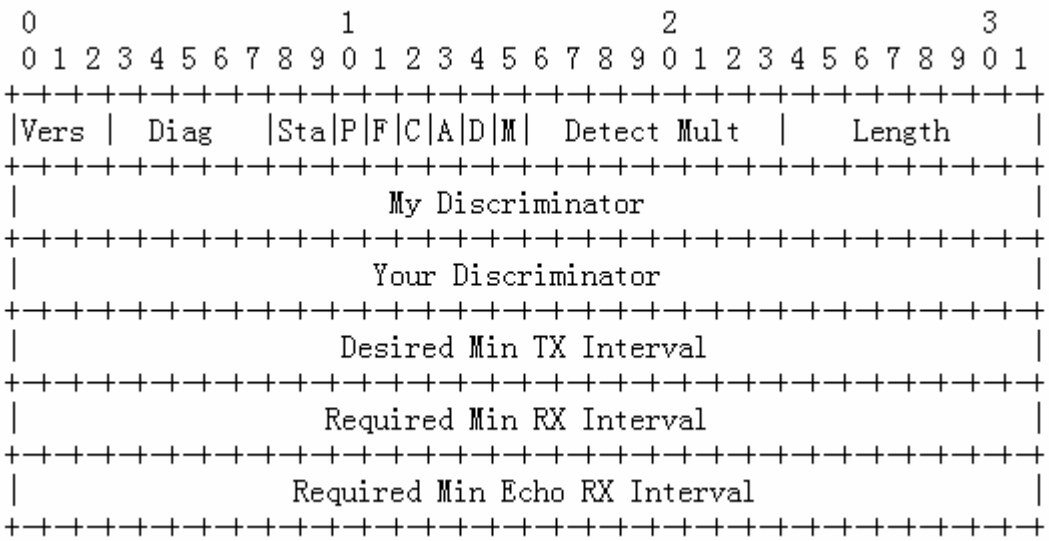
52.1 理解 BFD

52.1.1 BFD 概述

BFD(Bidirectional Forwarding Detection，双向转发检测)协议提供一种轻负载、快速检测两台邻接路由器之间转发路径连通状态的方法。协议邻居通过该方式可以快速检测到转发路径的连通故障，加快启用备份转发路径，提升现有网络性能。

52.1.2 BFD 报文格式

BFD 报文有两种类型分别是控制报文和回声报文。其中回声报文只有 BFD 会话本端系统关心远端不关心，因此协议没有规定其具体格式。协议只规定了控制报文的格式，目前控制报文格式有两个版本(版本 0 和版本 1)，BFD 会话建立缺省采用版本 1，但如果收到对端系统发送的是版本 0 的报文，将自动切换到版本 0 来建立会话，可以通过 show bfd neighbors 命令察看采用的版本。版本 1 的格式如图表 1：



图表 2BFD 控制报文格式

- Vers：BFD 协议版本号，目前为 1
- Diags：给出本地最后一次从 UP 状态转到其他状态的原因，包括:

  0—没有诊断信息

  1—控制超时检测

  2—回声功能失效

3—邻居通告会话 Down

4—转发面复位

5—通道失效

6—连接通道失效

7—管理 Down

- Sta：BFD 本地状态，取值为：0 代表 AdminDown，1 代表 Down，2 代表 Init，3 代表 Up；
- P：参数发生改变时，发送方在 BFD 报文中置该标志，接收方必须立即响应该报文
- F：响应 P 标志置位的回应报文中必须将 F 标志置位
- C：转发/控制分离标志，一旦置位，控制平面的变化不影响 BFD 检测，如：控制平面为 OSPF，当 OSPF 重启/GR 时，BFD 可以继续检测链路状态
- A：认证标识，置位代表会话需要进行验证
- D：查询请求，置位代表发送方期望采用查询模式对链路进行检测
- M：用于将来应用点到多点时使用，目前必须设置 0
- Detect Mult：检测超时倍数，用于检测方计算检测超时时间
- Length：报文长度
- My Discreaminator：BFD 会话连接本端标识符
- Your Discreaminator：BFD 会话连接远端标识符
- Desired Min Tx Interval：本地支持的最小 BFD 报文发送间隔
- Required Min RX Interval：本地支持的最小 BFD 报文接收间隔
- Required Min Echo RX Interval：本地支持的最小 Echo 报文接收间隔（如果本地不支持 Echo 功能，则设置 0）
- Auth Type：认证类型(可选)，目前协议提供有：

  Simple Password

  Keyed MD5

  Meticulous Keyed MD5

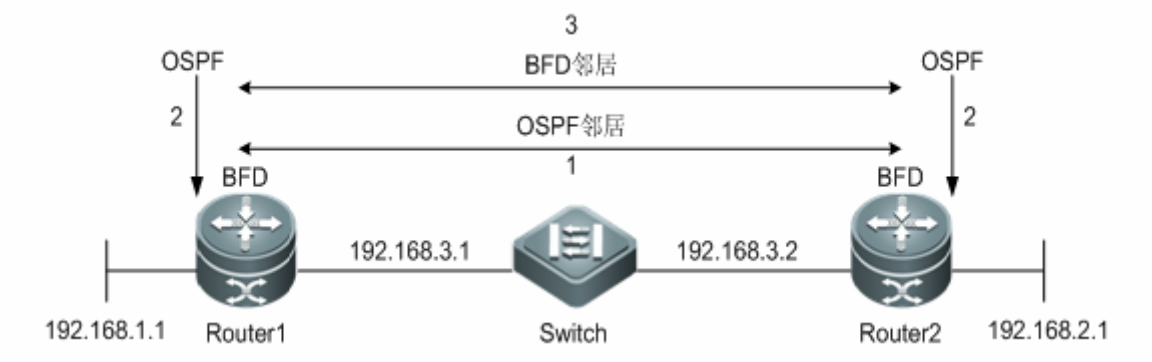
  Keyed SHA1

  Meticulous Keyed SHA1
- Auth Length：认证数据长度
- Authentication Data：认证数据区

注意	RGOS 从 10.3(4b3)版本开始，支持版本 1 和版本 0 的报文格式，缺省情况下会话发送报文采用版本 1，如果收到对端发送的版本 0 的报文，将自动切换到版本 0 来建立会话

52.1.3 BFD 工作原理

BFD 提供的检测机制与所应用的接口介质类型、封装格式、以及关联的上层协议如 OSPF、BGP、RIP 等无关。BFD 在两台路由器之间建立会话，通过快速发送检测故障消息给正在运行的路由协议，以触发路由协议重新计算路由表，大大减少整个网络的收敛时间。BFD 本身没有发现邻居的能力，需要上层协议通知与哪个邻居建立会话。



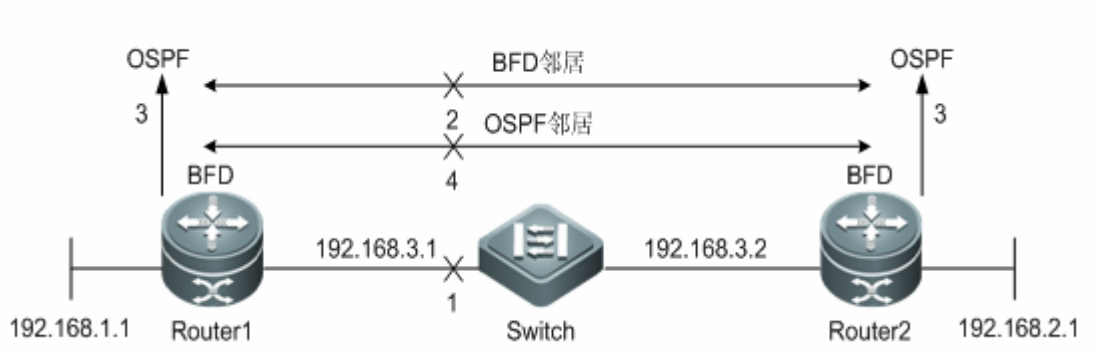
图表 3BFD 会话建立过程

图 2 简单的拓扑图，两台路由器通过一台二层交换机相连，两台路由器同时运行 OSPF 和 BFD。BFD 会话建立过程:

第 1 步:OSPF 发现邻居后并与邻居建立连接

第 2 步:OSPF 通知 BFD 与该邻居建立会话

第 3 步:BFD 与该邻居建立起会话



图表 4BFD 会话检测故障处理过程

图 3 显示 BFD 会话检测到故障后的处理过程:

第 1 步:Router1 与 Switch 之间的链路通信发生故障

第 2 步:Router1 和 Router2 之间的 BFD 会话检测到故障

第 3 步:BFD 通知本地运行的 OSPF 到邻居的转发路径发生故障

第 4 步:OSPF 进行邻居 Down 过程的处理，如果存在备份转发路径那么将进行协议收敛，从而启用备份转发路径

52.1.4 协议规范

目前 BFD 相关的协议规范有：

- draft-ietf-bfd-base-09：Bidirectional Forwarding Detection
- draft-ietf-bfd-generic-05：Generic Application of BFD
- draft-ietf-bfd-v4v6-1hop-09：BFD for IPv4 and IPv6 (Single Hop)
- draft-ietf-bfd-multihop-07：BFD for IPv4 and IPv6 (Multihop)

注意	10.3(5)版本不支持 draft-ietf-bfd-mpls-07 草案标准 RGOS10.3(5)版本不支持 draft-ietf-bfd-mib-06 草案标准

52.2 BFD 特性简介

以下章节描述了 BFD 的特性：

- BFD会话建立模式
- BFD检测模式
- BFD会话参数
- BFD认证方式
- BFD 支持与动态路由协议联动
- BFD支持与静态路由联动
- BFD支持与策略路由联动
- BFD支持与VRRP联动
- BFD支持VRF
- BFD支持的接口

52.2.1.1 BFD 会话建立模式

BFD 会话建立包含如下几种模式

1. 主动模式

在建立会话前不管是否收到对端发来的建立 BFD 会话的控制报文，都会主动发送建立 BFD 会话的控制报文。

2. 被动模式

在建立对话前不会主动发送建立 BFD 会话的控制报文，直到收到对端发送来建立 BFD 会话的控制报文。

注意	10.3(5)版本不支持被动模式，且不可配置

52.2.1.2 BFD 检测模式

BFD 可以包含如下几种检测模式：

1. 异步模式

在异步模式下，系统之间相互周期性地发送 BFD 控制报文，如果某个系统在检测时间内没有收到对端发来的 BFD 控制报文，就宣布会话为 Down

2. 查询模式

在查询模式下，假定每个系统都有一个独立的方法用来确认它连接到其他系统。这样一旦一个 BFD 会话建立起来以后，系统停止发送 BFD 控制报文，除非某个系统需要显式地验证连接性，在需要显式验证连接性的情况下，系统发送一个短序列的 BFD 控制包，如果在检测时间内没有收到返回的报文就宣布会话为 Down，如果收到对端的回应报文，表示转发路径正常。

3. 回声功能

本地系统周期性的发送 BFD 回声报文，远端系统通过它的转发通道将它们环回回来。如果本地在检测周期内连续几个回声报文都没有接收到，会话就被宣布为 Down。回声功能可以和上述两种检测模式一起使用。采用回声报文的检测功能，不需要远端系统的控制面参与，报文通过远端系统的转发面转回，减少了延迟，相对于发送控制报文可以更快的检测到故障。如果在异步模式下启用回声功能，可以大大减少了控制报文的发送，因为检测工作由回声功能完成；如果在查询模式下启用回声功能，在会话建立后可以完全取消发送控制报文。BFD 会话两点必须同时启用回声功能，否则回声功能将不生效。

注意	RGOS10.3(5)版本不支持查询模式 配置回声功能时，必须先通过 **no ip redirects** 命令关闭 IP 报文的重定向功能，以及通过 **no ip deny land** 命令关闭防止 Land-based DDOS 攻击功能 BFD 的回声功能只能在 BFD 会话工作在版本 1 时才能工作

52.2.2 BFD 会话参数

在 BFD 会话建立以后，可以修订 BFD 会话参数(如 Desired Min Tx Interval，Required Min RX Interval，Detect Mult 等)，修订后 BFD 会话将重新协商并采用最新参数进行会话检测，修订过程会话可以继续保持 UP 状态。

52.2.3 BFD 认证方式

BFD 协议允许采用如下方式进行认证：

1. Simple Password
2. Keyed MD5

3. Meticulous Keyed MD5
4. Keyed SHA1
5. Meticulous Keyed SHA1

> 注意 RGOS10.3(5)版本不支持 BFD 认证

52.2.4 BFD 支持与动态路由协议联动

路由协议通过与 BFD 联动，可以利用 BFD 相对于协议自身的“HEELO”机制更快速检测故障的优点，提高协议的收敛性能。一般情况下，检测故障的时间可以缩短到 1 秒以内。RGOS10.3(5)版本支持联动的路由协议包括：

1. RIPv1、RIPV2
2. OSPFv2
3. BGP

52.2.5 BFD 支持与静态路由联动

静态路由与 BFD 联动，可以避免在配置的静态路由不可达的情况下，路由选路不会选择该静态路由作为转发路径。如果存在备份路由转发路径，将可以快速地切换到该备份转发路径。

与动态路由协议不同，静态路由没有发现邻居的机制。因此，当配置 BFD 与静态路由关联，静态路由的下一跳可达性将依赖于 BFD 会话状态。如果 BFD 会话检测到故障，表示静态路由的下一跳不可达，则该静态路由将不安装到 RIB 中。配置时需要确保 BFD 会话邻居均启用静态路由关联 BFD 会话，否则 BFD 会话将无法建立。但如果已经有动态路由协议或者其他应用通知 BFD 与相应邻居创建会话，那么静态路由关联 BFD 会话也将使能。

如果 BFD 会话建立过程，远端系统删除 BFD 会话，将会造成 BFD 会话状态变为 Down，在这种情况下系统确保不影响静态路由的转发行为。

> 注意 10.3(5)版本只支持 BFD 与 IPv4 静态路由联动

52.2.6 BFD 支持与策略路由联动

策略路由与 BFD 联动，可以避免在配置的策略路由不可达的情况下，路由选路不会选择该策略路由作为转发路径。如果存在备份路由转发路径，将可以快速地切换到该备份转发路径。

与策略路由联动的方式等同于静态路由。通过 BFD 跟踪检测与指定邻居的转发路径，当会话检测到故障，将通知策略路由到达相应下一跳不可达，达到该下一跳的策略路由将不生效。

配置时需要确保 BFD 会话邻居都启用策略路由关联 BFD 会话，否则 BFD 会话将无法建立。但如果已经有动态路由协议或者其他应用通知 BFD 与相应邻居创建会话，那么策略路由关联 BFD 会话也将建立。

如果 BFD 会话建立过程，远端系统删除 BFD 会话，将会造成 BFD 会话状态变为 Down，在这种情况下系统确保不影响策略路由的转发行为。

注意	10.3(5)版本只支持 BFD 与 IPv4 策略路由联动

52.2.7 BFD 支持与 VRRP 联动

VRRP 与 BFD 联动可以替代 VRRP 自身的“HEELO”机制实现快速检测主备路由器的运行状态，加快了故障时主备路由器的切换，提高网络的性能。一般情况下，检测故障的时间可以缩短到 1 秒以内。

配置时需要确保 VRRP 两端的路由器都启用 VRRP 关联 BFD 会话，否则 BFD 会话将无法建立。但如果已经有动态路由协议或者其他应用通知 BFD 与相应邻居创建会话，那么 VRRP 关联 BFD 会话也将建立。

VRRP 还可以利用 BFD 来跟踪指定的邻居，如果 BFD 会话检测到与该邻居的转发路径发生故障，则自动降低 VRRP 优先级一定数额，触发主备路由器进行切换。该配置只有在动态路由协议或者其他应用通知 BFD 与相应邻居创建会话时，该会话才会创建。

52.2.8 BFD 支持 VRF

BFD 支持 VRF（VPN Routing and Forwarding，VPN 路由转发），允许 BFD 检测 PE(provider edge)和 CE(customer edge)之间转发路径的连通状态。

52.2.9 BFD 支持的接口

交换机产品，BFD 只允许在 Routed Port 和 SVI 接口上配置，不支持在 L3 AP 接口配置。

路由器产品，BFD 支持在同步口、异步口、ATM、串口、帧中继、POS、CPOS、拨号口、以太口及子接口、E1、信道化的 ATM、信道化的 CPOS、MPPP 接口配置。

52.3 配置 BFD

以下章节描述如何配置 BFD 特性：

- (必选)配置BFD的会话参数
- (可选)配置BFD的回声功能
- (可选)配置BFD的保护策略
- (必选)配置RIP与BFD联动
- (必选)配置OSPF与BFD联动
- (必选)配置静态路由与BFD联动
- (必选)配置策略路由与BFD联动
- (必选)配置VRRP与BFD联动

52.3.1 BFD 的缺省配置

功能特性	缺省值
BFD 会话建立模式	主动模式，不可配置
BFD 检测模式	异步模式，回声功能缺省打开
BFD 会话参数	没有缺省值，必须配置
BFD 认证方式	关闭，不可配置
BFD 支持与动态路由协议联动	关闭
BFD 支持与静态路由联动	关闭
BFD 支持与策略路由联动	关闭
BFD 支持与 VRRP 联动	关闭
BFD 支持 VRF	关闭

52.3.2 配置 BFD 会话参数

BFD 会话参数没有缺省值，必须进行配置，配置步骤如下：

	命令	作用
Step 1	Ruijie>**enable**	进入特权模式
Step 2	Ruijie**# configure terminal**	进入全局模式
Step 3	Ruijie(config)**# interface** *type number*	进入接口配置模式
Step 4	Ruijie(config-if)**#bfd interval** *milliseconds* **min_rx** *milliseconds* **multiplier** interval-multiplier	配置 BFD 参数在指定的接口 **interval** *milliseconds*，配置最小发送间隔,单位是毫秒 **min_rx** *milliseconds*，配置最小接收间隔，单位是毫秒 **multiplier** interval-multiplier，配置检测超时倍数

Step 5	Ruijie(config-if)**# end**	退出接口配置模式，回到特权模式

如果要删除 BFD 会话参数配置，在接口模式模下使用 **no bfd interval** 命令进行设置。

配置举例：

在 Routed Port 接口 FastEthernet 0/2 上配置 BFD 会话参数。

```
Ruijie# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Ruijie(config)# interface fastEthernet 0/2
Ruijie(config-if)# bfd interval 100 min_rx 100 multiplier 3
```

注意	配置时设置的参数需要考虑不同接口传输上的带宽差异。如果设置最小发送间隔和最小接受间隔过小，可能导致 BFD 占用过大带宽而影响本身的数据传输 交换机不允许在 L3 AP 接口下进行配置

52.3.3 配置 BFD 回声功能

BFD 回声功能缺省打开，如下步骤配置如何启用 BFD 的回声功能。会话建立后，启用回声功能不影响会话状态。回声功能关闭后，将不在发送回声报文，同时转发面也不再接收回声报文。

按照如下步骤配置 BFD 回声功能：

	命令	作用
Step 1	Ruijie>**enable**	进入特权模式
Step 2	Ruijie**# configure terminal**	进入全局模式
Step 3	Ruijie(config)**# interface** *type number*	进入接口配置模式
Step 4	Ruijie(config-if)**# bfd echo**	启用回声功能
Step 5	Ruijie(config-if)**# end**	退出接口配置模式，回到特权模式

如果要关闭 BFD 回声功能，在接口模式模下使用 **no bfd echo** 命令进行设置。

配置举例：

在 Routed Port 接口 FastEthernet 0/2 上配置 BFD Echo 功能。

```
Ruijie# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Ruijie(config)# interface fastEthernet 0/2
Ruijie(config-if)# bfd echo
```

在 BFD 异步模式启用 ECHO 功能后，可以采用较慢的频率发送 BFD 控制报文，按照如下步骤可以配置该参数

	命令	作用
Step 1	Ruijie>**enable**	进入特权模式

Step 2	Ruijie# **configure terminal**	进入全局模式
Step 3	Ruijie(config)#**bfd slow-timer** [*milliseconds*]	配置慢速定时器定时时间，单位毫秒。可配置范围从 1000 到 30000，未配置缺省值为 1000。
Step 5	Ruijie(config-if)# **end**	退出全局配置模式

如果要恢复慢速定时器的缺省值，在全局模式模下执行 **no bfd slow-time** 命令进行设置

配置举例：

#配置慢速定时器时间为 1400 毫秒。

```
Ruijie# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Ruijie(config)# bfd slow-timer 1400
```

> 注意　本端的回声报文发出,在对端设备转发面处理后返回到本端，这个过程可能由于对端设备拥塞造成回声报文丢失，引发会话检测失败。在这种情况下，需要配置相应的 QOS 策略来确保回声报文优先得到处理或者关闭回声功能。

52.3.4 配置 BFD 会话状态通告抑制时间

配置 BFD 会话状态通告抑制时间特性主要解决由于线路不稳定导致 BFD 会话在 DOWN 和 UP 状态之间频繁切换，从而引起关联的应用（比如静态路由）频繁的进行转发路径切换，影响业务的正常运行的问题。该特性允许用户配置通告给关联应用会话 UP 状态前所需 UP 状态稳定的时间。

按照如下步骤配置 BFD 会话状态抑制功能:

	命令	作用
Step 1	Ruijie>**enable**	进入特权模式
Step 2	Ruijie# **configure terminal**	进入全局模式
Step 3	Ruijie(config)# **interface** *type number*	进入接口配置模式
Step 4	Ruijie(config)#**bfd up-dampening** *milliseconds*	配置 UP 状态稳定时间
Step 5	Ruijie(config-if)#**end**	退出全局配置模式

如果要恢复缺省值，在接口配置模式下执行 **no bfd up-dampening** 命令进行设置

配置举例：

#配置为 BFD 会话状态抑制时间为 60，000 毫秒。

```
Ruijie# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Ruijie(config)# interface fastEthernet 0/2
Ruijie(config-if)# bfd up-dampening 60000
```

52.3.5 配置 BFD 的保护策略

BFD 协议是非常敏感协议，如果所启用 BFD 功能的设备受到攻击（比如:大量 Ping 报文攻击设备）而发生 BFD 会话震荡可以通过配置启用 BFD 的保护策略进行保护。但如果启用该设备 BFD 功能的同时打开该保护策略，会导致上一跳设备发出的 BFD 报文经过该设备时，该设备会将 BFD 报文丢弃，从而影响上一跳设备与其他设备的 BFD 会话建立。该功能及限制仅对交换机生效。

按照如下步骤配置 BFD 的保护策略：

	命令	作用
Step 1	Ruijie>**enable**	进入特权模式
Step 2	Ruijie**# configure terminal**	进入全局模式
Step 3	Ruijie(config)**#bfd cpp**	启用 BFD 的保护策略
Step 5	Ruijie(config-if)**# end**	退出全局配置模式

缺省情况下该保护策略启用，在全局模式模下执行 **no bfd cpp** 命令可以关闭该保护策略

#配置启用 BFD 的保护策略。

```
Ruijie# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Ruijie(config)# bfd cpp
```

52.3.6 配置 RIP 与 BFD 联动

RIP 协议周期性发送路由更新信息，当在指定时间内没有收到路由更新时，认为此条路由不再生效，这种方式不能快速响应链路故障。

在 RIP 上启动 BFD 检测功能后，将会为 RIP 路由信息源(RIP 路由更新报文的源地址)建立 BFD 会话，一旦 BFD 检测到邻居失效，RIP 路由信息将直接进入失效状态，不再参与路由转发。收敛的时间可以从 180 秒(RIP 定时器的缺省配置情况下)降到 1 秒内。

可以通过 **bfd all-interfaces** 命令使能所有接口允许 RIP 关联 BFD 应用，也可以进入接口配置模式通过 **ip rip bfd [disable]**命令使能或者关闭指定接口允许 RIP 关联 BFD 应用。

	命令	作用
Step 1	Ruijie>**enable**	进入特权模式
Step 2	Ruijie**# configure terminal**	进入全局配置模式
Step 3	Ruijie(config)**# router rip**	进入 Router 配置模式
Step 4	Ruijie(config-router)**# bfd all-interfaces**	使能所有接口允许 RIP 关联 BFD

Step 5	Ruijie(config-router)# **exit**	(可选)退出 Router 配置模式，回到全局配置模式
Step 6	Ruijie(config)# **interface** *type number*	(可选)进入接口配置模式
Step 7	Ruijie(config-if)#**ip rip bfd** [**disable**]	(可选) 使能或者关闭指定接口允许 RIP 关联 BFD 应用
Step 8	Ruijie(config-if)#**end**	(可选)退出特权模式
Step 9	Ruijie#**show bfd neighbors** [**details**]	(可选)显示 BFD 会话建立的信息，以及 RIP 是否关联到指定会话

如果要关闭所有接口允许 RIP 关联 BFD 应用，可以在 Router 模式模下执行 **no bfd all-interfaces** 命令进行设置。

配置举例：

#配置使能除了 FastEthernet 0/2 接口外其他所有允许 RIP 关联 BFD 应用

```
Ruijie#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
```

Ruijie(config)# **router rip**

Ruijie(config-router)# **bfd all-interfaces**

Ruijie(config-router)# **exit**

Ruijie(config)# **interface** FastEthernet 0/2

Ruijie(config-if)# **ip rip bfd disable**

Ruijie(config-if)#**end**

注意

在为 RIP 与 BFD 联动时，要求运行 RIP 的两台设备的路由信息源(RIP 路由更新报文的源地址)在同一个网段内，这样才能在邻居之间建立 BFD 会话

启用 RIP 和 BFD 联动前，必须确保 BFD 会话参数已经配置，否则将不生效

对于非 unnumbered 接口，如果邻居地址和本地地址不是直连地址，那么将无法启用与 BFD 关联

如果由于 IP 选路而导致创建 BFD 会话时指定的接口和实际 BFD 报文出接口口不一致将导致会话无法建立

如果创建 BFD 会话时指定的接口和实际 BFD 回来的报文入接口口不一致将导致会话无法建立

52.3.7 配置 OSPF 与 BFD 联动

OSPF 协议通过 Hello 报文动态发现邻居，当 OSPF 启动 BFD 检测功能后，将会为达到 FULL 关系的邻居建立 BFD 会话，通过 BFD 机制检测邻居状态，一旦 BFD 邻居失效，OSPF 会立刻进行网络收敛。收敛的时间可以从 120 秒(缺省情况非广播型网络 OSPFhello 报文的发送间隔为 30 秒，而邻居设备失效的时间是间隔时间的 4 倍，也就是需要 120s)降到 1 秒内。

可以通过 **bfd all-interfaces** 命令使能所有接口允许 OSPF 关联 BFD 应用，也可以通过进入接口配置模式通过 **ip ospf bfd [disable]**命令使能或者关闭指定接口允许 OSPF 关联 BFD 应用。

	命令	作用
Step 1	Ruijie>**enable**	进入特权模式
Step 2	Ruijie**# configure terminal**	进入全局配置模式
Step 3	Ruijie(config)**# router ospf** *process-id*	进入 Router 配置模式
Step 4	Ruijie(config-router)**# bfd all-interfaces**	使能所有接口允许 OSPF 关联 BFD 通过 no 命令关闭所有接口允许 RIP 关联 BFD
Step 5	Ruijie(config-router)**# exit**	(可选)退出 Router 配置模式，回到全局配置模式
Step 6	Ruijie(config)**# interface** *type number*	(可选)进入接口配置模式
Step7	Ruijie(config-if)**#ip rip bfd** [**disable**]	(可选) 使能或者关闭指定接口允许 OSPF 关联 BFD 应用
Step8	Ruijie(config-if)**#end**	(可选)退出特权模式
Step9	Ruijie**#show bfd neighbors** [**details**]	(可选)显示 BFD 会话建立的信息，以及 OSPF 是否关联到指定会话
Step10	Ruijie**#show ip ospf**	(可选)显示验证 OSPF 是否关联指定的会话

如果要关闭所有接口允许 OSPF 关联 BFD 应用，可以在 Router 模式模下执行 **no bfd all-interfaces** 命令进行设置。

配置举例：

#配置使能除了 FastEthernet 0/2 接口外其他所有允许 OSPF 关联 BFD 应用

```
Ruijie#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
```

Ruijie(config)**# router ospf** 123

Ruijie(config-router)**# bfd all-interfaces**

Ruijie(config-router)**# exit**

Ruijie(config)**# interface** FastEthernet 0/2

Ruijie(config-if)**# ip ospf bfd disable**

Ruijie(config-if)**#end**

注意

10.3(5)版本不支持 OSPFv3 与 BFD 联动

启用 OSPF 和 BFD 联动的前，必须确保 BFD 会话参数已经配置，否则将不生效

如果由于 IP 选路而导致 BFD 会话邻居指定的接口和实际 BFD 报文出接口不一致将导致会话无法建立.

如果创建 BFD 会话时指定的接口和实际 BFD 回来的报文入接口不一致将导致会话无法建立

OSPFv2/OSPFv3 的虚链路不支持进行 BFD 监控

52.3.8 配置静态路由与 BFD 联动

执行如下步骤配置静态路由与 BFD 联动.

	命令	作用
Step 1	Ruijie>**enable**	进入特权模式
Step 2	Ruijie**# configure terminal**	进入全局配置模式
Step 3	Ruijie(config)**#ip route static bfd** [**vrf** vrf-name] *interface-type interface-number gateway* [**souce** *ip-addess*]	配置静态路由关联 BFD 的会话邻居，其中 *interface-type interface-number* 和 *gateway* 指邻居所在的接口及 IP。在多跳的情况下需要配置 **souce** *ip-addess* 作为会话的源 IP，配置时首先确保该接口的 BFD 会话参数已经配置，请参见配置 BFD 会话参数
Step 4	Ruijie(config)#{ [**ip|ipv6**] **route** *prefix mask* {*ip-address* | *interface-type interface-number* [*ip-address*]}	配置静态路由，请确保配置输入的参数 *interface-type interface-number* 和 *ip-address* 必须确保与 Step3 配置的两个参数一致，这样才确保该静态路由与 BFD 关联
Step 5	Ruijie(config)**# end**	退出特权模式
Step 6	Ruijie**#show bfd neighbors** [**details**]	(可选)显示 BFD 会话建立的信息，以及静态路由是否关联到指定会话

如果要关闭静态路由关联 BFD 应用，可以在接口模式模下执行 **no ip route static bfd** [**vrf** *vrf-name*] *interface-type interface-number gateway* 命令进行设置

配置举例：

#配置使能静态路由关联 BFD 应用，通过 BFD 检测与 172.16.0.2 邻居间的转发路径

```
Ruijie# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Ruijie(config)# interface FastEthernet 0/1
Ruijie(config-if)# no switchport
Ruijie(config-if)# ip address 172.16.0.1 255.255.255.0
```

```
Ruijie(config-if)# bfd interval 50 min_rx 50 multiplier 3
Ruijie(config-if)# ip route static bfd FastEthernet 0/1 172.16.0.2
Ruijie(config-if)# ip route 10.0.0.0 255.0.0.0 FastEthernet 0/1
172.16.0.2
Ruijie(config-if)# end
```

52.3.9 配置策略路由与 BFD 联动

执行如下步骤配置策略路由与 BFD 联动。

	命令	作用
Step 1	Ruijie>**enable**	进入特权模式
Step 2	Ruijie**# configure terminal**	进入全局配置模式
Step 3	Ruijie(config)**# route-map** *route-map-name* [**permit** | **deny**] *sequence*	配置定义路由图,进入路由图配置模式
Step 4	Ruijie(config-route-map)**#match ip address** *access-list-number*	配置匹配访问列表
Step 5	Ruijie(config-route-map)**#set ip next-hop verify-availability** [*next-hop-address* [**track** *number*|**bfd** [**vrf** *vrf-name*] *interface-type interface-number gateway*]]	配置策略路由关联 BFD 的会话邻居，其中 *interface-type interface-number* 和 *gateway* 指邻居所在的接口及 IP，配置时首先确保该接口的 BFD 会话参数已经配置,请参见配置 BFD 会话参数 BFD 会话检测到故障，表示 *next-hop-address* 参数指定的下一跳不可达。 通过 no 命令删除该配置
Step 6	Ruijie(config-route-map)**#exit**	退出路由图配置模式
Step 7	Ruijie(config)**# interface** *type number*	进入接口配置模式
Step 8	Ruijie(config-if)**#ip policy route-map** *route-map*	在接口上应用策略路由
Step 9	Ruijie(config-if)**#end**	退出特权模式
Step10	Ruijie**#show bfd neighbors** [**details**]	(可选)显示 BFD 会话建立的信息，以及策略路由是否关联到指定会话
Step11	Ruijie**#show route-map**	(可选)显示验证策略路由是否关联指定的会话

如果要关闭策略路由关联 BFD 应用，可以在 router-map 模式模下执行 **no set ip next-hop verify-availability** ［*next-hop-address* ［**track** *number*｜**bfd** ［**vrf** *vrf-name*］ *interface-type interface-number gateway*］］命令进行设置

配置举例:

#配置使能静态路由关联 BFD 应用，通过 BFD 检测与 172.16.0.2 邻居间的转发路径

```
Ruijie#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Ruijie(config)# route-map Example1 permit 10
```

```
Ruijie(config-route-map)# match ip address 1
Ruijie(config-route-map)# set ip precedence priority
Ruijie(config-route-map)#set ip next-hop verify-availability
172.16.0.2 bfd FastEthernet 0/1 172.16.0.2
Ruijie(config-route-map)#end
Ruijie(config)#interface FastEthernet 0/1
Ruijie(config-if)#no switchport
Ruijie(config-if)#ip address 172.16.0.1 255.255.255.0
Ruijie(config-if)#bfd interval 50 min_rx 50 multiplier 3
Ruijie(config-if)#ip policy route-map Example1
Ruijie(config-if)#exit
```

注意	10.3(5)版本不支持 PBRv6 与 BFD 联动

52.3.10 配置 VRRP 与 BFD 联动

执行如下步骤配置指定 VRRP 组与 BFD 联动来检测主备路由器

	命令	作用
Step 1	Ruijie>**enable**	进入特权模式
Step 2	Ruijie# **configure terminal**	进入全局配置模式
Step 3	Ruijie(config)# **interface** *type number*	进入接口模式
Step 4	Ruijie(config-if)#**vrrp** *group-number* **ip** [*ip-address*[**secondary**]]	配置在指定接口上创建 VRRP 组及虚 IP
Step 5	Ruijie(config-if)# **vrrp** *group-number* **bfd** *ip-address*	配置 VRRP 组与 BFD 联动，*ip-address* 指定邻居 IP
Step 6	Ruijie(config)# **end**	退出特权模式
Step 8	Ruijie#**show bfd neighbors** [**details**]	(可选)显示 BFD 会话建立的信息，以及 VRRP 是否关联到指定会话
Step 9	Ruijie#**show vrrp**	(可选)显示验证 VRRP 是否关联指定的会话

如果要关闭VRRP关联BFD来检测主备路由器应用，可以在接口模式执行**no vrrp** group-number **bfd** 命令进行设置

配置举例:

#配置使能 VRRP 关联 BFD 应用，通过 BFD 检测主备路由器间的转发路径

```
Ruijie#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Ruijie(config)#interface FastEthernet 0/1
Ruijie(config-if)#no switchport
Ruijie(config-if)#ip address 192.168.201.11 255.255.255.0
```

```
Ruijie(config-if)#bfd interval 50 min_rx 50 multiplier 3
Ruijie(config-if)#vrrp 1 priority 120
Ruijie(config-if)#vrrp 1 ip 192.168.201.1
Ruijie(config-if)#vrrp 1 bfd 192.168.201.12
Ruijie(config-if)#end
```

执行如下步骤配置指定 VRRP 组通过 BFD 跟踪指定的邻居 IP

	命令	作用
Step 1	Ruijie>**enable**	进入特权模式
Step 2	Ruijie**# configure terminal**	进入全局配置模式
Step 3	Ruijie(config)**# interface** *type number*	进入接口模式
Step 4	Ruijie(config-if)**#vrrp** *group-number* **ip** [*ip-address*[**secondary**]]	配置在指定接口上创建 VRRP 组及虚 IP
Step 5	Ruijie(config-if)**# vrrp** *group-number* **track bfd** *interface-type interface-number ip-addess* [*priority*]	配置指定 VRRP 组通过 BFD 跟踪指定接口的邻居 IP 通过 no 命令删除该配置
Step 6	Ruijie(config)**# end**	退出特权模式
Step 7	Ruijie**#show bfd neighbors** [**details**]	(可选)显示 BFD 会话建立的信息，以及 VRRP 是否关联到指定会话
Step 8	Ruijie**#show vrrp**	(可选)显示验证 VRRP 是否启用与 BFD 联动跟踪指定邻居 IP

如果要关闭 VRRP 关联 BFD 跟踪指定的邻居 IP 的应用，可以在接口模式执行 no **vrrp** *group-number* **track bfd** *interface-type interface-number ip-addess* 命令进行设置.

配置举例:

#配置指定 VRRP 组通过 BFD 跟踪指定的邻居 192.168.1.3

```
Ruijie#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Ruijie(config)#interface FastEthernet 0/1
Ruijie(config-if)#no switchport
Ruijie(config-if)#ip address 192.168.1.1 255.255.255.0
Ruijie(config-if)#bfd interval 50 min_rx 50 multiplier 3
Ruijie(config)#interface FastEthernet 0/2
Ruijie(config-if)#no switchport
Ruijie(config-if)#ip address 192.168.201.17 255.255.255.0
Ruijie(config-if)#vrrp 1 priority 120
Ruijie(config-if)#vrrp 1 ip 192.168.201.1
Ruijie(config-if)#vrrp 1 track bfd FastEthernet 0/1 192.168.1.3
30
Ruijie(config-if)#end
```

52.3.11 显示 BFD 配置和状态

BFD 提供了如下的显示命令用于查看各种配置信息及运行时信息，各命令的功能说明如下：

命令	作用
show bfd neighbors [**vrf** *vrf-name*] [**ipv4** *ip-addess* [**details**]| **ipv6** *ipv6-addess* [**details**] |**client** {**bgp**|**ospf**|**rip**|**vrrp**|**static-route**|**pbr**} [**ipv4** *ip-addess* [**details**] | **ipv6** *ipv6-addess* [**details**]| **details**]]	显示 BFD 会话信息,信息的具体内容,请参见图表 4 会话显示字段说明
show vrrp	显示 VRRP 与 BFD 联动信息
show route-map	显示策略路由与 BFD 联动信息
show ip static route	显示静态路由与 BFD 联动信息
show ip ospf	显示 OSPF 与 BFD 联动信息
show ip rip database	显示 RIP 与 BFD 联动信息

> 说明　以上显示命令均能在除用户模式外的任何模式配置。

52.4 RIP 与 BFD 联动典型配置举例

52.4.1.1 组网需求

Router A、Router B 通过二层交换机 switch 互连，在设备上运行 RIP 协议来建立路由，同时使能允许 RIP 在双方接口上关联 BFD 应用。在 Router B 和二层交换机 swicth 之间的链路发生故障后，BFD 能够快速检测并通告 RIP 协议，触发协议快速收敛。

52.4.1.2 组网拓扑

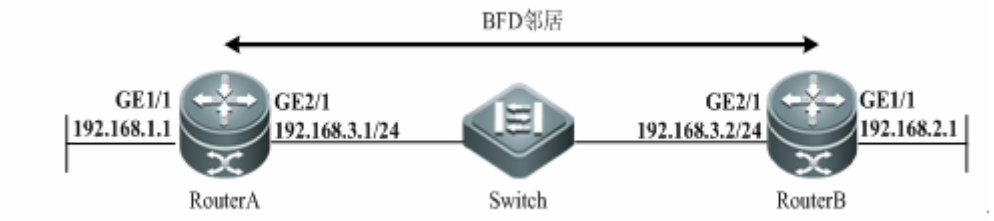


图 14 RIP 与 BFD 联动拓扑图

52.4.1.3 配置要点

1) 配置 RouterA

#在 RouterA 配置 Routed Port 接口 ge2/1、接口 IP、接口的 BFD 会话参数

```
Ruijie# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Ruijie(config)# interface GigabitEthernet2/1
Ruijie(config-if)# no switchport
Ruijie(config-if)# ip address 192.168.3.1 255.255.255.0
Ruijie(config-if)# bfd interval 200 min_rx 200 multiplier 5
# 配置 Routed Port 接口 ge1/1
Ruijie(config-if)# exit
Ruijie(config)# interface GigabitEthernet1/1
Ruijie(config-if)# no switchport
Ruijie(config)# ip address 192.168.1.1 255.255.255.0
```

#启用 RIP 协议并使能 RIP 协议关联 BFD 来检测 192.168.3.2 邻居

```
Ruijie(config-if)# exit
Ruijie(config-router)# router rip
Ruijie(config-router)# version 2
Ruijie(config-router)# network 192.168.3.0
Ruijie(config-router)# network 192.168.1.0
Ruijie(config-router)# passive-interface GigabitEthernet 2/1
Ruijie(config-router)# bfd all-interfaces
```

2) 配置 RouterB

#在 RouterA 配置 Routed Port 接口、接口 IP、接口的 BFD 会话参数

```
Ruijie# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Ruijie(config)# interface GigabitEthernet 2/1
Ruijie(config-if)# no switchport
Ruijie(config-if)# ip address 192.168.3.2 255.255.255.0
Ruijie(config-if)# bfd interval 50 min_rx 50 multiplier 3
# 配置 Routed Port 接口 ge1/1
Ruijie(config-if)# exit
Ruijie(config)# interface GigabitEthernet1/1
Ruijie(config-if)# no switchport
Ruijie(config)# ip address 192.168.2.1 255.255.255.0
```

#启用 RIP 协议并使能 RIP 协议关联 BFD 来检测 192.168.3.1 邻居

```
Ruijie(config-if)# exit
Ruijie(config-router)# router rip
Ruijie(config-router)# version 2
```

```
Ruijie(config-router)# network 192.168.3.0
Ruijie(config-router)# network 192.168.2.0
Ruijie(config-router)# passive-interface GigabitEthernet 2/1
Ruijie(config-router)# bfd all-interfaces
Ruijie(config-router)# end
Ruijie#
```

52.4.1.4 显示验证

1) 查看 RouterA BFD 会话建立情况

```
Ruijie# show bfd neighbors details
OurAddr         NeighAddr       LD/RD   RH  Holdown(mult)   State   Int
192.168.3.1  192.168.3.2  1/2     1       532 (3 )        Up      Ge2/1
  Local Diag: 0, Demand mode: 0, Poll bit: 0
MinTxInt: 200000, MinRxInt: 200000, Multiplier: 5
Received MinRxInt: 50000, Received Multiplier: 3
Holdown (hits): 600(22), Hello (hits): 200(84453)
Rx Count: 49824, Rx Interval (ms) min/max/avg: 208/440/332
Tx Count: 84488, Tx Interval (ms) min/max/avg: 152/248/196
Registered protocols: RIP
Uptime: 02:18:49
Last packet:    Version: 1              - Diagnostic: 0
                I Hear You bit: 1       - Demand bit: 0
                Poll bit: 0             - Final bit: 0
                Multiplier: 3               - Length: 24
                My Discr.: 2            - Your Discr.: 1
                Min tx interval: 50000  - Min rx interval: 50000
                Min Echo interval: 0
```

字段	解释
OurAddr	会话本地的 IP 地址
NeighAddr	会话邻居的 IP 地址
LD/RD	会话本地和远端标识
RH	会话对端是否响应本地
Holdown(mult)	会话本地未接收到hello报文的时间及会话超时检测

State	会话当前状态
Int	会话所在的接口号
Session state is UP and using echo function with 50 ms interval	会话是否采用 echo 模式以及 echo 的发帧时间间隔（该信息只有在工作在 Echo 情况下才会显示）
Local Diag	会话的诊断信息
Demand mode	会话查询模式是否激活
Poll bit	会话的配置是否修订
MinTxInt	会话本地配置的最小发送间隔
MinRxInt	会话本地配置的最小接收间隔
Multiplier	会话本地配置的超时检测次数
Received MinRxInt	会话远端配置的最小发送间隔
Received Multiplier	会话远端配置的超时检测次数
Holdown (hits)	会话检测时间及检测到超时的次数
Hello (hits)	会话协商后 hello 保文的接收最小间隔
Rx Count	会话本地接收到 BFD 报文的个数
Rx Interval (ms) min/max/avg	会话本地接收的最小间隔，最大间隔，平均间隔
Tx Count	会话本地发送到 BFD 报文的个数
Tx Interval (ms) min/max/avg	会话本地发送的最小间隔，最大间隔，平均间隔
Registered protocols	注册到该会话的应用协议类型
Uptime	会话保持 UP 的时间
Last packet	会话本地收到的最后一个 BFD 报文信息

图表 5 会话显示字段说明

2) 查看 RouterB BFD 会话建立情况

Ruijie# **show bfd neighbors details**

OurAddr NeighAddr LD/RD RH Holdown(mult) State Int

192.168.3.2 192.168.3.1 2/1 1 532 (5) Up Ge2/1

Local Diag: 0, Demand mode: 0, Poll bit: 0

MinTxInt: 50000, MinRxInt: 50000, Multiplier: 3

Received MinRxInt: 200000, Received Multiplier: 5

Holdown (hits): 600(22), Hello (hits): 200(84453)

Rx Count: 49824, Rx Interval (ms) min/max/avg: 209/440/332

Tx Count: 84488, Tx Interval (ms) min/max/avg: 153/249/197

Registered protocols: RIP

Uptime: 02:18:49

Last packet: **Version: 1** - Diagnostic: 0

I Hear You bit: 1 - Demand bit: 0

Poll bit: 0 - Final bit: 0

Multiplier: 5 - Length: 24

My Discr.: 1 - Your Discr.: 2

Min tx interval: 200000 - Min rx interval: 200000

Min Echo interval: 0

52.5 OSPF 与 BFD 联动典型配置举例

52.5.1.1 组网需求

Router A、Router B 通过二层交换机 switch 互连，在设备上运行 OSPF 协议来建立路由，同时使能允许 OSPF 在双方接口上关联 BFD 应用。在 Router B 和二层交换机 swicth 之间的链路发生故障后，BFD 能够快速检测并通告 OSPF 协议，触发协议快速收敛。

52.5.1.2 组网拓扑

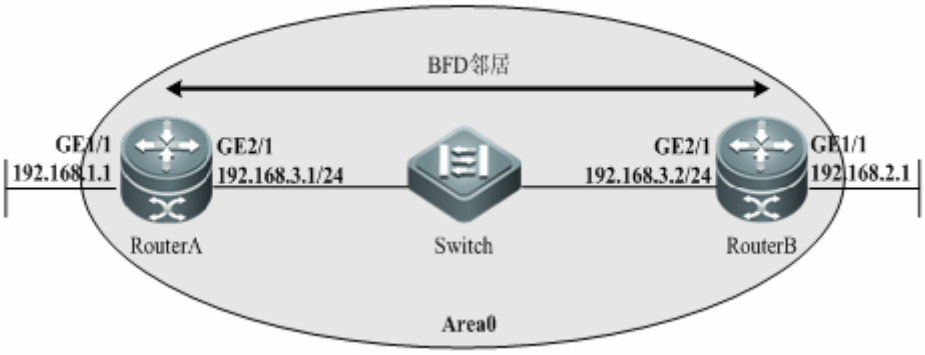


图 15 OSPF 与 BFD 联动拓扑图

52.5.1.3 配置要点

1) 配置 RouterA

#在 RouterA 配置 Routed Port 接口、接口 IP、接口的 BFD 会话参数

```
Ruijie# configure terminal
```

```
Enter configuration commands, one per line.  End with CNTL/Z.
Ruijie(config)# interface GigabitEthernet2/1
Ruijie(config-if)# no switchport
Ruijie(config-if)# ip address 192.168.3.1 255.255.255.0
Ruijie(config-if)# bfd interval 200 min_rx 200 multiplier 5
# 配置 Routed Port 接口 ge1/1
Ruijie(config-if)# exit
Ruijie(config)# interface GigabitEthernet1/1
Ruijie(config-if)# no switchport
Ruijie(config)# ip address 192.168.1.1 255.255.255.0
#启用 OSPF 协议并使能 OSPF 协议关联 BFD 来检测 192.168.3.2 邻居
Ruijie(config-if)# exit
Ruijie(config-router)# router ospf 123
Ruijie(config-router)# log-adjacency-changes detail
Ruijie(config-router)# network 192.168.3.0 0.0.0.255 area 0
Ruijie(config-router)# network 192.168.1.0 0.0.0.255 area 0
Ruijie(config-router)# bfd all-interfaces
Ruijie(config-router)# end
Ruijie#
```

2) 配置 RouterB

#在 RouterA 配置 Routed Port 接口、接口 IP、接口的 BFD 会话参数

```
Ruijie# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Ruijie(config)# interface GigabitEthernet 2/1
Ruijie(config-if)# no switchport
Ruijie(config-if)# ip address 192.168.3.2 255.255.255.0
Ruijie(config-if)# bfd interval 50 min_rx 50 multiplier 3
# 配置 Routed Port 接口 ge1/1
Ruijie(config-if)# exit
Ruijie(config)# interface GigabitEthernet1/1
Ruijie(config-if)# no switchport
Ruijie(config)# ip address 192.168.2.1 255.255.255.0
#启用 OSPF 协议并使能 OSPF 协议关联 BFD 来检测 192.168.3.1 邻居
Ruijie(config-if)# exit
Ruijie(config-router)# router ospf 123
Ruijie(config-router)# log-adjacency-changes detail
Ruijie(config-router)# network 192.168.3.0 0.0.0.255 area 0
Ruijie(config-router)# network 192.168.2.0 0.0.0.255 area 0
Ruijie(config-router)# bfd all-interfaces
Ruijie(config-router)# end
Ruijie#
```

52.5.1.4 显示验证

3) 查看 RouterA BFD 会话建立情况

```
Ruijie# show bfd neighbors details
OurAddr         NeighAddr       LD/RD   RH  Holdown(mult)   State   Int
192.168.3.1  192.168.3.2  1/2     1       532 (3 )        Up      Ge2/1
Local Diag: 0, Demand mode: 0, Poll bit: 0
MinTxInt: 200000, MinRxInt: 200000, Multiplier: 5
Received MinRxInt: 50000, Received Multiplier: 3
Holdown (hits): 600(22), Hello (hits): 200(84453)
Rx Count: 49824, Rx Interval (ms) min/max/avg: 208/440/332
Tx Count: 84488, Tx Interval (ms) min/max/avg: 152/248/196
Registered protocols: OSPF
Uptime: 02:18:49
Last packet:    Version: 1              - Diagnostic: 0
                I Hear You bit: 1       - Demand bit: 0
                Poll bit: 0             - Final bit: 0
                Multiplier: 3               - Length: 24
                My Discr.: 2            - Your Discr.: 1
                Min tx interval: 50000  - Min rx interval: 50000
                Min Echo interval: 0
```

4) 查看 RouterB BFD 会话建立情况

```
Ruijie# show bfd neighbors details
OurAddr         NeighAddr       LD/RD   RH  Holdown(mult)   State   Int
192.168.3.2  192.168.3.1  2/1     1       532 (5 )        Up      Ge2/1
Local Diag: 0, Demand mode: 0, Poll bit: 0
MinTxInt: 50000, MinRxInt: 50000, Multiplier: 3
Received MinRxInt: 200000, Received Multiplier: 5
Holdown (hits): 600(22), Hello (hits): 200(84453)
Rx Count: 49824, Rx Interval (ms) min/max/avg: 209/440/332 last: 66 ms ago
Tx Count: 84488, Tx Interval (ms) min/max/avg: 153/249/197 last: 190 ms ago
Registered protocols: OSPF
Uptime: 02:18:49
```

Last packet:	**Version: 1**	- Diagnostic: 0
	I Hear You bit: 1	- Demand bit: 0
	Poll bit: 0	- Final bit: 0
	Multiplier: 5	- Length: 24
	My Discr.: 1	- Your Discr.: 2
	Min tx interval: 200000	- Min rx interval: 200000
	Min Echo interval: 0	

52.6 静态路由与 BFD 联动典型配置举例

52.6.1.1 组网需求

Router A、Router B 通过二层交换机 switch 互连，在设备上配置静态路由来建立转发，同时使能允许静态路由在双方接口上关联 BFD 应用。在 Router B 和二层交换机 swicth 之间的链路发生故障后，BFD 能够快速检测并通告静态路由，触发系统将该静态路由从 RIB 中删除，从而避免选路错误。

52.6.1.2 组网拓扑

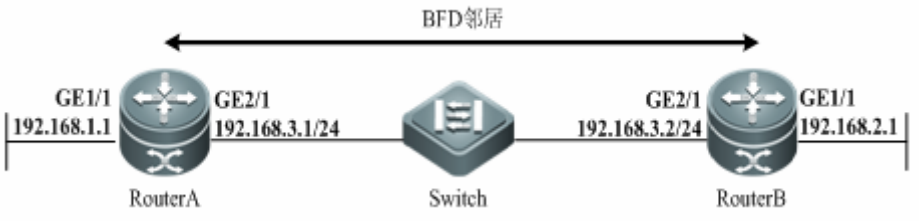


图 16 静态路由与 BFD 联动拓扑图

52.6.1.3 配置要点

1) 配置 RouterA

#在 RouterA 配置 Routed Port 接口 ge2/1、接口 IP、接口的 BFD 会话参数

```
Ruijie# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Ruijie(config)# interface GigabitEthernet2/1
Ruijie(config-if)# no switchport
Ruijie(config-if)# ip address 192.168.3.1 255.255.255.0
```

```
Ruijie(config-if)# bfd interval 200 min_rx 200 multiplier 5
# 配置 Routed Port 接口 ge1/1
Ruijie(config-if)# exit
Ruijie(config)# interface GigabitEthernet1/1
Ruijie(config-if)# no switchport
Ruijie(config)# ip address 192.168.1.1 255.255.255.0
#配置静态路由并关联 BFD 来检测 192.168.3.2 邻居
Ruijie(config-if)# exit
Ruijie(config)# ip route static bfd GigabitEthernet 2/1 192.168.3.2
Ruijie(config)# ip route 192.168.2.0 255.255.255.0 GigabitEthernet 2/1
192.168.3.2
Ruijie(config)# end
Ruijie#
```

2) 配置 RouterB

#在 RouterA 配置 Routed Port 接口、接口 IP、接口的 BFD 会话参数

```
Ruijie# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Ruijie(config)# interface GigabitEthernet 2/1
Ruijie(config-if)# no switchport
Ruijie(config-if)# ip address 192.168.3.2 255.255.255.0
Ruijie(config-if)# bfd interval 50 min_rx 50 multiplier 3
# 配置 Routed Port 接口 ge1/1
Ruijie(config-if)# exit
Ruijie(config)# interface GigabitEthernet1/1
Ruijie(config-if)# no switchport
Ruijie(config)# ip address 192.168.2.1 255.255.255.0
#配置静态路由并关联 BFD 来检测 192.168.3.1 邻居
Ruijie(config-if)# exit
Ruijie(config)# ip route static bfd GigabitEthernet 2/1
192.168.3.1
Ruijie(config)# ip route 192.168.1.0 255.255.255.0
GigabitEthernet 2/1 192.168.3.1
Ruijie(config)# end
Ruijie#
```

52.6.1.4 显示验证

1) 查看 RouterA BFD 会话建立情况

```
Ruijie# show bfd neighbors details
```

OurAddr NeighAddr LD/RD RH Holdown(mult) State Int

192.168.3.1 192.168.3.2 1/2 1 532 (3) Up Ge2/1

Local Diag: 0, Demand mode: 0, Poll bit: 0

MinTxInt: 200000, MinRxInt: 200000, Multiplier: 5

Received MinRxInt: 50000, Received Multiplier: 3

Holdown (hits): 600(22), Hello (hits): 200(84453)

Rx Count: 49824, Rx Interval (ms) min/max/avg: 208/440/332

Tx Count: 84488, Tx Interval (ms) min/max/avg: 152/248/196

Registered protocols: STATIC ROUTE

Uptime: 02:18:49

Last packet: Version: 1 - Diagnostic: 0

I Hear You bit: 1 - Demand bit: 0

Poll bit: 0 - Final bit: 0

Multiplier: 3 - Length: 24

My Discr.: 2 - Your Discr.: 1

Min tx interval: 50000 - Min rx interval: 50000

Min Echo interval: 0

2) 查看 RouterB BFD 会话建立情况

`Ruijie#` **show bfd neighbors details**

OurAddr NeighAddr LD/RD RH Holdown(mult) State Int

192.168.3.2 192.168.3.1 2/1 1 532 (5) Up Ge2/1

Local Diag: 0, Demand mode: 0, Poll bit: 0

MinTxInt: 50000, MinRxInt: 50000, Multiplier: 3

Received MinRxInt: 200000, Received Multiplier: 5

Holdown (hits): 600(22), Hello (hits): 200(84453)

Rx Count: 49824, Rx Interval (ms) min/max/avg: 209/440/332 last: 66 ms ago

Tx Count: 84488, Tx Interval (ms) min/max/avg: 153/249/197 last: 190 ms ago

Registered protocols: STATIC ROUTE

Uptime: 02:18:49

Last packet: Version: 1 - Diagnostic: 0

I Hear You bit: 1 - Demand bit: 0

Poll bit: 0 - Final bit: 0

Multiplier: 5 - Length: 24

My Discr.: 1 - Your Discr.: 2

Min tx interval: 200000 - Min rx interval: 200000

Min Echo interval: 0

52.7 策略路由与 BFD 联动典型配置举例

52.7.1.1 组网需求

Router A、Router B 通过二层交换机 switch 互连，在设备上配置策略路由来建立转发 l 路径，同时使能允许策略路由在双方接口上关联 BFD 应用。在 Router B 和二层交换机 swicth 之间的链路发生故障后，BFD 能够快速检测并通告策略路由，触发系统删除该策略路由，从而避免选路错误。

52.7.1.2 组网拓扑

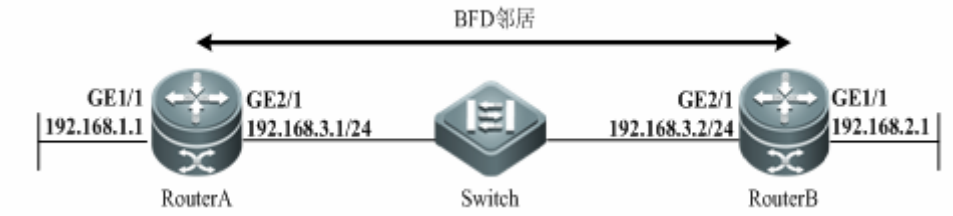


图 17 策略路由与 BFD 联动拓扑图

52.7.1.3 配置要点

1) 配置 RouterA

#在 RouterA 配置 Routed Port 接口 ge2/1、接口 IP、接口的 BFD 会话参数

```
Ruijie# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Ruijie(config)# interface GigabitEthernet2/1
Ruijie(config-if)# no switchport
Ruijie(config-if)# ip address 192.168.3.1 255.255.255.0
Ruijie(config-if)# bfd interval 200 min_rx 200 multiplier 5
# 配置 Routed Port 接口 ge1/1
Ruijie(config-if)# exit
Ruijie(config)# interface GigabitEthernet1/1
Ruijie(config-if)# no switchport
Ruijie(config)# ip address 192.168.1.1 255.255.255.0
```

#配置策略路由并关联 BFD 来检测 192.168.3.2 邻居

```
Ruijie(config)# ip access-list extended 100
Ruijie(config-ext-nacl)# permit ip any 10.10.10.0 0.0.0.255
```

```
Ruijie(config-ext-nacl)# deny ip any any
Ruijie(config-ext-nacl)# exit
Ruijie(config)# route-map Example1 permit 10
Ruijie(config-route-map)# match ip address 100
Ruijie(config-route-map)# set ip precedence priority
Ruijie(config-route-map)#set ip next-hop verify-availability
192.168.3.2 bfd GigabitEthernet 0/1 192.168.3.2
Ruijie(config)# end
Ruijie#
```

2) 配置 RouterB

#在 RouterA 配置 Routed Port 接口、接口 IP、接口的 BFD 会话参数

```
Ruijie# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Ruijie(config)# interface GigabitEthernet 2/1
Ruijie(config-if)# no switchport
Ruijie(config-if)# ip address 192.168.3.2 255.255.255.0
Ruijie(config-if)# bfd interval 50 min_rx 50 multiplier 3
# 配置 Routed Port 接口 ge1/1
Ruijie(config-if)# exit
Ruijie(config)# interface GigabitEthernet1/1
Ruijie(config-if)# no switchport
Ruijie(config)# ip address 192.168.2.1 255.255.255.0
```

#配置策略路由并关联 BFD 来检测 192.168.3.1 邻居

```
Ruijie(config)# ip access-list extended 100
Ruijie(config-ext-nacl)# permit ip any 10.10.11.0 0.0.0.255
Ruijie(config-ext-nacl)# deny ip any any
Ruijie(config-ext-nacl)# exit
Ruijie(config)# route-map Example1 permit 10
Ruijie(config-route-map)# match ip address 100
Ruijie(config-route-map)# set ip precedence priority
Ruijie(config-route-map)#set ip next-hop verify-availability
192.168.3.1 bfd GigabitEthernet 2/1 192.168.3.1
Ruijie(config)# end
Ruijie#
```

52.7.1.4 显示验证

3) 查看 RouterA BFD 会话建立情况

```
Ruijie# show bfd neighbors details
OurAddr         NeighAddr       LD/RD   RH  Holdown(mult)   State   Int
192.168.3.1  192.168.3.2  1/2     1       532 (3 )        Up      Ge2/1
Local Diag: 0, Demand mode: 0, Poll bit: 0
MinTxInt: 200000, MinRxInt: 200000, Multiplier: 5
Received MinRxInt: 50000, Received Multiplier: 3
Holdown (hits): 600(22), Hello (hits): 200(84453)
Rx Count: 49824, Rx Interval (ms) min/max/avg: 208/440/332
Tx Count: 84488, Tx Interval (ms) min/max/avg: 152/248/196
Registered protocols: PBR
Uptime: 02:18:49
Last packet:    Version: 1              - Diagnostic: 0
                I Hear You bit: 1       - Demand bit: 0
                Poll bit: 0             - Final bit: 0
                Multiplier: 3               - Length: 24
                My Discr.: 2            - Your Discr.: 1
                Min tx interval: 50000  - Min rx interval: 50000
                Min Echo interval: 0
```

4) 查看 RouterB BFD 会话建立情况

```
Ruijie# show bfd neighbors details
OurAddr         NeighAddr       LD/RD   RH  Holdown(mult)   State   Int
192.168.3.2  192.168.3.1  2/1     1       532 (5 )        Up      Ge2/1
Local Diag: 0, Demand mode: 0, Poll bit: 0
MinTxInt: 50000, MinRxInt: 50000, Multiplier: 3
Received MinRxInt: 200000, Received Multiplier: 5
Holdown (hits): 600(22), Hello (hits): 200(84453)
Rx Count: 49824, Rx Interval (ms) min/max/avg: 209/440/332 last: 66 ms ago
Tx Count: 84488, Tx Interval (ms) min/max/avg: 153/249/197 last: 190 ms ago
Registered protocols: PBR
Uptime: 02:18:49
```

Last packet: **Version: 1** - Diagnostic: 0

I Hear You bit: 1 - Demand bit: 0

Poll bit: 0 - Final bit: 0

Multiplier: 5 - Length: 24

My Discr.: 1 - Your Discr.: 2

Min tx interval: 200000 - Min rx interval: 200000

Min Echo interval: 0

52.8 VRRP 与 BFD 联动典型配置举例

52.8.1.1 组网需求

Router A、Router B 通过二层交换机 switch 互连，在设备上配置 VRRP，同时使能允许 VRRP 在双方接口上关联 BFD 应用来检测主备路由器。在 Router A 和二层交换机 swicth 之间的链路发生故障后，BFD 能够快速检测并通告 VRRP，触发 VRRP 主路由器降低优先级数额，引起主备切换，从而快速启用备份路由器。

Router A、Router B 分别通过 RouterC 和 RouterD 到达 Internet。RouterA 与 RouterC 之间以及 RouterB 与 RouterD 之间配置静态路由建立转发路径，同时也启用关联 BFD 来检测邻居。同时 Router A、Router B 设备上的 VRRP 还配置了通过与 BFD 联动来跟踪检测 RouterA 与 RouterC 和 RouterB 与 RouterD 之间转发路径，一旦检测失败将触发 VRRP 主路由器降低优先级数额，引起主备切换，从而快速启用备份路由器。

52.8.1.2 组网拓扑

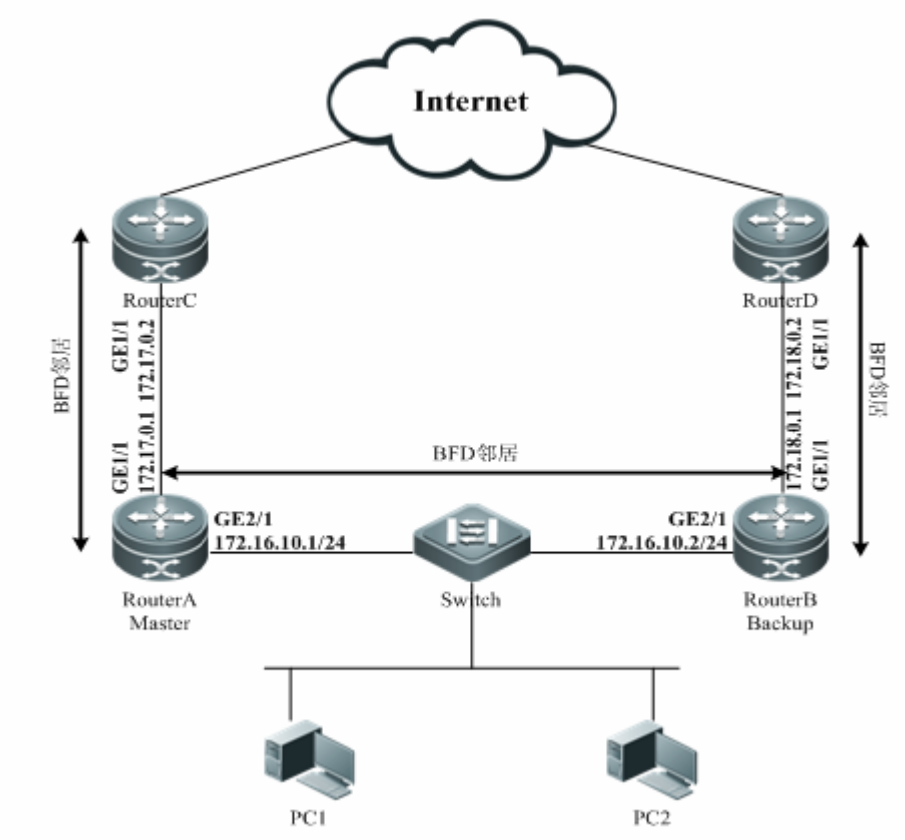


图 18 VRRP 与 BFD 联动拓扑图

52.8.1.3 配置要点

1) 配置 RouterC 略.

2) 配置 RouterD 略.

3) 配置 RouterA

#在 RouterA 配置 Routed Port 接口、接口 IP、接口的 BFD 会话参数

```
Ruijie# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Ruijie(config)# interface GigabitEthernet2/1
Ruijie(config-if)# no switchport
Ruijie(config-if)# ip address 172.16.10.1 255.255.255.0
Ruijie(config-if)# bfd interval 200 min_rx 200 multiplier 5
# 配置 Routed Port 接口 ge1/1
Ruijie(config-if)# exit
Ruijie(config)# interface GigabitEthernet1/1
Ruijie(config-if)# no switchport
Ruijie(config-if)# ip address 172.17.0.1 255.255.255.0
```

```
Ruijie(config-if)# bfd interval 200 min_rx 200 multiplier 5
```

#启用 VRRP 协议并使能 VRRP 协议关联 BFD 来检测 172.16.10.2 邻居，同时启用 BFD 跟踪检测 172.17.0.2 邻居

```
Ruijie(config-if)# interface GigabitEthernet2/1
Ruijie(config-if)# vrrp 1 timers advertise 3
Ruijie(config-if)# vrrp 1 ip 172.16.10.3
Ruijie(config-if)# vrrp 1 priority 120
Ruijie(config-if)# vrrp 1 bfd 172.16.10.2
Ruijie(config-if)# vrrp 1 track bfd GigabitEthernet 1/1 172.17.0.2 30
```

#配置静态路由并关联 BFD 来检测 172.17.0.2 邻居

```
Ruijie(config-if)# exit
Ruijie(config)# ip route static bfd GigabitEthernet 1/1 172.17.0.2
Ruijie(config)# ip route 0.0.0.0 0.0.0.0 GigabitEthernet 1/1 172.17.0.2
Ruijie(config)# end
Ruijie#
```

4) 配置 RouterB

#在 RouterB 配置 Routed Port 接口、接口 IP、接口的 BFD 会话参数

```
Ruijie# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Ruijie(config)# interface GigabitEthernet2/1
Ruijie(config-if)# no switchport
Ruijie(config-if)# ip address 172.16.10.2 255.255.255.0
Ruijie(config-if)# bfd interval 50 min_rx 50 multiplier 3
# 配置 Routed Port 接口 ge1/1
Ruijie(config-if)# exit
Ruijie(config)# interface GigabitEthernet1/1
Ruijie(config-if)# no switchport
Ruijie(config-if)# ip address 172.18.0.1 255.255.255.0
Ruijie(config-if)# bfd interval 200 min_rx 200 multiplier 5
```

#启用 VRRP 协议并使能 VRRP 协议关联 BFD 来检测 172.16.10.1 邻居，同时启用 BFD 跟踪检测 172.18.0.2 邻居

```
Ruijie(config-if)# interface GigabitEthernet2/1
Ruijie(config-if)# vrrp 1 timers advertise 3
Ruijie(config-if)# vrrp 1 ip 172.16.10.3
Ruijie(config-if)# vrrp 1 priority 90
Ruijie(config-if)# vrrp 1 bfd 172.16.10.1
Ruijie(config-if)# vrrp 1 track bfd GigabitEthernet 1/1 172.18.0.2 30
```

#配置静态路由并关联 BFD 来检测 172.18.0.2 邻居

```
Ruijie(config-if)# exit
Ruijie(config)# ip route static bfd GigabitEthernet 1/1 172.18.0.2
Ruijie(config)# ip route 0.0.0.0 0.0.0.0 GigabitEthernet 1/1 172.18.0.2
```

```
Ruijie(config)# end
Ruijie#
```

52.8.1.4 显示验证

1) 查看 RouterA BFD 会话建立情况

```
Ruijie# show bfd neighbors details
```

OurAddr NeighAddr LD/RD RH Holdown(mult) State Int

172.16.10.1 172.16.10.2 1/2 1 532 (3) Up Ge2/1

Local Diag: 0, Demand mode: 0, Poll bit: 0

MinTxInt: 200000, MinRxInt: 200000, Multiplier: 5

Received MinRxInt: 50000, Received Multiplier: 3

Holdown (hits): 600(22), Hello (hits): 200(84453)

Rx Count: 49824, Rx Interval (ms) min/max/avg: 208/440/332

Tx Count: 84488, Tx Interval (ms) min/max/avg: 152/248/196

Registered protocols: VRRP

Uptime: 02:18:49

Last packet: **Version: 1** - Diagnostic: 0

I Hear You bit: 1 - Demand bit: 0

Poll bit: 0 - Final bit: 0

Multiplier: 3 - Length: 24

My Discr.: 2 - Your Discr.: 1

Min tx interval: 50000 - Min rx interval: 50000

Min Echo interval: 0

OurAddr NeighAddr LD/RD RH Holdown(mult) State Int

172.17.0.1 172.17.0.2 2/3 1 532 (3) Up Ge2/1

Local Diag: 0, Demand mode: 0, Poll bit: 0

MinTxInt: 200000, MinRxInt: 200000, Multiplier: 5

Received MinRxInt: 50000, Received Multiplier: 3

Holdown (hits): 600(22), Hello (hits): 200(84453)

Rx Count: 49824, Rx Interval (ms) min/max/avg: 208/440/332 last: 68 ms ago

Tx Count: 84488, Tx Interval (ms) min/max/avg: 152/248/196 last: 192 ms ago

Registered protocols: VRRP,STATIC ROUTE

Uptime: 02:18:49

Last packet: **Version: 1** - Diagnostic: 0

I Hear You bit: 1 - Demand bit: 0

Poll bit: 0 - Final bit: 0

Multiplier: 3 - Length: 24

My Discr.: 2 - Your Discr.: 1

Min tx interval: 50000 - Min rx interval: 50000

Min Echo interval: 0

1) 查看 RouterB BFD 会话建立情况

`Ruijie#` **show bfd neighbors details**

OurAddr	NeighAddr	LD/RD	RH	Holdown(mult)	State	Int
172.16.10.2	172.16.10.1	2/1	1	532 (3)	Up	Ge2/1

Local Diag: 0, Demand mode: 0, Poll bit: 0

MinTxInt: 50000, MinRxInt: 50000, Multiplier: 3

Received MinRxInt: 200000, Received Multiplier: 5

Holdown (hits): 600(22), Hello (hits): 200(84453)

Rx Count: 49824, Rx Interval (ms) min/max/avg: 208/440/332 last: 68 ms ago

Tx Count: 84488, Tx Interval (ms) min/max/avg: 152/248/196 last: 192 ms ago

Registered protocols: VRRP

Uptime: 02:18:49

Last packet: **Version: 1** - Diagnostic: 0

I Hear You bit: 1 - Demand bit: 0

Poll bit: 0 - Final bit: 0

Multiplier: 3 - Length: 24

My Discr.: 1 - Your Discr.: 2

Min tx interval: 200000 - Min rx interval: 200000

Min Echo interval: 0

OurAddr	NeighAddr	LD/RD	RH	Holdown(mult)	State	Int
172.18.0.1	172.18.0.2	1/3	1	532 (3)	Up	Ge2/1

Local Diag: 0, Demand mode: 0, Poll bit: 0

MinTxInt: 200000, MinRxInt: 200000, Multiplier: 5

Received MinRxInt: 50000, Received Multiplier: 3

Holdown (hits): 600(22), Hello (hits): 200(84453)

Rx Count: 49824, Rx Interval (ms) min/max/avg: 208/440/332 last: 68 ms ago

Tx Count: 84488, Tx Interval (ms) min/max/avg: 152/248/196 last: 192 ms ago

Registered protocols: VRRP,STATIC ROUTE

Uptime: 02:18:49

Last packet:	**Version: 1**	- Diagnostic: 0
	I Hear You bit: 1	- Demand bit: 0
	Poll bit: 0	- Final bit: 0
	Multiplier: 3	- Length: 24
	My Discr.: 2	- Your Discr.: 1
	Min tx interval: 50000	- Min rx interval: 50000
	Min Echo interval: 0	

53 RLDP 配置

53.1 RLDP 概述

53.1.1 理解 RLDP

RLDP 全称是 Rapid Link Detection Protocol，是锐捷网络自主开发的一个用于快速检测以太网链路故障的链路协议。

一般的以太网链路检测机制都只是利用物理连接的状态，通过物理层的自动协商来检测链路的连通性。但是这种检测机制存在一定的局限性，在一些情况下无法为用户提供可靠的链路检测信息，比如在光纤口上光纤接收线对接错，由于光纤转换器的存在，造成设备对应端口物理上是 linkup 的，但实际对应的二层链路却是无法通讯的。再比如两台以太网设备之间架设着一个中间网络，由于网络传输中继设备的存在，如果这些中继设备出现故障，将造成同样的问题。

利用 RLDP 协议用户将可以方便快速地检测出以太网设备的链路故障，包括单向链路故障、双向链路故障、环路链路故障。

RLDP 是利用在链路两端交换 RLDP 报文来实现检测的，如下图所示：

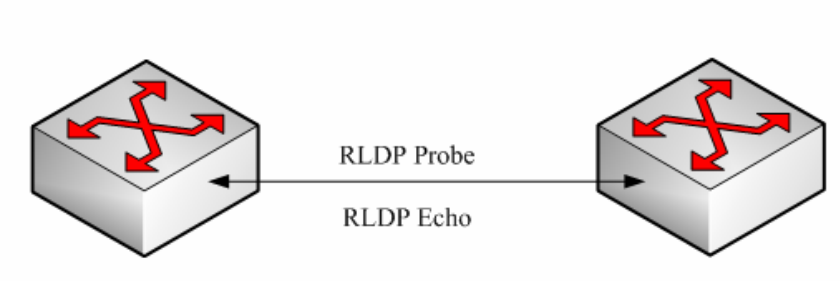


图 1

RLDP 定义了两种协议报文：探测报文(Probe)和探测响应报文(Echo).RLDP 会在每个配置了 RLDP 并且是 linkup 的端口周期性地发送本端口的 Probe 报文，并期待邻居端口响应该探测报文，同时也期待邻居端口也发送自己的 Probe 报文。如果一条链路在物理和逻辑上都是正确的，那么一个端口应该能收到邻居端口的探测响应报文以及邻居端口的探测报文。否则链路将被认定是异常的。

说明：

要使用 RLDP 的单向检测和双向检测功能，必须保证链路两端的端口都打开了 RLDP，并且不允许一个开启了 RLDP 的端口下连多个邻居端口，否则 RLDP 无法检测出每个邻居的链路健康状况。

53.1.2 典型应用

环路检测：

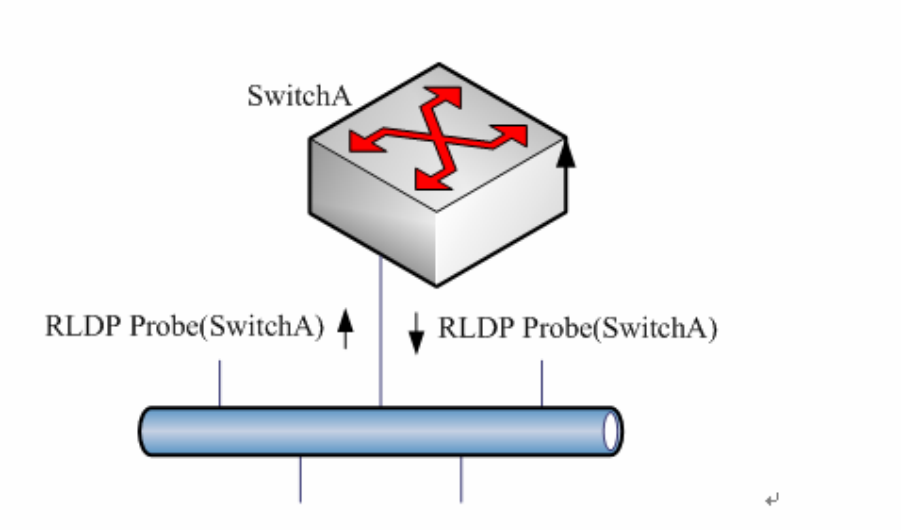


图 2 环路检测

所谓的环路故障是指端口连接的链路上出现了环路。如上图所示，RLDP 在某个端口上收到了本机发出的 RLDP 报文，则该端口将被认为是出现了环路故障，于是 RLDP 会根据用户的配置对这种故障做出处理，包括警告、设置端口违例、关闭端口所在的 svi、关闭端口学习转发等。

单向链路检测：

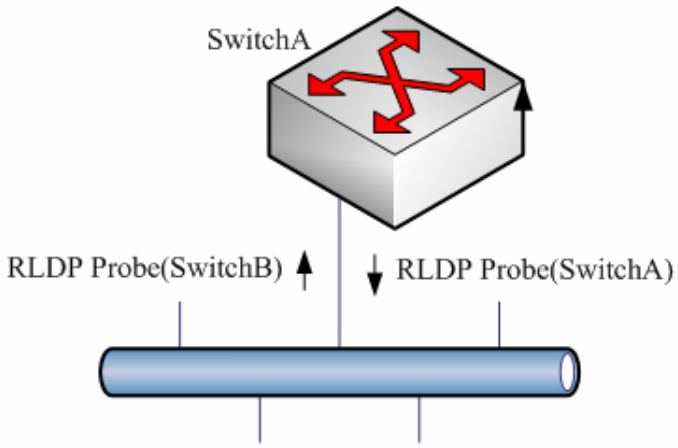


图 3 单向链路检测

所谓单向链路故障是指端口连接的链路只能接收报文或者只能发送报文（比如由于光纤接收线对接错误导致的单向接收或单向发送）。如上图所示，RLDP 在某个端口上只收到邻居端口的探测报文则该端口将被认为单向链路故障，于是 RLDP 会

根据用户的配置对这种故障做出处理。另外如果端口无法收到任何 RLDP 检测报文，也会被认为是发生了单向链路故障。

双向链路检测：

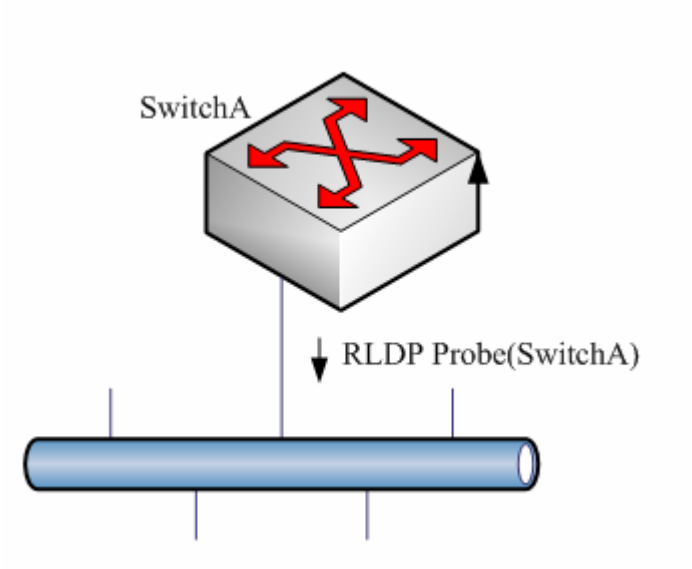


图 4 双向链路检测

所谓双向链路故障是指链路两端的帧收发都出现了故障。如上图所示，设备的端口在发出 RLDP 探测报文后，就一直无法接收到响应报文或邻居的探测报文，那么该链路将被认为是双向故障的。从故障性质上讲，双向故障实际上包含了单向故障。

说明：

如果链路两端有某一方未开启 RLDP 也会被诊断为双向或单向链路故障，因此配置双向链路检测或单向链路检测时需要管理员保证链路两端都开启了 RLDP，以避免出现错误的诊断信息。

53.2 配置 RLDP

我们将从以下几个章节描述如何配置 RLDP

- RLDP 的默认值
- 配置全局 RLDP
- 配置端口 RLDP
- 配置 RLDP 的探测间隔
- 配置 RLDP 的最大探测次数
- 恢复端口的 RLDP 状态

53.2.1 RLDP 的默认值

全局 RLDP 状态	DISABLE
端口 RLDP 状态	DISABLE
探测间隔	2S
最大探测次数	3 次

注意：

- RLDP 只能基于交换口(包括 AP)和路由口进行配置。
- RLDP 帧都是 untag。
- RLDP 故障处理类型中的 block 功能需要和 STP 互斥。也就是说如果用户配置了端口的故障处理类型为 blcok，则建议关闭 stp，否则由于 STP 无法识别单向链路，可能会出现 STP 允许端口转发，但 RLDP 却设置端口 block 的情况。

53.2.2 配置全局 RLDP

RLDP 有两个全局开关，一个是环路检测的全局开关，一个是单向检测和双向检测的全局开关。只有全局的 RLDP 打开，端口 RLDP 才能运行。

在全局配置模式下，按如下步骤打开全局的 RLDP 环路检测开关：

命令	作用
Ruijie(config)# **rldp loop-detect enable**	打开全局的 RLDP 环路检测功能开关。
Ruijie(config)# **end**	退回到特权模式。

如果要关闭全局的 RLDP，请使用该命令的 **no** 选项。

在全局配置模式下，按如下步骤打开全局的 RLDP 单向检测和双向检测开关：

命令	作用
Ruijie(config)# **rldp enable**	打开全局的 RLDP 功能开关。
Ruijie(config)# **end**	退回到特权模式。

如果要关闭全局的 RLDP，请使用该命令的 **no** 选项。

注意：

全局的环路检测开关默认打开，而全局的单向检测双向检测开关默认关闭。

53.2.3 配置端口 RLDP

RLDP 是基于端口运行的，因此用户需要显式配置那些端口需要运行 RLDP。另外在配置端口 RLDP 时，需要同时指定该端口的诊断类型以及故障处理方法。诊断类型包括：unidirection-detect（单向链路检测）、bidirection-detect（双向链路检测）、loop-detect（环路检测）。故障处理方法包括：warning（警告）、block（关闭端口学习转发）、shutdown-port（设置端口违例）、shutdown-svi（关闭端口所在的 svi）。

在配置模式下，按如下步骤设置端口 RLDP 功能：

命令	作用
Ruijie(config)# **interface** *interface-id*	进入接口模式
Ruijie(config-if)# **rldp port** {**unidirection-detect** | **bidirection-detect** | **loop-detect** } {**warning** | **shutdown-svi** | **shutdown-port** | **block**}	端口打开 RLDP，同时配置诊断类型和故障处理方法。
Ruijie(config-if)# **end**	退回特权模式

要关闭端口的 RLDP，请使用该命令的 **no** 选项逐一关闭已经配置的检测类型。

以下例子是在 GigabitEthernet 0/5 上配置 RLDP 并指定多个诊断类型和故障处理方法：

```
Ruijie# configure terminal
Ruijie(config)# interface gigabitEthernet 0/5
Ruijie(config-if)# rldp port unidirection-detect shutdown-svi
Ruijie(config-if)# rldp port bidirection-detect warning
Ruijie(config-if)# rldp port loop-detect block
Ruijie(config-if)# end
Ruijie# show rldp interface gigabitEthernet 0/5
port state     : normal
```

```
local bridge    : 00d0.f822.33ac
neighbor bridge : 0000.0000.0000
neighbor port   :
unidirection detect information:
action : shutdown svi
state  :  normal
bidirection detect information :
action : warnning
state  :  normal
loop detect  information    :
action : block
state  :  normal
```

配置端口检测有几个注意点：

- 路由口不支持 shutdown-svi 的错误处理方法，因此该方法在路由口发生检测错误时将不被执行。
- 配置环路检测时要求端口下连的邻居设备不能开启 RLDP 检测，否则该端口将无法做出正确的检测。
- 如果 RLDP 检测出链路错误，则会发出警告信息。用户可以通过配置 log 功能将这些警告信息发到 log 服务器，记录 log 的级别至少要保证可以记录 3 级日志。
- 由于产品特性的不同，某些产品对于 block 的端口仍然会将报文送 cpu, 这就导致在配置诊断类型为环路检测、故障处理方式为 block 时，当设备检测出环路并将端口 block 处理后，仍会有大量的报文送 cpu,这样就未能达到环路检测的效果，所以建议您在指定环路检测的诊断类型时选择 shutdown-port 的故障处理方法。
- RLDP 故障处理类型中的 block 功能需要和 STP 互斥。也就是说如果用户配置了端口的故障处理类型为 blcok，则建议关闭 stp，否则由于 STP 无法识别单向链路，可能会出现 STP 允许端口转发，但 RLDP 却设置端口 block 的情况。如果要和 STP 共用，我们建议将错误处理类型配置为”shutdown-port”。

注意:

端口的环路检测默认打开，默认的环路处理行为是 shutdown-port。

53.2.4 配置 RLDP 的探测间隔

打开了 RLDP 功能的端口将周期性地发出 RLDP Probe 报文。

在全局配置模式下，按如下步骤配置 RLDP 探测间隔：

命令	作用

Ruijie(config)# **rldp detect-interval** *interval*	配置探测间隔，interval 取值范围是 2-15s，默认是 3s。
Ruijie(config)# **end**	退回到特权模式。

如果恢复默认值，请使用该命令的 **no** 选项。

53.2.5 配置 RLDP 的最大探测次数

打开了 RLDP 功能的端口如果在最大探测期（最大探测次数×探测间隔）内仍然无法接收到邻居的报文，则该端口将被诊断为故障，具体的故障类型请参考概述章节。

在全局配置模式下，按如下步骤配置 RLDP 最大探测次数：

命令	作用
Ruijie(config)# **rldp detect-max** *Num*	配置最大探测次数，num 取值范围是 2-10,默认是 2 次。
Ruijie(config)# **end**	退回到特权模式。

如果恢复默认值，请使用该命令的 **no** 选项。

说明：

最多探测次数只有在单向链路检测和双向链路检测下才会起作用，如果某个端口只开启了环路检测，该值将失效。

53.2.6 恢复端口的 RLDP 状态

配置了 shutdown-port 故障处理的端口在出现故障后将无法主动恢复 RLDP 检测，如果用户确认故障已经解决了，则可以使用恢复命令重新启动被 shutdown 端口的 RLDP。该命令也会将其它有检测出错误的端口重新恢复。

在特权配置模式下，按如下步骤恢复端口的 RLDP 检测：

命令	作用
Ruijie# **rldp reset**	使所有 RLDP 检测失败的端口重新开始检测。

说明：

用户也可以在全局配置模式下使用 **errdisable recover** 命令来即时或定时重新启

动被 rldp 设置成违例的端口(即采用 shutdown-port 检测方式而处于错误状态的端口)的 RLDP 检测。需要注意的是，当配置了 rldp 的端口间不是线路直连的时候，也就是说中间还有一些中继设备，此时如果使用 **errdisable revover interval** 来定时恢复故障，需要将 rldp 的检测时间配置成大于 **errdisable recover interval** 的值，也就是 detect-interval* detect-max 的总时间要大于 **errdisable recover interval** 的值,以避免错误的判断。

53.3 查看 RLDP 信息

锐捷提供的可查看的相关 RLDP 的信息如下：

- 查看所有端口的 RLDP 状态
- 查看指定端口的 RLDP 状态

53.3.1查看所有端口的 RLDP 状态

在特权模式下使用如下命令查看 RLDP 的全局配置和所有配置了 rldp 检测的端口的检测信息：

命令	作用
Ruijie# **show rldp**	查看 RLDP 的全局配置和所有配置了 rldp 检测的端口的检测信息。

以下例子使用 **show rldp** 命令查看 rldp 所有端口的检测信息：

```
Ruijie# show rldp
rldp state        : enable
rldp hello interval  : 2
rldp max hello     : 3
rldp local bridge   : 00d0.f8a6.0134
------------------------------------------------
interface GigabitEthernet 0/1
port state:normal
neighbor bridge : 00d0.f800.41b0
neighbor port   : GigabitEthernet 0/2
unidirection detect information:
action : shutdown svi
state  : normal
interface GigabitEthernet 0/24
port state:error
neighbor bridge : 0000.0000.0000
neighbor port   :
bidirection detect information :
```

```
action : warnning
state  : error
```

从上述信息可以看到，端口 GigabitEthernet 0/1 配置了单向检测，并且当前未检测到错误，端口状态为正常(normal)。端口 GigabitEthernet 0/24 配置了双向检测，并且检测到了双向故障。

53.3.2查看指定端口的 RLDP 状态

在特权模式下使用如下命令查看指定端口的 RLDP 检测信息：

命令	作用
Ruijie**# show rldp interface** *interface-id*	查看 *interface-id* 的 rldp 检测信息。

以下例子使用 **show rldp interface GigabitEthernet** *0/1* 命令查看 fas0/1 端口的 rldp 检测信息：

```
Ruijie# show rldp int GigabitEthernet 0/1
port state     :error
local bridge   : 00d0.f8a6.0134
neighbor bridge : 00d0.f822.57b0
neighbor port   : GigabitEthernet 0/1
unidirection detect information:
action: shutdown svi
state :  normal
bidirection detect information :
action : warnning
state :  normal
loop detect  information   :
action: shutdown svi
state :  error
```

从上述信息可以看到，端口 GigabitEthernet 0/1 配置了三种检测类型：单向检测、双向检测、环路检测,故障时的错误处理分别为关闭端口所在的 svi、产生警告信息、关闭端口所在的 svi，其中环路检测发现了错误，使得当前的端口状态为 error，相应的，该端口所属的 svi 也被 shutdown。

54 DLDP 配置

54.1 理解 DLDP

54.1.1 DLDP 概述

DLDP 全称是 Data Link Detection Protocol，是一种基于快速检测以太网链路故障的检测协议。

一般的以太网链路检测机制都只是利用物理连接的状态，通过物理层的自动协商来检测链路的连通性。但是这种检测机制存在一定的局限性，在一些情况下无法为用户提供可靠的链路检测信息，比如在光纤口上光纤接收线对接错，由于光纤转换器的存在，造成设备对应端口物理上是 linkup 的，但实际对应的链路却是无法通讯的。再比如两台以太网设备之间架设着一个中间网络，由于网络传输中继设备的存在，如果这些中继设备出现故障，将造成同样的问题。

这样的问题，通常导致实际链路已经不通，但三层以上的各种协议却收敛很慢，主要依赖于各协议自身的收敛性能。

DLDP 将通过 ICMP echo 报文的检测来解决这类问题，具体工作原理可参见下面章节描述。

54.1.2 DLDP 基本概念及工作原理

DLDP 通过在三层接口（SVI、Routed Port、L3 AP）下不断的发出 IPv4 ICMP echo 进行通路检测，如果在指定时间内对端设备没有回应 ICMP reply，则 DLDP 认为这个接口通路出现问题，将该接口设置为“三层接口 DOWN”，于是触发各三层上的协议各种收敛、备份切换动作。

由于 DLDP 只是设置“三层接口 DOWN”，实际物理链路还是连通的（STP、802.1x 等二层协议还将继续正常通讯），因此 DLDP 还是可以继续发出 ICMP echo 报文，如果对端设备恢复响应了 ICMP reply，则三层接口恢复 UP，恢复正常通讯。

54.1.2.1 DLDP 检测的参数

DLDP 可以配置发送检测报文的时间间隔、重传次数，以便能适用更多样的网络环境。

当网络设备在“发送检测报文的时间间隔”×“重传次数”的时间周期内没有收到对端的应答报文，则认为三层接口 DOWN。一旦恢复正常通讯，则三层接口 UP。

说明:

1、一个三层接口下，DLDP 可以配置多个 IP 检测，当所有 IP 都没有 ICMP 响应时，才认为接口 DOWN；而一旦有一个 IP 恢复通讯，则认为接口恢复 UP。
2、DLDP 使用该三层接口的第一个 IP 地址作为报文的源 IP 地址进行通讯。

注意:

由于交换机有 CPP、NFPP 功能来控制 ICMP 报文的收发速率，所以在进行多个 IP 检测时，一定要注意 CPP、NFPP 所设置的收发包速率要大于 DLDP 报文的总收发包速率。
例如：缺省值情况下每个 DLDP 检测点所占用的 ICMP 报文速率为 10pps，当 DLDP 检测 IP 数目为 100 个时，整机设备的 ICMP 报文流量已经到达至少 1000pps 了。这时，对应的 CPP 的 ICMP 控制需要调到适当值。

54.1.2.2 DLDP 的被动模式

在实际网络连接中，如果两端设备都打开 DLDP，双方互发 ICMP echo 也是可以达到通路检测功能的，但这样明显存在多余重复的报文。

实际就只要有一端设备使用 ICMP echo 发包，另一端用同样的检测参数来确认报文的及时可达，一样能达到双方设备检测链路通路的效果，同时也节省了带宽资源和设备 CPU 资源的消耗。

于是，我们称主动发 ICMP echo 的为主动模式（缺省配置即是如此），被动接收对端 ICMP echo 发包的为被动模式。

注意:

无论是主动，还是被动模式，都需要确保对端设备的时间参数配置与本设备完全一致，以便同步链路状态效果的检测。否则可能因对端发包间隔、重传次数的不同步而引起链路状态的误判。

54.1.2.3 DLDP 的下一跳配置

在某些情形下，DLDP 需要检测非直连网段的 IP 可达性。这时需要配置该接口的下一跳 IP，以便 DLDP 能够通过 ARP 报文获取下一跳 MAC 地址，正确的封装 ICMP 报文发出。

但这种情形，一定要避免响应报文从其他链路回应的情形，这样就直接造成 DLDP 误判该接口没有收到 ICMP 应答。

54.1.2.4 DLDP 的恢复次数

在有些情形下，检测链路可能不太稳定，比如 PING 断了三次，通一次，又断了多次。如果按简单的逻辑，其中的 DLDP 检测就是 UP、DOWN 多次，实际可能更加剧了不稳定。

为此，我们扩展了“恢复次数”这个参数，缺省为 3 次，即只有该链路上 PING 通了 3 次才会 UP。这种情况下，虽然可能使链路检测会相对不那么灵敏，但增加了稳定性，因此相关参数在实际应用中还可根据实际网络情况进行调整。

54.1.3 协议规范

无

54.2 缺省配置

下表用来描述 DLDP 的缺省配置。

功能特性	缺省值
对端设备的检测 IP 及下一跳 IP	无
发送检测报文的时间间隔	100tick = 1s
重传次数	3 次
恢复次数	3 次
检测模式	主动

54.3 配置 DLDP

- （必选）配置DLDP的检测IP及相关参数
- （可选）被动模式

54.3.1 配置 DLDP 的检测 IP 及相关参数

缺省情况下，接口关闭 DLDP 功能，进入接口模式，按以下步骤开启 DLDP 检测功能：

命令	功能

Ruijie# **configure terminal**	进入全局配置模式
Ruijie(config)# **interface** *interface-name*	进入三层接口
Ruijie(config-if)# **dldp** *ip-address* [**next-hop** *ipv4-address*] [**interval** *tick*] [**retry** retry-num] [**resume** resume-num]	配置检测对端设备的 IP。 如果是跨网段的，请准确配置上 next-hop IP。 如果需要调整发送报文的时间间隔及重传次数，请设置 interval 及 retry 相关参数。 interval 范围：1-6000 tick 注：1 tick =10 毫秒 retry 范围：1-3600 如果需要设置设备链路的恢复次数，该次数表示链路从 DOWN 状态为 UP 状态前，需要收到连续的 DLDP 检测报文响应次数，请配置 resume 参数值。 resume 范围：1-200
Ruijie(config-if)**#end**	退回特权模式
Ruijie# **show running**	确认配置

要删除相关配置，可以在接口配置模式下执行 **no dldp** *ipv4-address* 命令来执行。

说明：

1、本功能借助 ICMP ECHO 报文来完成，需要对端设备打开 ICMP 响应功能；
2、跨网段检测，需要配置下一跳 IP 地址。

配置举例：

添加一个 DLDP 检测 IP

```
Ruijie# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Ruijie(config)# interface vlan 1
Ruijie(config-if)# ip-address 10.10.10.1 255.255.255.0
Ruijie(config-if)# dldp 10.10.10.2
Ruijie(config-if))# end
```

😐各产品支持情况	在 10.3(5)以及之后的 10.3 系列版本中，各款交换机产品均支持。

54.3.2 被动模式

缺省情况下，接口的 DLDP 功能为主动模式，进入特权模式，按以下步骤使 DLDP 转为被动模式：

命令	功能
Ruijie# **configure terminal**	进入全局配置模式
Ruijie(config)# **interface** *interface-name*	进入三层接口
Ruijie(config-if)**# dldp passive**	配置接口处于被动模式
Ruijie(config-if)**#end**	退回特权模式
Ruijie# **show running**	确认配置

要删除相关配置，可以在接口配置模式下执行 **no dldp passive** 命令来执行。

说明：

启动被动模式时，需要确保对端设备的时间参数配置与本设备完全一致，以便同步链路状态效果的检测。否则在被动模式中可能因对端发包的不同步而引起链路状态误判。

配置举例：

配置一个接口为 DLDP 被动模式

```
Ruijie# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Ruijie(config)# interface vlan 1
Ruijie(config-if)# ip-address 10.10.10.1 255.255.255.0
Ruijie(config-if)# dldp 10.10.10.2 interval 50 retry 10
#上述时间参数配置注意与对端设备的探测间隔、重传次数一致
Ruijie(config-if))# dldp passive
Ruijie(config-if))# end
```

😐各产品支持情况	在 10.3(5)版本中，各款交换机产品均支持。

54.4 查看 DLDP 信息

命令	功能
Ruijie(config-if)# **show dldp** [**interface** *interface-name*][**statistic**]	show dldp 查看 DLDP 的配置信息； show dldp statistic 用以查看检测链路的 down/up 统计次数。

配置举例：

#显示各个 DLDP 配置信息

```
Ruijie#show dldp
Interface  Type        Ip          Next-hop     Interval Retry
Resume State
---------    -------   ------------------
--------------------   --------- ------- ----------- --------
Vl2      Passive 192.168.6.3    192.168.2.2     10       5   3
Up
Vl3      Passive 192.168.7.3                 10       5   3
Up
Vl4      Passive 192.168.3.3    192.168.4.2     10       5   3
Up
```

#显示各个 DLDP 的 down/up 统计信息

```
Ruijie#show dldp statistic
Interface  Type        Ip        record-time  Up-count
Down-count
-----------  ---------- ------------------   ---------------
----------- ----------------
Vl2      Passive 192.168.6.3    2h34m5s    10      9
Vl3      Passive 192.168.7.3    30m34s     9       8
Vl4      Passive 192.168.3.3    1d2h3m52s  10      9
```

说明：

在 show dldp statistic 显示统计信息中包含 record-time 信息，该信息表示用于统计 up/down 次数的时间段长度。对应时间的显示格式为*y***d**h**m**s，这里的 y 表示年，d 表示天，h 表示小时，m 表示分钟，s 表示秒。

结合后面的 Up-count 、Down-count 参数，即说明在这段时间内，up/down 了多少次。

54.5 清除 DLDP 统计信息

命令	功能
Ruijie#**clear dldp** [**interface** *interface-name* [*ip*]]	清除指定监测点的 down/up 次数，并重新开始计数统计；用户也可以清除指定三层接口或所有三层接口上监测点的 down/up 次数。

55 TPP 配置

55.1 TPP 概述

TPP(Topology Protection Protocol，拓扑保护协议)是一个拓扑稳定保护协议。网络拓扑比较脆弱，当网络中存在非法攻击时，可能造成网络设备的 CPU 利用率异常、帧通路堵塞等现象，这些现象很容易引起网络拓扑的振荡。拓扑防护主要通过检测本地的异常现象（CPU 利用率异常、帧缓冲异常等）和检测邻居设备的异常现象来达到稳定网络拓扑的目的。与邻居设备的交互是通过发送特定的异常通告报文来实现的，该功能拥有较高的运行优先级，可以有效防止网络的拓扑振荡。

55.2 TPP 应用

拓扑防护主要是针对 MSTP 或 VRRP 以及其他分布式网络协议可能造成的网络拓扑振荡而产生的。MSTP 或 VRRP 等协议均使用定时报文通告机制来自动维护网络拓扑结构，自动适应网络中的拓扑变化。这也造成了网络拓扑易受攻击，当受到人为的网络攻击时，因 CPU 利用率过高或帧通路阻塞等原因，可能造成定时报文的短暂中断，从而造成网络拓扑发生错误的振荡，这给网络的正常通信造成极大危害。而拓扑防护功能正是为了最大限度的防止这种不必要的网络振荡，它与其他分布式协议（MSTP、VRRP 等）协同工作，从而使网络更加稳定、可靠。

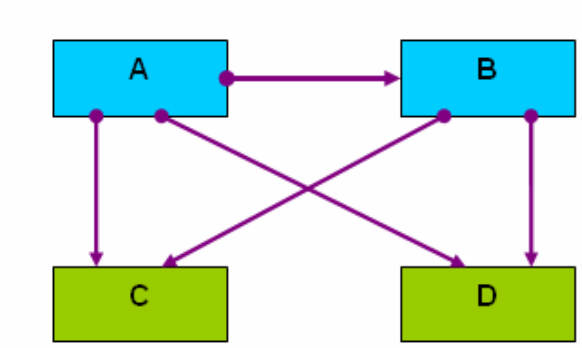


图 1

如上图的双核心拓扑，图中 A、B 为三层汇聚设备，C、D 为二层接入设备。A 为 MSTP 的根桥，各个网络设备的拓扑防护功能均开启。

三层汇聚设备 A 因遭受网络攻击造成 CPU 异常繁忙，从而导致 BPDU 报文不能正常发送。这时，拓扑防护功能检测到异常后，会向邻居设备发送异常通告报文，图中二层接入设备 C、D 的和三层汇聚设备 B 均接收到异常通告，这时，设备 C、D 的和设备 B 会根据异常通告信息，进行相应的防振荡处理。

设备 B 因遭受大量报文攻击造成 CPU 异常繁忙，这时造成收发包不正常，检测到异常后，它会向所有的邻居设备发送异常通告。设备 A 收到异常报文后，根据来

源，发现异常对其没有影响，不会进一步处理。而下游的二层接入设备 C、D 接收到异常报文后，发现异常会影响其拓扑的计算，因此做进一步的防御处理，从而保证网络拓扑保持稳定。

55.3 TPP 配置

TPP 配置包括全局功能配置和端口功能配置。全局功能配置用于使能设备的拓扑防护功能，默认情况下，全局拓扑防护功能使能，这时它会检测本地和邻居设备的运行情况，对产生的异常情况进行处理。不过它不会向邻居设备通告本地的运行情况。端口功能配置用于使能端口的拓扑防护功能，端口的拓扑防护功能使能时，表示对端的邻居设备关心本地设备的运行情况，因此当本地设备发生异常时，将通告该端口的对端邻居设备。默认情况下，所有端口的拓扑防护功能关闭。

说明：

拓扑防护功能适用于点对点链路的网络，且相邻网络设备都必须使能拓扑防护功能。另外，部署 TPP 功能时，通常需要通过 cpu topology-limit 命令配置 cup 利用率检测的阈值，当设备的 cpu 利用率超过该值时，系统会产生拓扑防护通告。我们建议该值设置在一个中等偏上的位置比较合适，比如 50-70,这时 TPP 能够较为准确地对网络情况进行判断。如果该值太低，则可能导致网络拓扑该切换的时候由于 TPP 的报警而不切换，如果该值太高，则可能系统已经繁忙到无法产生 TPP 告警，导致 TPP 功能失效。

55.3.1 配置全局拓扑防护

全局拓扑防护功能默认是使能的。使用该命令的 **no** 选项禁止全局拓扑防护。

配置命令如下：

命令	作用
Ruijie> **enable**	进入特权命令模式
Ruijie# **config terminal**	进入全局配置模式
Ruijie(config)# **topology guard**	使能全局拓扑防护
Ruijie(config)# **end**	退回特权命令模式
Ruijie# **copy running-config startup-config**	保存配置

使用 **no topology guard** 禁止设备的全局拓扑防护功能

55.3.2 配置端口拓扑防护

配置命令如下：

命令	作用
Ruijie> **enable**	进入特权命令模式
Ruijie# **config terminal**	进入全局配置模式
Ruijie(config)# **interface gi** *0/1*	进入接口配置模式
Ruijie(config-if)# **tp-guard port enable**	使能端口拓扑防护功能
Ruijie(config-if)# **end**	退回特权命令模式

使用 **no tp-guard port enable** 禁止端口的拓扑防护，该命令只适用于二层交换口和路由口。不适用于 AP 成员口。

说明：

全局拓扑防护作为拓扑防护的全局开关，当全局拓扑防护使能时，设备会检测本设备的运行参数，同时监听各个邻居设备的运行参数，但当本地出现异常现象时，它不会向邻居设备发送异常通告报文进行通告。端口的拓扑防护功能使能时，当本地出现异常现象时，会向该端口对端的邻居设备发送异常通告报文进行异常通告。

55.4 TPP 典型配置举例

如下图为双核心组网拓扑图：

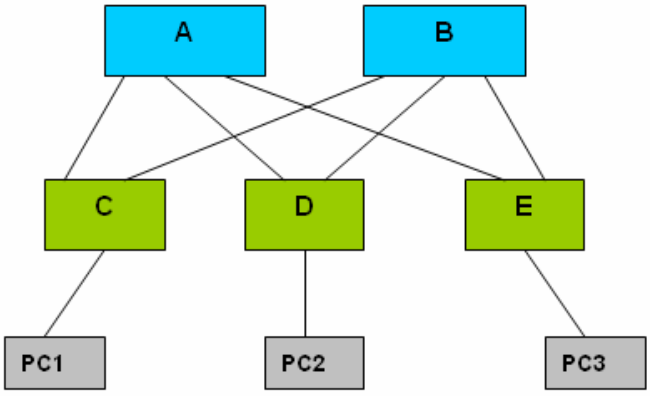


图 2

图中 A、B 为三层汇聚设备，C、D 和 E 为二层接入设备。

三层汇聚设备 A、B 和二层接入设备 C、D 和 E 均开启 MSTP，同时两台三层汇聚

设备开启 VRRP 协议。拓扑防护功能使 MSTP 和 VRRP 运行更加稳定，避免网络拓扑发生不必要的振荡。

对于三层汇聚设备 A、B 使能全局拓扑防护功能，同时使能各个端口的拓扑防护功能。对于各台二层接入设备 C、D 和 E 使能全局拓扑防护功能。

55.5 查看 TPP 信息

我们提供的可查看的 TPP 的相关信息如下：

查看设备的 TPP 配置及状态

55.5.1 查看设备的 TTP 配置及状态

在特权模式下使用如下命令查看查看设备的 TTP 配置及状态：

命令	作用
Ruijie**# show tpp**	查看设备的 TPP 配置及状态。

```
Ruijie #show tpp
tpp state        : enable
tpp local bridge  : 00d0.f822.35ad
-----------------------------------
```

56 文件系统配置

56.1 概述

所谓文件系统是指一个负责存取和管理辅助存储设备上文件信息的机构，交换机提供了串行 Flash 作为辅助存储器，用于存储和管理交换机的网络操作系统文件，以及一些交换机的配置文件。

文件数据在串行 Flash 上是以日志的格式保存的，每个文件都有一个文件头记录该文件的基本信息。当存储器被写满而无空间再去进行其他操作时，文件系统将自动整理碎片以及垃圾回收，以达到提供足够的空间进行文件操作，这个时间是相当快的，通常察觉不到，为了充分的利用有限的空间，文件系统还提供了数据压缩的功能以及数据节点索引信息。

56.2 配置文件系统

我们将从以下几个小节描述如何配置文件系统：

- 切换目录
- 复制文件
- 显示目录内容
- 格式化系统
- 创建目录
- 移动文件
- 显示当前工作路径
- 删除文件
- 删除空目录

56.2.1 文件系统配置指导

命令关键字对大小写不敏感，文件名对大小写敏感，最长文件名为 4096。

所有的文件名以及路径信息均不支持通配符操作。

注意：

在flash存储空间（不包括扩展flash的存储空间）为32M的设备上，当剩余空间低于512K时，为保证后续对flash文件系统操作正常，建议手动清理过时无用的文件。例如：当USB挂载时，操作系统就会对flash文件系统进行操作。因此当USB文件系统不能挂载时且flash剩余空间低于512K时，建议手动清理过时无用的文件然后

再尝试。
在flash存储空间（不包括扩展flash的存储空间）为512M的设备上，当剩余空间低于4M时，为保证后续对flash文件系统操作正常，建议手动清理过时无用的文件。
例如：当USB挂载时，操作系统就会对flash文件系统进行操作。因此当USB文件系统不能挂载时且flash剩余空间低于4M时，建议手动清理过时无用的文件然后再尝试。
当flash存储剩余空间低于欲拷贝文件大小的110%时，为保证拷贝能成功完成，建议手动清理过时无用的文件。
例如：当想拷贝10MB的文件，操作系统将利用一部分flash存储空间来作为这10MB数据的管理空间，因此当flash存储剩余空间低于11MB时，建议手动清理过时无用的文件然后再尝试。

56.2.2 切换目录

从当前目录切入到指定目录。

在特权用户模式下，按如下步骤使用该命令：

命令	作用
Ruijie# **cd** *directroy*	进入指定的 directory 目录。
Ruijie# **cd** *../*	进入上一级目录
Ruijie# **cd** *./*	进入本级目录

以下例子是进入根目录下的 MNT 目录的 Document 目录：

```
Ruijie# cd mnt/document
```

之后再进行的操作将在 MNT/Document 目录下进行

56.2.3 复制文件

复制文件到一个目录中或者一个文件中。

在特权用户模式下，按如下步骤配置 **copy** 命令可以实现文件到目录或文件到文件的拷贝：

命令	作用
Ruijie# **copy flash:** *filename* **flash:** *directoryname*	将文件复制到指定的目录中。
Ruijie# **copy flash:** *filename* **sour** *directoryname*	复制文件到指定的文件中。

以下是分别拷贝到一个目录和拷贝到一个文件的实例：

```
Ruijie# copy flash:config.tex flash:tmp/
Ruijie# copy flash:con_bak.txt flash:config.text
```

56.2.4 显示目录内容

显示当前工作目录下或者指定目录下的信息：

命令	作用
Ruijie# **dir**	显示当前目录下的内容。
Ruijie# **dir** *directory*	显示指定目录的内容。

以下例子是显示当前目录内容和指定目录内容的举例：

```
Ruijie# dir
Ruijie# dir ../bak
```

56.2.5 格式化系统

在特权用户模式下，按照下面的配置命令格式可以格式化文件系统要管理和操作的设备：

命令	作用
Ruijie# **makefs dev** *devname* **fs** *fs_name*	为名字为 *fs_name* 的文件系统格式化名字为 **dev** 设备。

以下例子是格式化 dev 目录下的第一个 MTD 设备，供 JFFS2 文件系统使用：

```
Ruijie# makefs dev/dev/mtd/mtdblock/1 fs jffs2
```

按照上面的配置将为 JFFS2 文件系统格式化 MTDBLOCK 目录下的一个设备，清空设备的数据供文件系统使用。

56.2.6 创建目录

在特权用户模式下，按如下步骤在指定的位置创建需要的目录：

命令	作用
Ruijie# **mkdir** *directoryname*	创建目录

以下例子是在根目录下面创建一个 BAK 目录：

```
Ruijie# mkdir bak
```

56.2.7 移动文件

在特权用户模式下，将指定的文件移动到指定的目录或者文件：

命令	作用
Ruijie# **rename flash:** *old_filename* **flash:** *new_filename*	将名字为 *old_filename* 文件命名成名字为 *new_filename* 的文件 。

56.2.8 显示当前工作路径

在特权用户模式下，按如下步骤设置可以显示当前的工作路径信息：

命令	作用
Ruijie# **pwd**	显示当前的工作路径信息

56.2.9 删除文件

在特权用户模式下，按如下步骤设置来完成永久删除一个文件的功能：

命令	作用
Ruijie# **del** *filename*	删除指定的文件。

下面的例子是删除 MNT 目录下一个叫做 large.c 的临时文件：

```
Ruijie# del mnt/large.c
```

56.2.10 删除空目录

在特权用户模式下，按如下步骤设置可以永久删除指定的空目录：

命令	作用
Ruijie# **rmdir** *directoryname*	删除一个空目录。

以下的例子是删除一个空的目录 MNT：

```
Ruijie# rmdir mnt
```

57 系统管理配置

CPU 利用率查看

配置任务列表

- 显示系统 CPU 利用率
- 配置 CPU 利用率触发门限

显示系统 CPU 利用率

使用 **show cpu** 命令将显示系统总的 CPU 利用率和每个任务 CPU 利用率的相关信息。

命令	作用
Ruijie# **show cpu**	查看系统 CPU 利用率信息

缺省情况下，交换机的名称为 Ruijie。

下面举例显示 **show cpu** 运行的结果：

```
Ruijie# show cpu
=======================================
    CPU Using Rate Information
CPU utilization in five seconds: 25%
CPU utilization in one minute  : 20%
CPU utilization in five minutes: 10%
  NO   5Sec   1Min   5Min   Process
  0     0%     0%     0%    LISR INT
  1     7%     2%     1%    HISR INT
  2     0%     0%     0%    ktimer
  3     0%     0%     0%    atimer
  4     0%     0%     0%    printk_task
  5     0%     0%     0%    waitqueue_process
  6     0%     0%     0%    tasklet_task
  7     0%     0%     0%    kevents
  8     0%     0%     0%    snmpd
  9     0%     0%     0%    snmp_trapd
 10     0%     0%     0%    mtdblock
 11     0%     0%     0%    gc_task
 12     0%     0%     0%    Context
```

```
13    0%    0%    0%    kswapd
14    0%    0%    0%    bdflush
15    0%    0%    0%    kupdate
16    0%    3%    1%    ll_mt
17    0%    0%    0%    ll main process
18    0%    0%    0%    bridge_relay
19    0%    0%    0%    d1x_task
20    0%    0%    0%    secu_policy_task
21    0%    0%    0%    dhcpa_task
22    0%    0%    0%    dhcpsnp_task
23    0%    0%    0%    igmp_snp
24    0%    0%    0%    mstp_event
25    0%    0%    0%    GVRP_EVENT
26    0%    0%    0%    rldp_task
27    0%    2%    1%    rerp_task
28    0%    0%    0%    reup_event_handler
29    0%    0%    0%    tpp_task
30    0%    0%    0%    ip6timer
31    0%    0%    0%    rtadvd
32    0%    0%    0%    tnet6
33    2%    0%    0%    tnet
34    0%    0%    0%    Tarptime
35    0%    0%    0%    gra_arp
36    0%    0%    0%    Ttcptimer
37    8%    1%    0%    ef_res
38    0%    0%    0%    ef_rcv_msg
39    0%    0%    0%    ef_inconsistent_daemon
40    0%    0%    0%    ip6_tunnel_rcv_pkt
41    0%    0%    0%    res6t
42    0%    0%    0%    tunrt6
43    0%    0%    0%    ef6_rcv_msg
44    0%    0%    0%    ef6_inconsistent_daemon
45    0%    0%    0%    imid
46    0%    0%    0%    nsmd
47    0%    0%    0%    ripd
48    0%    0%    0%    ripngd
49    0%    0%    0%    ospfd
50    0%    0%    0%    ospf6d
51    0%    0%    0%    bgpd
52    0%    0%    0%    pimd
53    0%    0%    0%    pim6d
54    0%    0%    0%    pdmd
55    0%    0%    0%    dvmrpd
56    0%    0%    0%    vty_connect
57    0%    0%    0%    aaa_task
58    0%    0%    0%    Tlogtrap
```

```
 59     0%     0%     0%    dhcp6c
 60     0%     0%     0%    sntp_recv_task
 61     0%     0%     0%    ntp_task
 62     0%     0%     0%    sla_deamon
 63     0%     3%     1%    track_daemon
 64     0%     0%     0%    pbr_guard
 65     0%     0%     0%    vrrpd
 66     0%     0%     0%    psnpd
 67     0%     0%     0%    igsnpd
 68     0%     0%     0%    coa_recv
 69     0%     0%     0%    co_oper
 70     0%     0%     0%    co_mac
 71     0%     0%     0%    radius_task
 72     0%     0%     0%    tac+_acct_task
 73     0%     0%     0%    tac+_task
 74     0%     0%     0%    dhcpd_task
 75     0%     0%     0%    dhcps_task
 76     0%     0%     0%    dhcpping_task
 77     0%     0%     0%    dhcpc_task
 78     0%     0%     0%    uart_debug_file_task
 79     0%     0%     0%    ssp_init_task
 80     0%     0%     0%    rl_listen
 81     0%     0%     0%    ikl_msg_operate_thread
 82     0%     0%     0%    bcmDPC
 83     0%     0%     0%    bcmL2X.0
 84     3%     3%     3%    bcmL2X.0
 85     0%     0%     0%    bcmCNTR.0
 86     0%     0%     0%    bcmTX
 87     0%     0%     0%    bcmXGS3AsyncTX
 88     0%     2%     1%    bcmLINK.0
 89     0%     0%     0%    bcmRX
 90     0%     0%     0%    mngpkt_rcv_thread
 91     0%     0%     0%    mngpkt_recycle_thread
 92     0%     0%     0%    stack_task
 93     0%     0%     0%    stack_disc_task
 94     0%     0%     0%    redun_sync_task
 95     0%     0%     0%    conf_dispatch_task
 96     0%     0%     0%    devprob_task
 97     0%     0%     0%    rdp_snd_thread
 98     0%     0%     0%    rdp_rcv_thread
 99     0%     0%     0%    rdp_slot_change_thread
100     4%     2%     1%    datapkt_rcv_thread
101     0%     0%     0%    keepalive_link_notify
102     0%     0%     0%    rerp_msg_recv_thread
103     0%     0%     0%    ip_scan_guard_task
104     0%     0%     0%    ssp_ipmc_hit_task
```

```
105     0%     0%     0%    ssp_ipmc_trap_task
106     0%     0%     0%    hw_err_snd_task
107     0%     0%     0%    rerp_packet_send_task
108     0%     0%     0%    idle_vlan_proc_thread
109     0%     0%     0%    cmic_pause_detect
110     1%     1%     1%    stat_get_and_send
111     0%     1%     0%    rl_con
112    75%    80%    90%    idle
```

在上面的列表中，开头的 3 行分别表示系统在最近 5 秒钟、最近 1 分钟、最近 5 分钟内总的 CPU 利用率情况（包括 LISR、HISR 和任务）。下面则是具体的 CPU 利用率分布情况。其中，每一列的含义如下：

- No：序号
- 5Sec：每一行表示的任务最近 5 秒钟内的 CPU 利用率
- 1Min：每一行表示的任务最近 1 分钟内的 CPU 利用率
- 5Min：每一行表示的任务最近 5 分钟内的 CPU 利用率
- Process：任务名称

表格的前 2 行比较特殊，分别表示所有 LISR 的 CPU 利用率和所有 HISR 的 CPU 利用率，从第 3 行开始，就表示任务的 CPU 利用率了。最后一行是 idle 线程的 CPU 利用率，跟 Windows 下的“System Idle Process”一样，表示系统的空闲状态。在上面的例子中，idle 线程 5 秒内的 CPU 利用率为 75%，说明当前 CPU 有 75%是处于空闲状态。

配置 CPU 利用率日志信息触发门限

要手工配置 CPU 利用率日志信息触发门限,可以使用 cpu-log，使用该命令可以配置 CPU 利用率日志信息的触发门限。

命令	作用
cpu-log log-limit *low_num high_num*	配置 CPU 利用率日志信息触发高门限和触发低门限

默认情况下该高门限为 100%，低门限为 90%。

下面的示例是将 CPU 利用率触发低门限配置为 70%，高门限配置为 80%。

```
Ruijie# configure terminal                    // 进入全局配置模式
Ruijie(config)# cpu-log log-limit 70 80       //设置 CPU 利用率触发门限
```

若 CPU 利用率高于 80%将显示如下信息：

```
Oct 20 15:47:01 %SYSCHECK-5-CPU_USING_RATE: CPU utilization in one minute : 95% , Using most cpu's task is ktimer : 94%
```

若 CPU 利用率低于 70%将显示如下信息：

```
Oct 20 15:47:01 %SYSCHECK-5-CPU_USING_RATE: CPU utilization in one minute :68% , Using most cpu's task is ktimer : 60%
```

```
Oct 20 15:47:01 %SYSCHECK-5-CPU_USING_RATE: The CPU using rate has
down!
```

系统内存状态查看

配置任务列表

- 显示系统总的内存使用情况
- 显示系统所有内存池的使用情况

查看系统内存使用情况

使用 **show memory** 命令将显示系统总的内存使用情况和内存状态的相关信息。

命令	作用
Ruijie# **show memory**	查看系统内存使用情况

缺省情况下，交换机的名称为 Ruijie。

下面举例显示 **show memory** 运行的结果：

```
Ruijie#show memory
Buddy System info ( Active: 4, inactive: 2, free: 19958 ) :
Zone : DMA
watermarks : min 429, low 1716, high 2574
watermarks : min 0, low 0, high 0
watermarks : min 0, low 0, high 0
Totalpages : 25780(103120KB), freepages : 19958(79832KB)
System Memory Statistic:
Total Objects : 89128, Objects Using size 20428KB.
System Total Memory : 128MB, Current Free Memory : 81066KB
slabinfo - (statistics)
============================================================
cache                  |objects                        |slabs
|statistics
------------------------------------------------------------
memory-cache-name   |active   number   size    |active number
order |high   alloc  grown  reaped error
============================================================
kmem_cache        86       87       132      3      3      1
86    86    3     0     0
ssp_rx_packet_pool   2044    2044    2048    1022   1022   1
2044   2044   1022   0     0
ssp_emergen_mem     0       1024    512     0     128    1
```

```
1024   1024   128    0     0
tcp_tw_bucket        0       0       224      0      0      1
0      0      0      0      0
tcp_bind_bucket      1       59      32       1      1      1
1      1      1      0      0
tcp_open_request     0       0       96       0      0      1
0      0      0      0      0
ip_mrt_cache         0       0       576      0      0      1
0      0      0      0      0
tuncache             0       0       132      0      0      1
0      0      0      0      0
tsfnpool             0       0       12       0      0      1
0      0      0      0      0
ARP table            0       0       132      0      0      1
0      0      0      0      0
bst_pool             9       72      20       1      1      1
9      9      1      0      0
sock                 91      91      1184     7      7      4
91     131    7      0      0
clipri               0       0       32       0      0      1
0      0      0      0      0
clieol               1       5       3072     1      1      4
2      81     1      0      0
clipsr               0       8       512      0      1      1
6      37864  1      0      0
waitqueue_block      0       0       20       0      0      1
0      0      0      0      0
hookbait_pool        140     177     32       3      3      1
140    140    3      0      0
skb_head             34      45      224      3      3      1
34     34     3      0      0
long_sk_data         0       0       10048    0      0      4
0      0      0      0      0
sk_data              1       3       2432     1      1      2
1      1      1      0      0
blkdev_requests      1024    1050    96       35     35     1
1024   1024   35     0      0
jffs2_inode_cache    37      67      24       1      1      1
37     37     1      0      0
jffs2_node_frag      3598    3654    28       58     58     1
3603   3611   58     0      0
jffs2_raw_node_ref   7445    7488    16       96     96     1
7445   7445   96     0      0
jffs2_tmp_dnode      0       3640    36       0      65     1
3601   3635   65     0      0
jffs2_raw_inode      0       39      68       0      1      1
```

```
1     11    1     0     0
jffs2_raw_dirent     0       53       40       0      1      1
1     3     1     0     0
jffs2_full_dnode     3602     3666     16       47     47     1
3611   3637   47     0     0
jffs2_i              9       10       768      2      2      1
10    11    2     0     0
devfsd_event         0       0        20       0      0      1
0     0     0     0     0
file_lock_cache      0       0        88       0      0      1
0     0     0     0     0
dnotify_cache        0       0        20       0      0      1
0     0     0     0     0
kiobuf               0       0        64       0      0      1
0     0     0     0     0
cdev_cache           0       0        160      0      0      1
0     0     0     0     0
bdev_cache           1       20       160      1      1      1
1     1     1     0     0
mnt_cache            9       40       64       1      1      1
9     11    1     0     0
inode_cache          128     132      640      22     22     1
128    153    22     0     0
dentry_cache         141     144      128      6      6      1
141    168    6     0     0
filp                 101     120      128      5      5      1
101    101    5     0     0
names_cache          0       2        4096     0      2      1
2     69    2     0     0
buffer_head          0       0        96       0      0      1
0     0     0     0     0
fs_cache             87      118      32       2      2      1
93    96    2     0     0
files_cache          87      99       416      11     11     1
93    96    11    0     0
mc_bitmap            176     180      128      6      6      1
176    176    6     0     0
mdbg_cache           2883    3030     16       15     15     1
2883   6492   15    0     0
size-33554432(DMA)   0          0          33554432 0       0
8192   0     0     0     0     0
size-33554432          0          0          33554432 0       0
8192   0     0     0     0     0
size-16777216(DMA)   0          0          16777216 0       0
4096   0     0     0     0     0
size-16777216          0          0          16777216 0       0
```

```
4096   0      0      0      0      0
size-8388608(DMA)     0          0          8388608   0        0
2048   0      0      0      0      0
size-8388608          0          0          8388608   0        0
2048   0      0      0      0      0
size-4194304(DMA)     0          0          4194304   0        0
1024   0      0      0      0      0
size-4194304          0          0          4194304   0        0
1024   0      0      0      0      0
size-2097152(DMA)     0          0          2097152   0        0
512    0      0      0      0      0
size-2097152          0          0          2097152   0      0      512
0      0      0      0      0
size-1048576(DMA)     0          0          1048576   0        0
256    0      0      0      0      0
size-1048576          0          0          1048576   0      0      256
0      0      0      0      0
size-524288(DMA)      0          0          524288    0      0      128
0      0      0      0      0
size-524288           0          1          524288    0      1      128
1      1      1      0      0
size-262144(DMA)      0          0          262144    0      0      64
0      0      0      0      0
size-262144           7          7          262144    7      7      64
7      7      7      0      0
size-131072(DMA)      0          0          131072    0      0      32
0      0      0      0      0
size-131072           18         18         131072    18     18     32
18     18     18     0      0
size-65536(DMA)       0          0          65536     0      0      16
0      0      0      0      0
size-65536            34         39         65536     34     39     16
39     44     39     0      0
size-32768(DMA)       0          0          32768     0      0      8
0      0      0      0      0
size-32768            56         57         32768     56     57     8
57     64     57     0      0
size-16384(DMA)       0          0          16384     0      0      4
0      0      0      0      0
size-16384            72         74         16384     72     74     4
74     77     74     0      0
size-8192(DMA)        0          0          8192      0      0      2
0      0      0      0      0
size-8192             24         24         8192      24     24     2
24     34     24     0      0
size-4096(DMA)        0          0          4096      0      0      1
```

```
0     0     0     0     0
size-4096           148      148      4096     148    148    1
148    2962   148    0     0
size-2048(DMA)       0        0        2048     0      0      1
0     0     0     0     0
size-2048           63       64       2048     32     32     1
63     403    32     0     0
size-1024(DMA)       0        0        1024     0      0      1
0     0     0     0     0
size-1024           188      188      1024     47     47     1
188    473    47     0     0
size-512(DMA)        0        0        512      0      0      1
0     0     0     0     0
size-512            160      161      512      23     23     1
160    440    23     0     0
size-256(DMA)        0        0        256      0      0      1
0     0     0     0     0
size-256            92       104      256      8      8      1
95     1244   8      0     0
size-128(DMA)        0        0        128      0      0      1
0     0     0     0     0
size-128            248      288      128      11     12     1
272    787    12     0     0
size-64(DMA)         0        0        64       0      0      1
0     0     0     0     0
size-64             16519    16560    64       414    414    1
16534  17990  414    0     0
size-32(DMA)         0        0        32       0      0      1
0     0     0     0     0
size-32             49854    49914    32       845    846    1
49866  90468  846    0     0
============================================================
```

上面列表中的信息包含如下几个部分：

1. 页分配 Buddy System 信息：

命令		说明
Active		文件系统正在使用的页数，不可回收
Inactive		文件系统已用完的页，可回收
Free		所有区的空闲页总和
Zone		分区名，现在就只有 DMA 和 Normal，如果物理内存总量在 256M 以内，则只有 DMA 区
Zone	watermasks 1	带 DMA 申请标志申请 DMA 池所用的水线

DMA		min：水线的下限，当空闲内存低于它时不允许申请 low：低水线值，当空闲内存低于它时触发回收线程进行回收 high：高水线值，当空闲内存高于它时回收线程停止回收
	watermasks 2	不带 DMA 申请标志申请 DMA 池所用的水线
	watermasks 3	无用
Zone Normal	watermasks 1	Normal 分区所有的水线（上例中由于内存大小低于 256M，因此无 Normal 分区）
	watermasks 2	无用
Totalpages		该区的总页数，括号里是换算出来的字节数
Freepages		该区空闲可用的页数，括号里是换算出来的字节数

2. 系统内存信息：

命令	说明
Total Objects	缓冲池总共申请的对象数
Object using size	所有对象占用的内存空间(不包括内存的管理结构)
System Total Memory	系统的物理内存总量
Systrem Free Memory	系统总共剩余内存，包括空闲页空间和缓冲池的所有空闲空间

3. 缓冲池信息：

命令		说明
Cache	Memory-cache-name	缓冲池的名字
Objects	Active	该池正在使用中的对象数
	Number	该池总的对象数
	Size	该池对象的大小，单位 Byte
Slabs	active	该池正在使用的 slab 数（里面至少有一个对象正在使用中）
	number	该池总共的 slab 数
	order	该池的一个 slab 大小，单位为页
Statisti	High	该池使用高峰时，同时被使用的对象数

cs	Alloc	该池被申请的次数
	Grown	该池增长的次数，一次增长该池增加一个 slab
	Reaped	该池被回收的次数，一次回收该池减少一个 slab
	Error	错误次数，无用

门限配置

门限概述

门限（Threshold）是用来了解系统状态的特殊值。有 CPU 利用率门限、内存利用率门限。每类都有两级的门限，第一级门限为告警门限，第二级门限为严重告警门限。门限除了可以使用 CLI 进行设置和读取以外，还可以使用 MIB 进行设置和读取。

57.1.1 门限基本概念

57.1.1.1 CPU 利用率门限

CPU 利用率门限用来指示当前 CPU 的使用状况。可供 MIB 软件读取，没有相关的 log。

57.1.1.2 内存利用率门限

内存利用率门限用来指示当前内存的使用状况。可供 MIB 软件读取，没有相关的 log。

缺省配置

下表用来描述门限的缺省配置。

功能特性	缺省值
CPU 利用率门限	告警门限为 90，严重告警门限为 100
内存利用率门限	告警门限为 90，严重告警门限为 100

配置门限

以下章节描述如何配置门限：

- 配置CPU利用率门限
- 配置内存利用率门限
- 查看配置

配置 CPU 利用率门限

	命令	作用
Step 1	Ruijie# **configure terminal**	进入全局配置模式。
Step 2	Ruijie(config)# **threshold set cpu [M1 | M2 | slot** *n* **| member** *n***]** *warning_value critical_value*	配置指定设备的 CPU 利用率告警门限和严重告警门限，范围为 1~100。

如果要恢复到缺省值，可用 **no threshold set cpu** 配置命令恢复配置。

配置内存利用率门限

	命令	作用
Step 1	Ruijie# **configure terminal**	进入全局配置模式。
Step 2	Ruijie(config)# **threshold set memory [M1 | M2 | slot** *n* **| member** *n***]** *warning_value critical_value*	配置指定设备的内存利用率告警门限和严重告警门限，范围为 1~100。

如果要恢复到缺省值，可用 **no threshold set memory** 配置命令恢复配置。

查看配置

门限提供了如下的显示命令用于查看各类门限值，各命令的功能说明如：

命令	作用
show threshold cpu	显示设备的 CPU 利用率门限
show threshold memory	显示设备的内存利用率门限

> 说明　以上显示命令均能在除用户模式外的任何模式配置。

门限典型配置举例

配置步骤

使用如下命令将 CPU 利用率的两级门限设置为 80 和 90，温度的两级门限设置为 50 和 80。

```
Ruijie# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Ruijie(config)#threshold set cpu member  1 80 90
Ruijie(config)#threshold set temperature member 1 60 80
```

显示验证

使用 show threshold 查看各个门限的值

```
Ruijie#show threshold  cpu
Device       Warning     Critical
Member 1:        80          90
Ruijie#show threshold memory
Device       Warning     Critical
Member 1:        90         100
Ruijie#show threshold temperature
Device       Warning     Critical
Member 1:        60          80
```

由上述信息可以看出，CPU 利用率门限和温度门限已被设置为指定值，而内存利用率门限依然为缺省值。

58 系统日志配置

58.1 概述

设备在运行过程中，有各种状态变化如链路状态 UP、DOWN 等，发生一些事件如收到异常报文、处理异常等。锐捷产品日志提供一种机制，在状态变化或发生事件时，就自动生成固定格式的消息（日志报文），这些消息可以被显示在相关窗口（控制台、VTY 等）上或被记录在相关媒介（内存缓冲区、FLASH）上或发送到网络上的一组日志服务器上，供管理员分析网络情况和定位问题。同时为了方便管理员对日志报文的读取和管理，这些日志报文可以被打上时间戳和序号，并按日志信息的优先级进行分级。

58.1.1 日志报文格式

锐捷产品的日志报文格式如下：
<priority> seq no: timestamp sysname
%ModuleName-severity-MNEMONIC: description

依次是：<优先级> 序号 时间戳 设备名 模块名-严重性-信息简写：信息内容
优先级值＝设备值＊8＋严重性

例子：

```
<189> 226:Mar  5 02:09:10 S3760 %SYS-5-CONFIG_I: Configured
from console by console
```

注意:

在用户窗口打印的日志报文是不带优先级字段的，优先级字段只出现在发送给 Syslog Server 的日志报文中。

58.2 日志配置

58.2.1 日志开关

日志开关默认情况下是打开的，如果关闭日志开关，设备将不会在用户窗口打印日志信息，不会将日志信息发送给 Syslog 服务器，也不会将日志信息记录在相关媒介（内存缓冲区、FLASH）上。

要打开或关闭日志开关，请在全局配置模式下执行以下命令：

命令	作用
Ruijie(config)# **logging on**	打开日志开关
Ruijie(config)# **no logging on**	关闭日志开关

注意：

一般情况下，不要关闭日志开关，如果觉得打印的信息太多，则可以通过设置不同设备日志信息的显示级别来减少日志信息的打印。

58.2.2 设置日志信息显示设备

打开日志开关以后，日志信息不仅可以在控制台上显示，也可以发送给不同的显示设备。

要设置接收日志的不同显示设备，请在全局配置模式或特权用户层下执行以下命令：

命令	作用
Ruijie(config)# **logging buffered** [*buffer-size* | *level*]	将日志记录到内存缓冲区
Ruijie# **termninal monitor**	允许日志信息显示在 VTY 窗口上
Ruijie(config)# **logging** *host*	将日志信息发送给网络上的 Syslog Sever
Ruijie(config)# **logging file flash:***filename* [*max-file-size*] [*level*]	将日志信息记录到扩展 FLASH 上

Logging Buffere：将日志信息记录到内存缓冲区。日志内存缓冲区是循环使用的，即如果内存缓冲区满以后，最早的日志信息将被覆盖。要显示内存缓冲区中的日志信息，请在特权用户层执行命令 **show logging** 。要清除内存缓冲区中的日志信息，请在特权用户层执行命令 **clear logging** 。

Termninal Monitor：允许日志信息在当前 VTY(如 Telnet 窗口)上显示。
Logging Host：指定接收日志信息的 Syslog Server 地址，锐捷产品 允许配置最多 5 个 Syslog Server。日志信息将被同时分给配置的所有的 Syslog Server。

注意：

要将日志信息发送给 Syslog Server，必须打开日志信息的时间戳开关或序号开关，否则日志信息将不会被发给 Syslog Server。

Logging File Flash：将日志信息保存到 FLASH 中，日志文件名不要带文件类型的后缀名。日志文件后缀为固定为 TXT，配置文件后缀名将被拒绝。

More Flash：Filename 命令可以查看 Flash 日志文件的内容

注意：

部分设备支持扩展 FLASH，如果设备存在扩展 FLASH，日志信息将记录在扩展 FLASH 中。如果设备没有扩展 FLASH，日志信息将记录在串行 FLASH 中。

58.2.3 启用日志信息时间戳开关

要在日志信息上添加或取消时间戳，请在全局配置模式下执行以下命令：

命令	作用
Ruijie(config)# **service timestamps** *message-type* [**uptime** | **datetime**]	启用日志信息中的时间戳
Ruijie(config)# **no service timestamps** *message-type*	关闭日志信息中的时间戳

时间戳格式有两种：设备启动时间(Uptime)或者设备日期(Datetime)。请根据需要选择不同类型的时间戳。

消息类型：Log 或 Debug，Log 信息是指在严重性级别在 0－6 之间的日志信息，Debug 信息是严重性级别为 7 的日志信息。

注意：

如果当前设备不存在 RTC，配置设备时间无效。自动采用启机时间做为日志信息时间戳。

58.2.4 启用日志信息系统名开关

默认情况下，日志信息不带系统名。要日志信息加上或取消系统名，请在全局配置模式下执行以下命令：

命令	作用
Ruijie(config)# **no service sysname**	在日志报文中取消系统名。

Ruijie(config)**# service sysname**	在日志报文中添加系统名。

58.2.5 启用日志信息统计功能开关

默认情况下，日志信息统计功能关闭。要打开或者关闭日志信息统计功能，请在全局配置模式下执行以下命令：

命令	作用
Ruijie(config)**# no logging count**	关闭日志信息统计功能，并清除日志信息统计数据。
Ruijie(config)**# logging count**	打开日志信息统计功能。

58.2.6 启用日志信息序列号开关

默认情况下，日志信息不带序列号。要日志信息加上或取消序列号，请在全局配置模式下执行以下命令：

命令	作用
Ruijie(config)**# no service sequence-numbers**	在日志报文中取消序列号。
Ruijie(config)**# service sequence-numbers**	在日志报文中添加序列号。

58.2.7 设置用户输入与日志信息输出同步

默认情况下，用户输入与日志信息输出不同步，当用户正在输入字符时，如果有日志信息输出，用户输入将被打断。要设置用户输入与日志信息输出同步，请在线路配置模式下执行以下命令：

命令	作用
Ruijie(config-line)**# logging synchronous**	设置用户输入与日志信息输出同步；
Ruijie(config)**# no logging synchronous**	取消用户输入与日志信息输出同步；

58.2.8 设置日志信息速率控制

默认情况下，日志信息不进行速率控制。要进行日志信息速率控制，请在全局配置模式下执行以下命令：

命令	作用

Ruijie(config)# **logging rate-limit** *rate*	设置对日志信息进行速率控制
Ruijie(config)# **no logging rate-limit**	取消对日志信息进行速率控制

58.2.9 设置日志信息显示级别

要限制在不同设备上显示的日志报文个数，可以通过设置不同设备上允许显示日志信息的严重性级别来实现。

要设置日志信息显示的级别，请在全局配置模式下执行以下命令：

命令	作用
Ruijie(config)# **logging console** *level*	设置允许在控制台上显示的日志信息级别
Ruijie(config)# **logging monitor** *level*	设置允许在 VTY 窗口(如 telnet 窗口)上显示的日志信息级别
Ruijie(config)# **logging buffered** [*buffer-size* | *level*]	设置允许记录在内存缓冲区的日志信息级别
Ruijie(config)# **logging file flash:***filename* [*max-file-size*] [*level*]	设置允许记录在扩展 FLASH 上的日志信息级别
Ruijie(config)# **logging trap** *level*	设置允许发送给 Syslog Server 的日志信息级别

锐捷产品的日志信息分为以下 8 个级别：

关键字	等级	描述
Emergencies	0	紧急情况，系统不能正常运行
Alerts	1	需要立即采取措施改正的问题
Critical	2	重要情况
Errors	3	错误信息
warnings	4	警告信息
Notifications	5	普通类型，不过需要关注的重要信息
informational	6	说明性的信息
Debugging	7	调试信息

值越小，级别越高，即 0 级别的信息是最高级别的信息。

当指定设备设置允许显示的日志信息级别以后，所有等于或低于所设置值级别的日

志信息将被允许显示。如配置命令 **logging console *6*** 以后，所有级别为 6 或小于 6 的日志信息将被显示在控制台上。

控制台默认允许显示的日志信息级别为 7。

VTY 窗口默认允许显示的日志信息级别为 7。

默认发送给 Syslog Server 日志信息级别为 6。

默认允许被记录在内存缓冲区的日志信息级别为 7。

默认允许被记录在扩展 FLASH 中日志信息级别为 6。

可以通过特权命令 **show logging** 来查看允许在不同设备上显示的日志信息级别。

58.2.10 设置日志信息的设备值

设备值是构成发送给 Syslog Server 报文优先级字段的一部分，指示产生信息的设备类型。

要设置日志信息的设备值，请在全局配置模式下执行以下命令：

命令	作用
Ruijie(config)# **logging facility** *facility-type*	设置日志信息的设备值
Ruijie(config)# **no logging facility** *facility-type*	恢复日志信息的设备值为默认值。

下面是各种设备值的含义：

```
Numerical Code          Facility
         0          kernel messages
         1          user-level messages
         2          mail system
         3          system daemons
         4          security/authorization messages
         5          messages generated internally by syslogd
         6          line printer subsystem
         7          network news subsystem
         8          UUCP subsystem
         9          clock daemon
        10          security/authorization messages
        11          FTP daemon
        12          NTP subsystem
        13          log audit
        14          log alert
        15          clock daemon
        16          local use 0  (local0)
```

```
17          local use 1  (local1)
18          local use 2  (local2)
19          local use 3  (local3)
20          local use 4  (local4)
21          local use 5  (local5)
22          local use 6  (local6)
23          local use 7  (local7)
```

锐捷产品默认设备值为 23。

58.2.11 设置日志报文的源地址

默认情况下，发送给 Syslog Server 的日志报文源地址为发送报文接口的地址，可以通过命令来固定所有日志报文的源地址。

可以直接设定 Log 报文的源 IP 地址，也可以设定 Log 报文的远端口。

要设置日志报文的源地址，请在全局配置模式下执行以下命令：

命令	作用
Ruijie(config)# **logging source interface** *interface-type interface-number*	设置日志信息的源端口
Ruijie(config)# **logging source ip** *A.B.C.D*	设置日志报文的源 ip 地址

58.2.12 设置发送用户 LOG 信息

默认情况下，用户登录与退出，以及执行配置命令，均不输出 LOG 信息。如果需要输出用户登录/退出的 LOG 信息，或者需要输出配置命令的 LOG 信息。请在全局配置模式下执行以下命令：

命令	作用
Ruijie(config)# **logging userinfo**	设置用户登录/退出的 LOG 信息；
Ruijie(config)# **logging userinfo command-log**	设置执行配置命令时，发送 LOG 信息

58.3 日志监控

为了监控日志信息，请在特权用户模式下执行以下命令：

命令	作用
Ruijie# **show logging**	查看内存缓冲区中的日志报文，以及日志相关统计信息
Ruijie# **show logging count**	查看系统中各模块日志信息统计情况

Ruijie# **show logging reverse**	按从新到旧顺序，查看内存缓冲中的日志报文，以及日志相关统计信息
Ruijie# **clear logging**	清除内存缓冲区中的日志报文
Ruijie# **more flash:***filename*	查看扩展 FLASH 中的日志文件

注意：

show logging count 命令输出信息中，显示的时间戳格式，是该日志信息最后一次输出时的时间戳格式。

58.3.1 日志配置举例

以下配置一个启用日志功能的一个典型例子：

```
Ruijie(config)# interface gigabitEthernet 0/1
Ruijie(config-if)# ip address 192.168.200.42 255.255.255.0
Ruijie(config-if)# exit
Ruijie(config)# service sequence-numbers                    //启用序列号
Ruijie(config)# service timestamps debug datetime    //启用 debug 信息时间戳，日期格式
Ruijie(config)# service timestamps log datetime  //启用 log 信息时间戳，日期格式
Ruijie(config)# logging 192.168.200.2     //指定 syslog server 地址
Ruijie(config)# logging trap debugging          //所有级别的日志信息将发给 syslog server
Ruijie(config)# end
```